i2 iBase documentation

Welcome to the i2 iBase documentation, where you can find information about how to use and administer i2 iBase.

i2 iBase is an intuitive intelligence data management application that enables collaborative teams of analysts to capture, control, and analyze multi-source data in security-rich workgroup environments. It addresses the analyst's daily challenge of discovering and uncovering networks, patterns and trends in today's increasing volumes of complex structured and unstructured data. i2 iBase provides a multi-user data sharing environment that combines rich analysis and visualization capabilities with dissemination tools.

Licensing i2 iBase 10 requires a valid license. The software licensing documentation provides more information.

Support
The i2 iBase support page contains links to the release notes and support articles.

- i2 iBase support

Installing i2 iBase

You can install i2 iBase using an Installation Manager.

Before you begin

You should review the release notes on the i2 iBase support page to ensure your setup meets the prerequisites and system requirements.

If you want to use Analyst's Notebook 10 or later, with iBase 9.2.1 or earlier, see installing i2 iBase Notebook Extension.

About this task

i2 iBase Geographic Information Systems Interfaces are installed with iBase. If you intend to use a mapping application with iBase, then in most cases you must install the mapping application first.

Procedure

1. Extract the product files from your downloaded distribution.
2. Using Windows Explorer, browse to the root of the distribution and run `Setup.exe`.
3. Follow the prompts. You will be asked for the setup type:
   - **Typical**

     Installs iBase User along with its documentation.
   - **Complete**

     Installs:
     - iBase User and iBase Designer.
     - All tools apart from the iBase Index Service Configuration tool and iBase Database Replication.
     - All of the help documentation.

© N. Harris Computer Corporation

- **Custom**

  You can select which parts of iBase you require.

4. Follow the prompts to complete the installation.

   Note: If you installed the Coordinate Extensions option, as part of a custom installation, then the following message might display when you first start iBase:

   ```
   An application plug-in failed to load: i2 iBase Bulk Coordinate Converter.
   ```

   This message indicates that a required environment variable does not exist. To resolve this problem, either log off from Windows, or restart the computer.

5. If you intend to use Microsoft SQL Server to store your iBase databases, you need to ensure that the connection to SQL Server is not blocked by a firewall or similar.

   To check the port, open the SQL Server Configuration Manager and navigate to **SQL Server Network Configuration > Protocols for MSSQLSERVER**. Right-click **TCP/IP** and select **Properties**. The port is set in the **IP Addresses** tab; the default port is 1433.

What to do next

If you want to make changes to your installation, go to the **Control Panel** on your computer and click **Programs and Features**. Highlight i2 iBase in the programs list and click **Change > Next** to open the **Program Maintenance** options.

Select **Modify** and click **Next**. Then from the **Custom Setup** screen, select any additional features that you would like to install. Click **Next** to modify the installation.

Installation and application data folders

When you install iBase, you can install it in the folder suggested by the installer or to a folder of your choice. Regardless of where you choose to install the product, any data that is used by the i2® application is automatically copied to the application data folder as defined by the version of Microsoft™ Windows™ that you are running. These are hidden Windows folders.

The application data folder is defined by the version of Microsoft Windows that you are running. Users also have a folder for storing files such as iBase templates. The folder can also contain shortcuts to other folders that contain per user application data.

Per machine data

Data that is specific to the machine on which iBase is installed is held in the per machine application data area given previously. This is a copy of data in `C:\Program Files`. You should not use any data held in the Program Files area. If you choose to copy configuration files from one machine to another, then you should always overwrite the files in the application data area.

Data of this type consists of configuration files such as:

Folder	Files or folders
`i2\i2 iBase <n>\en- us\Configuration`	`Iconlist.txt`, `Military Iconlist.txt`, `Combined Iconlist.txt`, `FTSexclude.txt`, `WSexclude.txt`
`i2\i2 iBase <n>\en-us\CommandGroups`	`CommandGroups.mdb`

© N. Harris Computer Corporation

Folder	Files or folders
`i2\i2 iBase <n>\en-us\Settings`	`Settings.xml` (as set by options in the Options dialog) (**Note:** All users have read/write access to this file unless you change the permissions on the file).
`i2\i2 iBase <n>\en-us \WorkgroupTemplates`	`*.idt` files (the default workgroup templates and any templates that you want to make available to all users).
`i2\i2 iBase <n>\en-us\Mapping`	Mapping configuration files. For information on the mapping configuration files, see the release notes for iBase GIS Interfaces.
`i2\i2 iBase <n>\en-us\Scheduler`	`Scheduler.mdb` (you can specify an alternative location).

Per user data

Application data that is specific to a user of the machine is copied to, or created in, the per user application data folder given previously.

Installing iBase Extended Features

When you use the Custom setup to install iBase, you can choose which options to include from the available iBase Extended Features.

The iBase Installation Manager offers the Custom setup option when you first install iBase, and if you want to modify an existing installation. Click the Extended Features icon in the list, and select the features you want to include in the installation.

Feature Name	Description
Coordinate Extensions Option	Enables you to convert coordinates from one format to another, and provides tools to perform geographical queries.
Database Replication Option	Enables the copying of data from a central iBase database to one or more iBase databases. (Chargeable component)
GIS Interfaces	Enables use of GIS mapping applications. Analysis tools in the mapping application allow visual analysis of entities and links plotted on maps.
Plate Analysis	A dedicated environment designed to help manage and analyze Automatic License Plate Recognition (ALPR) data. Plate Analysis only works with SQL Server databases.

© N. Harris Computer Corporation

Feature Name	Description
Tools	Includes Audit Viewer to view and query the iBase audit logs. Database Configuration allows you to manage connections to SQL Server databases, and Repair Compact repairs damaged iBase Microsoft Access databases.
Scheduler	iBase Scheduler enables automatic scheduling for imports and exports. It comprises several distinct components that can be set up to comply with a variety of network and database configurations.
Chart Item Extractor	Extracts data from Analyst's Notebook charts for storage in iBase databases.

Note: Chargeable components can be used if you have the associated license.

Installing i2 iBase from the command line

iBase is installed using a Microsoft Windows Installer. You can use the msi command line options to install iBase components.

Before you begin

Ensure that all iBase prerequisites are installed before you install iBase using the command line.

Procedure

To install iBase using the command line:

1. Open a command prompt with administrator privileges.
2. Navigate to the location of the iBase msi file.

   Note: You can also provide the absolute file path to the msi file.
3. Enter the command that specifies the components you would like to install in the following format:

   `msiexec /i "i2 iBase 10.msi" <ADDITIONAL OPTIONS>`

   Where the i2 iBase specific additional options are:

   - INSTALLLEVEL - specify the install level of a feature set. Using this option ensures that you get all options at a level and all the options available at lower levels, ensuring all prerequisite features are present.
   - ADDLOCAL - specify specific features to install
   - I2_LANGUAGE - specify the language that is installed
   - I2LIC_ENABLED - specify whether Product Access Management is enabled.
   - I2LIC_SERVERS - identifies the location of the Product Access Management servers.
   - I2LIC_BROADCASTS_ENABLED - checks for license servers with available licenses on the network.

   For the i2 iBase specific values that are available, see iBase components and language codes. For a list of the Microsoft specific options use `msiexec /h`.

© N. Harris Computer Corporation

Examples:

Install iBase User silently

```
msiexec /i "i2 iBase 10.msi" /qn
```

Install iBase User in French

```
msiexec /i "i2 iBase 10.msi" I2_LANGUAGE=fr
```

Install iBase Client components

```
msiexec /i "i2 iBase 10.msi" INSTALLLEVEL=200
```

Install iBase User and Designer with Plate Analysis and Scheduler

```
msiexec /i "i2 iBase 10.msi"
 ADDLOCAL=ThirdParty,User,Designer,ANPR,Scheduler,SchedulerService /qn
```

iBase components and language codes

The following options can be used in command line installs of iBase.

Feature Names for iBase and iBase components

Feature Name	Default Installation Level	Description	Prerequisite Feature
ThirdParty	50	Microsoft and other third-party merge modules that are required to use iBase features.	
User	100	The iBase User client components.	ThirdParty
UserExamples	100	iBase User example files.	ThirdParty, User
Help	100	Help files for iBase User, and iBase Designer.	ThirdParty
Designer	200	Design and administer iBase databases and security files.	ThirdParty
DesignerExamples	200	iBase Designer example files.	ThirdParty, Designer
CoordinateExt	200	Enables multiple coordinate systems in a database.	ThirdParty

© N. Harris Computer Corporation

Feature Name	Default Installation Level	Description	Prerequisite Feature
GISArcGIS	200	Interface to ArcGIS Pro.	ThirdParty
GISMapInfo	200	Interface to MapInfo.	ThirdParty
ANPR	200	Plate Analysis Features.	ThirdParty
Tools	200	Includes Audit Viewer and tools for maintaining iBase databases.	ThirdParty
Scheduler	200	Schedule batch imports and exports.	ThirdParty
SchedulerService	200	Run scheduled tasks.	ThirdParty, Scheduler
ChartItemExtractor	200	Extracts charts into individual properties allowing them to be stored in iBase.	ThirdParty
Server	300	iBase Server components for the search service.	ThirdParty
Replication	300	Use Microsoft SQL Server merge replication to enable distributed use of iBase (Chargeable Component).	ThirdParty

For details of the features available when you use the Custom setup, see Installing iBase Extended Features.

Supported language options

Language	languageCode
English	en
Arabic	ar-SA
Chinese (Simplified Han)	zh-Hans
Chinese (Traditional Han)	zh-Hant

© N. Harris Computer Corporation

Language	languageCode
Czech	cs
French	fr
German	de
Italian	it
Japanese	ja
Polish	pl
Portuguese (Brazil)	pt-BR
Spanish	es

Designing i2 iBase

i2 iBase provides powerful solutions for configuring, capturing, controlling, analyzing and displaying complex information and relationships in link and entity data. iBase is both a database application, and a modeling and analysis tool. You design and maintain your iBase databases using i2 iBase Designer.

Licensing information

i2 iBase is a licensed product and requires a valid software license. Depending on your organization's arrangements, you might need to provide a license directly or connect to a license server.

There are a number of ways you might provide your license information in order to activate iBase.

Standalone licenses

Standalone licenses are stored on the same workstation as your installation, and are provided in the form of license keys or license files.

Network licenses

You can use a floating license when your workstation is connected to a license server on your network. These licenses can also be borrowed for offline access, and returned when no longer required.

Internet licenses

When you have internet access, you can use a license from i2's internet license server.

Support

For detailed information about licensing, refer to the i2 Software licensing guide.

If you need further help with iBase licensing, contact your system administrator or i2 Support.

© N. Harris Computer Corporation

Introducing iBase Designer

The Database Window is a complete view of your database. The Database Explorer tree view on the left of the Database window shows all the objects in the database, and the Details window on the right displays information on the selected object.

Types of object in the database

Name of your database

Describes the database and statistics for the database, for example how many entity types are defined in the database.

Entity Types

Lists the entity types defined in this database. The Details window shows the number of fields, the default icons and the setting of the option in the **Selected in 'Expand' list** column (this determines which entities are displayed when you expand to a chart).

Link Types

Lists the link types in this database. The Details window shows the number of fields, the color used for the link and the setting of the option in the **Selected in 'Expand' list** column.

To view the fields and how they are defined, expand the Link Types, and then click the required link type.

Labeling Schemes

Lists the labeling schemes in the database, along with the total number of labels defined for use in iBase.

Code Lists

Shows the number and type of code lists in the database.

For details of the code lists of a specific type, such as their names and the number of items they contain, Expand Code Lists, and then click the list type, such as Pick Lists.

To see the actual values in a code list, Expand the list type and select a specific list.

Chart Attributes

Shows the chart attributes in the database. The Details window lists the symbols, prefixes and suffixes, and whether these are displayed on charts.

Standard Fields

Shows the standard fields that are defined in the database.

Functions

Lists some of the tools available in iBase Designer. For example, click Datasheet Manager to list the custom forms in this database. These tools are also available on the Tools menu.

Note: Depending on the database design some of these object types might be empty.

© N. Harris Computer Corporation

Logging on to iBase Designer

When you log on to iBase Designer, you open a security file that defines the permissions for the user account to which you are logged on. To close a security file, you should log off.

About this task

When you are logged on to a security file, depending on the permissions of the user account, you can:

- Open one of the databases controlled by the current security file
- Maintain security policies, user groups and users
- Create a new database
- Make changes to iBase as installed on the local computer, for example to the:
  - Plug-ins available on the computer
  - Basic charting, and advanced settings for using iBase
  - Recently-used databases listed on the File menu

Procedure

1. Select **File** > **Logon**

   Note: Select **Logon As** if you usually log on using your Windows credentials but on this occasion want to log on using an iBase user name and password.
2. In the Security File dialog, browse for the security file to open (the file name will end with .ids).
3. Click **Open**. The Logon dialog is displayed if you have an iBase user name and password. If you use your Windows user name and password, then the dialog is only displayed if you are able to log on as one of several iBase users.
4. If the Logon dialog is displayed:
   - Enter your iBase user name and password
   - Select the iBase user from the list
5. Click **OK** to open the security file.

Logging on as a different user

Depending how Windows security is set up at your site, you might be prompted to select the user to log on as. To avoid repeating this step each time you log on, you might have turned on the **Remember my selection** checkbox in the Logon dialog.

Procedure

To cancel this selection:

1. Log on in the usual way (you do not need to open the database).
2. On the General page, turn off **Remember user for Windows single sign-on**
3. Log off and then log on again. You will then be prompted to select the iBase user log on as.

© N. Harris Computer Corporation

Adding objects

Database objects such as Entity and Link types are used to describe the data that is stored in your database. The results of your analysis depend on the types of object that are defined.

There are three ways of adding a database object, such as a link type or a chart attribute:

- In the Database Explorer window, right-click the object type and from the shortcut menu, select **New**.
- Double-click the icon in the Database Explorer window or the large icon in the Detail window.
- In the Detail window, right-click and from the shortcut menu, select **New**.

To edit objects, expand the object type and then select the specific object, such as a named pick list:

- Right-click and from the menu, select **Edit**.
- Double-click the specific object.

For entity and link types, you can also edit the object by copying and pasting fields into it, or by changing the order in which they are displayed with the Reorder Fields command on the shortcut menu.

Note: To delete objects, expand the object type and then select the specific object, such as a named pick list, then right-click and from the shortcut menu, select **Delete**.

Adding new fields

You can add new fields to an entity type or link type in either the Database Explorer or the Detail windows by selecting the entity or link type, right-clicking and from the shortcut menu, selecting **New Field**.

You can also add new fields by:

- Copying a field from another entity or link type. For details, see Copying and pasting between objects.
- Adding a standard field, which will be applied to all entity and link types.

Copying and pasting between objects

If you have fields that contain the same information in multiple objects, you can save time by copying the information.

Procedure

You can copy fields, for example, between entity and link types, like this:

1. In the Database Explorer window, select the entity type or link type. This displays its fields in the Details window.
2. In the Details window, select the field that you want to copy, right-click and from the shortcut menu, select **Copy**.
3. In the Database Explorer window, select the object to which you want to add the field. For example, to paste a field into a link type, ensure the link type fields are displayed in the Details window.
4. In the Details window, right-click and from the shortcut menu, select **Paste**.

© N. Harris Computer Corporation

Copying and pasting between databases

If you have multiple databases, you might want to duplicate objects and fields. If both databases share the same security file you can copy information.

Procedure

You can copy both fields and objects between databases like this:

1. In the same iBase Designer window, open the databases that you want to copy between.
2. Select the entity type, link type or field that you want to copy, then right-click and select **Copy**.
3. In the other Database, select the correct object in the Database Explorer. For example:
   - To paste an entity type or standard field, ensure that Entity Types or Standard Fields are selected in the Database Explorer window.
   - To paste a field into an entity or link type, ensure the entity or link type fields are displayed in the Details window.
4. In the Details window, right-click and from the shortcut menu, select **Paste**.

   Note: You copy code lists between databases by importing. In the Database Explorer window, select **Code lists**, and then select either **Pick lists**, **Icon lists** or **SCC lists**. Right-click and select **Import**.

Creating a security file

In order to control access to your iBase database, you must create a security file. You use the security file to control who can access the iBase and iBase Designer applications, and any iBase databases secured by the security file.

About this task

The first step in designing the security for your iBase system is to create a new security file (`.ids` file). You always create the security file as an MS Access file but you can choose to convert it to SQL Server format later.

Try to avoid creating multiple security files. You can provide groups with varied but restricted access to many databases secured by the same security file.

To create your own security file, you must first create or choose a folder to hold it. Only one security file can be created in each folder, but you can create subfolders to hold security files if required.

A good strategy is to create a shared folder on a server computer with high availability to all likely users. All users of a database must have access to the folder to log on and open databases.

The default location for the database or the database connection file is the same folder as the security file.

Note:

- For Access databases, you must allow enough disk space to hold the database files (Access databases are limited to 2 GB).
- For SQL Server databases, the database files that are held in the folder with the security file are connection files that hold only enough administrative data to allow connection to the SQL Server database system. These connection files are much smaller than full databases, typically in the range 50 kilobytes through 1 MB. Full use of iBase facilities imposes some other requirements on the computer that is running SQL Server.

© N. Harris Computer Corporation

Procedure

To create a security file:

1. Start iBase Designer.
2. From the **File** menu, select **New Security File**. The Create New Security File dialog is displayed.
3. Browse to the folder you for your security file.

   Note: If you want the security file to be accessible from any computer on the network, you must use a UNC path. For example: `\\Server1\Databases\Fraud.ids`
4. In the **File name** field, enter a name for the file.
5. Click **Save**. iBase Designer creates the `ids` file and displays a message that says that you are logged on to the new security file as, for example, SYSADMIN (with the password SYSADMIN).

   Important: Do not use this user or this password for an operational database. Create another user with system administration rights, with a different name and password, then delete the SYSADMIN user. The path that is used to create the security file is displayed in the status area at the bottom of the iBase Designer window.
6. Click **OK**.

What to do next

Now that the security file is available, you can start the following tasks:

- Create groups and users. For more information, see Creating Groups and Adding Members.
- Change the administrative password. See Changing the administrative password.

After you create a security file, you must give authorized users access to it at the Windows™ level, and also protect it from accidental deletion.

You must also include it in any backup schedules for the database. For more information, see iBase backup policies.

You can move or copy a security file to a different server computer. For more information, see:

- Moving Access databases or security files
- Moving SQL Server databases or security files

Creating an example database

After you log on to a security file, you can create a new database.

About this task

Every iBase database contains several types of information, such as:

- Database management information

  Information on statistics and access control to the database. The statistics are held within the database. The separate security file can be used to make access control unique to one database or consistent across several databases.
- Entities and links

  iBase stores data that is organized as entity and link types; link types define the relationship between entities. Fields are the basic building blocks of the data, and their types vary in complexity.

© N. Harris Computer Corporation

Note: Each entity and link type has a separate table in the database. Entities and links are stored as records in those tables.

- Folder objects

  Folder objects support the use of the database, for example, browse definitions, labeling schemes, sets, and queries.

- Data sheets

  You can define data sheets to enter specific types of data.

Procedure

To create an example database:

1. Select **Create New Database** and click **OK**.
2. In the **Name** box, enter a name.
3. For **Database Type**, select **MS Access**.

   Choosing **MS Access** simplifies the initial setup of a database. However, you can create an SQL Server database. For an SQL Server database, you must specify how to connect to the database server. If you know these details, you can choose to create an SQL Server database now. If you do not know these details, create an MS Access database and then convert (upsize) it to SQL Server later.
4. Click **OK** to create an empty database with the name you have entered.

Results

iBase Designer creates a database file with extension `.idb` in the folder that contains the security file.

- For Microsoft™ Access databases, this file is the actual database.
- For SQL Server databases, this file is a connection file to the actual database that is managed by the server.

The configuration options for security files and databases are described in Configuration Options for an iBase System.

Access control

There are various internal features of an iBase database that can be used to apply levels of security. All features are optional and can be added as their usefulness becomes apparent.

Auditing is included here but it can have other uses, including review of analysis steps that are taken to reach a result. In SQL Server databases, auditing is also used with alerting to give users information on why an alert is raised.

The need for security

Security has several aspects, which can depend on legislative and operational circumstances.

You might need to apply security for several reasons, such as:

- Restricting access to sensitive data.
- Providing a record of how data was added, changed, viewed, or exported to other systems.

The benefits of an appropriate security policy and its implementation can include:

© N. Harris Computer Corporation

- Assurance that data is protected from deliberate or accidental change.
- Assurance that sensitive data is protected from inappropriate viewing or other use.
- Simple and appropriate working environments for staff in different functional areas or operational groups.

Applying security

The fundamentals of computer and database security apply to iBase. The first levels of protection are control of physical access and proper use of login identifiers and passwords. Full information about physical access control is outside the scope of this document, but you should not ignore the need for it.

User names and passwords

With iBase, you use user names and passwords to control access to databases, either database by database or in groups of databases (controlled by the same security file). You can choose to use Windows user names (single sign-on) or iBase user names.

All user names and passwords are stored in encrypted form, in security files, typically held in the same folder with the databases that they control.

After a user has access to a database, they might be subject to the following types of security:

- Database permissions, to read or alter data.
- Command access or denial, and usage monitoring.
- Data access, for viewing or changing records.
- Folder objects, to provide private storage of analysis methods.

Each of these types of security is applied through a dedicated type of security group, with a common method of defining their membership. See Types of Group for details.

Groups and their membership are held in the same security files as user identifiers and passwords.

Auditing

A final aspect of security is monitoring the use of databases, user access, and use of commands. iBase supports monitoring through audit logs and provides a dedicated application for viewing and analyzing logs.

Comparison of Access and SQL Server databases

You can use iBase with both SQL Server and Microsoft Access databases. This allows you to work with the scale of data appropriate to your analysis. iBase automatically recognizes the type of database and you can switch between them within an iBase session.

Access should only be used as the supporting database if the number of simultaneous users is five or less. When a database of more than 200 MB is accessed by a number of users simultaneously then consideration should be given to using SQL Server.

Using SQL Server increases the size of database that can be created and analyzed beyond the theoretical 2 GB limit of iBase using Access. However, there is no definitive maximum database size because this depends on factors such as: the nature of the data, the configuration of the server, the type of analysis, and acceptable response times. The most significant factor is the amount of memory on the server.

There are some extra capabilities when the database is stored in SQL Server format. These additional features are summarized below:

© N. Harris Computer Corporation

Search 360

Provides extra and more powerful features to Word Search, in particular the ability to search for words or phrases, allowing for typing errors, spelling mistakes, missing spaces, and so on. See Setting up Search 360.

Queries

SQL Server databases allow:

- Queries to be run that count the number of different entities linked to a specific entity (called distinct counts)
- The use of Any Link/Entity queries as source to other queries
- You to run more flexible queries that use semantic types

Alerting

SQL Server databases allow users to set up alert definitions to monitor items of interest in the database, such as single records or the results of queries, and receive alerts when any changes are detected. For details, see Configuring alerting.

Bulk import

SQL Server databases allow you to import large volumes of data more quickly than using the standard import mechanism. See Overview of Bulk Import for details.

XML import

SQL Server databases allow you to import from an XML data source when you work in iBase Designer.

Note: An XML import is a type of bulk import.

XML export

SQL Server databases allow users that work in iBase to export data as XML.

Note: An XML export is a type of database subset.

Use of Security Classification Codes

SQL Server databases allow you to classify each record with a security classification code so that access is restricted on a record by record basis. For details, see Using Security Classification Codes. This feature requires an Extended Access Control license.

Use of cases

In a SQL Server database, you can partition your database by case so that access to data is restricted on a case by case basis. See Creating case-controlled databases for details.

Audit level 5

This additional audit level allows you to log when entity and link records are accessed or viewed, without change to the data.

For example, it logs all records which have been in a human readable form in the session whether charted, printed, shown, and so on. This feature does not necessarily log all records that were requested. This keeps the audit file smaller and is a more accurate reflection of what the user may have seen.

Audit history

© N. Harris Computer Corporation

You can audit changes to the data in the database by selecting the **Audit History** option. This is available regardless of the audit level of the database.

Configuration options for an iBase system

There are various ways of configuring the iBase system. The differences between theses configurations are illustrated below.

There are two ways of holding security information:

- In a security file (in Microsoft Access format).
- In an SQL Server database, to which you connect at logon time by using a security connection file. The connection file contains only enough information to allow users access to the security database on the server.

Both files are `.ids` files and users see no difference in how they log on to them.

There are also two ways of holding the user and administrative data for the database:

- In an Microsoft Access database file.
- In an SQL Server database that is opened by using a connection file. The connection file contains only enough information to allow users access to the main database on the server.

Both the database and the database connection file are `.idb` files, and users see no difference in how they open them.

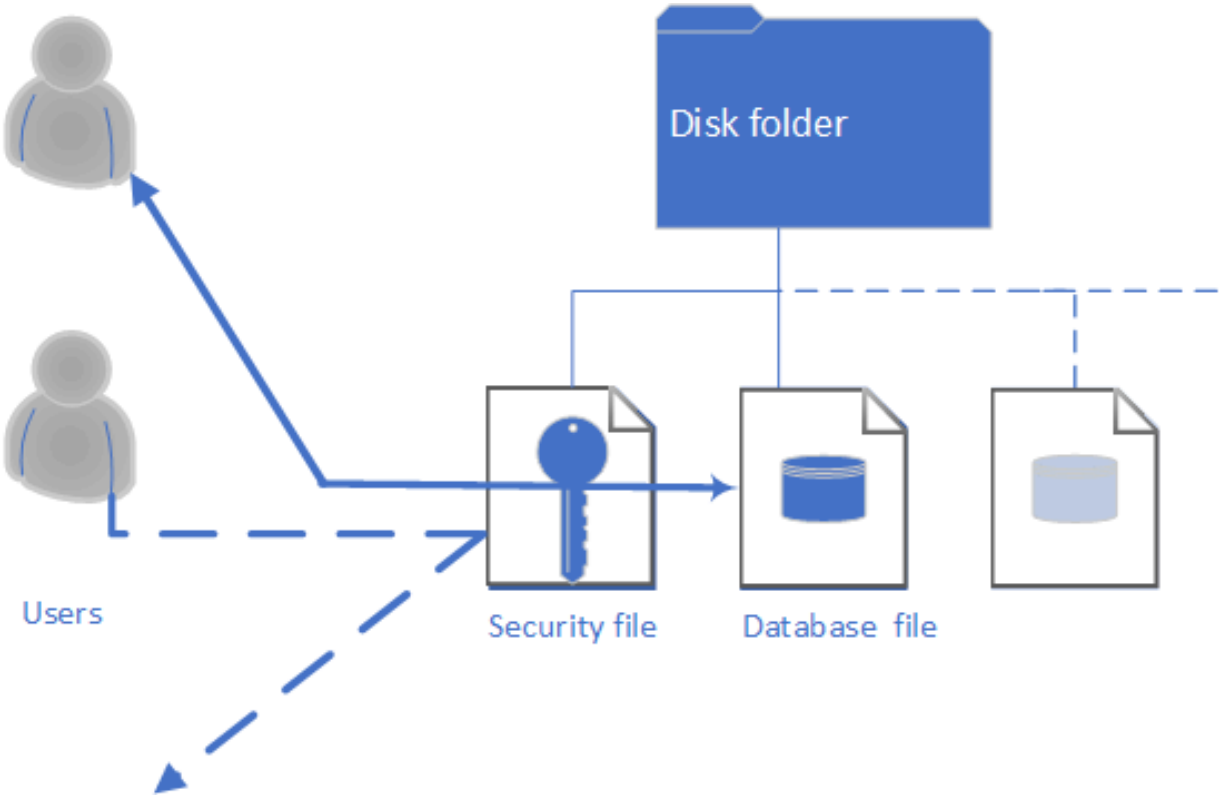
Note: The name for the database(s) on the server match, or partially match, the name chosen for the connection file, subject to naming conventions on the server.

This means that there are various ways of configuring the iBase system. The following sections illustrate the differences between these configurations.

© N. Harris Computer Corporation

Microsoft Access

The simplest possible arrangement is to have one security file and one database file held in a disk folder, typically shared to network users. The diagram shows two users: the upper one able to read and write data to the database, the lower one denied all access to the database.



The important points to remember are:

- Users gain access to databases through the security file, by logging on with an appropriate user name and password.
- A security file can control access to several databases.
- Each database is associated with just one security file.
- Each database records which security file is used to access it.

This configuration provides the lowest level of security. The Microsoft Access files are encrypted and password protected to resist intrusion from casual users, but might not be proof against attack from a determined technical person.

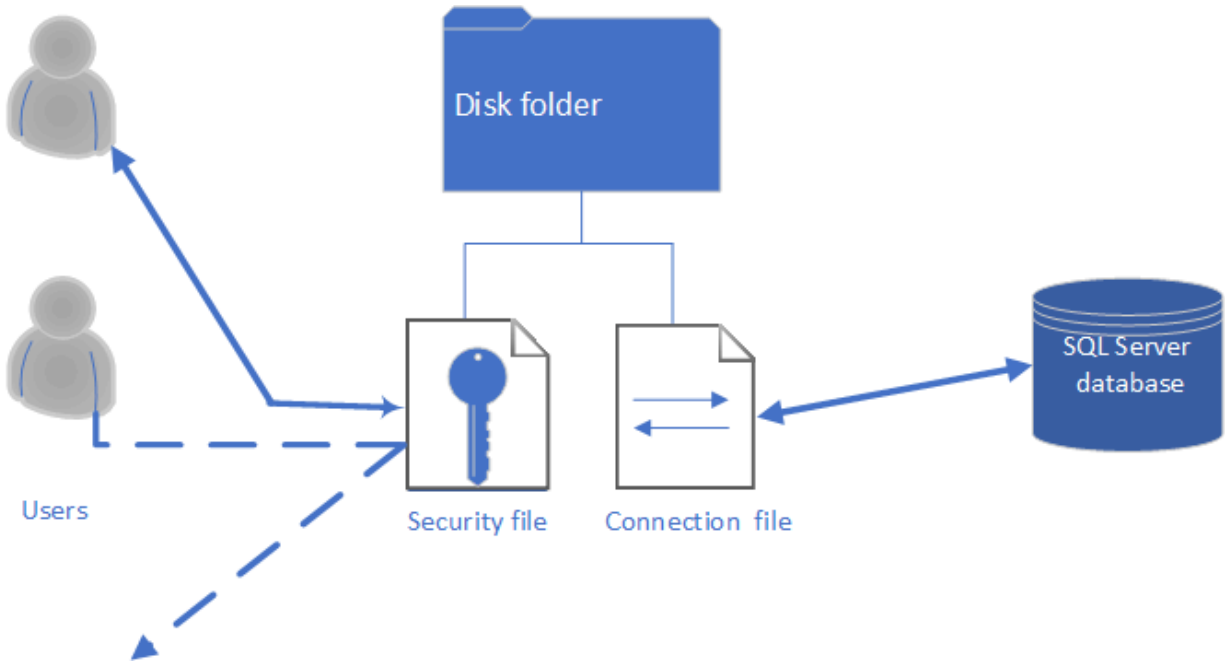
This simple configuration can be extended in several ways: by adding many databases, by adding more users, and by allowing those users different degrees of access to the databases ranging from full administration rights, through varying levels of ability to change or inspect data, down to no access at all.

It is also possible to create additional security files for other databases provided the security files are in separate folders.

© N. Harris Computer Corporation

SQL Server database

The simplest possible SQL Server arrangement is an extension of the one shown for an Microsoft Access database. There is still a security file, but there is now a connection file in place of the database file, and there is a server to hold the SQL Server database.



From the user's point of view nothing has changed, because they see a connection file that appears to be a database. The points to remember are still:

- Users gain access to databases through the security file.
- A security file can control access to several databases.
- Each database has just one security file associated with it.
- Additionally, users can make use of functionality specific to SQL Server databases.

For an administrator, there are differences:

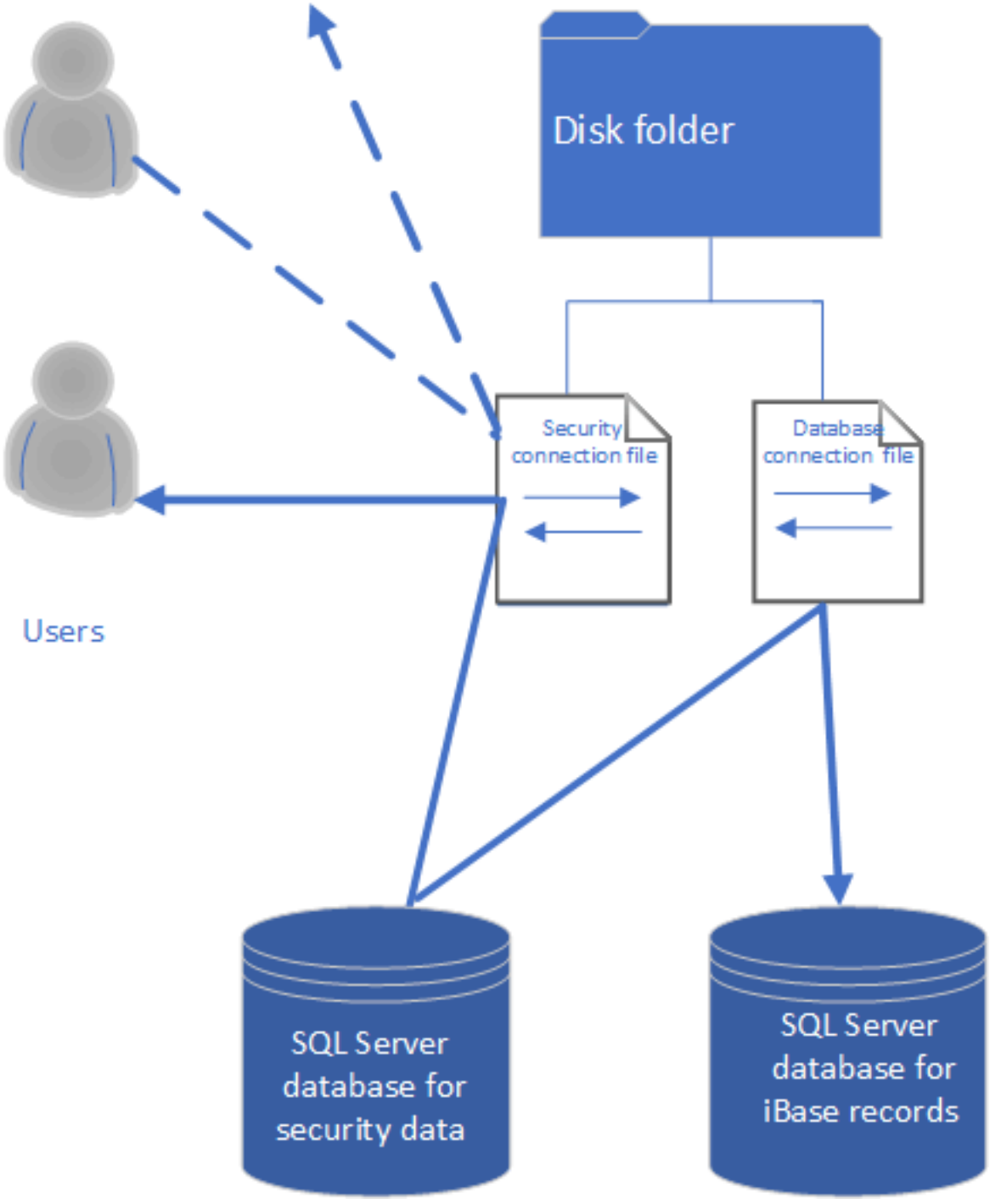
- There is a server running SQL Server, with consequent changes in performance and administration.
- There is an opportunity to centralize the operation and administration of multiple databases, and automate more of the routine administration.
- There is an opportunity to operate another level of security between the user and the data.

This configuration is more secure than Microsoft Access.

© N. Harris Computer Corporation

SQL Server database and security

In this option, there are now two connection files, one to replace the Microsoft Access security file and one to replace the Microsoft Access database. There is a server to hold the SQL Server databases that contain the security data and the main database.



© N. Harris Computer Corporation

Of the three options, this is the most secure configuration. However, from the user's point of view nothing has changed, because they see connection files that appear to be a security file and database. The points to remember are still:

- Users gain access to databases through the security file regardless of its file type.
- A security file can control access to several databases regardless of their file type.
- Each database has just one security file associated with it regardless of its file type.

For an administrator, there are differences:

- There is a server running SQL Server, with consequent changes in performance and administration.
- There is an opportunity to centralize the operation and administration of multiple databases, and automate more of the routine administration.
- There is an opportunity to operate another level of security between the user and the data.
- There is an opportunity to use replication to distribute copies of iBase data and database objects between servers and keep these synchronized.

This configuration provides the highest level of security.

Mixed iBase systems

The similarity in the user view of iBase operating with Microsoft Access and SQL Server database is intentional and real. You can operate iBase with a mixture of database types, even securing Access and SQL Server databases with the same security file.

When you want to migrate an Microsoft Access database or security file to SQL Server, by upsizing, you can do so in place, so that users need not see a difference until you choose to add features that are only available in SQL Server databases.

SQL Server clients, servers, and networks

You can run iBase on a system configured in a number of ways.

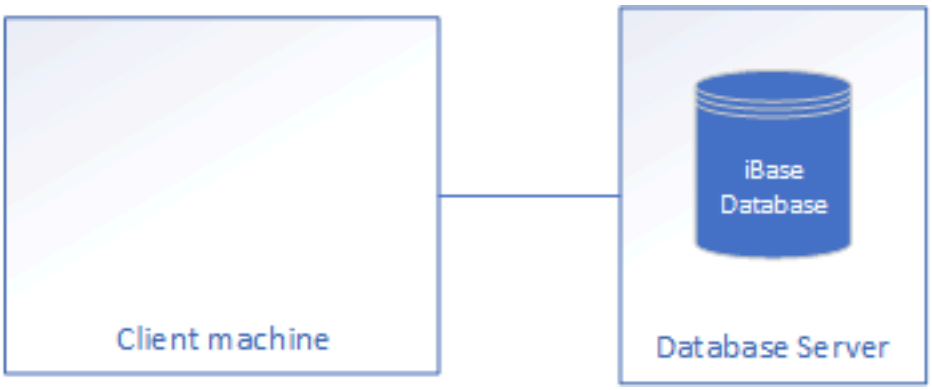
The system uses a combination of these elements:

- SQL Server database server for managing access to the databases
- iBase clients, or,
- An application server that is running iBase with thin clients, if you are using Terminal Services/Citrix

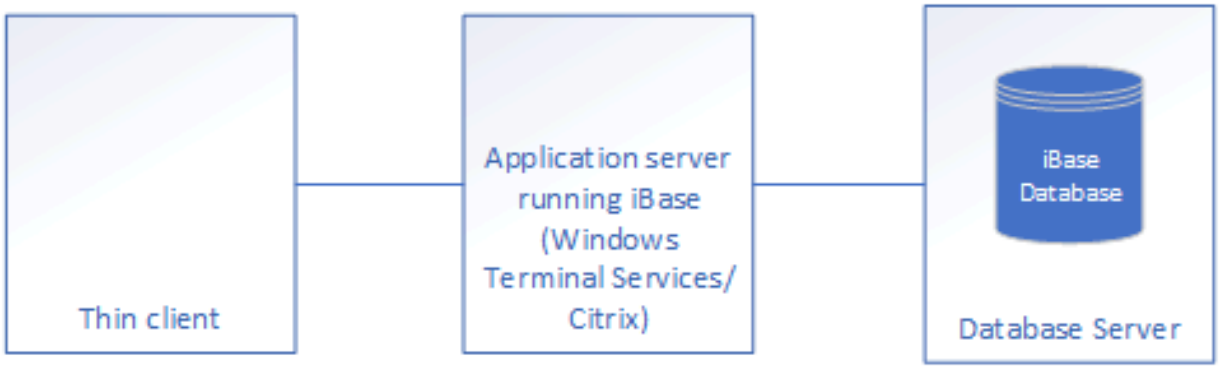
The following figures summarize the possible configuration options for iBase.

iBase Standard configuration option 1

© N. Harris Computer Corporation



iBase Standard configuration option 2



Hardware specifications and supported operating systems

Hardware specifications and the supported operating systems for a particular release of iBase are defined in its system requirements.

Client machines

The client should be sized to suit all the applications that it is intended to run alongside iBase. Mapping products in particular can place heavy demands on the host's processor and memory resources. The type of iBase usage that is expected on the client machine should also be taken into account: manual data entry places much lower stress on the client than analytical use or large data imports.

Server machines

iBase data is stored in database files that are managed by SQL Server. When your administrator first installs Microsoft™ SQL Server, they are prompted for the location of the program files and the data files. The default is that both sets of files are placed on the boot drive of the server. It is important to ensure that the data files are stored on the dedicated data partition. Typically this is a dedicated set of disks in a RAID 5 configuration. SQL Server stores the database files created by iBase in the default location for the database files.

© N. Harris Computer Corporation

We suggest that your SQL Server administrator use a RAID 1 configuration for the system disks and transaction logs, and RAID 5 for the data. The major activity in an iBase Standard installation is reading data and RAID 5 offers a performance advantage in reading. RAID 5 requires a minimum of three disks. The more disks used, the better the performance.

If the read auditing of activity is turned on, it is advantageous to place files for the iBase Audit Log database, both main data file and transaction log file, on a disk array with good write performance such as RAID 1. For maximum performance this should be on a separate disk controller.

Network requirements

Analysts use large amounts of data that must be transferred from the server to the client across the network. For example when starting up, finding, charting, and so on. A measure of the suitability of the network is latency: that is how long a packet of data takes to get from the server to the client and vice versa. Most local area networks should have low latency. Poor network performance leads to poor iBase performance when you browse, query, chart, map, and export to Data Miner.

Note: Deployment of iBase clients over a wide area network (WAN) is not supported. The architecture of iBase Standard requires relatively large volumes of network traffic. However, because the data flows in relatively small packets, the effect of latency, which is usually higher on WANs is more pronounced. The effect is that not only would client performance be slow and inconsistent but iBase would also disrupt other services that run over the WAN. As an alternative, i2 offers support for iBase WAN deployment using terminal services emulation.

The example user guide database

An example database that is called `User Guide.idb` is installed with iBase. This is a Microsoft Access database and demonstrates many of the features of iBase and iBase Designer. Typically each user needs a separate copy of the database if used for training purposes. There is a command to reset the database to its original state.

Some features are specific to SQL Server databases only and you might need to convert (upsize) each copy of the database to SQL Server before your users can use these features. Alternatively, users might be able to do this for themselves, using the Database Upsize wizard, if you provide them with details of how to connect to the SQL Server machine.

Before users upsize their copy of the database, they must to rename the `User Guide.idb` file so that it is unique on the SQL Server instance. For example, they might add their initials to the file name: `User Guide EB.idb`. They will also need to rename the other files associated with the database (with the suffixes `.dot`, `.doc`, `.ant`, and `.idx`).

See Upsizing a Database to SQL Server for further details.

Copying the User Guide database to a user application data area

The User Guide database is copied automatically to the user's application data area when they first open the database. This gives each user a separate copy of the database and ensures that the security file is in the correct location.

To copy the database and open it, each user selects from the Programs group on the Windows **Start** menu:

i2 iBase > **Documentation** > **iBase User Guide Database**

The first time that they open the database in this way, the `Examples` folder is copied to their application data area. `User Guide.idb` is opened from this location, and a shortcut to this location is added to the user's `My Documents` folder (or equivalent depending on the version of Microsoft Windows).

© N. Harris Computer Corporation

Standard user accounts for the User Guide database

These are the standard user accounts for the User Guide database:

User account	Password	Role
General	General	A user with all the permissions required for general use.
SYSADMIN	SYSADMIN	A full system administrator.
DataEntry	DataEntry	A Data Entry User with restricted menu functionality and access to fewer links.
Analyst	Analyst	An analytical user with read-only access.

Reverting to a clean copy of the User Guide database

At any point, a user can restore the database to an unmodified state by replacing the database in their application data area (see above) with the database as installed initially. If the database has been upsized to SQL Server, it restores the Access database but leaves the SQL Server database on the server.

To revert the database:

1. Select the following from the Programs group on the Windows `Start` menu:

   i2 iBase > **Documentation** > **iBase User Guide Database**
2. Click Yes when prompted to reset the database.

   Note: Reverting to an unmodified User Guide database means that you lose any changes that you made to the User Guide database. For example, you delete all entities, links, sets, queries, and other folder objects that you created or modified.

   Note: Any files that are created independently, such as export files remain.

Moving the User Guide database to a new location

If you move the User Guide database from the standard location, you need to open it in iBase Designer to reregister the location of the security file that controls it.

© N. Harris Computer Corporation

Managing i2 iBase

The Managing i2 iBase documentation provides information to help you configure and administer iBase.

Opening a database

If you are a database administrator or a security administrator, you can open a database, or create a new one, after logging on to a security file. You can have several databases open in the same session provided that the databases are associated with the same security file.

You can also choose to work without an open database, for example to work with the Security Manager or to check and repair databases.

There is an example database, the User Guide database. See About the example database for details.

Note: You will not be able to open the database if someone else has already opened it in any other i2 application. Examples of such applications are iBase, iBase Designer, and i2 Analyst's Notebook.

Listing records

You can list the records for a selected entity or link type so that you can review the data and take action as required. All the field values are shown, as well as the label for the entity or link type as defined in the default labeling scheme.

Procedure

1. In the left pane of the Database window, select an entity or link type.
2. Right-click and select **Records**.

   The records of that type are displayed. You can sort the records by the values in specific columns or change the order of the columns.
3. To change the number of records that are displayed, enter the number of records in the **Number of records to be displayed** box, and then click **Refresh**.

   Note: If there are a very large number of records, you may prefer to use a different method of viewing the records. For example, by using Find or Query in iBase.
4. To copy the information about records in the list, select the required records, and click **Copy to Clipboard**. You can then paste the records into another application, such as Microsoft Word or Microsoft Excel. Each record becomes a separate paragraph in a document or a separate row on a spreadsheet.

Menus and record lists

In record lists and from the icons that are used in records, you can work on the selected records by right-clicking and selecting an action from the menu. The available commands depend on the record list and the current selection.

Show, Show With, Show Records

There are three ways of opening and viewing a record:

- **Show** - displays the selected record, either in the show record view, or the default data sheet.
- **Show With**
  - Select **Show With Show Record** to display the selected entity.

© N. Harris Computer Corporation

  - Select **Show With `datasheet name`** to use the indicated data sheet.
- **Show Records** - lists two or more records so that you can compare the selected records and browse their field values.

Show History

Displays the audit history so you can view the changes to the current records and find out who made those changes.

Note: Only available in databases that are set up to use this feature. See your system administrator.

Links

View the links and the link end entities for the selected record.

Matching Records

Finds any records that match the selected record, and then displays them. iBase searches for matching records using the fields that are defined as discriminators in the entity type.

Note: It might take a while to retrieve and display the records. To pause the retrieval, press the `Esc` key.

iBase Link Chart

Shows the links and link end entities for the record in an iBase Link Chart.

Add to Set

Adds the selected records to a new or existing set.

Set Membership

Lists the sets to which the record belongs.

Create Report

Sets up the report wizard to create a report on the selected record. It uses the default report definition for the entity type selected (if there is one).

Create Link

Create links between two or more selected records. Only available when you select multiple entity records:

- With two entities selected, the entities are placed at either end of the link.
- With more than two entities selected, the entities are placed at the End 2 of the link, leaving you to specify the End 1 entity.

Chart

Add the selected records to an Analyst's Notebook chart:

- **Chart > Add to Chart** - create chart items for the selected records.
- **Chart > Expand** - create chart items for the selected records and then expands them as specified in the Charting Settings dialog to add associated records to the chart.

Add Alert

Monitor activity on a record or changes to the results of a query by adding an alert.

Note: Only available in databases that have been set up to use this feature. See your system administrator.

© N. Harris Computer Corporation

Properties

Shows the properties for the selected record. These include the record's system properties such as its creation date, the name of the user who created it, and its record identifier (unique record number).

About the example database

An example database called `User Guide.idb` is installed with the product. This can be used to understand the way that databases can be set up, allowing you to apply these concepts to your database.

Procedure

1. To open the User Guide database:
   a. From the Programs group on the Windows Start menu, select **i2 iBase** > **Documentation** > **iBase User Guide Database**. iBase is started and the Logon dialog is displayed.
   b. Enter your user name and password and click **OK**. Unless your database administrator has made a change and told you about it, these are the standard user IDs for the User Guide database:

User ID	Password	Role
general	general	A user with all the permissions required to perform general work.
SYSADMIN	SYSADMIN	A full system administrator
DataEntry	DataEntry	A Data Entry User with restricted menu functionality and access to fewer links
Analyst	Analyst	An Analytical user with read-only access

   c. Click **OK**.

When you use the User Guide Database for the first time, the database is automatically copied to your application data area. For example: `C:\Users\<your username>\AppData\Local\i2\i2 iBase\<language>\ Examples\User Guide Database`

This means that you can change the Microsoft Access database and the records as you want. This option is not available if you are using an SQL Server database.

At any point, you can restore the database to an unmodified state:

- From the Programs group on the Windows Start menu, select **i2 iBase** > **Documentation** > **iBase User Guide Database**, and click **Yes** when prompted to reset the database.

Note: Reverting to an unmodified User Guide database will mean that you will lose any changes that you made to the User Guide database. For example, you will delete all entities, links, sets, queries, and other folder objects that you created or modified. Any files created independently, such as export files will remain.

© N. Harris Computer Corporation

2. To upsize the User Guide database to SQL Server:
   a. Check that the server does not have an existing database called User Guide. If it does, rename the User Guide.idb file, for example to User Guide 2.idb. You will also need to rename the other files associated with the database (with the suffixes .dot, .doc, .ant and .idx).
   b. Copy the User Guide Database folder to a suitable place. It is located in the following folder (where n is the version number of the product): `C:\Program Files\i2 iBase n\Resources\<language>\ Examples\User Guide Database`
   c. Start iBase Designer, and then log on to the security file `User Guide.ids` as user SYSADMIN and cancel the option to open a database or create a new one.
   d. Select **Tools** > **Database Administration** > **Upsize** > **Database to SQL Server**.
   e. Accept the option to make a backup.
   f. Enter the name of the SQL Server machine and a login and password that has the dbcreator role on the server. Do not use the server name (local) since other clients will not be able to use the database. This server name is intended only for local use on the server computer. If the database name does not appear when you refresh the list, type in the machine name of the server.
   g. Click **Finish.** The database will be copied to the server using the name of the .idb file and the .idb file will become a connection file for the database.
   h. If you want your users to use Search 360, follow this additional step: in iBase Designer, use the option **Search Administration** on the **Tools** > **Search** menu to build a full index for all fields of all entities and links. An SQL Server administrator will also need to set up the index service for the database.

Security files, users, and groups

A security file controls who is allowed to log on to iBase and, after they have logged on, what they are allowed to do in both iBase and iBase Designer. Permissions are defined by creating user groups and assigning users to those groups.

In iBase, you use a security file to control who has access to an iBase database and the type of access they have. There are two formats for this file depending on the degree of security required.

For an introduction to these formats and to the different ways of configuring an iBase system, see Configuration Options for an iBase System.

Access to data and to parts of iBase itself is controlled by creating users and assigning them to user groups. Permissions are defined for user groups and inherited by users according to their membership of one or more user groups.

There are different types of group that control:

- Read/write access to records
- Commands on the iBase menus
- Access to entity types, link types, and fields
- Access to Security Classification codes (requires an Extended Access Control license and an SQL Server database)
- Access to named folder control objects; folder control objects include report definitions, import specifications, queries, and so on

The use of cases also determines how security works at your site. For example, in a database that is partitioned by case, a user only has access to the data in a case if they are assigned to the case, and

© N. Harris Computer Corporation

they are prevented from updating that data if the case is closed. For more information on cases, see Creating and Managing Cases in iBase.

An extra aspect to security is the physical security of the iBase installations and the network, and permissions to iBase files and folders.

Security files

The first step in designing the security for your iBase system is to create a new security file. You use the security file to control who can access the iBase and iBase Designer applications, and any iBase databases secured by the security file.

Note: The security files that are supplied with iBase are examples only. You can inspect the contents of these files, but you should create a new security file before you implement your own security scheme and creating databases.

Only users whose details are known to a security file can start an iBase session or open a database. When they log on to a security file, they can only open one of the databases secured by that security file. A security file secures access to any database that is created in a session started by logging on to the security file.

You should keep the security file in the same folder as the database files that it secures. You can move both the security file and its databases as required. For more information on moving files, see Moving and Copying Databases.

Creating a security file

In order to control access to your iBase database, you must create a security file. You use the security file to control who can access the iBase and iBase Designer applications, and any iBase databases secured by the security file.

About this task

The first step in designing the security for your iBase system is to create a new security file (`.ids` file). You always create the security file as an MS Access file but you can choose to convert it to SQL Server format later.

Try to avoid creating multiple security files. You can provide groups with varied but restricted access to many databases secured by the same security file.

To create your own security file, you must first create or choose a folder to hold it. Only one security file can be created in each folder, but you can create subfolders to hold security files if required.

A good strategy is to create a shared folder on a server computer with high availability to all likely users. All users of a database must have access to the folder to log on and open databases.

The default location for the database or the database connection file is the same folder as the security file.

Note:

- For Access databases, you must allow enough disk space to hold the database files (Access databases are limited to 2 GB).
- For SQL Server databases, the database files that are held in the folder with the security file are connection files that hold only enough administrative data to allow connection to the SQL Server database system. These connection files are much smaller than full databases, typically in the range

© N. Harris Computer Corporation

50 kilobytes through 1 MB. Full use of iBase facilities imposes some other requirements on the computer that is running SQL Server.

Procedure

To create a security file:

1. Start iBase Designer.
2. From the **File** menu, select **New Security File**. The Create New Security File dialog is displayed.
3. Browse to the folder you for your security file.

   Note: If you want the security file to be accessible from any computer on the network, you must use a UNC path. For example: `\\Server1\Databases\Fraud.ids`
4. In the **File name** field, enter a name for the file.
5. Click **Save**. iBase Designer creates the `ids` file and displays a message that says that you are logged on to the new security file as, for example, SYSADMIN (with the password SYSADMIN).

   Important: Do not use this user or this password for an operational database. Create another user with system administration rights, with a different name and password, then delete the SYSADMIN user. The path that is used to create the security file is displayed in the status area at the bottom of the iBase Designer window.
6. Click **OK**.

What to do next

Now that the security file is available, you can start the following tasks:

- Create groups and users. For more information, see Creating Groups and Adding Members.
- Change the administrative password. See Changing the administrative password.

After you create a security file, you must give authorized users access to it at the Windows™ level, and also protect it from accidental deletion.

You must also include it in any backup schedules for the database. For more information, see iBase backup policies.

You can move or copy a security file to a different server computer. For more information, see:

- Moving Access databases or security files
- Moving SQL Server databases or security files

Upsizing a Security File to SQL Server

In order to convert a Microsoft Access security file to SQL Server format, you need to upsize it. Upsizing creates an SQL Server database leaving the .ids file in the database folder as a connection file for the SQL Server database.

About this task

The SQL Server database will be added to SQL Server Enterprise Manager as <file>_ Sec, where <file> is derived from the name of the .ids file.

Once you have upsized the security file, you can rename the security connection file if required, and review its database properties: select **File Security** > **File Properties**.

© N. Harris Computer Corporation

You need to log on as a security administrator or as a system administrator to perform the following steps.

Note: Before starting the conversion, you need to discuss the server to use and the security mode with your SQL Server administrator.

Procedure

To convert the security file:

1. Log on using the Microsoft Access security file.
2. Click **Cancel** when you are prompted to open or create a database.
3. Select **Tools** > **Database Setup** > **Upsize** > **Security File to SQL Server**. The Upsize Security File dialog is displayed.

   Note: A backup of the Microsoft Access security file is made automatically, and you are informed when this is completed. The backup file has the file extension . ids.bak (or .ids.bak1 if there is an existing BAK file in the folder).
4. Enter the name of the SQL Server instance.

   Note: Only select the (local) option from the **Server** list if the database is for personal use.
5. Select the security mode. This will be Windows Authentication unless your SQL Server administrator directs otherwise.
6. If you are using iBase Database Replication, you must enter a site identifier in the **Identifier** box that is unique to the sites in your replicated iBase system. The identifier for the security file and its database will generally be the same.

   Note: Database identifiers are optional if you are not using iBase database replication.
7. Click **OK** to validate the settings and perform the upsize, and then click **OK** when it completes. The path of the security connection file will be displayed in the status area with (SSE) after the file name to indicate that it is SQL Server format.
8. Copy the backup of the Microsoft Access security file to a safe location. It is particularly important to keep this file if you are using i2 iBase Database Replication.

What to do next

When you have created the security file, you must protect it by making it read-only or by setting appropriate security permissions.

Viewing the Properties of the Security File

You can view the properties of a security file, whether in Microsoft Access or SQL Server format. To display the properties, select **File** > **Security File Properties**.

Microsoft Access security files

As a security administrator, you can view, and change, the properties of an Access security file.

Property	Description
Security File	The location of the Microsoft Access security file.

© N. Harris Computer Corporation

Property	Description
Identifier	A code that uniquely identifies the security file. The identifier is optional for Microsoft Access security files.

SQL Server security connection files

Property	Description
Security File	The location of the connection file for the iBase security database.
Database Type	The security data is held in an SQL Server database.
Database name	The name of the SQL Server database, which may be different to the name of the connection file that you use when you log on.
Server	The name of the server machine on which the security database is stored.
Login Name	The SQL Server login used when iBase connects to the SQL Server instance.
Password	The password is never displayed.
Use Windows Authentication	The mechanism used for validating attempts to connect to the SQL Server instance that holds the database. Windows authentication is used when the checkbox is turned on, and SQL Server authentication when the checkbox is turned off.
Identifier	A code that uniquely identifies the security database and is, typically, the same as the main iBase database. This is mandatory for SQL Server connection files in a replicated iBase system where the identifier should be unique across all the replicated databases.

Support for Unicode characters

Although iBase supports Unicode, the security file and database might not. Because security files are typically in Microsoft Access format, they will support Unicode characters in user names, passwords, group names and so on. However, these user names will not be supported in a non-Unicode enabled SQL Server database when saved as part of the entity and link records, for example in system fields such as Created By.

© N. Harris Computer Corporation

If the SQL Server database does not support Unicode then user names must not contain any Unicode characters.

Important: To avoid possible problems, you should convert all non-Unicode security files and databases.

Viewing security settings and properties

The security files that are supplied with iBase are examples only. You should inspect the contents of these files, but create a new security file before you implement your own security scheme and creating databases.

Viewing the properties of a security file

Any user can review the properties of the security file:

- In iBase Designer, select **File Security File Properties**.

The Security File Properties dialog displays the format of the security file and its location. If the security file is SQL Server format, it displays the connection details for the associated SQL Server database. See Creating an SQL Server Security File for further details.

Viewing the existing security settings

To view the existing settings in the security file:

1. In iBase Designer, log on to the security file for the relevant database. If your installation does not yet have a customized security file, you can view the one supplied with the iBase User Guide database. See About the User Guide Database.
2. Select **Security > Security Manager**. To display the permissions of a Database Management group, select the group and click **Edit**.

In an existing security file, other types of group might exist. To display their permissions, select one of these commands from the **Security** menu:

- **System Commands Access Control**
- **Data Access Control**

You can also produce a Security Design report:

1. Select **Security > Security Design report**.
2. Select the information to include in the report.

About the settings in a new security file

If you have a newly created a new security file, it has one administrative user called, SYSADMIN (with password SYSADMIN). This user has all database management permissions as a result of membership of the System Administrators group. The security file also contains a range of database management groups but no security policy.

Database management groups are necessary to all iBase databases. A user who does not belong to any database management group has only read-only permissions in the database. To grant a user additional permissions, you must add them as member of a database management group with the extra permissions they need.

There are three default database management groups. In the Security Manager, click the **Groups** tab to display these. You can display and edit the described permissions to match your own needs:

© N. Harris Computer Corporation

Group	**Members of the group...**
System Administrators	Have full database permissions.
Users	Able to create and modify data, and perform analysis by creating and saving sets, queries, and so on.
Guests	Cannot modify any data but can create and save analysis items.

The default security file also contains optional System Commands Access Control groups. The following groups are defined (but these are examples only, and can be added to, changed or deleted as required):

Group	**Members of the group can...**
Browse Users	Run queries, search text, and chart data to Analyst's Notebook. In SQL Server databases, they can add alert definitions and view the history of records.
Data Entry Commands	In SQL Server databases, they can add alert definitions and view the history of records.
Read Only Users	Perform basic and advanced analysis that involves searching, defining queries, sets, and scored matching. In SQL Server databases, they can add alert definitions and view the history of records.
Standard Users	Perform basic and advanced analysis, create records and define reports.
Super Users	Perform all iBase tasks (including import, export, and manage cases), apart from destructive operations on groups of records: batch edit, batch delete, merging entities, purge, and restore soft deleted records.

To view these groups, in iBase Designer, select **Security > System Commands Access Control**.

Logging on and changing your password

To complete any security work, you must log on to the security file for the relevant database or group of databases, with a user account with the Security Administrator role.

About this task

There are various ways of logging on in iBase Designer:From the menu, select

© N. Harris Computer Corporation

Procedure

1. Log on to iBase Designer
   - If you have a newly created a security file, you are already logged on.
   - Select **File** > **Open Database** and then browse for the database you want to open. If you are not using single sign-on, you are prompted to enter your iBase user name and password.

     You do not need to open a database, but opening a database can be the most convenient way to identify the security file for that database. You can close the database immediately after you log on.
2. Change the administrative password:
   a. Select **Security** > **Security Manager**
   b. Select the user, for example SYSADMIN, and click **Edit**.

      Note: Do not use the user SYSADMIN for an operational database. Create another user with system administration rights, with a different name and password and then delete the SYSADMIN user.
   c. Change the password as required. If you have just created a new security file, then the default password for a new SYSADMIN user is the same as the user name.

      Note: You can change your own permissions by adding additional group membership or permissions. The User Permissions dialog shows you the changes to database management permissions but, like any other user, you do not see the effect of these changes until you log off and log on again.

Creating an SQL Server security file

You always create security files in Microsoft™ Access format and then convert them to SQL Server. The main reason for converting to this format is to increase the security of your security data.

Before you begin

The conversion leaves a security connection file (the `.ids` file) in the iBase database folder and create an SQL Server database with the name `<file>_Sec` on the designated server (where `<file>` is the name of the `ids` file). Users connect to the SQL Server database by logging on to the connection file. The connection file holds just the information necessary to allow the user to connect to the SQL Server instance.

Note: If you copy security connection files to client machines, it might compromise the security of your system and adds to the administrative workload. You should keep the connection file, in the same folder as the database connection file, in a central location. If it becomes necessary to copy it, then the file name and path must be identical on each machine to which you copy it.

About this task

For SQL Server databases, the name of the security file is used to generate the name of the SQL Server database so you might want to discuss the naming convention to use with your SQL Server administrator and, if necessary, rename the security file before you upsize it. Although, you can always rename a security connection file, you cannot rename the associated SQL Server database. See SQL Server Database Names for details.

© N. Harris Computer Corporation

Procedure

To upsize a security file:

1. Using iBase Designer, log on as a security or system administrator using the Microsoft Access security file you want to convert.
2. Click **Cancel** at the prompt to create or open a database.
3. Select **Tools** > **Database Setup Upsize Security File to SQL Server**.
4. Click **OK** when you are informed that a backup has been made. This is a backup of the original Microsoft Access security file and has the file extension `.ids.bak` (appended with a number, such as `.ids.bak1`, if there is already a file with this extension in the folder).
5. In the Upsize Security File dialog, enter the server name. Do not select the **Local** option from the **Server** list.
6. Select the security mode as directed by your SQL Server administrator.
7. Skip the **Identifier** field - identifiers for security files are only used in iBase database replication.
8. Click **OK** to validate the settings and perform the upsize, then click **OK** when the upsize is complete. If you want to review the connection details and ID of the security connection file, select **File Security File Properties**. The path of the security connection file will also be displayed in the status area with (SSE) after the file name to indicate that it is SQL Server format.
9. Back up the connection file (`.ids` file). If you lose the connection file, you are not able to connect to the SQL Server database.

What to do next

You can protect the SQL Server security connection file by making it read-only or by setting appropriate security permissions.

This will allow any user in iBase Designer to view the properties of the connection file but prevents anyone, including iBase administrators, from changing the SQL Server connection details. This applies to SQL Server files only.

You should also ensure that the security connection file is included in any backup schedules for the database folder. For more information, see Backing Up iBase Databases.

Creating users and groups

The first step in designing security for your iBase is to identify the groups of administrators and users that you want to use. iBase is supplied with some default groups and an administrative user, all of which are created in any new security file.

You can add new groups and users, and modify or remove the supplied groups and users. You must modify the administrative user, if only to set a secure password in place of the default. If you do not use single sign-on, and before you add any users, you might need to define a security policy to control passwords and how users log on.

As an administrator, you need to create a user account for each individual who uses iBase, or access an iBase database from Analyst's Notebook®. This allows them to log on to the security file and open the database with the lowest possible level of access to the data. You define what they can do in iBase by setting up database management groups with specific permissions and assigning users to those groups.

Users gain the permissions that are accumulated from all database management groups of which they are a member. Further control is possible by creating other types of group.

© N. Harris Computer Corporation

Creating a security policy

A security policy sets restrictions on the user accounts that are set up to access iBase. The security policy specifies rules for adding and changing passwords that apply only to user accounts with iBase usernames and passwords.

About this task

New security files do not have a security policy because by default none of the settings on the **Security Policy** page of the Security Manager are turned on.

The absence of a security policy means that:

- Minimum password length is four-characters.
- No restriction on the characters that are used to make up passwords.
- Passwords never expire.
- No limit to the number of attempts to log on.
- Last used username is displayed at the next logon.
- No password history (although a new password cannot be the same as the current password).

Note: Although a security policy is part of the security file, it is not replicated even if you choose to replicate the security file. Enabling each site that is involved in iBase Database Replication to maintain their own security policy. However, the password history is replicated as it is possible that users might need to log on and change their account details at any of the sites.

Procedure

To view a security policy or change its settings:

1. In iBase Designer, Select **Security** > **Security Manager** > **Security Policy**.
2. Enter the requirements for new iBase passwords.
   - **Minimum password length**

     Enforce a minimum number of characters for the password, 1 - 20 characters.
   - **Minimum password age**

     Prevent the user from changing their password for a specified number of days.

     Note: This restriction can be overridden by turning on **Reset password at next logon**.
   - **Maximum password age**

     Force the user to change their password after a specified number of days has passed. By default, passwords never expire.
   - **Show password expiry reminder**

     Remind the user to change their password for a specified number of days before the expiry date.
   - **Enforce password history**

     Prevent the user from changing their password back to one used previously. The new password is compared to all previous passwords. Set the passwords remembered option to limit the number of passwords that are used in validating the new password.
   - **Lock out user after**

© N. Harris Computer Corporation

Control the number of times the user can enter an incorrect password before their account is disabled.

Note: You can unlock the account in the User settings by turning on **Account is active**.

- **Reset account lock-out after**

  Automatically unlock an account that has been disabled as a result of too many failed logon attempts.

  Note: Administrative accounts are automatically reset after thirty minutes.

- **Enforce complex passwords**

  Force the user to select a password of a suitable complexity.

- **Hide last username when logging on**

  Hide the name of the last user to use iBase. By default, last used username is displayed at the next logon.

- **Enforce SQL Server certificate validation for connections**

  This option is unchecked by default, and iBase connects without completing certificate validation. Check this option to enforce SQL Server certificate validation for users logging into iBase.

  Note: If you use this option you must configure the certificates on the SQL Server in order to connect to your SQL Server databases.

Note: The changes that you make do not affect existing passwords unless you require users to change their passwords when they next log-on.

3. Click **Apply** to save your changes. The changes come into effect when you log off.
4. If you are editing an existing policy, and change the password settings, select whether you want to force users to change their password when they next log-on.

Password settings

The password settings only apply to accounts with iBase user names and passwords. Some considerations are described below.

Note: Turning on a checkbox next to a password setting selects that setting. If you do not specify a setting, then the standard settings apply as described above. The standard settings are not the values displayed next to the checkboxes.

Password age

A user cannot change their password until it reaches the minimum password age, unless an administrator forces the user to reset it (for example, by turning on **Reset password at next logon**). This helps prevent a user from changing their password back to one previously used.

Password history

Specifies whether a password history is stored to prevent the reuse of old passwords. New passwords are validated against the user's complete password history unless you enter a specific number of previous passwords. If this option is not used, then the new password is only checked against the current password.

Account lock-out

© N. Harris Computer Corporation

You can choose to unlock the account automatically after a period of time, or you can unlock it manually using the User dialog in iBase Designer. Administrative user accounts are always reset after 30 minutes.

Note: Failed logons that result in an account lock-out are recorded in the audit log.

Creating groups and adding members

You set the permissions for all users by adding groups and defining the permissions for each group. Users acquire permissions by becoming a member of one or more groups.

Adding groups to a security file

Users gain the database management permissions that are accumulated from all database management groups of which they are a member. There is a similar combination of permissions or restrictions for the user's membership of each other type of group.

If you are adding many users, you might want to consider this work flow:

1. Identify the different types of user who will use iBase. Each type of user is represented by one or more Database Management groups, and possibly by other groups of different types depending on the complexity of your security arrangements.
2. Define the permissions for each of the required groups.
3. Define a template user to represent each user type and assign them to the correct groups.
4. Finally, create each user and assign permissions by copying the permissions of the appropriate template user.

Creating groups

You use groups to grant basic permissions to users. You may find that you want to create a group of each type for a particular purpose.

About this task

For example, you might create four groups with Analysts in the names and use these groups to define the basic database management permissions and, optionally, access to commands, access to data, and grouping of folder objects for analysts.

All groups have users as members. A particular user can be a member of any number of groups, of any type. The user gains the permissions that are defined for all the groups in which they are a member.

Procedure

1. Select **Security** > **Security Manager** > **Groups**, and expand the group type.
2. Click **New**.

Creating database management groups

Database Management groups grant permissions to users. For each Database Management group, you define the permissions of the group by turning on one or more checkboxes.

Users who become members of the group inherit these permissions:

Add Entity/Link Records

ON: Entity and link records can be added, either individually or by importing. Labeling schemes can be created.

© N. Harris Computer Corporation

OFF: Records cannot be created, members can still find, browse, and show the records in the database.

Update Entity/Link Records

ON: Records created by the user can be edited.

OFF: Records cannot be changed in any way. This includes batch editing, assigning new icons, and merging.

Delete Entity/Link Records

ON: Records created by the user can be deleted.

OFF: Records cannot be deleted, either individually or by using batch delete.

Update/Delete Entity/Link Records created by other users

ON: Members can edit and delete:

- Any record in the database that they can see
- Entries on pick lists
- Entries on icon lists

OFF: Members cannot edit or delete any of the items listed opposite.

Add Folder Objects

ON: Members can add or save, for example:

- Sets
- Queries
- Report definitions
- Import and export standard and batch specifications
- Charting schemes (providing they also have the Add Entity/Link Records permission)

OFF: Members can run queries, reports, and so on, either by using definitions created by others or by using new definitions of their own but they cannot save their own definitions.

Update Folder Objects

ON: For folder objects created by a member, a member can edit existing objects (see the above list). Members can also edit the contents of existing sets, including appending records to existing sets.

OFF: Once a member has added a new folder object, they cannot edit it.

Delete Folder Objects

ON: Members can delete folder objects that they have added.

OFF: Once a member has added a new folder object, they cannot delete it.

Update/Delete Restricted Folder Objects created by other users

ON: Members can update and delete restricted folder objects in the database. (**Note:** Do not grant this permission to non-administrative users if it is important to preserve folder object restrictions. See Folder Object Control Groups for details.)

© N. Harris Computer Corporation

OFF: Members cannot update or delete restricted folder objects created by others. Members can only update or delete the objects that they create.

Update/Delete Public Folder Objects created by other users

ON: Members can update and delete any public folder object in the database.

OFF: Members cannot update or delete public folder objects created by others. They can only update or delete folder objects that they create.

Database Creator, Database Administrator, Security Administrator

System roles that are only for administrative users. See Administrative Users for details.

Audit Administrator

This role allows members of the group to view restricted audit logs in Audit Viewer.

For additional details, see Database Management Groups.

Creating optional groups

To create a System Commands Access Control group, a Data Access Control group or a Folder Object Control group, you first create a group that has no defined permissions. Once the group exists, you can define the permissions.

Procedure

1. Select **Security** > **Security Manager** > **Groups**, and expand the group type, and click **New**.
2. Define the permissions for the group by using the following commands on the **Security** menu:
   - **Systems Command Access Control**
   - **Data Access Control**

   Note: There is no command for defining the permissions for Folder Object Control groups. The use of these groups is defined entirely by the members that belong to it, and are specific to the database. For more information, see Folder Object Control Groups for details.

Adding a user to several groups

You can use the Security Manager to add a user to multiple groups.

Procedure

To add a user to several groups:

1. Select **Security > Security Manager > Users**.
2. Right-click on the user, and select **Edit**.
3. On the Account page, make sure that **Account is active** is turned on.
4. On the Permissions page, turn on or off the required group checkboxes to define the user's membership of the groups.

   Note: You can deny access for this user by turning off the **Account is active** checkbox. An inactive user is unable to log on to the security file but the security file still holds a record of their password and group membership.
5. Click **OK** to save the details of the new user and then log off from the security file. The users will only be able to use their new permissions when you log off and when they next log on.

© N. Harris Computer Corporation

Adding several users to a group

If you add a group, you can add multiple users at the same time.

Procedure

To add several users to a group:

1. Select **Security** > **Security Manager** > **Groups**, and expand the group type.
2. Right-click on the group, and select **Edit**.
3. Click the **Users** tab to list the users defined in the security file.
4. Select each user you want to add to the group.
5. Click **OK**.

   Note: The users will only be able to use their new permissions when you log off and when they next log-on.

Reviewing the permissions defined in the security file

The permissions available to groups and users in a security file can be reviewed by users that have permission. You can also create reports that detail the information that you need.

About this task

You can see the permissions that are granted to a database management group or user by:

- On the Groups page of the Security Manager, expand the group type, select the required group, and click **Edit**.
- On the Users page, select the group and click **Edit** > **Show User Permissions.**. The database management permissions that are granted to the user by virtue of their membership of one or more groups are displayed.

Users do not see the effect of any changes to their permissions until they log off and log on again.

Procedure

You can also produce a Security Design report:

1. Select **Security menu** > **Security Design report**.
2. To include information, for example user information in the report, turn on the relevant box.

Creating users

If you have large numbers of users to create, you might find it easiest to add the security groups first, and then create template users to represent the different types of user in your organization so that you can copy their permissions.

Users can use their Windows credentials to automatically log on to iBase if the Windows account is domain-based and accessible through Active Directory.

A user is automatically logged on if their Windows credentials match an iBase account for either their Windows user name or the name of a Windows group to which they belong. In iBase Designer only, you can override single sign-on and log on as an iBase user by logging on using **Logon As**.

Note: Even if the iBase user account is for a Windows group, the audit log will always record the network details of the individual users, such as the computer names.

© N. Harris Computer Corporation

A user cannot log on automatically if they belong to two or more Windows groups and there is an iBase user account for each group. In this situation, the user is prompted to select the user to log on as. However, there is a **Remember my selection** option in the Logon dialog so that users do not need to repeat the selection each time. (Users can reset this option in the iBase Options dialog.)

Note: Security policies do not apply to this type of user.

Creating single sign-on user accounts

Single-sign on users use their Windows credentials to authenticate with iBase. Users that use single sign-on do not need to provide their details to access iBase.

Procedure

To add a user that will automatically log on with single sign-on:

1. Select **Security** > **Security Manager** > **New**.
2. Select **Windows User / Group**.
3. Enter the Windows user name, in any of these formats:
   - DisplayName (example: `FirstName LastName`)
   - ObjectName (example: `Group1`)
   - UserName (example: `Username1`)
   - ObjectName@DomainName (example: `Username1@Domain1`)
   - DomainName\ObjectName (example: `Domain1\Username1`)
4. Click **Check Name** to verify the name. The name is converted to the format `<domain name>\<account name>`. Successfully verified user names are displayed underlined.
5. Enter the basic details of the user account:

Option	Description
Account is active	By default, an account is active when created but you can suspend it by turning off the checkbox to prevent them from logging on.
Restricted Audit Log	This setting is applicable only if you use Audit Viewer. It is used to restrict other users from viewing the audit logs of the current user.
Account expires after	The user can log on up to and including the specified date.
Default Category	Enter the name of the default folder in which the user will save their folder objects. This is described in more detail in Default Categories for Users.

6. Define the permissions for the user by assigning the user to one or more groups on the Permissions page. For more information, see Assigning Users to Groups.

© N. Harris Computer Corporation

7. **Optional:** Enter contact details for the user, such as their email address, on the Information page. For more information, see Contact Details for Users. This topic also describes how contact information can be used in iBase.
8. Click **OK** to create the new user.

Creating iBase user accounts

iBase users have specific iBase accounts in addition to their Windows credentials. Users that use iBase user accounts need to provide their details to access iBase.

Procedure

To add a user who will log on by entering an iBase user name and password:

1. Select **Security** > **Security Manager** > **New**.
2. Enter a name up to 255 characters in length.

   Note: For more information, see Unicode support if the name you want to enter contains any Unicode characters.
3. Select **iBase Security**.
4. Enter and then confirm the password. The requirements that the password must meet are determined by the security policy.

   Note: In iBase, users can change their own passwords by selecting **Change Password** from the **File** menu. If required, you can remove this permission by defining System Commands Access Control Groups.
5. Enter the remaining details of the user account:

Option	Description
Account is active	By default, an account is active when created but you can suspend it by turning off the checkbox to prevent them from logging on.
User cannot change password	When this option is selected, the user will never be prompted to change their password regardless of the security policy and, in fact, will be unable to do so. Use this setting for accounts set up for services such as Scheduler.
Restricted Audit Log	This setting is applicable only if you use Audit Viewer or Audit History. The audit log generated by this user will have restricted access and only authorized users will be able to view it. Authorized users will have the Audit Administrator role.
Account expires after	The account will be accessible up to and including the specified date.
Reset password at next logon	When turned on, it forces the user to change their password when they next log on. The

© N. Harris Computer Corporation

Option	Description
	checkbox is then turned off after the user has done this.
Password never expires	Turn on to create an account that will not be affected by the security policy of this security file. Use this setting for accounts that will be used for services such as Scheduler.

6. In the **Default Category** box, enter the name of the default folder in which the user will save their folder objects. For more information, see Default Categories for Users
7. Define the permissions for the user by assigning the user to one or more groups on the Permissions page. For more information, see Assigning Users to Groups.
8. **Optional:** Enter contact details for the user, such as their email address, on the Information page. For more information, see Contact Details for Users. This topic also describes how this information can be used in iBase.
9. Click **OK** to create the new user.

Default categories for users

You can manage the folder objects that users create (such as queries and sets) by requiring them to save them in specific categories. Categories appear to users as folders, with similar behavior to Windows Explorer folders, and users navigate the folders in a similar way.

About this task

Categories should be organized systematically for each iBase database, for example, by departments, functions or projects. A category can contain several levels of subcategories. If required, each user can have their own personal category in which their folder objects are saved or it can be shared with other users. If there is no personal category for the user, and the user does not specify a category when saving, then a folder object is saved in the default category specified in the Options dialog.

There are two ways of handling what happens when a user clicks **Save**:

- Either, you set up iBase to prompt the user to specify a category for each new folder object that they save. The category defaults to their default category (or to the General category if the user does not have a default category).
- Or, the folder object is automatically saved in their default category (or in the General category if the user does not have a default category).

Users can move folder objects between categories but they cannot rename existing categories.

Folder objects comprise:

- Browse definitions
- Queries and scored matching definitions
- Report definitions
- Import specifications and import batch specifications
- Export specifications and export batch specifications
- Charting schemes

© N. Harris Computer Corporation

- Sets
- Mapping configurations (if iBase GIS Interfaces are in use)

There are different types of access for folder objects.

Procedure

To define the name of the default category for a user:

1. In the Security Manager in iBase Designer, select the user and click **Edit**.
2. In the User dialog, enter the name in the **Default Category** box. The default category is the value as specified in the Options dialog if you leave the **Default Category** box blank.

What to do next

You can edit the name at any point but this will not update existing folder objects in the database. To apply the new category name to existing folder objects, you need to ask the user to recategorize them.

Contact details for users

More details of the user can be recorded to provide contact information for other iBase users who need to establish the history of a particular record or modification, or for use by Alerting. This information is available in the Property dialog of each record, and also in the Show dialog and datasheets if owner hyperlink fields are added to the entity and link types. This feature is intended for users who have iBase user accounts rather than users who use single sign-on.

Before you begin

About this task

Contact information consists of a user's:

- Full name
- Location (which is predefined by the security or system administrator)
- Email address (used for email alerts, however, iBase cannot check whether the email address is valid)
- Telephone number
- Notes

Contact details are stored in the security file not in the database.

Note: It is simplest to ask your users to add their own details: in iBase, select **Change User Information** from the **File** menu.

You might want to assign each user a location, such as a site or department, as part of their contact details.

You can derive the location from the user name; however, the advantage of using the location field in the contact details is that the user can keep the same user name even if they move location. Typically the location is the physical location of the user rather than the location of the database.

You can structure the location name to facilitate wildcard search in the Audit Viewer.

© N. Harris Computer Corporation

Procedure

You must predefine the locations by adding contact information for a sample user from each location:

1. In the Security Manager in iBase Designer, select an existing user and click **Edit**.
2. Click the **Information** tab.
3. Enter the location name, up to a maximum of 50 characters.
4. Click **OK**.

   Note: As a security or system administrator, you can also add and modify location names in the User Information dialog in iBase.

What to do next

When you add a record or updating an existing one, you can:

- Make yourself the owner by typing $ (if it is not already displayed); your user name is inserted when you save the record.
- Select a different user as the owner:
  1. Click **Browse** next to the "owner" field to display the list of possible owners. If you know the first few characters of the name, enter these first - this will then scroll down to that position in the list.
  2. Double-click a name to select that person as the owner and close the dialog.

Users can find out who the owner of a record is when using either the Show dialog or a datasheet. There are two ways of displaying the contact details for the owner of the record:

- Click the user name shown in the owner, or a similarly named, field. The user name is displayed as a hyperlink.
- Click **Edit** and then double-click the user name.

The record owner might be a different person to the user who created or updated the record. To find out who these users were:

1. Right-click on the record in any record list, and from the menu, select `Properties` to display the Properties dialog.
2. Click the **User Information** button to display the contact details for the person who created or updated the record.

Assigning users to groups

You set up permissions by adding groups and defining the permissions for each group. Users then acquire permissions by becoming a member of one or more groups.

Before you begin

For example, users gain database management permissions by accumulating them from all database management groups of which they are a member. The same principle applies to the other types of group. By default, new users have the lowest level of access defined in the database.

To assign a user to one or more groups:

Procedure

1. Select **Security** > **Security Manager**.

© N. Harris Computer Corporation

2. On the **Users** page, select the user and click **Edit**.
3. Click the **Permissions** tab.
4. **Optional:** To assign a user to the same groups as another user, click **Copy Permissions** and then select an active user with the required permissions. The checkboxes display the updated group membership.
5. Change the user's group membership by turning on or off the checkbox for each required group:

-

  Database management

  Database management groups allow the user to add, modify, and delete records and folder objects or administer the database. You must assign the user to at least one database management group. To check which permissions are granted, click **Show User Permissions**.

  Note: A user who does not belong to any database management group has read-only access to the records in the database, and is unable to create any folder objects.

-

  System access control

  System Commands Access Control groups deny the user access to specific menu commands or audit their use of specific commands.

-

  Data access control

  The user is automatically made a member of all the Data Access Control groups defined in the database. This means that they automatically acquire the lowest level of permissions. Review these groups and, if required, turn off the checkboxes to grant the user additional permissions.

-

  Folder object control

  To allow the user to save folder objects in restricted categories, turn on the required Folder Object Control checkboxes.

  Note: At least one administrative user should have membership of all the groups of this type. This is required for a full view of the restrictions on folder objects and the ability to change each restriction individually.

6. Click **OK** to save your changes.

© N. Harris Computer Corporation

Administrative users

You can create members of the default system administrators group and grant all administrative powers to members of that group. Alternatively, you can create new groups that divide administration into separate roles.

This section describes the roles, how to give each role to a group, and the common tasks that the roles enable a user to perform.

Roles

When you are editing a database management group, the administrative roles that are shown in the System Roles area of the Permissions page of the Group dialog are:

- Database Creator
- Database Administrator
- Security Administrator

Each of these roles can be given to a group by turning on the relevant checkbox, along with any wanted data or folder object permissions.

There is another specific role, System Administrator. This role is given to a user only by membership of a group in which all system roles (apart from Audit Administrator) and database permissions are given. That is, the System Administrator requires all the permissions from the Security Manager dialog, not just those in the System Roles area. The System Administrator has a few powers that are not available to any other combination of roles and database permissions, so is more powerful than you could predict by adding the individual roles or permissions.

There are various other combinations of roles that you can assign by turning on the checkboxes for more than one role, but these combinations work exactly as predicted by combination of the individual roles.

A description of the capabilities of users with assigned system roles, starting with none and ending with the most powerful, System Administrator is given below.

User with no system roles

Without any system role, anyone with an iBase Designer license can:

- Log on to iBase Designer and log off.
- View their current user permissions.
- Create a security file.
- View the properties of the security file.
- Run these commands:
  - `Repair/Compact Security File`, as long as exclusive access is obtained to the file.
  - `MRU List Manager`
  - `Plug In Manager`, depending on permissions defined in System Commands Access Control groups and file-level security permissions on the `Settings.xml` file.
  - `Options`, modifying some of the settings in the Options dialog depends on file-level security permissions on the `Settings.xml` file.

Users without any system role cannot open the database in iBase Designer.

© N. Harris Computer Corporation

Audit Administrator role

A user with this permission can view the records that are displayed and modified by other users who are defined as having a restricted audit log.

Database Creator role

Users with only this system role are not able to administer the new database. Therefore, this role is most useful when databases are created from templates.

The Database Creator role is required to use iBase Designer to:

- Create templates from databases secured by the current security file.
- Create new databases.
- Manage templates using the Template Manager, depending on file level security permissions.

The Database Creator role is required to use iBase to:

- Create new databases.
- Create database subsets.

Note: The Audit Administrator role is also required to see the audit history of users with restricted audit logs, in iBase and in Audit Viewer.

Security Administrator role

Users with only this system role can open the database and view the database schema but they cannot change it or view any records in the database.

The Security Administrator role is required to use iBase Designer to:

- Move a security file.
- Modify the properties of the security file, such as the server details and security file identifier, in the Security File Properties dialog.
- Use the Security Manager.
- Produce a Security Design report.
- Upsize a security file to SQL Server.
- Create Database Design reports* and view database statistics*.
- Use Data Access Control*.
- Use System Commands Access Control.
- Open and close databases to allow the security administrator to perform the tasks marked with an asterisk (*) above.

This system role is also required to:

- View the audit log.

  Note: The Audit Administrator role is also required to see the audit history of users with restricted audit logs, in iBase and in Audit Viewer.

Database Administrator role

The Database Administrator role is required to use iBase Designer to:

- Open and close databases.

© N. Harris Computer Corporation

- View database properties and database statistics
- Upsize an iBase Access database to an SQL Server database.
- Check database and data integrity using Schema Integrity Check, Link Integrity Check and Valid End Types
- Work on the database schema, such as add entity types, fields, pick lists, and assign semantic types.
- Create Database Design reports
- Run `Update Database Schema`
- View records in iBase Designer
- Set up database functions (such as text search)
- Activate and run Bulk Import

Note: Extra permissions are required for exporting and importing data (including Bulk Import). Both export and import require Add Entity/Link Records, Update Entity/Link Records and Update/Delete Entity/Link Records. Importing also requires Update/Delete Entity/Link Records created by other users

In iBase, this system role allows a user to:

- Purge and restore soft deleted records
- Initialize databases for mapping or database subsets
- Synchronize database subsets

Note: The Audit Administrator role is also required to see the audit history of users with restricted audit logs, in iBase and in Audit Viewer.

Security Administrator and Database Administrator roles

Both the Security Administrator and Database Administrator roles are required to use iBase Designer to:

- Update specific database properties, such as the audit level SQL Server details, and turn on case control.
- Create, modify, and delete SCC lists.

Note: The Audit Administrator role is also required to see the audit history of users with restricted audit logs, in iBase and in Audit Viewer.

System Administrators

A user with all the system roles, apart from Audit Administrator, and all the database management permissions is referred to as the System Administrator. This user can use iBase Designer to:

- Import and export data.
- Obtain the database password from the Advanced page of the Options dialog.
- See statistics for all the cases in a case-controlled database (even if they select a single case).

In iBase, this user can:

- Edit existing text in a Multi-Line Text (Append Only) field.
- Delete alert definitions belonging to other users, in particular alert definitions without owners (for example belonging to deleted users or users removed from cases).
- Select any case in a case-controlled database.
- See statistics for all the cases in a case-controlled database (even if they select a single case).

© N. Harris Computer Corporation

Note: The Audit Administrator role is also required to see the audit history of users with restricted audit logs, in iBase and in Audit Viewer.

Editing and deleting users

Edit and delete users in the current security file by using the Security Manager.

You manage the users defined in the current security file by using the Security Manager and User dialogs. For example, you can:

- Change user passwords.
- Add and remove users from groups to control users' permissions.
- Prevent users from logging on.
- Delete unwanted users.
- Record personal details.

Editing a user

To edit the details of an existing user:

1. From the **Security** menu in iBase Designer, select `Security Manager`.
2. In the Security Manager, select the user and click **Edit**.

You can change any of the details; however the changes are not retrospective. For example, changing a user name does not update the user name shown in any existing records added by that user.

You can also change the user's permissions by changing their membership of the different groups. For example, turning on a checkbox grants the permissions of the group to the user. The user will only gain these permissions when you log off and they next log on.

You can also:

- Remove or restore the user's access by turning off or on the **Account is active** checkbox. An inactive user is unable to log on to a security file. Their alert definitions are set to expired and can be deleted by a system administrator; any alerts that are raised by those alert definitions are not deleted. Inactive users cannot be added as subscribers to new alert definitions.

For information on adding a user to a case, see Giving and Revoking Access to Cases.

Making a user inactive versus deleting a user

You can delete a user if required. However, after you have deleted the user, you are not able to do any of the following:

- Selectively restore or purge soft deleted records that belong to that user.
- View audit log activity, or the audit history, exclusively for that user.
- View or modify their alert definitions and any alerts raised by these definitions.

To delete a user:

- In the Security Manager, select a user and click **Delete**.

Note: Deleting a group does not delete the users within the group. It removes the group membership only.

© N. Harris Computer Corporation

Types of group

There are four types of group that can be used to control the access that users must data in an iBase database and to the iBase features.

Database Management groups

Database Management groups allow you to define groups with basic permissions to affect data records or folder objects (sets, queries, and so on), and entire databases or security files.

Database Management groups are defined in the Group dialog:

- From the **Security** menu in iBase Designer, select **Security Manager** and, on the Groups page, click **New**.

The Permissions page of the Group dialog divides permissions into these areas:

- Entity/Link Records
- Folder Objects
- System Roles

Entity/Link Records

In this area of the Group dialog, you can give the group members permission to manipulate entities and links. This applies only to records that they create. To allow group members to update or delete records that are created by other users, turn on the **Update/Delete Records created by other users** checkbox.

Typically, you want to give Add and Update permissions to all data entry staff. You might want to give Delete permission to all data entry staff, which enables them to remove records that they personally created, for example to correct mistaken or duplicate entries. In some cases, you might want to restrict both the **Delete** and **Update/Delete data Created by Other Users** permissions to supervisory or senior staff roles.

Folder Objects

In this area of the Group dialog, you can give the group members permission to manipulate folder objects (sets, queries, and so on). This applies only to folder objects that they create. To allow group members to update or delete folder objects belonging to other users, turn on these checkboxes:

- **Update/Delete Restricted Folder Objects created by other users** (for details of restricted folder objects, see Folder Object Control Groups)
- **Update/Delete Public Folder Objects created by other users**

Note: Even if you give a Database Management group every permission in the Entity/Link Records and Folder Objects area, you can still restrict what a user does by making that user a member of other types of group. For example, you can use System Commands Access Control groups to hide some or all of the commands that implement actions of a type enabled in the Database Management group.

System Roles

In this area of the Group dialog, you can give the group members one or more of the administrative roles, or grant them permission to view restricted audit logs. These roles are not modified in any way by the other types of iBase security groups. See Administrative Users for details of the administrative roles.

© N. Harris Computer Corporation

What users see

In general, users without a particular permission can start iBase and related applications but the affected commands in menus and shortcut menus appear as unavailable (dimmed or gray).

Some affected menu commands, mostly those linked to folder objects (sets, queries, and so on), remain available but, when selected, these commands display a dialog saying that the user has insufficient permissions to continue.

Note: In addition, you can define System Commands Access Control groups and deny commands to hide unavailable commands. See System Commands Access Control Groups.

In the Audit Viewer, users without the Security Administrator system role, see commands as available but are unable to open any log.

System Commands Access Control groups

System Commands Access Control Groups can be used to deny and hide specific iBase commands to users.

System Commands Access Control groups allow you to:

- Deny use of iBase commands that would otherwise be available to users because of their membership of one or more Database Management groups.
- Hide iBase commands and toolbar buttons that are not available because of a user's membership of one or more Database Management groups. Where it is not possible to hide these, a message is displayed `You do not have the necessary permissions to perform this action`.
- Record the user's reason for using a particular command.
- Log the use of the command in the audit log.

To display the System Commands Access Control dialog:

- Click From the **Security** menu in iBase Designer, select `System Commands Access Control`.

Existing security groups are listed in the left of the dialog. See Creating Groups and Adding Members if there no groups of this type defined in the security file.

Note: You can also deny use of iBase functionality to all the users of the local machine, rather than just to the members of a specific user group.

Access to basic menu commands in iBase

A user with full database management permissions (such as SYSADMIN) always has access to the following menu commands in iBase, even when they are denied access to all the system commands listed in the following section:

- Find, list, and show records
- Use iBase Link charts
- Create reports
- For links, view the valid end types
- Lists sets, add records to sets, and view set membership
- List labeling schemes and set a default labeling scheme
- Search for duplicate and matching records
- Examine their user details and the database properties

© N. Harris Computer Corporation

- Set session defaults and change the settings in the Options dialog
- Export data to Microsoft Excel using the Excel Interface
- Define folder objects as common folder objects (only of use when there is a Schema Update license)

Denying access to menu commands in iBase

iBase has several hundred commands including some with very similar names, which would make administration tricky and tedious if you had to make individual decisions for each command. To reduce this complexity, the commands are divided into groups.

Advanced Analysis

Denies access to Scored Matching, Field Calculator, starting Analyst's Notebook from iBase, sending data to Analyst's Notebook charts, and commands for Mapping Configurations and sending data to maps.

Alerting

SQL Server databases only: denies access to the commands in the Database Explorer for adding alert definitions. Users are still able to receive alerts.

Basic Analysis

Denies access to queries, combining sets and analyzing sets, and the Coordinate Query Builder.

Batch modification

Denies access to commands that affect batches of records: Merge Entities, Batch Edit and Batch Delete.

Charting

Denies access to all the commands on the shortcut menu in Analyst's Notebook that apply to existing records in an iBase database. For example: users cannot expand records, use the Timeline Wizard, find common neighbors, populate cards, expand records and so on. It also prevents a user from opening Analyst's Notebook while iBase is open. It does not restrict the use of iBase link charts.

Note: Users in Analyst's Notebook can continue to add new records to the iBase database, and add the records created during the session to sets but cannot expand them.

Charting Schemes

Removes or denies access to the commands for creating, editing, and saving charting schemes, as well to the commands on the shortcut menu for categorizing, listing and renaming them as folder objects.

Note: Users can still send data to Analyst's Notebook for charting and are prompted to select a charting scheme as usual.

Code lists

Removes the **Code Lists** command from the **Edit** menu so that users cannot change items on pick lists or icon lists.

Create Link/Entity

Removes the commands and toolbar buttons for adding new entity or link records whether using a standard dialog, a datasheet or Analyst's Notebook.

Database Statistics

© N. Harris Computer Corporation

Removes the commands for Database Statistics, Database Design Report, and Security Design Report.

Define Analysis

Users can chart existing queries but they cannot define new queries in iBase or Analyst's Notebook. Also, in iBase, they cannot open, categorize, list or rename queries, or use the Coordinate Query Builder.

Labeling Scheme

Users can still list the labeling schemes and select a default labeling scheme but they cannot add, delete, edit or rename labeling schemes, alter the contents of a labeling scheme or copy them.

Report Definitions

Users can still produce reports but they cannot add, edit, delete, categorize, list or rename report definitions.

Soft Delete

Removes the commands on the **Edit** menu for restoring and purging soft deleted records.

Tools

Removes the commands on the **Tools** menu in iBase for editing the MRU list and activating plug-ins.

View History

SQL Server databases only: prevents users from displaying the audit history both in iBase and in Audit Viewer. If alerting is used, it prevents users from displaying the alert details.

To deny access to the commands in a command group:

- In the System Commands Access Control dialog, select the group on the left and then turn on the required checkboxes on the Access Denied page to deny access to those commands.

You can inspect the detailed definitions of these groups by looking in a supplied, unsecured Access database, `CommandGroups.mdb`. This is in the application data area of your installation (see Installation and Application Data Folders for details). The command groups, their descriptions, and their definitions are in the _CommandGroup table.

Do not attempt to change these definitions, at least not without obtaining advice from your supplier. If you make changes to `CommandGroups.mdb`, then you need to apply it to the current security file by selecting **Database Setup** > **Update Command Groups** from the **Tools** menu.

Recording the reason for an action

You can require the user to enter a reason for using a particular command in iBase, or an iBase command when working in Analyst's Notebook. The reason is recorded in the audit log; however, the records affected by the command are only recorded if you set the audit level of the database to level 5.

Audit Analysis

Members of this group are prompted to enter a reason whenever they open a database or perform any analysis on iBase records, such as:

- Run a folder object such as a browse definition, report definition, query, import specification and so on
- Use any iBase command when the database is open in Analyst's Notebook

© N. Harris Computer Corporation

- Use any charting commands when in iBase
- Use any mapping commands when in iBase
- Use the Field Calculator dialog
- Copy data to the clipboard
- Export data using the Excel Interface dialog
- Use the Coordinate Query Builder

Audit Charting

Members of this group are only prompted to enter a reason when they work with iBase data on charts, specifically:

- Open Analyst's Notebook
- Use any iBase command when the database is open in Analyst's Notebook
- Use any charting commands when in iBase

Audit Data Exposure

Members of this group are prompted to enter a reason when they use any command in iBase that may result in data being printed (for example by exporting or reporting); or use iBase data in Analyst's Notebook, or i2 iBase Geographic Information System Interfaces.

Data Auditing: create, edit, delete

Members of this group must enter a reason for adding, editing, or deleting records before they can save the record. They are also prompted to do this when merging entities, batch editing and deleting, and assigning icons.

To prompt the user to record a reason for an action:

- In the System Commands Access Control dialog, select the group and then turn on the required checkbox on the Reason for Action page.

Auditing the commands used

You can record the commands used by a user in the audit log:

- In the System Commands Access Control dialog, select the group on the left and then turn on the required checkbox on the Audit page.

The three command groups are identical to the groups on the Reason for Action page. See above for details of the commands covered by each group.

What users see

Users do not see the commands that you have denied, so named menus (such as **File**) and shortcut menus become shorter, and some submenus might disappear entirely.

Note: Although some command groups deny commands for listing folder objects, users can still see which folder objects exist by using the Details window of the Database Explorer.

© N. Harris Computer Corporation

Data Access Control groups

Data Access Control (DAC) groups control permissions related to entity types, link types, fields, and records in each database. This allows a very fine control of how individual items of data are made visible to, or changed by, groups of users.

Among the possibilities are:

- Denying all access or change to all records for a particular entity type or link type. For example, you could deny access to all witness statements if those statements are stored in a particular entity type.
- Hiding administrative fields in records or making administrative fields read-only to certain groups of users.
- With SQL Server databases only, making selected records of various entity types or link types inaccessible according to the security classification (SC) code given to each record. For example, you could create DAC security groups called Public and Security Cleared, then use Data Access Control to deny the Public group access to records given an SCC level of Secret.

Note: New users are automatically made a member of all the existing Data Access Controls groups. This automatically gives them the lowest level of permissions defined for the database. You should review this default group membership whenever you add new users.

Note: After you make changes to a Data Access Control group in a database that uses alerting, log off and reopen the database as soon as possible, in either iBase or iBase Designer. This applies the security changes to any existing alert definitions.

Implications of using Data Access Control groups

You should consider carefully how you might want to use a scheme using conditional access.

A simple scheme with few combinations of access permissions is likely to work better than a scheme that implements many levels of restriction. Remember each combination of restricted data access has potential consequences for users by blocking access to queries using those restricted entities, links, or fields. This is true of all folder objects related to data items, not just queries.

You can start with a simple scheme and increase its complexity if needed.

What the user sees in iBase

The consequences of placing a user in a Data Access Control group can be wide ranging and can mean that different users see different databases.

Restriction	Details
Denied tables (entity types and link types) are entirely invisible to the user	This means that the user does not see the records for those types, and does not see even that the entity or link types exist. They are not able to run queries or reports for the denied entity or link type.
Read-only tables appear as normal, but do not have editing or creation options	For example a read-only entity does not have New command in the shortcut menu and the New and Edit buttons are unavailable in Show dialogs and data sheets for that entity.

© N. Harris Computer Corporation

Restriction	Details
Denied fields are entirely invisible to the user	For example, a denied field does not appear in Show dialogs, datasheets or when setting up browse definitions.
Read-only fields appear in data entry forms in the same form as equivalent system or calculated fields	For example, a read-only text field appears with a gray background even when other fields are editable.
Denied records (entities and links), denied because of an SC code, are made invisible on a record by record basis, so other entity and link records remain visible	For example, if there are some Crime entity records denied to a user, they may see other Crime records and they will always see that the Crime entity type exists, even if all records are denied to them.

Note: For details of further limitations that apply to matching records and merged entities, see Using Security Classification Codes.

The effects mentioned previously are direct and predictable. There are also effects that may seem less predictable, but are required to avoid users deducing what is hidden from them:

- All folder objects with references to denied tables become unavailable to the user. For example, users cannot see or use a query that refers to a denied entity or link.
- Users see data sheets, statistics, and design reports that match the entities and links that they see. Denied fields, entities, and links do not appear.
- If there are denied fields, users see a Show dialog without those fields. Users do not have access to a data sheet using a denied field.

What the administrator sees in iBase

You might see the following effects where users do things that are reasonable given their view of the database:

- Duplicate records that are created by users who have entities or links hidden by SCC restrictions.
- Users creating private queries to perform related tasks.
- Users in different Data Access Control groups see different results from performing the same analysis.

Creating Data Access Control groups

To create a Data Access Control group:

1. In iBase Designer, log on and open the database for which you want to set up Data Access Control. You do not have to open the database to create the group and add members to it but you do need to open it to define the permissions for the group.
2. Use the Security Manager dialog to create one or more Data Access Control groups, and make users into members of those groups. See Creating the optional types of group for details.
3. From the **Security** menu, select **Data Access Control**.

© N. Harris Computer Corporation

4. In the Data Access Control dialog, select the group and then define the group's access. You must repeat these steps in each database that is secured by the security file - the groups are defined in the security file but the permissions of each group are defined in the database.

   - **Tables**

     Lists the entity and link types in the database schema. To hide all records for a specific entity or link type, turn on the checkbox.

   - **Fields**

     Lists the fields of all the entity and link types in the database schema, including standard fields and mandatory fields. To hide a field in records of a specific entity or link type, turn on the appropriate checkbox.

     Note: You are warned if you deny access to a mandatory field (or if you make a denied field mandatory). If you choose to deny access to this field (or make a denied field mandatory), you prevent members of the group from adding records of the entity or link type.

     Note: Users who plot data on maps might need write access to the fields containing coordinate values if an iBase or Microsoft Access geocoding database is used.

   - **Read-Only Tables**

     On this page, records are visible to members of the selected security group but are protected from change.

   - **Read-Only Fields**

     On this page, data in the specified field is visible to members of the selected security group but are protected from change.

     Note: Users who plot data on maps need write access to the fields containing coordinate values if an iBase or Microsoft Access geocoding database is used, unless a more powerful user populates those fields for them.

   - **Security Classification Codes**

     The Security Classification Codes page lists the SC codes defined in the database schema. Turn on checkboxes for all classifications that you want to be denied to members of the selected security group. If any classification name appears in more than one SCC list, the denial of records applies to all records with that classification regardless of the list in which it appears.

     For further information, see Using Security Classification Codes.

     There is no Security Classification Codes page if you open an Access database or if the database is case controlled.

Where Data Access Control groups are stored

The relationship to database contents means that the full definition of a Data Access Control group is stored in two parts. The name and membership of each group is stored in the security file. The restrictions on members of each group are stored in the database, because it is in the database that the restrictions and their linkages to entities, links, and fields are stored.

If you create a new database from a template based on a database with DAC restrictions, the new database has no data access restrictions, but it does have access to the security groups in the relevant security file and any SCC lists in the template. This allows you to reproduce the security settings more easily than at first creation.

© N. Harris Computer Corporation

Folder Object Control groups

You can use Folder Object Control (FOC) groups to control access to queries, sets, and all other folder objects that require restricted access. When a member of a FOC group wants to save a report definition, for example, they can choose to restrict access to themselves and other members of the group. They can also choose to save it as a public or private object if they want.

The user's viewpoint

In understanding how this type of group works, it is important to consider the user's viewpoint:

- A user sees only the Folder Object Control groups in which they are a member. The user can see and change folder objects that are saved with restrictions based on those groups.
- If there are other FOC groups, the user does not see those groups and cannot directly change objects that are saved with restrictions based on those groups. Depending on database permissions, the user might be able to make the object public.

Here are some design and management ideas:

- Ensure that at least one administrative user has membership of all FOC groups. This access is required for a full view of the restrictions on folder objects and the ability to change each restriction individually.
- Do not give the permission **Update/Delete Restricted Folder Objects created by other users** to non-administrative users if it is important to preserve folder object restrictions. For more information, see Creating mandatory Database Management groups for details.

Managing Folder Object Control groups

In iBase Designer, you use the Security Manager to create Folder Object Control groups, and to declare appropriate users as members of those groups. See Creating the optional types of group for details. The members of the groups define how the groups are used.

You can also delete FOC groups. If you delete a FOC group, you must use iBase Designer to open each database that is secured by the security file before the change is fully applied. What happens in the databases is that the deleted FOC group is removed from all folder objects to which it has been applied. If that FOC group was the only group applied to an object, the object becomes public.

There is no other specific management activity.

Where Folder Object Control groups are stored

The relationship to database contents means that the full definition of a Folder Object Control group is stored in two parts. The name and membership of each group is stored in the security file. The restrictions on members of each group are stored in the database because it is in the database that the folder objects and their linkage to the groups are stored.

If you create a new database from a template based on a database with FOC restrictions, the new database has no folder object access restrictions, but it does have access to the security groups in the relevant security file and any folder objects in the template. This behavior allows you to reproduce the security settings more easily than at first creation.

Security Classification codes

With SQL Server databases only, you can choose to make selected entity or link records inaccessible according to the Security Classification (SC) code that is given to each record. Who can access records

© N. Harris Computer Corporation

with specific Security Classification codes is determined by membership of a Data Access Control group. Each group denies members of the group access to specific SC codes.

Requirements for using

To apply security on a record by record basis, and allow filtering by SC code:

- Each entity or link type in the database requires a field of type Security Classification Code (SCC). You can add only one Security Classification Code field per entity or link type.
- The classification levels that apply to entity and link records are defined as an SCC list, each classification level is represented as a Security Classification (SC) code. Multiple SCC lists can be added to the schema if different entity and link types use a different range of classification levels.
- The SCC list must be assigned to the SCC field in the relevant entity and link types.

Security classification codes and cases

If you want to use SC codes to classify your data, then you cannot also use cases. If you decide that you need to partition your data by case, then the conversion to case-control removes all the SC codes in your database. For more information about cases, see Creating a Case-Controlled Database.

Restricting SCC lists to accessible items only

By default, an SCC list displays all the SC codes on the list regardless of the current access to records classified with those SC codes. All the codes are provided to allow users to add a record and select an SC code for it that then denies that user access to the record they entered or updated.

Note: The user continues to have access to the record while it remains listed on their screen.

To prevent users from entering records with SC codes that represent security levels that are denied to them:

- Select **iBase Designer** > **File** > **Database Properties** > **Advanced** and turn on **Restrict SCC lists to accessible items only**.

SCC limitation in charting merged entities

When a chart contains denied entities or links, all the labels and data records for merged entities become available, including the labels and data records for denied entities and links. However, if the record is accessed in iBase, the message `This record has been deleted` is displayed.

Defining access to security classification codes

An SCC field on its own cannot enforce security. You need to add a Data Access Control group for each of the security classifications (in the Security Manager) and assign users to those groups. The group initially allows access to everything in the database; you must use **Data Access Control** to deny access.

Procedure

1. Create a Data Access Control group and assign users to this group. For more information, see Creating Users and Groups for details.
2. Select **Security** > **Data Access Control**.

   Note: This menu command requires a license for Extended Access Control.
3. Click **Security Classification Codes**.

© N. Harris Computer Corporation

Note: If the Security Classification Codes tab is not present, you cannot use SC codes, either because you are working in an Access database, or the database is case-controlled.

4. Select the group that you defined.
5. For the members of the selected group, turn on each Security Classification code that you want to deny.
6. You must repeat these steps in each database that is secured by the security file. The groups are defined in the security file but the permissions of each group are defined in the database.

Configuring and maintaining databases

You can create and maintain iBase databases, whether in Microsoft™ Access or SQL Server format.

Create databases

You can create different types of iBase database for operational use.

The different types of database include:

- Empty databases without any schema. You must define the schema, or copy and paste it from another database.
- Databases that use the schema of another database (the new database is based on a database template). You can add to the schema, or modify and delete the objects in it.
- Databases that partition their data by case. A database of this type must contain a minimum of one case and the only access to the data is through the case or cases to which the user is assigned. See What is case control?.

The database is either MS Access or SQL Server:

- Microsoft™ Access databases offer most of the database features of iBase and creating each database is simpler. However, they are only suitable for small amounts of data (up to 2 Gb) and small numbers of simultaneous users (up to 5 or 6). For more information, see Before Creating a Database.
- SQL Server databases are more suitable for large databases with large numbers of simultaneous users. They also provide additional features.

Before creating a database

Consider the following points, at least before you create your first operational database, and then review your decisions as you create other databases.

Should the database be SQL Server or Microsoft™ Access?

Microsoft Access databases offer most of the features of iBase and creating each database is simpler. However, over time and with growing size or numbers of databases, you might find that administration becomes difficult.

For personal use, and especially for use with a portable computer, an iBase Access database might be the best choice. You can always upsize any iBase Access database to SQL Server, allowing a straightforward transfer of all data and folder objects to the new format.

In general, the advantages of SQL Server databases make it the preferred choice:

- The advantages include the ability to work with larger databases, more users, better performance with large databases, and a higher level of data security with more flexible access control.

© N. Harris Computer Corporation

- You need to use one or more of the features specific to SQL Server databases. For a summary of these features, see Comparison of Access and SQL Server Databases.

The different combinations of Microsoft Access and SQL Server databases and security files are summarized in Configuration Options for an iBase System.

System requirements

All iBase installations can use Access databases. Multi-user sites need only a shared disk folder on a suitable server.

If you decide to use SQL Server, you need the following before you create an iBase database:

- SQL Server instance on a server or locally
- Suitable logins on that server

For more information about SQL Server logins, see Access control.

Identifying other database requirements

There are also some standard decisions to make for each database.

- Should the database have Federal Information Processing Standards (FIPS) compliance enabled?

  If you are working in environments that enforce FIPS compliance, you must enable FIPS when you create your database. FIPS enablement cannot be modified later because this changes the encryption algorithms, and existing users are prevented from logging in.

Before you create database records, you should consider the following questions:

- Do you want to identify records in this database uniquely when combined with records from other iBase databases?

  If so, you need to choose a text string, up to 5 characters long, that is unique to this database and that can be guaranteed to remain unique as new databases are created. (This is mandatory for replicated iBase databases.)
- Should the data be read-only to users?

  For example, this state might be appropriate if the database is used only for analysis of historical data collected from other databases. Database administrators can change this setting at any time, but you might prefer to make such a database read-only from the time of creation and change it to an editable state only when necessary for a specific task. Only the data is read-only, users (depending on their permissions) can still add, modify, and delete folder objects, such as queries.
- Should the database be partitioned by case?

  Do you want to restrict access to the records in the database on a case-by-case basis? If so, you need to create a case-controlled database. However, this setting cannot be changed at a later date. For details of how case control works, see What is case control?

With the exception of case control and FIPS, most other decisions can be made now or easily modified after the database has been in use for some time. For example:

- What level of auditing is appropriate?

  A low or intermediate level of detail is often a good starting point, because it is easy to modify settings for operational databases.
- Should audit logs contain a cross-reference for records from external data sources?

© N. Harris Computer Corporation

If you do not have this need, there is nothing to do now. If you want this functionality, the process is complex and extends across database design, configuration choices, and auditing.

With the answers and information that is prompted by these questions, you are ready to create the database.

Logging on to the correct security file

You must be logged on to the correct security file when you create the database. The new database can only be accessed through this security file. In iBase Designer, the name of the security file is displayed in the second area from the left of the Status Bar at the bottom of the application window.

Note: Each database shares a unique identifier with the security file used when you create the database. You can only use the database with this security file (or with a copy of the security file).

Database templates

You can create a new empty database from a template that is created from an existing iBase database. Creating databases in this way reduces the time that is taken to define core components.

Depending on the type of database, the template contains:

- Entity types, link types, fields, and standard fields
- Pick lists, icon lists, and SCC lists
- Datasheets
- Charting and labeling schemes
- Folder objects such as report definitions and queries
- Mapping configurations
- Common folder objects

The template does not contain anything that relies on the existence of specific records. For example, it does not contain:

- Sets
- Alert definitions
- Database subset definitions
- Data Access Control group permissions
- Cases (even if the source database is case-controlled)
- Support files, such as Analyst's Notebook templates
- Access permissions for folder objects (permissions are always set to Public unless you are using iBase database replication)

A template is saved with the file extension `idt`.

It is important to make sure that the template you select for use is up-to-date. It can sometimes be difficult to change the schema of a database that is in constant use or is off-site. A Schema Update utility is available to reduce the time that is taken to apply schema changes. For more information, see Updating Database Schemas.

© N. Harris Computer Corporation

Templates and database formats

You can create templates from both Microsoft™ Access and SQL Server format databases, and create a database of any format from that template.

However, a Microsoft Access database that is created from a template based on an SQL Server database does not contain any objects that rely on SQL Server. For example:

- Queries containing semantic conditions or distinct counts
- Import specifications and Import Batch specifications

Note: A template that is created from a case-controlled database is also case controlled. You can never create a Microsoft Access database from this type of template.

Where database templates are stored

Templates are stored in either the `Templates` or `WorkgroupTemplates` folder. By default the workgroup folder contains the database templates that are supplied by i2 and the `Templates` folder contains the ones that are created by the user locally.

Database templates are always created in the `Templates` folder. To distribute a database template for general use, you need to copy it to the `WorkgroupTemplates` folder. For more information, see Installation and Application Data Folders for details of paths.

Any user can change the path of their `Templates` folder.

Note: To prevent users from moving the templates folder, change the permissions for the `Settings.xml` file. See Location of Templates, Icons, and other Files for details.

Backing up database templates

Make sure that the folder in which you keep your database templates is included in any backups that are made of the iBase system.

Creating a database template

Database templates hold no data records but that do contain definitions of database objects to allow databases to be created quickly that match frequently used configurations. You can use any database that you can access to create a database template.

Procedure

To create a template from the database:

1. In iBase Designer, log on to the security file associated with the database but do not open the database.
2. Select **File** > **New Database Template**.
3. Use the file browser to locate and select your database. You can only create a template from a database associated with the current security file. The name of the security file is displayed in the second area from the left of the status bar at the bottom of the application window.
4. Review the entity and link types listed in the dialog to check that you have selected the correct database.
5. In the **Template Name** box, enter a name. You might also want to include the version number of the template in the file name. For example `Crime_v1_0`.
6. Click **OK** to create the template. Templates are always saved in your folder for user templates.

© N. Harris Computer Corporation

What to do next

You can now create a new database from this template or use it to update the schema of a copy of the database (from which you created the template).

Creating a Database

Databases can be created in both iBase User and iBase Designer.

Before you begin

Before you start to create a database, check that:

- You are logged on to a suitable security file (see Logging on to the correct security file for details).
- The security file is stored in the correct folder because iBase Designer creates the new database file (`.idb` file) in the same folder as the security file (`.ids` file).

When you are ready to create the database:

Procedure

1. In iBase Designer or iBase User, select **File > New Database**.
2. In the **Name** box, enter a unique name for the database. When you choose the name, consider:
   - Whether the name uniquely identifies the database, not only within your iBase system but also when the database is used with other iBase databases from other organizations, which is possible if maps and Analyst's Notebook® charts are created using data from multiple iBase databases.
   - For SQL Server databases, the name you choose is used to generate the name of the SQL Server database so you might want to discuss the naming convention to use with your SQL Server administrator. For more information, see SQL Server Database Names.
3. From the **Database Type** list, select the file type of the database you want to create:
   - **Microsoft Access**

     Creates a Microsoft™ Access database. Click the **Details** tab to continue.
   - **SQL Server**

     If you have a suitable server available, you can create an SQL Server database. To do this:

     a. For the **Database Type**, select **SQL Server**.

     b. Enter a **Server** name in the box to use a known server. Only select the **local** option, if available, if the database is for personal use.

     c. Choose how your computer connects to that server, using one of these options:
        - If your SQL Server database administrator has given you a login name and password for SQL Server, type these items in the **Login Name** and **Password** boxes. Each iBase user connects to the server using this login.
        - Turn on the **Use Windows Authentication** checkbox if you want to use integrated security, where SQL Server accepts the fact that a user has logged on to a Windows™ domain as sufficient permission to connect to the server. If you choose this option, the SQL server login entered above is never used, and each user that attempts to connect to use the iBase database is validated by the server using their network credentials.

© N. Harris Computer Corporation

The different methods of authenticating a connection are described in detail in Authenticating Connections to SQL Server.

d. Click the **Details** tab to display the Details page.

4. On the Details page, add a **Title** for your database. The title will appear in the title bar of the application window when the database is open in iBase.
5. **Optional:** Enter a **Description** of the database. You might want to enter a brief description that is seen by users opening the database. You might also want to record the name of its database template and the version number of the schema or template.
6. Set the **Audit Level** which you want to log changes.

   Level 1 means that the audit log collects the lowest level of detail, and level 5 the highest. (If you are creating an Microsoft Access database, the highest setting is 4.) Level 4 and higher collect large amounts of data about user activities so you should use these levels with care, and monitor the size of the log file as the database is used.

   You have now set enough properties to create a blank database.
7. Depending on your requirements:
   - Click **OK** to close the dialog and create the database now.
   - Specify a template for the database. For more information, see Creating a database from a template.
   - Set advanced properties for your database. For more information, see Setting advanced properties.

What to do next

The final step is to control the type of access allowed to the database folder and its files. By default, Windows users do not have sufficient permissions to log on and open the new database.

Creating a database from a template

Database templates contain standard components. Creating a database from a template reduces the time that is taken, and ensures that databases for a specific task are created consistently.

Procedure

To create a new database from a template:

1. Ensure that you are logged into iBase, but have no databases open.
2. Select **File > New Database**.
3. Click the **Template** tab.
4. Select a type from *Blank*, *User* (user created) or *Workgroup* (system created) templates. You have the option to create and delete templates from this tab. You can also click **View** if you want to see the entity types, link types, and fields in a selected template.
5. Click **Use Database** to create a database from an existing template. You can browse to any database accessible from the currently logged in security file. A temporary template is created and given a type of *Database*. This template is automatically deleted if you change to a different template, create a database using the template, or close the New Database dialog.

   Note: You can also create a template from a different database, and use that template instead. For more information, see Creating a template from an existing database.

© N. Harris Computer Corporation

6. Click the **Configuration** tab, and select the database type.
7. Click the **Details** tab, and enter the name of the database and some information about the purpose of the database or its contents.
8. Click the **Advanced** tab, and enter the details:
   - **Database Identifier**

     Optionally, enter a short string of text in the **Database Identifier** box. Do this if you want to identify entity and link records as belonging to this database. This database identifier is only necessary if you plan to perform operations outside iBase on records taken from different databases.

     Attention: The use of a database identifier has an impact on performance since the database identifier is appended to the record identifier on every record.
   - **Extra Detail Field for Audit Log**

     Type the name of a field (in this database) in the **Extra Detail Field for Audit Log** box if you want the audit log to record the value of this field when recording actions that affect records.
   - **Soft Delete**

     Turn on the **Soft Delete** checkbox if you want to use a two-stage process for deleting records. With **Soft Delete** turned off, all delete operations take place immediately. If the **Soft Delete** checkbox is turned on, all **Delete** commands mark records for deletion and make those records unavailable for most analysis, but do not delete the records.
   - **Read Only**

     Turn on the **Read Only** checkbox if you want to make the entire database read-only, and prevent any changes to records. Users can still create sets, queries, and other folder objects.
   - **Security Classification Codes / Case Control**

     Determines whether the database uses Standard Security Classifications or restricts information based on specific cases. If you select **Standard (SCC)**, you can additionally opt to **Restrict SCC lists to accessible items only**. Turn on this option to restrict any lists of Security Classification Codes to accessible ones only. This setting applies when you add or edit a record that includes an SCC list.
   - **First Day of Week**

     Displays the first day of the week as set for this database. This defaults to <System>, which is Sunday for Microsoft Access databases. For SQL databases, this is derived from the current locale as set on your machine or via the locale ID of the SQL Server machine.

     You should only need to change this if the locale on the SQL Server machine is different to your local machine or you are working with statistics and you want your week to start on a different day.

     Note: The start day of the week may affect calculations on dates and date parts.
9. Click **OK** to create the database with the settings you have chosen.

© N. Harris Computer Corporation

Setting advanced properties

When you create a database, you can set certain advanced properties. The following information describes the properties that need to be set before data is entered.

About this task

Database Identifier

If you want to identify entity and link records as belonging to this database, enter a short string of text (up to five characters). This is appended to the identifier of each new record; for example, `PER475\GEN` where `GEN` is the database identifier. This identifier is only necessary if you plan to perform operations outside iBase on records taken from different databases or if you use iBase database replication.

Note: If you intend to use this feature, you must enter this string before you create any records in the database. The field remains editable after database creation and the addition of records, but any change you make will mean that records created before and after the change will have different database identifiers.

Attention: The use of a database identifier has an impact on performance since the database identifier is appended to the record identifier of every record.

Extra Detail Field for Audit Log

If you want the audit log to record the value of a particular field when recording actions that affect individual records, enter the name of a field (from this iBase database).

The audit log always records the iBase Record ID so this extra recorded field is a free choice from data entered in iBase or imported from another database. Typically, the database designer sets up the schema so that the named field or standard field contains an external reference number or some other way of assessing the history or validity of a record.

For example, this feature can be used to maintain an audit log with details of data and record identifiers imported from external databases.

Standard (SCC) Control

Leave the **Standard (SCC) Control** option selected. This gives each user access to all the records in the database, depending on their user permissions. For details of creating a database that is partitioned by case, see Creating a Case-Controlled Database.

Other properties can be set, or changed with caution, at any time during the life of the database. For more information about all of these properties, see Summary of the Database Properties.

Procedure

To set advanced database properties:

1. Click the **Advanced** tab to display the Advanced page.
2. Select the properties that you would like to use.
3. Click **OK** to create the database with the settings you have made.

What to do next

For any open database, you can view the properties by displaying the Database Properties dialog. When viewed in iBase Designer, you can change the Title, Description and, with caution, the settings displayed on the Advanced page.

© N. Harris Computer Corporation

Summary of the database properties

The properties of the database provide detailed information about the database.

At any time you can view the properties of the database in iBase Designer, by selecting **File** > **Database Properties**.

Title

The title for the database, as displayed in the application title bar.

Description

The description of the database, as displayed when any user first opens the database.

File

The location of the database (`.idb`) file.

Version

The database version number.

Audit Level

The detail level at which the audit log collects data on changes to the database and security file. You can change the audit level: level 1 means that the audit log collects the lowest amount of detail and level 5 collects the highest amount of detail (SQL Server databases only). Level 4 and higher collect large amounts of data about user activities so you should use these levels with care, and monitor the size of the log file.

Audit History

In SQL Server databases only, all updates to data, including code lists, are logged and can be viewed either in Audit Viewer or in the iBase History. In a database that is set to audit level 5, records that are viewed but not updated are also logged.

Note: This property is automatically turned on if the database is initialized for alerting and cannot be turned off while alerting is in use.

Database Logo

You can replace the default database logo with an alternative image. Click **Select...** to browse the available supported file types in your Windows Pictures folder. Select an image no larger than 0.5MB and click **OK**. To remove the image from the database, click **Clear** and **Yes** to confirm. Your image is replaced with the default iBase icon.

Configuration

The configuration page shows details of the database file and format, and the security mode. You can change the authentication mode when connecting to the SQL Server instance on this page or by using the Database Configuration tool (see Managing SQL Server Connection Settings).

Database Type

The file format of the database, either Microsoft Access or SQL Server.

Database Name

The name of the SQL Server database on the server. See SQL Server Database Names.

Note: You cannot rename an SQL Server database in iBase Designer. See SQL Server Database Names for further details.

© N. Harris Computer Corporation

Server

The name of the database server. You can change to a different server provided that the database exists on that server. Enter a name in the field to use a known server. Only select the **(local)** option if the database is for personal use.

Note: This and the following changes do not take effect until you reopen the database.

Login Name, Password

An SQL Server login name and password is displayed if SQL Server authentication is used to secure access to the SQL Server instance. See Authenticating Connections to SQL Server for details.

For security reasons, the login that is used to create the database might be different from the one used after creation. After creation, you might prefer to change the login to one with a lower level of SQL Server permissions. After creation, you might also want to change the authentication mode by turning on the **Use Windows Authentication** checkbox.

Use Windows Authentication

The **Use Windows Authentication** checkbox is turned on if Windows authentication (integrated security) is used to secure access to the SQL Server. Each user that attempts to connect to use the iBase database is validated by the server using their network credentials. See Authenticating Connections to SQL Server for details.

Advanced

The Advanced page displays the current setup of the database, which you can change with caution. See Advanced settings for details.

Database Access Token

Use the Database Access Token page to obtain a copy of the token for the current database. The token is used by some applications, such as the Search 360 Indexer, to connect to SQL Server databases. You can save time by copying the token from this page and pasting it into the application, rather than having to regenerate the token in the Database Configuration tool.

Passwords for Microsoft Access databases

A 20-character password is generated for you when the Microsoft Access database is created. You should keep a record of this password. The password is the same for all the Microsoft Access databases created from the same security file.

If you are using iBase Designer and want to see the password, select **Tools** > **Feature Availability** > **Options** > **Advanced**. This option is not available in iBase User.

Advanced database settings

In addition to the details that must be entered to identify and secure your database, you can set additional options that determine how your database operates. To view the advanced settings, select **Database Properties** > **Advanced**.

Setting a database identifier

© N. Harris Computer Corporation

A database identifier is a code of up to 5 characters in length that is appended to the identifier of each record in the database. This identifier can be used to track the origin of records when they are exported into other systems.

About this task

The database identifier is only necessary if you plan to interact with records taken from different databases. If you are using iBase database replication, a database identifier is mandatory.

Important: The use of a database identifier has an impact on performance since the database identifier is appended to the record identifier on every record.

Procedure

To set a database identifier:

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, enter your **Database Identifier**.

   Note: The database identifier can be up to 5 characters in length and must be alphanumeric.
3. Select **OK**.

Display database details

You can choose whether or not to display information about your iBase database when you log on.

About this task

iBase database details are displayed by default when you log on to iBase User, or from Analyst's Notebook and Analyst's Notebook Premium. You can change this setting in **Database Properties**.

Procedure

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, deselect the checkbox next to **Display database details after log on**. You can reinstate the setting at any time.
3. Click **OK**.

Adding extra details for auditing

When records are being audited, you can specify a field that is included in all audit entries. You can use this field to include extra information.

About this task

When the audit level is set high enough to log information about item creation, modification, and deletion, iBase searches for a field on the item with a name that matches the text in the **Extra Detail Field for Audit Log** field. If the field is available, the value from the field is entered into the 'Extra detail' column in the audit log.

For example, if you set the free-text field to AKA, then create a person who is called `Robert`, with the **AKA** field is set to `Bob`. The audit entry for that item creation has `Bob` as the **Extra Detail** value. If you then edit `Robert` to be called `Jonathan` and as part of that same edit, change **AKA** to `Jon`. The audit entry for the modification has `Jon` as the **Extra Detail** value.

Procedure

To set a field to be used for extra details:

© N. Harris Computer Corporation

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, enter the name of the field to use **Extra Detail Field for Audit Log**.
3. Select **OK**.

Enabling soft delete

Deleting records is a permanent and irreversible operation unless soft delete is enabled for your database. When soft delete is enabled, deleted records, whether deleted individually or in batches, are removed from the user's view of the database but you have an opportunity to undo the deletion.

About this task

Soft deleted records do not appear in search results or in record lists (when listing and browsing records). A soft deleted record is a record that is marked in the database to prevent it appearing in general use, but without removing the information completely.

If you have system administrator rights:

- You can view the soft deleted records that are not purged.
- You can restore soft deleted records.
- You can permanently remove soft deleted records.

If soft delete is not enabled, then deleting records is a permanent and irreversible operation.

Note: If records are currently soft deleted, you are unable to disable soft delete on your database without resolving these records.

Procedure

To enable soft delete:

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, select **Soft delete**.

Setting the database to read only

A database can be set to read only to prevent changes to the records that it contains. When a database is set to read-only, sets, queries, and other folder objects can still be created.

About this task

You might want to set a database to read only for a number of reasons, for example:

- An investigation might be complete
- The records might be imported from a different data source that is regularly updated

Procedure

To set the database to read only:

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, select **Read Only**.
3. Click **OK**.

Enabling alerting

© N. Harris Computer Corporation

When you enable alerting, users can create alert definitions to monitor records and Query results.

About this task

Alerting is only available for SQL Server databases. For more information, see Configuring alerting.

Procedure

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, select the checkbox next to **Alerting**, or deselect the checkbox to prevent alerting.
3. Click **OK**.

Turning on Security Classification Codes control

Standard (SCC) security restricts record access based on membership of specific user groups. By restricting access on a need to know basis, restricted data can be accessed by authorized users but not generally available.

About this task

Security Classification Codes must be set up before you enable SCC control.

Note: If you would like to restrict a set of records for access by the same group of users, the alternative to Standard (SCC) control is to enable Case Control.

Procedure

To turn on Security Classification Code control:

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, enter your **Restrict SCC Lists to Accessible Items Only**.
3. Select **OK**.

Turning on Case control

Case control is used to partition the records in your database into a number of cases. Partitioning your database allows you to provide groups of users with access to the records in particular cases.

About this task

For more information about case control, see Working with cases.

Procedure

To turn on case control:

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, select the **Case Control** option.

   Note: Case control is only supported on SQL Server, and the options are available if database replication is not installed.

Setting the first day of the week

© N. Harris Computer Corporation

The first day of the week is used in calculations on dates and date parts. A default first day of the week is taken from your system, but can be set to match your needs.

About this task

The first day of the week defaults to **<System>**:

- For Microsoft™ Access databases the first day of the week is Sunday
- For SQL databases, the first day of the week is derived from the current locale as set on your machine or using the locale ID of the SQL Server machine

This setting only needs changing if the locales on the SQL Server machine and your local machine differ, or you want your week to start on a different day.

Procedure

To change the first day of the week:

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, select the **First Day of Week**.
3. Select **OK**.

Global setting validation in a replicated environment

To prevent issues, certain database settings must be used consistently in environments that use either database replication, or database subsetting. When you are working in replicated environments, you are notified if these settings are not consistent, and must resolve the issue before the database can be used.

About this task

Audit level

The amount of information that is stored about user interactions with the data that is stored in your system. As different audit levels record different amounts of information, it is impossible to synchronize data in environments where the audit levels differ.

Audit history

The types of interaction that are recorded in the audit logs. As different audit history settings record different types of information, it is impossible to synchronize data in environments where the audit history is different.

Field attachments

Whether documents and images can be associated with specific fields rather than stored in fields that are set up as document or image field types. To prevent these attachments from potentially being lost, you cannot synchronize databases unless their field attachment setting matches.

Note: You can use field attachments to change how entities are displayed on charts based on the information of interest. This information is accessible when the record is open, but is not searchable.

For more information about field attachments, see Enabling field attachments.

Field security

Whether you can set the SCC access to a record at a field level. To prevent information from potentially being visible when it should be restricted, you cannot synchronize databases unless their field security setting matches.

© N. Harris Computer Corporation

For more information about the field security, see Enabling field security.

Field confidence

Whether you can grade the information in a record at a field level. To prevent this grading information from potentially being lost, you cannot synchronize databases unless their field confidence setting matches.

For more information about field confidence, see Enabling field confidence.

Soft delete

Soft delete allows records to be removed from view without removing their details from the database.

For more information about the soft deleting, see Enabling soft delete.

Note: Soft delete has to be enabled in replicated databases.

Allowing non-adjacent filtered pick lists

Pick lists can be set up to display different values based on the selection of a value in a parent pick list. Normally, to filter a pick list, you must have the parent list field placed directly next to the child list, but you can select to allow filtering if both fields are available.

Procedure

1. In iBase Designer, open **File** > **Database Properties** > **Advanced**.
2. Select **Allow non-adjactent filtered pick lists**.
3. Restart any open iBase clients to see the changes.

SQL Server database names

The names that you choose for the security (`ids`) file and database (`idb`) file in iBase are used to generate the names of the SQL Server databases. For this reason, you might want to discuss the naming convention to use with your SQL Server administrator.

Main iBase database

A complete logical iBase database (for entity and link data) contains two Microsoft™ SQL Server databases:

- An iBase database:

  Typically the database name is similar to the name of the connection file, but is subject to modification to comply with SQL Server naming rules.

  The database name always contains an underscore (_). For example, if the requested database name is `Intelligence`, SQL Server uses the name `Intelligence_` and the connection file remains `Intelligence.idb`. Additionally, any spaces in database names are replaced by underscores (_). T
- An Audit Log database:

  The Audit Log database is the database name with _log added at the end, for example `Intelligence__log`. (Notice the double underscore in this single-word database name.)

These two databases are always present.

© N. Harris Computer Corporation

iBase security database

Optionally, iBase security data can be held in an SQL Server database. The SQL Server name follows the rules for the main iBase database but is appended with _sec. For example, if the name of the Access security file is `Intelligence.ids` then the SQL Server name is `Intelligence__sec`.

Renaming SQL Server databases

To rename an SQL Server database that contains entity and link data (not security data), create a new database in iBase Designer with the wanted name. The name must uniquely identify the database within your iBase system and also when used with third-party iBase databases. You must be logged on to the correct security file when you do rename a database. The connection file that is required by iBase to connect to the database on the server is also created. To move the data to the new database, your SQL Server administrator must make a backup of the SQL Server database that you want to rename and then restore the backup over the new database.

Attention: You cannot rename an SQL Server security database in this way. You lose the connection between the security file and the databases that it secures and prevent your users from opening the databases.

Upsizing a database to SQL Server

Existing iBase databases can be upsized from Access to SQL Server. The original Microsoft Access connection file is replaced with a connection file for SQL Server, but the content of the upsized database and text search indexes remains unchanged

About this task

Important: Upsizing a database is an irreversible operation.

Procedure

1. Select **Tools** > **Database Setup** > **Upsize** > **Database to SQL Server**.
2. In the first page of the wizard select the database.
3. Enter a name for the backup file. If you do not want to create a backup to disk, delete the suggested name. Click **Next**.
4. In the next page enter the SQL Server information and click **Next**.

   Note: Only select the (local) option from the Server drop-down list if the database is for personal use.
5. Check the information in the third page and click **Finish**.As the database is upsized a list of operations is displayed. When the upsize is complete, scroll through this list and check that all values in the Status column are "Success". When you close list, iBase Designer opens the SQL Server database.
6. Check that the settings in the Configuration and Advanced pages of the Database Properties dialog are as expected; they are for an SQL Server database.
7. Close the database.
8. Select **Tools** > **Database Administration** > **Schema Integrity Check**.
9. Select the upsized database and complete each page of the wizard. When you have completed the wizard, the SQL Server database is opened and ready to use.

© N. Harris Computer Corporation

Database subsets

A database subset is a portion of records in the database that are copied into a separate database. This collection of records are selected by creating a database subset definition that consists of the results of queries and sets.

You might want to create a database subset for a number of reasons:

Creating an environment that matches your current production environment for testing or training.

Adding a smaller amount of real data from a production environment lets you test changes to the database, or train users in as close to the production environment as possible.

Working with a set of data that relates to a specific department or organization.

By creating an environment that only contains specified data allows sanctioned data to be shared.

A database subset can be created from a query at any time, unlike the information in a case, that is assigned as the data is added.

To create a database subset:

1. Define the records to include using a subset definition.
2. Create the database subset in either Microsoft™ Access or SQL Server.

The database subset can then be used independently, and if required, you can synchronize any changes with the original database.

Creating a database subset definition

The records in a database subset are selected by creating a database subset definition. When you have created the definition, you can use it to export the data you selected as XML, or you can create a database containing the selected records.

Procedure

To define the records in a database subset:

1. Log on as a user with permission to add folder objects, and open the database.
2. Select **File** > **Data** > **Database Subsets** > **Database Subset Definitions**.
3. Click **New**.
4. Select the records by adding queries and sets to the definition. The queries and sets form a part of the definition and deleting any of these sets or queries, as opposed to just removing them from the definition, invalidates the definition and any database subsets created from it.

   Note: If the subset definition is being used to create database subsets in Microsoft Access, you can use parameterized queries and the values required to run these queries are entered when the database subset is created (or synchronized). If you include parameterized queries, then you must enter values for them when creating database subsets (and when synchronizing). Advanced subsets cannot be created using subset definitions that include parameterized queries.
5. Click **Save** to save the definition.

To create a database subset from your definition:

1. Select the type of database storage to use for your subset:

© N. Harris Computer Corporation

- To create a subset in a Microsoft Access database, select **Create Subset**, and follow the instructions in Creating database subsets (Microsoft Access).
- To create a subset in a Microsoft SQL Server database, select **Create Advanced Subset**, and follow the instructions in Create advanced database subsets (SQL Server).

Results

The database subset definition is created.

At any stage, you can:

- Change the definition by adding new sets and queries or by removing them (during synchronization the database subset will be re-created).
- Rename and move the sets and queries that are listed in the definition (this updates the definition).
- Rename the definition.
- Move the definition to a different folder.

You can also delete the definition if it is:

- No longer required to create new database subsets.
- No longer required to update database subsets at the end of synchronization.

Creating database subsets (Microsoft Access)

You can create a database subset from the records that are included in the results of running queries or sets that are specified in a database subset definition. If you use the **Create Database subset** option, the subset database will be in Microsoft Access format.

Before you begin

Before you can create a database subset, you need to specify the records that you want to copy to the new database by creating a database subset definition.

Note: Only database administrators can initialize the database for database subsets.

Procedure

To create a database subset:

1. Log on as a user that has the Database Creator role.
2. Open the database from which you want to create the database subset.
3. Select **File** > **Data** > **Database Subsets** > **Create Database Subset**.
4. In the **Identifier** box, enter a unique ID for the database subset. The ID is up to five alphanumeric characters long. Previously-used identifiers are listed in the **Utilized Identifiers** list.
5. In the **Name** box, enter a name that will be used for both the subset security file and subset database.
6. A new user account with system administrator permissions will be created in the subset security file. Enter the username and password for this account. This account will be used to synchronize the database subset with the main database and to log on to the database subset if no other user accounts are added to the security file.

© N. Harris Computer Corporation

Note: Any records added to the database subset will have this user as their "Create User". You may therefore want to select a username that will be meaningful once these records are uploaded to the main database.

7. In **Destination folder**, browse to the folder where you want to create the subset security file and database. You can create a new folder if you have sufficient Windows permissions. The folder you use can contain only one iBase database and security file.
8. In **Subset Definition**, browse for the definition that defines the data to be copied to the new database. At this stage, it is not possible to know whether the definition is still valid or whether the total number of records exceeds 50,000 (the maximum allowed records).
9. Click **Create** to continue. You will be warned if the definition is invalid because it contains deleted queries or sets, or if the total number of records exceeds the 50,000 record limit.
10. Click **OK** to create the database subset. If the definition contains any parametrized queries then you will be prompted for the values. You can click **Cancel** but doing so will also cancel the creation of the database subset.

Synchronizing database subsets

Database subsets are used remotely, and the records they contain must be synchronized with the main database regularly. Although in most cases, records are modified either in the subset or in the main database, you might need to resolve conflicts that arise.

Before you begin

You must connect to that database as a system administrator of the database subset to ensure that you have access to all records in the database subset and the necessary permissions.

During synchronization, you can choose whether the database subset expires after synchronization is complete.

About this task

Synchronization begins by identifying the records that are needed to repopulate the database subset by examining the queries and sets in the database subset definition. If the definition comprises any parameterized queries, then you are prompted for the parameter values. If you cancel entry of these values, then synchronization is also canceled. This step is not necessary for database subsets that are set to expire.

The main database is then updated in three phases:

1. All newly created records in the database subset are added to the main database, with the same record identifiers, create date and time, and the same create user.
2. Any updated records in the database subset are copied to the main database:
   - An update to a database subset record is applied to the main database if the main database record is unchanged since the last synchronization.
   - If a record is changed in both the main and the subset databases since the last synchronization, then the conflict is resolved by applying the rule that is selected by the user.
   - All soft deleted records in the database subset are ignored. They do not delete the corresponding record in the main database.
3. Finally, either:

© N. Harris Computer Corporation

- The database subset is updated with the changes and additions that are made in the main database.
- Or, if the database subset is set to expire, then all the entity and link records are deleted, and the database subset is set to read only.

Procedure

To synchronize a database subset:

1. Back up the main database if it is an Microsoft™ Access database. This step is unnecessary for SQL Server databases because updates are committed to the main database after each phase of the synchronization process.
2. Log on to iBase as a user with the Database Administrator role, and then open the main database.
3. Select **File > Data > Database Subsets > Synchronize Database Subset**.

What to do next

When a conflicting change is made in the same record (in any field in that record) in both the main database and database subset, you can choose to:

- Keep the changes made to all the main records that are in conflict, and discard all the changes that are made to the corresponding subset records
- Keep all the changes made to the subset records that are in conflict, and discard all the changes that are made to the corresponding main records

The user decides without knowing which records are affected or what the conflicts are, and the rule that is selected applies to all records with conflicting changes.

Notes:

- If the main record is deleted, and is changed in the subset, then it is either restored and updated (to match the subset record) or re-created (depending on whether it was soft deleted or purged).
- Restoring or re-creating a link always results in its link ends being restored or re-created if necessary.
- Restoring or re-creating an entity results in its links being restored or re-created; but only for those links where the other end of the link is still active.

Create advanced database subsets (SQL Server)

You can create a database subset from the records that are included in the results of running queries or sets that are specified in a database subset definition. If you use the **Create Advanced Subset** option, the subset database will be in Microsoft SQL Server format.

Before you begin

Before you can create a database subset, you need to specify the records that you want to copy to the new database by creating a database subset definition.

Note: Only database administrators can initialize the database for database subsets.

Procedure

To create an advanced database subset:

1. Log on as a user that has the Database Creator role.

© N. Harris Computer Corporation

2. Open the database from which you want to create the database subset.
3. Select **File** > **Data** > **Database Subsets** > **Create Advanced Subset**.
4. In the **Name** box, enter a name that will be used for both the subset security file and database subset. This also generates the **Database Name** displayed in the **SQL Server (subset)** section.
5. The subset security file will be generated with the same users as the master database.
6. Enter the Server connection URL in the **Server** box, and enter your database credentials, these can be:
   - An exact duplicate of the credentials used to access the master database.
   - A specified user name and password
   - Windows Authentication

   Tip: Test your connection each time you change the server or the credentials used to access it.
7. In **Destination folder**, browse to the folder where you want to create the subset security file and database identifier. You can create a new folder if you have sufficient Windows permissions. The folder you use can contain only one iBase database identifier and security file.
8. In **Subset Definition**, browse for the definition that defines the data to be copied to the new database. At this stage, it is not possible to know whether the definition is valid.
9. Confirm the iBase password that was entered when you logged into the database.
10. Click **Create** to continue. You will be warned if the definition is invalid if it contains parameterized queries, deleted queries or sets, or if the total number of records exceeds the record limit (5 million).
11. Click **OK** to create the database subset.

Advanced synchronize

Synchronizing databases, uploads the data from the database subset to the main database and downloads new and updated records in the subset definition to the database subset. You can update the database subset using the original subset definition or you can select a different subset definition.

About this task

When you synchronize an advanced database subset, the records are compared, any records that have been updated either the main database or the database subset is updated in the other location.

A conflict occurs when an entity or link is changed in both the main database and the database subset. To resolve the conflict, you need to decide which record you want to keep. You can either:

- Discard the subset record changes, keeping the changes to the record in the main database and lose the information in the record from the database subset.
- Keep the subset record changes, keeping the information in the record in the database subset and overwriting the changes in the main database.

If the main record is deleted, then it is:

- Restored and updated to match the subset record if Soft Delete is in use.
- Re-created if the record is deleted or purged.

Restoring or re-creating a link always results in the link ends being restored or re-created if necessary. Restoring or re-creating an entity also restores or re-creates any associated links if the other end of the link is still active.

During synchronization, the following error messages might be displayed:

© N. Harris Computer Corporation

- The database subset has expired. - You cannot reuse an expired database subset. Re-create it from its database subset definition.
- The database subset has an incompatible schema. - The database subset is invalid because the schema of the main database was changed after the database subset was created. To fix this problem, use the **Database Schema Update** option in iBase Designer.
- The database subset is read-only. - Use iBase Designer to change the database properties of the database subset so that it is no longer read-only. Although you can change the Read-only property in an expired database, you cannot reuse it.
- This is not a valid database subset. - The selected database subset is either not a database subset or it might be a subset of a different database. You can set the database subset to expire if you do not need it any longer. This deletes the contents of the database subset and mark it as read-only. The database subset can never be reused.

When you synchronize an advanced database subset with the main database:

- Newly created entities and links in the database subset are added to the main database, with the same record identifier, create date or time, and create user.
- All (soft) deleted records in the database subset are ignored - they have no effect on the main database.
- Records in the main database are updated to match the changes in the database subset if there are no conflicts.
- If a record has changed in both the main database and database subset, since the last synchronization, then conflict resolution is applied. See below for details.

At the end of synchronization, you are informed of the changes made to the main database:

- The number of new records added to the main database.
- The number of records updated in the main database with changes made in the database subset
- If Soft Delete is used: the number of records restored as a result of conflict resolution
- If Soft Delete is not used: the number of records that are re-created as a result of conflict resolution
- The total number of conflicts resolved (at record level)

When synchronization is complete, an updated database subset, re-created using the latest version of the subset definition, is available for reuse in the field. Alternatively, the database subset is set to read-only if the database subset was set to expire.

Procedure

To synchronize an advanced database subset:

1. Log on using a user account that has the Database Administrator role and permission to add records, update records, delete records, and update or delete records created by other users.
2. Open the database from which the database subset was created.
3. Select **File** > **Data** > **Database Subsets** > **Advanced Synchronize (SQL Server)**.
4. Browse for the database subset containing the records that you want to load.
5. Enter the iBase username and password used to access the database subset.

   Note: This user account should also have the Database Administrator role and permission to add records, update records, delete records, and update or delete records created by other users.
6. **Optional:** Use the **Options** to determine whether field attachments and records that have been deleted are included in the synchronization.

© N. Harris Computer Corporation

7. Click **Next**.
8. Decide how you want to handle any conflicts between changes made in the main database and those made in the database subset. By default, synchronization will never overwrite changes in the main database.

   Note: At this stage it is not possible to know whether there are actually any conflicts.
9. Click **Next**.
10. View the list of changes and use **Ignore Selected** to stop selected records from being updated.
11. **Optional:** Decide whether to update field attachments where they differ using **Include field attachments when repopulating**.
12. **Optional:** To discard the subset after uploading the records to the main database, turn on **The database subset should expire after synchronization**.
13. Click **Syncronize**.

Advanced synchronization behavior

Subset synchronization is a two-step process. First, the changes in the subset are pulled into the original database, and then the subset is repopulated from the original database that uses the current version of the subset definition.

This process leads to the following synchronization behavior for records that match the subset definition:

Main	**Subset**	**Synchronization result**
-	-	No changes to synchronize so both remain unchanged.
Added	-	Record added to the subset if the record matches the subset criteria.
-	Added	Record added to the main database.
Changed	Unchanged	Subset record is updated.
Unchanged	Changed	Original record is updated.
Changed	Changed	The user's preference of conflict resolution behavior is used.*
Soft-deleted	Unchanged	If the subset definition includes soft-deleted records, the record is marked as soft-deleted in the subset. If the subset definition does not include soft-deleted records, the record is removed from the subset.

© N. Harris Computer Corporation

Main	Subset	Synchronization result
Soft-deleted	Changed	Record updated in the main database to match the subset changes.
Unchanged	Soft-deleted	Record marked as soft-deleted in the main database. If the subset definition includes soft-deleted records, the record is marked as soft-deleted in the subset. If the subset definition does not include soft-deleted records, the record is removed from the subset.
Changed	Soft-deleted	Record re-created in the subset including the changes.
Changed	Changed then soft-deleted	The user's preference of conflict resolution behavior is used.*
Changed then soft-deleted	Changed	The user's preference of conflict resolution behavior is used.*
Purged/Deleted	Restored from a soft-deleted state	Record re-created in the main database.
Purged/Deleted	Unchanged	Record removed from the subset.
Purged/Deleted	Changed	Record re-created in the main database.
Changed then Purged/Deleted	Changed	Record re-created in the main database but without any changes made to the main record since the subset was last populated.
Unchanged	Purged/Deleted	Record re-created in the subset.
Changed	Purged/Deleted	Record re-created in the subset.

* If the changes in the main database include security changes that remove your permission to update the record, you are unable to synchronize.

Note: If the subset definition is changed, or records are added to the subset that do not match the criteria in the definition. New or updated records will be added to the master database and subsequently removed from the subset.

© N. Harris Computer Corporation

Creating case-controlled databases

If you need to restrict access to data on a case by case basis, you must to create a case-controlled database. Because you cannot convert a case-controlled database to a standard (SCC) controlled database, consider carefully whether you require this facility.

Note: You cannot use cases and Security Classification (SC) codes in the same database. You cannot use cases in a replicated database (or if iBase database replication is installed).

What is case control?

All data in a case-controlled database is partitioned by case. Every record belongs to a single case, and each user is assigned to one or more cases. Data cannot be shared between cases - data can only be entered, whether by manual entry or by importing, when a single case is selected. However, records from two or more cases can be analyzed together, for example by running queries and reports.

A record in a case-controlled database:

- Belongs to a single case.
- Might be duplicated across cases, such as a repeated telephone number, address, vehicle, but with distinct case ownership - updating one record does not update the other.
- Can only be edited or deleted when working in a single case.
- Cannot be linked to records in other cases.
- Is always read-only if it belongs to a closed case (but still appears in searches and queries).

A user in a case-controlled database:

- Sees only the records in the cases to which they are assigned.
- Sees all the records to which they have access when they work in multi-case analysis mode.
- Can only update records (either manually or by importing) in a single, open case. When working in a case, all reports, queries, browses, and so on, apply just to the records in that case.

Note: Sets are not specific to any case. A set can contain records from all the cases assigned to the user and, if the access to the set is Public, might also contain records added by other users, from other cases. However, a user only ever sees the records that belong to the current case (or all their assigned cases if working in multi-case analysis mode).

The history of each case can be recorded in the audit log. Actions include: Case Added, Case Closed, Case Deleted, Case Name Changed, Case Reopened.

Defining access to cases

All users, including database administrators, must be assigned to one or more cases before they can open a case-controlled database in iBase.

You do not need to assign system administrators to cases. Any system administrator can, in iBase Designer, add cases and assign users to the cases. They can also assign all the users who are members of a Data Access Control group.

See Creating and Managing Cases in iBase for details of creating cases and assigning users to them.

Displaying case names

By default, the case name is not displayed as part of the record - to display the case name in iBase you need to add a case field to each entity and link type. You might prefer to use a standard field for this.

© N. Harris Computer Corporation

A case field is useful to users working in multi-case mode who might want to know which case a record belongs to. Case fields are less useful to users who work only in one case at a time because the name of the case is displayed in the title bar of the iBase window at all times.

Note: It is not possible to change the value of a case field.

Creating a case-controlled database

You can create a new case-controlled database. This is similar to creating a standard database.

About this task

In the Create New Database dialog:

1.
2.

Procedure

1. In iBase Designer, select **File** > **New Database**.
2. On the **Configuration** page, select SQL Server as the database type.
3. On the **Advanced** page, select **Case Control**. This option is not available in the following situations:
   - If you also select a database template that uses SCC control - you must to create the database first and then convert it to case control later (before any records are entered).
   - If iBase database replication is installed on the machine.

   Note: After you create the database, you are not able to select the **Standard (SCC) Control** option, on the **Advanced** page of the Database Properties, which would allow the use of security classification (SC) codes. It is not possible to create a case-controlled database from a template that uses standard (SCC) control and vice versa. A template always inherits this setting from the database used to create it.

Converting a database to case control

SQL server databases without database replication can be converted to case-controlled databases. This allows copies of the data to be used remotely without connection and synchronized when connection is available.

About this task

Turning on case control is an irreversible process. Ensure that you back up your database before you experiment with this facility on a database that contains data. Where there is no data yet, you might create a template from your database instead of backing up.

After you convert to case control:

- You cannot convert the database back to standard (SCC) control.
- Menu commands and options that apply to security classification (SC) codes and SCC lists are no longer available.
- Any records that are classified by using security classification codes are stripped of those codes and assigned to a default case.

Note: You can preserve the classifications represented by security classification codes, for example to place Confidential records in a separate case to other records. Before you turn on case control, you

© N. Harris Computer Corporation

would need to export the Confidential records that you want to allocate to a different case. If required, you can delete those records from the database - this means that the Confidential records will not become part of the default case. After the database is case controlled, import the Confidential records into the required case(s).

Procedure

To convert an existing database to case control:

1. If you have an Access database, convert the database to SQL Server format. For more information, see Upsizing a Database to SQL Server.
2. Select **File** > **Database Properties**.
3. On the Advanced page, select **Case Control** and click **OK**.

   Note: This option is not available if iBase database replication is installed on the machine.
4. Confirm that you want to proceed with the conversion to case control and the removal of existing SC codes and SCC lists.
5. Enter the name of the default case. All existing records, folder objects, and alert definitions will be moved to this case. You can use export and import to move records between cases.
6. If alerting is in use, assign all users who have created alert definitions to the default case. You must do this before you close the database.

   Warning: Alert definitions without owners are deleted when you close the database.

iBase cases

Cases can be created in either iBase or iBase Designer. Cases can be accessed by people who are assigned to the case.

Before a user can open a case-controlled database, you as a system administrator must assign them to one or more cases. When a user opens a case-controlled database, they then select:

- Either, a single case to obtain read/write access to the case, if the case is not closed and that they belong to a user group that grants add, modify, and delete permissions. (Some analysts only ever require read-only access to the data.)
- Or, all their cases (by turning on **Multi-Case Analysis**) to obtain read-only access to all the cases assigned to them.

Regardless of the type of access, a user who selects all their cases, when they open a database, only ever has read-only access to their cases.

When no cases are defined, only administrators with the Database Administrator and Security Administrator roles can open the database in iBase; for example, to create a new case. To add data to the new case, they must select the case by selecting **File** > **Change Case**.

Note: To obtain information on the cases in a database, run a Security Design report.

Creating new cases

Cases can be created and updated by any administrator with both Security Administrator and Database Administrator roles.

1. To create a new case:
   - In iBase Designer: Select **New** > **Case** from or in the Database Explorer, right-click on Cases, and select **New**.

© N. Harris Computer Corporation

- In iBase: If required, select a single case on opening the database and select **New** > **Case**. Creating a case does not select it. To change to the new case: select **File** > **Change Case**. The name of the current case is shown in the title bar of the application window.

2. Enter the details of the case.
3. Assign users to the case.

Giving and revoking access to cases

You authorize users to work on a case when you create or edit it. You can either assign users one at a time or you can assign all the users who are members of a Data Access Control group.

Note: The Users page lists the users who are assigned individually to the case. If a user has access to a case because they belong to a group that is assigned to that case, then they are not listed on the Users page of the Case.

You do not need to assign system administrators to cases. Administrators (with both the Database Administration and Security Administration roles) always have access to all cases.

New or amended access to a case only takes effect the next time a user opens the database.

When you revoke access to a case, note:

- If a user has access to a case because they belong to a group that is assigned to a case, then you can only revoke access by removing the user from the group.
- If alerting is in use, then the user is removed from any alert definitions that they own. These alert definitions remain active for other users. A system administrator can delete the alert definitions if required.

Listing existing cases

You can view a list of all the cases in the database in two places:

- In iBase Designer: Click **Cases** in the left pane of the Database Explorer to list the cases in the right pane.
- In iBase: Log on with an account with both the Security Administrator and Database Administrator roles and select a database to open.

To obtain information on who is assigned to which case, run a Security Design report, select **Security** > **Security Design Report**.

Modifying cases

In both iBase Designer and iBase, you can modify the description of a case, change who has access to it, and open or close it (see the following sections). You can only change the name of the case in iBase Designer. When you change the name, all the records belonging to the case are updated as well.

In iBase, for users with current sessions, the changes take effect next time they open the database.

To modify a case:

- In iBase Designer, right-click on the case in the Database Explorer, and select **Edit**.
- In iBase, select from the **Edit** menu, select `Case`. You can only modify the case that you selected when you opened the database.

© N. Harris Computer Corporation

Viewing case statistics

You can view statistics by case or by database. To view statistics:

- In iBase Designer, select **File** > **Database Statistics**.
- In iBase, select **File** > **Properties Database Statistics**.

In iBase Designer, record statistics are displayed by case (both open and closed) for the whole database. For example, click the **Entity Types by Case** tab to see the number of entity records by type for each case. The number of entity records for the database is displayed on the Entity Types page.

In iBase, **Database Statistics** display only the statistics for the cases that are accessible to the user. For example:

- A user who selects a single case sees statistics for that single case.
- A user who works with all their cases sees statistics for entities in all their cases on the **Entity Types** page, and a breakdown by case on the **Entity Types by Case** page.
- A system administrator always sees statistics for all the cases in the database, even if they select a single case when opening the database.

You can also print the database statistics with a breakdown by case if required.

Closing and reopening cases

In both iBase Designer and iBase, you can close cases. When you close a case, the closure date is recorded. The closure date is taken from the date set on the user's machine. If required, you can reopen cases, and the closure date is retained until you close it again.

A closed case is read-only and no one can edit the records that belong to the case. However, any user who is assigned to the case can select it when they open the database. When more than one case is selected, operations such as finding and querying includes results from both closed and open cases.

To review the complete history of a case, use the audit log.

To close or reopen one or more cases:

- In iBase Designer: in the right pane of the Database Explorer, right-click on a case, and from the menu, select **Edit**.
- In iBase: log on as an administrator with the Security Administrator and Database Administrator roles, select the case that you want to close or reopen, and select **Edit** > **Case** .

You can then change the status of the case on the **General** page.

Deleting cases

When you delete a case in iBase Designer, you delete all the records that belong to that case, all the entries in the audit log for those records, and all alert definitions.

Before you delete a case:

1. Archive the records in the case. Either by backing up the database, or alternatively, exporting all records to text files.
2. Archive the audit log for the case, making sure that you have archives that cover the period of the case.

To delete a case in iBase Designer:

- In the Database Explorer, right-click on the case, and select **Delete**.

© N. Harris Computer Corporation

Routine maintenance

There are several areas in iBase that require routine maintenance to ensure that your database continues to run correctly. Where possible, you can use tools that are provided in iBase Designer to run maintenance tasks.

The commands for routine database maintenance are available from the **Tools** menu in iBase Designer:

Maintaining database tables and indexes

All databases and security files operate more slowly as deletions and changes increase the fragmentation of the data.

For Microsoft™ Access databases and security files, use the relevant **Tools** > **Database Administration** > **Repair/Compact** option.

For more information about maintaining tables and indexes in SQL Server databases, see Performance Tuning in iBase Designer.

Maintaining search indexes

Depending on the type of search, the method of maintaining the search index varies:

- For Search 360, ensure that:
  - The Index Service is scheduled to run regularly.
  - The `IBaseIndexDB` database and Searching `Config.xml` configuration file are included in your backup schedule.
  - The transaction log is monitored, and cleared when it becomes too large.
- For Word Search indexes, run **Tools** > **Search** > **Word Search Indexing** each time that you want to update the index.
- For Full-Text Search indexes, you can use **Tools** > **Search** > **Full-Text Search Indexing** to set up ongoing updates, either with a regular schedule or in response to changes in the database content. On a less regular basis, you might want to respond to user comments or new types of recorded data by updating the lists of excluded words or synonyms.

For more information, see Setting Up Search.

Check for responsiveness and integrity of the database.

If users report slow performance or recurring errors in normal operation, it might indicate fragmented disk files or some kind of corruption.

In iBase Designer, you can use the commands: **Repair/Compact Database File**, **Schema Integrity Check**, and **Link Integrity Check**. There might also be causes external to the database system, such as other processes that run on the server or client computers or poor network connections.

For more information, see Checking a database.

Managing databases that use Soft Delete

In databases using soft deletion of records, purge or restore records as required. For more information, see Batch delete.

Managing databases that use cases

In databases that use cases, add new cases, give and revoke access to cases, and close old cases as required. For more information, see iBase cases.

© N. Harris Computer Corporation

Monitoring iBase usage

Monitor the following regularly:

Audit logs

Use the external iBase Audit Viewer to monitor usage and identify any repeated events such as failed logons or repeated editing or analysis activities that might indicate user difficulties.

Pick lists

You might find that users are frequently typing to supply alternatives to entries suggested in a pick list. You can add these entries to the pick list if required.

Datasheets

Datasheets provide alternative ways of creating or viewing records. Their effectiveness might need monitoring. For example, you might find that analysts want to view an entity in association with selected fields from a particular type of link and linked entity, or that data entry might be made faster by reordering and regrouping the fields of information.

Select **Tools** > **Datasheet Manager** to start editing or creating datasheets.

Reviewing database design, statistics, and security

At any time, you can use iBase Designer to view, or change, the database properties that are chosen when you created the database, and view data statistics and a database design report. With default access control, all users of iBase can view, but not change, database properties, database statistics, and a database design and statistics report. Select the relevant command from **File** > **Properties**.

The security design report can have several forms, but always lists security groups, users, and their consequent permissions or restrictions. You can choose to include user information if required.

The security design report presents all the information held in the security file to which you are logged on. The report does this first by group, listing the group's properties (if any) and user membership; then by user, listing the accumulated permissions of the user, possibly gained by membership of several groups, and the groups of which the user is a member.

If you have databases open, the report includes the use made of Data Access Control groups in the active database.

Note: The security design report does not include details for the use made of Folder Object Control groups.

Routine maintenance on the database servers

You need to maintain adequate free space on disk for databases, search indexes, audit logs, and any linked documents. This is largely a matter of using tools supplied with Windows™ to monitor both the free space and the size of the files that are growing most quickly to reduce that free space.

The strategy that you adopt for databases might vary from moving old data to archives with iBase batch export, batch delete, or creating new databases to hold current data for each year or other time period. For audit logs, the external iBase Audit Viewer provides a way to view, archive, and delete old audit records.

You need to maintain adequate backups of the database, security file, and audit logs. You should schedule backups for a time when no users are using the database. See Backing Up iBase Databases.

© N. Harris Computer Corporation

Database backup procedures depend upon the type of database, Microsoft Access or SQL Server. Back up each security file frequently, as a complete file. Back up audit logs using the external iBase Audit Viewer to identify data for archiving. For further details, see the Audit Viewer help.

Routine maintenance in SQL Server

For large SQL Server databases, disk operations have a significant effect on the performance of the database. To reduce the amount of data that is read from disk during queries, iBase applies indexes to the data. Over time as data is added to and deleted from the database the indexes become fragmented and larger than they need to be. This reduces performance because more data blocks are read into memory to process a query. Eventually, without corrective action, the result is queries that run many times slower than in a newly indexed database.

You need to maintain the indexes of an SQL Server database. The larger the database, the more benefit comes from regular maintenance of the indexes:

- For databases that use legacy indexing or that are tuned in SQL Server, this is done by your SQL Server administrator.
- For databases that use indexes that are optimized in iBase (and that are not tuned in SQL Server), this can be done in the Performance Tuning dialog in iBase Designer. See Performance Tuning in iBase Designer for details.

The indexes should be rebuilt regularly, at intervals that are determined by your SQL Server administrator who is able to measure the fragmentation of the indexes using tools in Enterprise Manager or Management Studio. For example, a database that is updated with imports that use the Bulk Import method might require reindexing after each bulk import.

Each rebuild takes some time and should be scheduled to take place when the database is not in use.

Batch delete

Users can delete records from the database either individually or in batches. When an entity record is deleted, all links to that entity are also deleted - the link end entities are not deleted.

Deleting records is a permanent and irreversible operation unless soft delete is enabled for your database. When soft delete is enabled, deleted records, whether deleted individually or in batches, are removed from the user's view of the database but you have an opportunity to undo the deletion.

You can use Batch Delete with or without soft delete.

Note: You can deny users access to the **Batch Delete** menu command by using a System Commands Access Control group.

Soft delete

Soft deleted records do not appear in search results or in record lists (when listing and browsing records), but are not removed from the database.

For system administrators, soft deleted records:

- Can be restored using **Restore Deleted Records** in iBase.
- Can be permanently removed using **Purge Deleted Records** in iBase.

If soft delete is not enabled, then deleting records is a permanent and irreversible operation.

To check the setting of soft delete:

1. In iBase Designer, select **File** > **Database Properties**.

© N. Harris Computer Corporation

2. On the Advanced page, view the **Soft Delete** setting.

Batch Delete

To delete batches of records:

- In iBase, select **Edit** > **Batch Delete**.

The deletion can take a while to complete if you are deleting many records. The speed of the deletion depends on several factors:

- Whether Data Access Control is used and is restricting access to the records selected for deletion.
- The format of the database (Microsoft™ Access or SQL Server)
- The audit level if the database is in SQL Server format

The following information describes these factors.

Batch Delete and Data Access Control

The actual records that are deleted depend on whether Data Access Control restricts access to the records selected for deletion.

Consider this example: there are a 1000 telephone entities in the database with many telephone call links between them. A user has permission to view all 1000 telephone records but has restricted access to the telephone call links. In fact, of the 1000 telephone entities, only 200 of them have unrestricted telephone call links. This means that although the user has full access to all the telephone entities, they can only delete the 200 telephone entities with unrestricted telephone call links.

Entity 1 access restriction	**Link access restriction**	**Entity 2 access restriction**	**Delete entity 1 and/or link?**
None	None	None	Yes
None	None	Read-only table	Yes
None	None	Hidden table	No
None	None	Record is restricted (using an SC code)	Yes
None	Any restriction	Any restriction	No
Any restriction	Any restriction	Any restriction	No

Note: Any restriction includes making a table or field read-only, hiding a table or field, or applying a Security Classification (SC) code to deny access to a record.

If Batch Delete encounters a record with restricted access, iBase skips that record. It does not report that it encountered a record that it could not delete. At the end of the operation, it reports on the number of records that it successfully deleted.

© N. Harris Computer Corporation

Batch Delete in Access databases

After batch deletion starts in an Access database, you can press the **Esc** key to stop the deletion but you cannot cancel the deletion of records already deleted. A message is then displayed that tells you how many records have been deleted.

Batch Delete in SQL Server databases

How Batch Delete works in a SQL Server database depends on the audit level of the database. Batch Delete runs fastest with the audit level set to 1, 2 or 3:

Audit level 1, 2 or 3

After batch deletion starts, you can press the **Esc** key to cancel the deletion and, provided that **Esc** is pressed before the deletion finishes, no records are deleted.

Audit level 4 or 5

After batch deletion starts, you can press the **Esc** key to stop the deletion, however you cannot cancel the deletion of records that have already been removed. A message is then displayed that tells you how many records have been deleted. An entry is made in the audit log for each deleted record.

Backing up iBase

Like any critical system, it's important to make and maintain regular backups of your iBase databases. This part of the documentation describes how and when to make backups, and provides guidance on establishing a backup policy.

As well as the database itself, which can be in Microsoft Access or SQL Server format, iBase uses a number of files to store configuration and other data. Backing up iBase requires you to consider all parts of the system.

Database folder

iBase's database folder contains the main database files (for Access) or connection files (for SQL Server). This folder also contains the security file, the log file, and the word search index for Microsoft Access databases.

Shared application data folder

iBase stores files that are shared by all users in Windows' shared folder for application data. Specifically, the folder contains database templates (although the installation can be customized so that workgroup templates are held in a different folder) and icon lists.

Note: Users do not write to this folder, but to their own areas for application data. See Installation and Application Data Folders for details.

SQL Server databases

For an iBase installation that uses SQL Server, you must back up the databases that store the iBase data. These databases include the main database, the security database, and the audit log database.

For more information, see Backing up SQL Server databases.

© N. Harris Computer Corporation

Other information

Every installation of iBase is different, and you might have additional files or databases that you need to back up. For example, for Microsoft Access databases, there might be separate folders for archive log files. For SQL Server databases, there might be separate databases that contain archived data.

Backing up the system

Every iBase installation is different, but the procedure for backing up the system is similar for all of them. The following instructions provide a general guide to backing up your iBase system.

Identify the iBase databases

If you don't know where all the iBase databases to be backed up are located, you can find them by checking the MRU List Manager from the **Tools** menu in iBase or iBase Designer on users' workstations.

The list of files that you obtain from the MRU List Manager provide the paths and names of the databases. Locating the iBase security (`.ids`) and database (`.idb`) files will reveal most of the others. The security file is usually located in the same folder as the databases it secures.

Determine the types of database in use

The iBase security and database files will help you to identify the locations of any SQL Server databases that iBase uses. To determine whether the security and database files are SQL Server or Access:

1. In an iBase client (*not* iBase Designer), log in as a system administrator, and open each database that you identified earlier.
2. Select **File > Security File Properties** and **File > Database Properties** in turn.

In the case of SQL Server security and database files, the properties dialogs will display the database name and the SQL Server name. In general:

- An iBase database name will not contain any spaces, and will always contain at least one underscore character.
- An iBase database will also have an associated audit log database that has the same name as the database, with a suffix of `_Log`.
- An iBase security file database is suffixed with `_Sec`.

You must make a note of the names of the SQL Server and the iBase databases, ready for the next part of the process.

Pause scheduled operations

Before backing up, ensure that the Scheduler service on the server that runs scheduled tasks is not performing imports or exports. Then, place the service in the "stopped" state to prevent future imports or exports from running.

If the Scheduler is in the process of running an import or export, wait for it to complete before you stop the service.

Note: For information on how to stop a Windows service, refer to Microsoft's documentation.

© N. Harris Computer Corporation

Perform the backup

When you have a full list of all the files and databases involved, you can start the backup process. Ensure that you make copies of all the files, and that you store them in a safe location. You will need both network and database admin permissions in order to back up your system successfully.

To back up the files:

1. Log in to the server with network administrator rights.
2. Identify the files to be backed up from the list that you prepared earlier.
3. Back up the files to a safe location.
4. Restore the backup to a test environment to ensure all key files have been recorded.

To back up a SQL Server iBase database:

1. Log in to the system with database administrator rights.
2. Verify that there are no users accessing the database.
3. Set the user access option on the iBase SQL Server database to single-user.

   Provided that you don't log out of SQL Server, only you will have access to the iBase databases, preventing changes from being made during the backup.
4. Perform the backups (including any Analysis Services databases and Audit Log databases for this database).
5. If you're backing up in preparation for another activity (an upgrade, for example), then take the database offline.

Note: Changing the user access option to single-user does not prevent users from accessing the database altogether if you log off. However, it does restrict the amount of change that can occur if a user does access the database. If you're upgrading multiple databases, perform steps 3 and 4 sequentially for each database in turn before you log out.

If you need help with backing up iBase, contact i2 Support.

Backing up SQL Server databases

SQL Server provides tools for performing and automating backups of your iBase databases. You can also use any third-party backup tools that you're familiar with. The important thing is that you back up the right files at suitable intervals.

In an iBase SQL Server database, there are several types of data that you need to back up:

Main database

A *main* SQL Server database is one that contains iBase entity and link data. Its name is based on the name of the `.idb` file in the database folder.

For each main database, you must decide on your backup regime depending on how you populate it:

Continuous updates

If the iBase database is populated by users that enter data continuously, then backing up the database is in two parts. You must back up the data file in which SQL Server keeps its data, and the file in which it keeps a log of all of the changes that are made to the database. SQL Server can use this Transaction Log file to recover changes made between main backups. The Transaction Log itself can be backed up during the working day.

© N. Harris Computer Corporation

Regular bulk updates

If the database is populated by periodically loading a set of data such as daily changes, then you can turn off the Transaction Log mechanism in SQL Server. You need only to back up the data file in which SQL Server keeps its data.

Note: Significant data can also be held in database subsets on users' own workstations.

Security database

If you have upsized any of your main databases' security files to SQL Server, then each one has a corresponding security database. Its name is based on the name of the `.ids` file in the database folder, appended with `_Sec`.

The security database stores user information and group membership information for all the users in the system. Loss of this information can result in an inaccessible database until it is recreated and the user, group, and extended access control information is rebuilt.

After creation, the information within the security database rarely changes, so you only need to make backups when changes take place - if you switch alerting on, for example.

Audit data

If audit information is vital to your organization - if you are using alerting, for example - then in addition to backing up the main database and the security database, you must also back up the audit log database and its Transaction Log file.

You can use iBase Audit Viewer to archive audit information to a separate SQL Server database on a different ("linked") server.

Connection files

Connection files to main and security iBase databases in SQL Server store only the configuration details for logging into and accessing the databases. The loss of either of these files and the absence of a backup can involve complicated recovery process before users are able to gain access to the iBase database again.

Report templates, database templates, icons

iBase stores its template and icon files in the `Program Files\i2 iBase <n>\Resources` folder. You should back up all these files to avoid unexpected behavior from iBase if they are inadvertently lost.

Search 360 indexes

A separate SQL Server database, `IBaseIndexDB`, contains configuration information for the service that builds and updates Search 360 indexes. This database might be on a separate server.

When you back up this database, you should also back up the configuration file, `Searching Config.xml`, in the data folder on the local machine - specifically, `C:\ProgramData\i2\i2 iBase\<language>`.

Note: The backup must also include the files that hold the search catalogs and indexes that Full-Text Search uses. Backing up and restoring these files is separate from SQL Server backup and recovery, but you should coordinate any recovery process of databases and files to ensure synchronization.

For detailed information on backing up SQL Server databases, see the Microsoft SQL Server documentation.

© N. Harris Computer Corporation

Restoring SQL Server databases and security files

When you back up SQL Server databases, you must always back up the associated connection (`.idb`) files. When you restore those databases, you must always restore the corresponding connection files.

This rule also applies to security (`.ids`) files, which are also connection files if the security database is in SQL Server format.

Note: Sometimes, SQL Server prevents the Application Role of the restored database from being re-created. If you try to open the database in iBase Designer and cannot connect, you might see an error message to that effect. See this i2 Support article to resolve the issue.

iBase backup policies

A backup policy is essential to ensure that you can recover your data. It is important to establish a backup policy that covers all the elements of an iBase installation.

Consider implementing a backup policy that includes the following:

- A daily backup of the `.idb` and `.ids`.
- A daily backup of the main database and the `_Log` database.
- A daily backup of the `_Sec` database, if it has been upsized to SQL Server.
- A daily check on the free disk space for the SQL Server.
- If the security file is in Access format, a weekly check that the size of the `.ids` file is less than 1GB. If it is larger, you can use repair and compact and Audit Viewer to purge some older log files.
- A monthly review of statistics for the main database and the `_Log` database.
- A six-monthly review of the size of your different SQL Server databases. If the server memory is insufficient, you might see performance issues when processing large numbers of records.

In addition to your regular backup cycle, there are other occasions when you should also make a backup. Some examples include:

- Before you upsize a Microsoft Access database (or security file) to SQL Server
- Before and after you import data using Bulk Import
- Before you delete the records held in a case
- Before you convert a database to case control
- Before you use **Update Database Schema**
- Before you synchronize a database subset with an iBase database in Microsoft Access format

Attention: It is safest to perform backups while no users are performing operations on the database and no services are running.

Some iBase operations can take place over a relatively long time and affect multiple database records. Examples of such operations are data imports, batch edit, batch delete, merge, or deletion of entities with many links.

If a backup was performed during such an operation, and the database is subsequently restored from the backup, the operation restores data on which work was in progress and is potentially in an incomplete state.

© N. Harris Computer Corporation

Client file types

An installation of an iBase client application on a user's workstation includes several files that are important to the operation of the software. You must take these files into account when performing a backup.

The following list contains names and descriptions of all the types of files that might be present in a client installation.

`scheduler_database_name.mdb`

This file is the database that controls the imports and exports that iBase Scheduler performs. The location of this file is specified during the installation of iBase Scheduler.

`CommandGroups.mdb`

This file defines the set of command groups for use in security files. It is installed with iBase on each workstation, but might have been modified to define a different set of command groups.

The file is usually located in the `Resources` subfolder of the iBase installation folder, but iBase might have been directed to look for it in another folder.

`IconList.txt`

This file defines the icons that are available to iBase, and the mappings between the displayed name and the operating system file name.

The file is usually located in the `Resources` subfolder of the iBase installation folder, but iBase might have been directed to look for it in another folder.

`WSExclude.txt`, `FTExclude.txt`

These files define the exclusion lists for iBase's Word Search and Full-Text Search features. The files are usually located in the `Resources` subfolder of the iBase installation folder.

`i2.wor`, `i2.mxd`, `i2.apr`

These files define GIS workspaces that iBase uses. They are installed with iBase GIS Interfaces on each workstation, but might have been modified since installation.

Database file types

An installation of an iBase database on a server includes many files that are vital to the operation of the software. You must take these files into account when performing a backup.

The following list contains names and descriptions of all the types of files that might be present in a server installation.

File name	Description
`security_file_name.ids`	iBase security file (or iBase security connection file for SQL Server).
`database_name.idb`	iBase database (or iBase database connection file for SQL Server).
`database_name.idl`	iBase audit log. This file exists for all iBase Access databases, and for iBase SQL Server

© N. Harris Computer Corporation

File name	Description
	databases that have been upsized. Located in the same folder as the iBase database.
`database_name.idx`	Word search index for iBase Access databases. Located in the same folder as the iBase database.
`security_file_name.ids.bak, security_file_name.ids.bak1, ...n`	Backups of the security file created by an iBase version upgrade, or when the security file was upsized to SQL Server.
`database_name.idb.bak, database_name.idb.bak1, ...n`	Backups of the database file created by an iBase version upgrade or when the database was upsized to SQL Server.
`database_name*.doc, database_name*.dot`	iBase report templates. Located in the same folder as the iBase database.
`database_name*.ant`	Analyst's Notebook template for use with the database. Located in the same folder as the iBase database.

Moving and Copying Databases

The procedures used to move and copy databases are slightly different for Microsoft Access and SQL Server databases. The principles of moving a database applies equally to copying a populated database.

An iBase system might contain several related databases, for example copies on laptops of a main database on a central server. In this situation, you might want to define the folder objects that are shared by all the databases before you create or copying the main database. You might also want to consider Database Subsets as an alternative to copying the database.

Note: For greater control over these folder objects, use the separately-licensed Schema Update utility; for details, see Handling updates to the schema and folder objects.

There are several reasons why you might need to move, copy, or rename a database and its security file. Among these reasons are:

- Migration to a different computer or server
- Providing a copy for use at another site or on a laptop

  Warning: Consider using database subsets where only a portion of the records is required.
- Routine backup

When you copy or renaming a database, you should select a name that uniquely identifies it within your iBase system and also when used with third-party iBase databases.

The procedures for moving or copying a database and its security file are different for Microsoft Access and SQL Server databases and are described separately in:

- Moving a Microsoft Access Database or Security File
- Moving an SQL Server Database or Security File

© N. Harris Computer Corporation

You always need both the Windows permissions to move the files, and the ability to log on in iBase Designer as an iBase Security Administrator. If you are moving an SQL server database, you also require an SQL Server login name and password for connecting to each of the relevant SQL Server instances.

Note: Each database records the location of the security file that protects it. Each database is secured by only one security file but there might be several databases secured by the same security file. There must be only one security file in any one folder. The folder should be shared and referenced by a UNC path.

Handling external files

Databases can make references to, or otherwise use, external files. Many of these files must to be moved or copied with the database:

Hyperlink fields

As file names given in Hyperlink fields within records.

For a database with multi-user access across a network, good practice would mean that all such files are held in a shared folder and named in the field using a UNC path, such as: `\\server\sharedfolder\Report99.doc`.

- No action is needed if this is true and the shared folder is accessible to users of the database in its new location.
- If UNC names are not used, copy the files to a corresponding drive letter on the destination computer, as discussed next for a single user database.

For a single user database, it is possible that the files are held on a local disk and named using a drive letter and local folder names, as in this example: `C:\Artwork\House.bmp`.

You must copy these files to a similar location on the destination computer. This might not always be possible if there is a conflicting use of drive letters.

Support files

As support files, such as chart and report templates, held in the same disk folder as the database `.idb` file.

You must copy or move these files if you copy or move the `.idb` file so that the files stay together in one folder.

Audit log files

As a log file with extension `.idl`, only present for an iBase Microsoft Access database.

You can move this file if you want to maintain a single log file for the database. If you do not move this file, iBase creates a new log file in the new location.

Word Search index

As a Word Search index with extension `.idx`, only present for an iBase Microsoft Access database that uses Word Search.

You do not need to move this file. You can use iBase Designer to create a new index file in the new location.

You must make and restore true binary copies of all files mentioned in this section, using any convenient method supported by Microsoft Windows. If all you do is make copies for backup and occasionally restore from these copies to the original location, there is no special iBase procedure

© N. Harris Computer Corporation

to follow. The procedures for handling external files are the same for both Access and SQL Server databases.

Handling updates to the schema and folder objects

If you have a Schema Update license, you can keep the copies of a main database, for example held on laptops, in step with changes made to the main database. Changes could include the addition of new fields, new pick lists, or changes to folder objects such as import specifications.

To facilitate the maintenance of copy databases on laptops, you can mark the folder objects that you want to be able to update in the future as common folder objects. These objects can then be added to, updated and deleted from the copy databases--- standard folder objects cannot be maintained in this way.

Common folder objects can also be used to facilitate the updating of data in a copy database. For example, you could:

1. Add import specifications and an import batch specification to the main database, and export the data from the main database to create import files for use with the import batch specification.
2. Mark the import specifications and import batch specification as common folder objects.
3. Save a template from the main database to give to your laptop users.
4. Each user applies the template to their copy of the database. This adds the specifications to their database.
5. Each user runs the import batch specification to load the new and amended records.

For further information, see Using Common Folder Objects.

After the move

After a database is moved, users must find the new location of any moved files. After users open a moved database, iBase records any change of connection file location in the most recently used (MRU) section of their **File** menu.

What happens in subsequent use depends on the relative positions of the connection and security files:

- If the security and connection files are in the same folder, users see no change from behavior before the move.
- If the security and connection files are in different folders, users see a Security File browser each time that they need to log on and must navigate to the security file. (Where possible, you should always keep the security file and database in the same folder.)

Moving Access databases or security files

Move or rename an Access database or security file.

The necessary iBase database files to move are:

- The security file, with extension `.ids`
- The database file, with extension `.idb`

Moving or renaming the security file

If you move the security file to another location or rename it, you must open each related database in iBase Designer to update the stored location.

© N. Harris Computer Corporation

What happens when you open the database depends on the location of the security file. The possibilities are:

- If there is a security file in the same folder as the database file, iBase Designer opens that file immediately, even if it is the wrong security file.
- If it is the wrong security file, an error message notifies the user that the database is not associated with the current security file.
- If there is no security file, iBase Designer displays a Security File browser for you to locate the moved security file and click **Open**.

An example message:

```
Incorrect Security File
You normally connect to this database via a different security file
(\\ SERVER\databases\my_security.ids).
Are you sure you want to connect via this security file
(\\SERVER2\databases2\my_security.ids)?
```

If you confirm that you want to use the new file, iBase Designer stores the location. You can close the database immediately, or continue working.

If you have other databases that are linked to this security file you can open and close them for update now, without needing to log on each time.

Moving or renaming the database file

After a database file is moved or renamed, there is no need to open an Access database in iBase Designer if the security file is in its original location.

As mentioned for moved security files, you (and users in iBase) need to confirm the location of a security file that is not in the same folder as the database file. However, keep the security file in the same folder as the databases it secures.

Note: Renaming a database file prevents any existing Analyst's Notebook charts from accessing that database.

Note: If your organization creates maps or Analyst's Notebook charts that use data from multiple iBase databases, the name of the database must be unique.

What users see

At their next use of each database, users must find the new location of the moved or renamed file.

After users open a moved database, iBase records any change of database location in the recently used (MRU) section of their **File** menu.

What happens in subsequent use depends on the relative positions of the database and security files:

- If the security and database files are in the same folder, users see no change from behavior before the move.
- If the security and database files are in different folders, users see a Security File browser each time that they need to log on and must locate the security file.

© N. Harris Computer Corporation

Moving SQL Server databases or security files

For a SQL Server iBase database, moving or renaming the database also involves making changes to its connection and security files. In some cases, iBase Designer can make the changes automatically. In others, you might need to intervene.

An iBase SQL Server database consists of:

- The security file, with extension `.ids`
- The database connection file, with extension `.idb`
- The Microsoft™ SQL Server database that holds the main iBase database
- Optionally, extra Microsoft SQL Server databases that hold the iBase audit log, archived audit logs, and the Search 360 indexes

If the security file has been upsized to SQL Server, the database also consists of:

- The security *connection* file, with extension `.ids`
- The SQL Server database that holds the security data

It is simplest for administrators and users if you keep the security file (or security connection file) in the same folder as the database connection files that it secures. Share the folder and reference it by a UNC path.

Note: Do not copy connection files to client workstations. This might compromise the security of your system and adds to the administrative workload. Keep the `.ids` file in the same folder as the `.idb` file, in a central location.

Moving or renaming the security connection file

If you move or rename a SQL Server security connection file, you must open each related database in iBase Designer to update the stored location.

Note: If you move the security and database connection files to the same folder, you can update both locations in one operation by opening the database.

What happens when you open the database depends on the location of the `.ids` file relative to the `.idb` file. The possibilities are:

- If there is a `.ids` file in the same folder as the database connection file, iBase Designer opens that file immediately, even if it is the wrong security file.
- If there is no `.ids` file, a Security File browser is displayed for you to locate the moved security file and click **Open**.

When you open the database, log on as a user with the Security Administrator role. iBase Designer then recognizes that the security file is in a new location, and asks if you want to store that new location in the database.

An example message:

```
Incorrect Security File
You normally connect to this database via a different security file
(\\SERVER\databases\my_security.ids).
Are you sure you want to connect via this security file
(\\SERVER2\databases2\my_security.ids)?
```

If you confirm that you want to use the new file, iBase Designer stores the location. You can close the database immediately, or make any changes that you want.

© N. Harris Computer Corporation

If you have other database connection files and databases linked to this security file, you can open and close them for update now, without needing to log on each time.

Moving or renaming the database connection file

You can move or rename database connection files, but you should not copy them to client workstations. After you move or rename a database connection file:

- In iBase Designer, open the database connection file in its new location. The new location is stored in the database. There is no accompanying message.

Note: Renaming a database connection file prevents any existing Analyst's Notebook charts from accessing that database.

Note: If your organization builds up maps or Analyst's Notebook charts that use data from multiple iBase databases, the name of the database connection file must be unique.

Moving the SQL Server database

To move a Microsoft SQL Server database to another server:

1. Disable Search 360 and Alerting.
2. Backup the iBase `.idb` and `.ids` files.
3. Backup the iBase database in Microsoft SQL Server Management Studio.
4. You should install the versions of iBase and SQL Server on your new server; or upgrade your Production environment before moving servers.
5. Copy the iBase `.idb` and `.ids` files to the new server.
6. Use Microsoft SQL Server Backup and Restore to copy the database from server to server. You must use the same name for the database on the new server as it had on the old server.

   Note: It is possible to rename the database if it is the main database containing the entity and link data. For details, see SQL Server Database Names. You cannot rename the database containing the security data.
7. Use the Database Configuration utility to open the database connection file that connects to the database, and update the connection details for the new server. See Managing SQL Server Connection Settings.

   Note: If the security (`.ids`) file has been upsized, you might need to provide the Database Configuration utility with the *database* password in order to remake the connection with the security database. See the next section for information.
8. Open the database in iBase Designer to confirm that the connection is working correctly.
9. If you are using Bulk Import, Alerting, or Search 360 for this database, you must set up the new server. See:
   - Bulk import details
   - Configuring alerting
   - Setting up Search 360

Note: If you move a full-text indexed database to another server, you must also install the Microsoft Search Service on the new server if you want to continue to use Full-Text Search.

© N. Harris Computer Corporation

Updating an upsized security file

An upsized security (`.ids`) file stores the location of the SQL Server database that contains the security data. If, after you copy the database to a new server, the `.ids` file no longer has access to the old environment, the Database Configuration Utility displays this message:

```
iBase was unable to connect to the security database. Please enter the
 database password for this security file to confirm that you have security
 administrator permissions.
```

To find the password and open the database on the new server:

1. In the *old* environment, open the `.idb` connection file in iBase Designer.
2. Go to **Tools > Feature availability > Options**.
3. In the **Options** dialog, click the **Advanced** tab and make a note of the value in the **Database Password** field.
4. Return to the new environment and launch the Database Configuration Utility.
5. Open the `.ids` file and enter your administrator credentials. This results in the message requesting the password.
6. Enter the password that you noted in Step 3.
7. Click **OK** to open the security database in your new environment.

Database schema updates

Schema changes to an operational database on a server are typically made and tested in a temporary copy of the database before application to the operational database itself. You can use the **Update Database Schema** command in iBase Designer to manage this process, making the changes and then applying them to the other databases by applying a new database template.

This process is only suitable for compatible databases. A compatible database is any database that is created from the same database template or any copy of a database. These databases are compatible because their entity types, link types, fields, and standard fields share underlying table names, column names, and identifiers. For example, you cannot make a database 'compatible' by adding an apparently identical entity type because the entity type might not have the same table ID as the other databases.

A source database becomes incompatible with the other databases if you turn on case control - any action that you take must be repeated in all the related databases. Adding, modifying, or deleting entity types, link types, fields or standard fields does not make it incompatible because these changes can be updated to the target databases by saving a template.

A target database becomes incompatible if there is a conflict between the identifiers in the source and target databases. For example, if you manually add an entity type to the target database that has the same identifier as a different entity type in the source database. It also becomes incompatible with the source database if you turn on case control when the source database is not case-controlled.

Updating the original schema

Elements of a database schema that can be updated:

- Entity types, link types, fields, and standard fields
- Datasheets
- Pick lists, icon lists, and SCC lists
- Common folder objects, such as import specifications, report definitions, queries, charting schemes and so on (but not labeling schemes).

© N. Harris Computer Corporation

You can add to and edit these items as required.

Attention: Removing entity types, link types, fields or standard fields from the schema of an operational database deletes the data held for those database objects.

Creating a template for a schema update

To create a template that captures the updates to a database schema, including any changes to the common folder objects, create a template from the database that contains the required updates.

You should always test the new template before you apply it to the operational database or any copy databases. To do this, create a copy of the operational database and apply the update template to it (using the steps in the following section). Only when you verify that the database was updated correctly, should you apply these steps to your operational database.

Note: You can also create new databases from this template if required. Any database created from the template contains both the ordinary folder objects and the common folder objects.

Updating the schema of a database from a template

After you create a suitable template, you can apply the new schema to the operational database and to any copies of it. Before you start, make sure that you have:

- A backup of the databases
- Permissions to create and delete files in the same folder as the main database `.idb` file

To apply the schema change:

1. In iBase Designer, log on as a database administrator and open the database.
2. From the **Tools** menu, select `Database Design Update Database Schema`. An empty Update Database Schema dialog is displayed.

   Note: You cannot display this dialog if you are a member of a Data Access Control group that denies access to any tables or fields in the database.
3. Select the template that contains the schema changes.

   After you select a template, you can review the entity types, link types, and fields in the template by clicking

   .
4. On the Additions and Modifications page, and the Deletions page, review the changes that are listed. For example, the Additions and Modifications page summarizes the changes made to:
   - Entity types and their fields
   - Link types and their fields
   - Standard fields
   - Datasheets
   - Pick lists, icon lists, and SCC lists
   - Common folder objects (listed separately for each type of folder object)
   - Semantic Type Library (but specific changes are not listed)

© N. Harris Computer Corporation

5. If required, click

to save a list of the schema changes in a file that you can print later.

6. Click **Update** when you are ready to apply the changes. When this is finished, you are warned if any folder objects were renamed because they have the same name as a common folder object in the template.

Common folder objects

You can simplify the administration of several common databases, by defining a core set of folder objects (common folder objects).

Common folder objects across all the databases:

- Have identical names
- Are in the same categories
- Have an identical definition
- Are set to Public access (unless you are using iBase database replication in which case the original access setting on the folder object is preserved)

Any authorized user can define folder objects as common items.

There is otherwise no visible difference between an ordinary folder object and a common folder object. For this reason, you might want to use a naming convention for common folder objects or keep them in a specific category.

How common folder objects are updated

In order for the Common Folder Objects option to be available in the iBase **Tools** menu, you need to have the Schema Update Option installed. You can modify your iBase installation in the usual way from the Windows Control Panel. From the Custom Setup page in the installation wizard, select the Schema Update Option, under Extended Features.

Common folder objects are updated by running the `Schema Update` command in iBase Designer. This command applies changes held in a database template to the schema of the database in which it is run.

When a folder object, such as a report definition or a charting scheme, is defined as a common folder object, it can be:

- Added to databases that do not already contain it
- Updated with the changes held in a database template
- Removed from a database if it exists in the database but not in the template

Ordinary folder objects remain unchanged (but are renamed if they have the same name as a common folder object).

To update a compatible database with the current folder objects, create a template from the database containing the folder objects, and then apply that template to the other database. For more information, see Updating Database Schemas.

Defining a common folder object

To define an existing folder object as a common folder object:

© N. Harris Computer Corporation

- From the **Tools** menu in iBase, select `Common Folder Objects`. The Common Folder Objects dialog is displayed. Click **Help** in the dialog for information on how to use the dialog.
- Dependent on a set. Being data-dependent, sets cannot be saved in a template.
- Dependent on a folder object that is not selected as a common folder object (or that is deleted).

A folder object cannot be defined as a common folder object if it is:

The settings that are made in the Common Folder Objects dialog are saved in the database. Redisplaying the dialog displays the common folder objects defined in the database.

Any template that is saved from the database, distinguishes between ordinary and common folder objects.

Effect of adding, modifying, and removing common folder objects

What happens when you define a new folder object as a common folder object in the source database on folder objects in the target database is summarized below:

- In the source database, add a folder object and define it as a common folder object

  An identical common folder object is added to the target database. If any ordinary folder object with the same name exists, then the object is not overwritten but it is renamed by adding an underscore to the beginning of the name.

  Note: The access permission is not copied, unless you are using iBase database replication.
- In the source database, modify a common folder object

  The common folder object in the target database is updated to match the definition in the source database, including any updates to the name or category. If the common folder object was renamed in the source database, then any ordinary folder object in the target database with the same name is not overwritten. It is renamed by adding an underscore to the beginning of the name.

  Note: The access permission is not copied, unless you are using iBase database replication.
- In the source database, make a common folder object into an ordinary folder object

  The common folder object is deleted from the target database.
- In the source database, delete a common folder object

  The common folder object is deleted from the target database.

Note: You are informed if any name changes are made during the update process. The renamed folder objects are identifiable as they appear at the top of any lists (because of the underscore prefix).

Checking a database

You can check a database after upsizing, or after large changes or prolonged editing, or at any time that you suspect problems.

There are several ways to check a database and, if necessary, repair any inefficiencies or errors found. In approximate order of use, you should use these commands:

- **Database Statistics** (or **Database Design Report**)
- **Repair/Compact Database File**
- **Schema Integrity Check**
- **Link Integrity Check**

© N. Harris Computer Corporation

Except for **Database Statistics** (or **Database Design Report**), all these methods work with a database that is not open in iBase Designer.

Reviewing the database statistics

Use the **Database Statistics** command to review the database statistics. These statistics provide a quick way of seeing how many entity and link records a database contains.

About this task

One statistics report is only a count of the records, but seeing reports with identical counts, before and after any conversion process, provides a quick confidence check that all data is transferred to the new database.

Procedure

1. Select **File** > **Database Statistics**.
2. **Optional:** If you want to make a paper record for later reference, click **Print**.

Repairing and compacting the database file

Use the **Repair/Compact** command to reduce the fragmentation of tables in a Microsoft™ Access database where many records have been changed or removed. This command removes only the space that is marked as being unused in the database. For SQL Server databases, it only compacts the connection file.

About this task

You must be logged on to the relevant security file but have the database closed.

Procedure

1. In iBase Designer, select **Tools** > **Database Administration** > **Repair/Compact** > **Database File**.
2. Check that the result is success.

What to do next

Whether or not you see a successful report, complete the other checks described next:

- Checking the integrity of the schema
- Checking the integrity of the links

Checking the integrity of the schema

Use the Schema Integrity Check wizard to check the integrity of the database schema, that is, whether the database structure follows the rules set up in the database design.

About this task

The wizard reports any problems found and offers to fix those that it can repair. Some repairs can involve additions to the schema to make them consistent. For example, the wizard might add indexes or fields (that can be blank or use the default field value), or change the size of fields. You can choose to abandon repairs at any point up to final approval, allowing you to assess what the repairs would mean. Eventually, you must repair problems to avoid the possibility of prolonged and greater errors.

© N. Harris Computer Corporation

Note: Schema Integrity Check is unable to check or repair the indexes if you have run the Performance Tuning wizard but not yet completed the reindexing process. For further information, see Performance Tuning in iBase Designer.

Procedure

To use this wizard:

1. Log on to the relevant security file as a database administrator but do not open the database.
2. From the **Tools** menu in iBase Designer, select **Database Administration Schema Integrity Check**. A list of database files is displayed.
3. Select a database from the list. If necessary, select **More Files** to display a file browser where you can locate the database.
4. Click **Next**. The next page displays a list of all entities and links, together with details of system and user data for each.

   Note: You can expand the list into a tree by clicking the + signs. At first viewing, the **All** option is selected; showing you all entities, whether or not they have errors. (Any entries without a tick or check mark to the right of the checkbox have an error.)
5. Select the **Errors** option to see only problems. In a properly functioning database, the list for Errors should be empty. If there are errors, you see a message below the list as you move the mouse pointer over the errors with an X in a red circle. Typical messages include: `Incorrect schema size and Index missing`.
6. If there are errors, turn on the checkbox for each of the errors that you want the wizard to repair.
7. Click **Next**. The next page displays a list of all errors that you have selected for repair, and the corresponding repairs that the wizard will perform. You can click **Back** to alter your selection or **Cancel** to abandon all changes.
8. Click **Finish** to perform the listed repairs, if any.
9. The wizard performs the repairs and then displays **Close**. Click **Close**.

What to do next

The database is opened, whether or not you asked for any repairs. You might want to check the integrity of links.

Checking the integrity of the links

Use the Link Integrity Check wizard to check the integrity of the link records for a database, that is, whether the data held for links is consistent with that held for the entity records at the ends of the links. You should check the integrity of the database schema before you check link integrity.

About this task

In a properly functioning database, the pages of the Link Integrity Check wizard should be empty. However, issues can occur and the wizard reports any problems found with links, or the entities they reference, and offers some repair options. Most repairs are safe and non-destructive, but some repairs might involve removing invalid data. You can use the list of proposed repairs to further inspect suspect data and perhaps recover it by other means. After repair, you should look at places where the wizard has added entities and links, possibly with blank mandatory fields, and decide how to make these records usable. Eventually, you must repair problems to avoid the possibility of misleading analysis based on faulty data.

© N. Harris Computer Corporation

Procedure

To use the Link Integrity Check wizard:

1. Log on to the relevant security file as a database administrator but do not open the database.
2. In iBase Designer, select **Tools > Database Administration > Link Integrity Check**. The Link Integrity Check wizard opens.

Wizard Page	Action
Step 1	Select a database from the list and click **Next**.
Step 2	Any link IDs listed on this page are missing one of the two link records that hold direction information. You can select list items to view more information that is displayed in the other panes on this page. All listed items are checked for repair by default, you can deselect any record that you do not want the wizard to repair.
Step 3	Any link IDs listed on this page are missing the description records that hold their link attributes. All checked items are repaired with blank data, which is the only possible repair. Deselect any record that you do not want the wizard to repair.
Step 4	Any link IDs listed on this page are missing both of the required records, and are selected by default to be deleted. This is the only repair available. You can choose to uncheck any link IDs that you do not want the wizard to delete, make a note of the details and fix the problem by other means.
Step 5	The entity IDs listed on this page cannot be found in the database, although they should be a link end record. The wizard can repair any missing entity records with blank data, which is the only possible repair.
Step 6	Links should only represent the relationship between two end records. Any link IDs listed here appear to connnect more than two ends, and cannot be repaired. Make a note of these issues for further investigation and fix the problem by other means.
Step 7	Any link IDs listed on this page have inconsistent record status values across the link ends table, and the link's own table. For consistency, the wizard addresses the problem by replacing the

© N. Harris Computer Corporation

Wizard Page	Action
	values in the link ends table with the value in the link table. All link IDs are checked for repair by default, they can be unchecked if required.
Step 8	You can review the list of fixes that you selected in the previous steps, and take the appropriate actions.

3. Click:
   - **Cancel** To abandon all repairs.
   - **Back** If you want to select a different set of repairs in earlier pages of the wizard.
   - **Finish** To perform the listed repairs.
4. Click **Close**. The database is opened, whether or not you asked for any repairs. What you do next depends on whether you repaired errors:
   - If the wizard did not report errors, the database is ready for use.
   - If the wizard reported errors and you chose to repair them, close the database and run the wizard again. (Some errors are only revealed after the wizard has made its first repairs.)
5. Run the wizard again until you see no errors. After two uses with repairs performed, the third use of the wizard should always be error free.

What to do next

After repairs, you might need to add data for any records that are created with blank data fields or to replace removed entities or links, perhaps by importing data from a suitable source.

Managing SQL Server databases

iBase provides the capabilities to store data in Microsoft SQL Server databases and Microsoft Access databases. Microsoft Access should be used as the supporting database only if the number of simultaneous users is five or less. When a database of more than 200 Mb is accessed by a number of users simultaneously then consideration should be given to using SQL Server.

Upgrading an iBase database to SQL Server

You can use iBase Designer to convert a Microsoft Access database to SQL Server format. The upsize process creates an SQL Server database and an `.idb` file that contains the connection details. For details of this process, see Upsizing a Database to SQL Server.

Managing the security of the data in an SQL Server database

iBase supports SQL Server's **Transparent Data Encryption** (TDE) feature. TDE ensures data is encrypted at rest; that is the physical files on the disk where SQL Server stores your data are encrypted. Data in the server's memory or on the network is not encrypted. You can add a layer to protect data in transit over the network with **Transport Layer Security** (TLS). iBase does *not* support the **Always Encrypted** feature, introduced in SQL Server 2016.

For detailed information about configuring the security of the overall system, see Managing access to data and functions.

© N. Harris Computer Corporation

Populating the SQL Server database

If you need to import very large quantities of data, then you might want to consider using bulk import or XML import. Bulk import makes use of the SQL Server BULK INSERT statement and requires the database and server to be configured before it can be used. For further information, see Overview of Bulk Import.

Attention: You cannot use the general SQL Server tools to populate iBase SQL Server databases. The iBase application must have complete control of the data in the database to ensure the integrity of the entities and the links between them. Any data that is not entered or imported by iBase tools can render the whole database corrupted.

Keeping data safe and available (backup)

This is probably the most complex area of managing a database installation, and iBase with SQL Server is no different. SQL Server provides tools for completing the backups and automating them, although your SQL Server administrator might use other backup tools if the right files are backed up at suitable intervals.

With your SQL Server administrator, you must to decide on your backup regime. This can depend on how the iBase SQL Server databases are populated: for example, whether the database is populated by users entering data continuously or by users importing large sets of data. For further information, see Backing Up iBase Databases.

Note: Perform database backups at a time when no users are using the database. This is because some iBase operations can take place over a relatively long time and affect multiple database records. Examples of such operations are data imports, batch edit, batch delete, merge, or deletion of entities with many links. If the backup was performed during such an operation and the database is subsequently restored from the backup the restore operation restores data on which work was in progress at the time the backup was taken and is therefore potentially in an incomplete state. It is safest if backups are completed when no users are performing operations on the database.

Modifying the database schema

Your SQL Server administrator cannot modify the schema of an iBase SQL Server database using SQL. The schema is part of the structure of iBase, and must remain unchanged to ensure data integrity and the success of future upgrades. The only way that you can modify the schema is to use iBase Designer.

Note: It is possible for an SQL Server administrator to modify the indexes of an iBase database to improve performance in areas such as querying although there is a tool for doing this in iBase Designer, see performance tuning section for details. Completing this step manually needs careful planning, and your SQL Server administrator should keep detailed notes and take SQL scripts of the changes to default indexing. Completing this step manually prevents the use of the iBase Designer Performance Tuning wizard.

Note: Before you modify the indexes, your SQL Server administrator must to stop the Microsoft Search service if it is used to continuously update the Full-Text Search indexes in iBase. Other services, such as alerting, are stopped automatically when you open the database in iBase Designer.

Performance tuning in SQL Server

The performance of an iBase SQL Server database can be maintained by regular reindexing in SQL Server. A decline in performance might become apparent after the database grows larger than possibly 10 to 15 GB, and is most noticeable when you run iBase queries. If you are using a database upgraded from iBase 4, then you might be able to improve the performance of queries by optimizing the database indexes. A tool for doing this is available in iBase Designer, see Performance Tuning for details.

© N. Harris Computer Corporation

If you are already using query-optimized indexes (which is the case for databases created or upsized in iBase 5) and query performance is still poor, you need to discuss the problem with your SQL Server administrator. Setting aside issues with hardware and network infrastructure, the decline in performance might occur for various reasons in Microsoft SQL Server:

- Frequent data imports caused the data and indexes to become fragmented
- Databases that are set to grow/shrink automatically on the same disk became fragmented
- Inserting, updating, or deleting large amounts of data caused the SQL Server database statistics to become out-of-date

There are a number of steps that an SQL Server administrator can take to address these problems:

- Data and index fragmentation can be addressed by rebuilding or defragmenting the indexes on the database tables. An SQL Server administrator can do this while the database is online but, for the best results, it is preferable to first take the database offline.
- Operating system fragmentation can be resolved by defragmenting the disk files. This can be done by a server administrator rather than by an SQL Server administrator. It also requires the database to be taken offline so that the files can be moved around the physical disk.
- If automatic statistics updating is disabled, an SQL Server administrator can update them manually.

Effect of auditing on performance

Standard auditing of updates and deletions has a low impact on performance. However, the read auditing that can be configured as an option for iBase SQL Server databases does have an impact. The design of this auditing is such that only records, which have been displayed, charted, or reported are audited. This means that activities such as finding and querying do not run noticeably slower. Activities that result in a revealing a record, such as charting, can take more time to complete. If you intend to use read auditing extensively, it is possible to configure the Audit log database to write to files on disks with fast write performance (see Server machines for details).

Read audit places a higher load on the network and so network performance is more important when using this option. The read audit logs grow relatively quickly and should be archived regularly.

SQL Server Replication and iBase

See Setting Up iBase Database Replication in SQL Server for details of how to set up SQL Server replication for iBase databases.

For more information on hardware requirements, see SQL Server Clients, Servers and Networks.

You can use iBase installation to work with data in both SQL Server and Microsoft Access database formats. This allows you to work with the scale of data appropriate to your analysis. iBase automatically recognizes the type of database and you can switch between them within an iBase session.

Authenticating connections to SQL Server

All users connect to an iBase SQL Server database using the same SQL Server login identifier (ID) and password, which is saved as part of the database properties.

The SQL Server login is used:

- when any iBase user logs on to a security file and opens the database
- when any iBase administrator upsizes a database from Access to SQL Server format, creates a new database or uses the Database Configuration utility

© N. Harris Computer Corporation

The identity of the user attempting to connect is authenticated by using one of the following mechanisms (as defined as part of the SQL Server login):

- SQL Server authentication
- Windows authentication, sometimes called integrated security, where SQL Server accepts the fact that a user has logged on to a Windows domain as sufficient permission to connect to the server. (This is a more secure method than SQL Server authentication because it uses the Kerberos authentication protocol.)

You can also inspect the server and login names in the Database Properties dialog in iBase Designer.

Before you can create or upsize a database, the SQL Server login name and password must be configured in Microsoft SQL Server, for example by your SQL Server administrator. As a minimum, the login must have the `dbcreator` server role.

Creating databases

After creating an iBase SQL Server database, the SQL Server login and password are stored, encrypted, in the connection file (`.idb` file).

It is your choice whether all iBase administrators who create databases use the same SQL Server login and password, or whether each iBase administrator has an individual login. Individual logins make it easier for the SQL Server administrator to trace the owner of a database on the server, so you might prefer this option if several users are likely to create databases.

Changing the SQL Server login after database creation

Because the SQL Server login is used when any user logs on to a security file and opens the database, you might prefer to change the login after you create the database to an SQL Server login with a lower level of permissions or to use Windows authentication instead.

You can do this using the Database Configuration utility. This is a much safer method than changing settings while a database is open, using the Database Properties dialog.

If you choose to change the login that is used to a less powerful one suitable for use by iBase users, you must to ask your SQL Server administrator to grant iBase users permissions on the new database.

Note: You could add this login, which should be mapped to a Windows user group, to the model database. This ensures that members of this group are automatically given database access rights to any database created in iBase.

Upsizing a Database to SQL Server

You can upsize (convert) an iBase Microsoft™ Access database to SQL Server format. You must have a backup of the original database if you want continued access to the Microsoft Access version of the database.

Before you begin

Before you can upsize a Microsoft Access database, you need:

- An iBase logon for the original database with at least the Database Administrator role.
- Exclusive access to the database that you are upsizing.
- A backup of the iBase database that you are upsizing, or sufficient space to make a disk copy if you want the upsize process to make a copy for you.

© N. Harris Computer Corporation

- A printout of the database statistics for the Access database--- you might want to compare these with the statistics of the upsized database.
- The identity (network name) of the server on which Microsoft SQL Server is running.
- The login name and password of an SQL Server user that belongs to the dbcreator server role. For more information, see Managing access to data and functions.
- Sufficient disk space and time to complete the operation.

Note: The upgraded databases use twice the disk space of the original iBase database.

About this task

The upsize process creates an SQL Server database and an `.idb` file, which contains the connection details.

Make sure that you have a backup of the database that you intend to upsize. If this is an operational database, it is a good idea to restore the backup and make sure that you can read the restored version before you complete the upsize.

You can complete the upsize from any iBase client machine. For large databases, however if possible, run this iBase Designer session on the server machine to reduce network traffic.

Note: If you are upsizing any database that is likely to exist already on the server, such as the supplied example database `User Guide.idb`, you may need to rename the original database (`.idb`) file to a name expected to be unique on the server. For example, you might rename the database file `User Guide.idb` to `User GuideAB.idb`. After the upsize is completed, rename the database connection file created by the upsize back to its original name to make sure that any report templates work. For example, you would rename the connection file `User GuideAB.idb` back to User `Guide.idb`.

Procedure

1. Start iBase Designer.

   Note: Do not open any database.
2. Select **File** > **Logon**.
3. In the Security File browser dialog, navigate to the folder and select the security file used to secure the database you are upsizing.
4. Click **Open**.
5. When you successfully log on, click **Cancel** in the i2® iBase dialog. You cannot have the database open when upsizing.

   Note: You might want to open the database briefly, to confirm that you have used the correct security file and, perhaps, from the **File** menu to select `Database Statistics` and view or print the information so that you can compare it with statistics for the database after upsizing. Close the database before you continue. You are now ready to upsize the database:
6. From the **Tools** menu in iBase Designer, select **Database Setup** > **Upsize** > **Database to SQL Server**.
7. Select a database from the list. If necessary, select the entry **More Files** and click **Next** to display a file browser where you can locate the database.
8. Name the backup file or, if you do not want a disk backup file, delete the suggested name to leave an empty field. Click **Next** to continue. A backup is created if required.
9. Enter the name of the server or select it from the **Server** list.

© N. Harris Computer Corporation

Note: Do not use the aliases (**local**) or '.' because they refer to the client machine when the connection file is opened remotely.

10. Enter the logon details for the SQL Server instance on the server. Use SQL Server authentication for the upsizing, not Windows™ authentication. See Authenticating Connections to SQL Server for details.
11. Click **Next** to continue. Your choices are checked and any problems are reported. For example, if the database exists on the server, you must choose another server, or exit and change the name of the original database, before you restart the process. Provided there are no problems, the settings for the new database are displayed.
12. Check that these settings are what you want and click **Finish** The upsizing process starts and progress is displayed by listing each stage with a time and success or failure.
13. Click **Close**. When the upsize process is complete, the iBase database file is overwritten with a file of the same name and extension. For example, `User Guide.idb` is now a connection file to an SQL Server database, and it is likely to be significantly smaller than before. The new SQL Server database is opened automatically.
14. **Optional:** Close the database and change the name of the connection file back to the original database name.
15. Before you use the database, check the database properties to see that the settings in the Configuration and Advanced pages of the Properties dialog are what you expected.
16. If success is reported for all stages of the upsize process, there is no reason to expect problems. It is still wise to check the upsized database as described fully in Checking a Database and summarized here:
    a. Select **Tools** > **Database Administration** > **Schema Integrity Check**. Select the new database and complete each page of the wizard. When you finish the wizard, the database is reopened. Close the database.
    b. Select **Tools** > **Database Administration** > **Link Integrity Check**. Select the new database and complete each page of the wizard. When you finish the wizard, the database is reopened.

What to do next

If you want to use Word Search with the upsized database, you must to rebuild the index. The original index (`.idx`) file is no longer be used by the upsized database. However, it might be required if you plan to allow continued access to the Microsoft Access version of the database.

Note: When users create or upsize to a new SQL database, a Database Access Token is silently generated. Before the Search 360 indexer can be manually run, or a schedule set up, you must change and record the Database Access Token using the database configuration tool. See Managing SQL Server Connection Settings.

Managing SQL Server Connection Settings

You use the Database Configuration utility (iBaseConfig) to manage SQL Server settings held in an iBase connection file (whether a security connection file or a database connection file).

About this task

You can change:

- The name of the server that holds the database
- The server login name and password for all users if SQL Server authentication is used

© N. Harris Computer Corporation

- The security mechanism that is used: SQL Server authentication or Windows™ authentication (integrated security)
- Database Access Tokens

Typically, you use the Database Configuration utility when you use SQL Server tools to change the server instance or login details for existing databases. For example:

- After you create a database, you can change the SQL Server login that is used by the iBase application to one with fewer permissions.
- After you use backup and restore tools to move a database from one server to another, you can reestablish a connection between iBase and SQL Server.

You can inspect many of these details in the Database Properties dialog within iBase or iBase Designer. The advantage of using the Database Configuration utility is that it displays these settings without opening the database on the server, so that you can specify a different server and test the connection.

Note: You must update any copies of the connection files held on other machines. Users are unable to connect to the server if the path or file name is different and see the message: The security file has failed an integrity check. Access is denied.

Procedure

1. In iBase Database Configuration, enter the following details and then click **Next**:

Option	Description
Security File Name	Enter the name of the security (`.ids`) file or the security connection file that secures the database connection file. If you want to change the connection details for a security connection file, leave **Database File Name** blank.
Database File Name	Enter the name of the database connection (`.idb`) file. By entering a database file name, you change the connection details for the database that contains the entity and link data rather than the security data.
User Name, Password	Enter the user name and password of an iBase System Administrator (that is, a member of an iBase database management group with all permissions granted).

When you click **Next**, the connection file is opened, the connection settings are read, and the database and SQL Server information is displayed.

2. You can change many of the settings, for example if you move the database to another server or want to change the method of login to an existing server. However, you cannot change the database type or database name.

- **Server**

© N. Harris Computer Corporation

Specify the name of the server. You must enter a name that can be seen from network client computers. If you are working on the server computer, this means that you cannot choose (local) or its equivalent presentation as a single period (.).

- **Login Name, Password**

  After selecting a server, you must choose the authentication method to be used for connection to the SQL Server instance. You can use either SQL Server or Windows authentication.

  To use SQL Server authentication, enter the SQL Server login name and password. You can enter the details of any user who has the appropriate access rights on the server.

- **Use Windows Authentication**

  To use Windows authentication, turn on the **Use Windows Authentication** checkbox. Each iBase session will log on to the database using the Windows login name with which the user started their Windows session.

Note: The Database Name box displays the name of the Microsoft™ SQL Server database that the connection file (.idb file) connects to. It is not possible to change this name. This prevents a user from connecting to a database where they do not have access by using a connection file for which they do have access permissions.

Click **Test** to check that the details are valid. When you click **Next** the Database Access Tokens are displayed.

3. To create Database Access Tokens, SQL Server users must have `db_owner` database role and `Alter Any Application Role` permissions on the database.

   If you change a token on a database that has Search 360 enabled, you will receive a notification when you click "Generate". You either need to update the Database Access Token in the Configure Database dialog of the iBase Service Configuration tool, or add the new token to the Search 360 Indexer command line arguments.

4. Click **Save** to update the connection file. A summary of its actions is then displayed. A typical summary looks like this:

```
Test connection succeeded.
Server Name
Server Login Name
Server Login Password
Integrated Security setting
Unicode setting
Security access token
Database access token

Completed.
```

Performance Tuning in iBase Designer

iBase automatically indexes certain system tables when an iBase SQL Server database is created or upsized. It will also index those columns within user-defined tables where the

Before you begin

Running Performance Tuning on an SQL Server database requires `VIEW DEFINITION` permission on the SQL Server database. You need to grant this permission to the user mapped to the SQL Server log in. You can use an SQL script similar to this:

```
GRANT VIEW DEFINITION TO username
```

© N. Harris Computer Corporation

For example, if users connect to iBase using Windows™ authentication, and the user who is running Performance Tuning is called iBaseAdmin and is a member of the YourDomain domain:

```
GRANT VIEW DEFINITION TO [YourDomain\iBaseAdmin]
```

You should revoke this permission after you run Performance Tuning:

```
REVOKE VIEW DEFINITION TO username
```

In addition, if present remove the Full-Text Search index. It is not possible to run the Performance Turning wizard while a Full-Text Search index exists.

About this task

Query optimized indexing is of significant benefit even if your database has no user-defined indexes. The index rules are used whenever you:

- Create an SQL Server database.
- Upsize an existing database to SQL Server.

Note: Upsizing removes any indexes that were created manually in Microsoft™ Access.

Procedure

1. In iBase Designer, select **Tools** > **Database Setup** > **Performance Tuning**.

   Note: If necessary, you can stop the process and resume it later. However, until you complete this process, the database is only partially indexed and some parts of iBase might perform slowly. Also, certain commands such as `Schema Integrity Check` do not display, check, or repair the indexes.
2. You can use the Schema Integrity Check to restore missing indexes on user-defined tables in an SQL Server database. In iBase Designer, log on as a database administrator but do not open the database, and from the **Tools** menu, select **Database Administration** > **Schema Integrity Check**. For information on using this dialog, see Checking the integrity of the schema. If the database uses the original iBase index rules, the command restores the indexes to conform to those rules. If the database uses the query optimized index rules, the command restores the indexes to conform to the query optimized rules.

Designing a database

The best design for a database is determined by the type of data you need to capture, and the intended type of analysis.

Before you start to design an iBase database, it is essential to define the requirements for the database, for example:

- What data is to be stored?
- How is that data to be entered?
- How is the data to be used?
- Who will use the database?

An overview of the design process is given below.

© N. Harris Computer Corporation

Entities and links

The modeling and analysis facilities in i2 applications are based on the concepts of entities and links. Entities are real-world objects, the things that are being represented, such as vehicles, people, and addresses.

Links represent relationships between entities, such as owner, associate, and marital status. Entities and links are defined in iBase Designer as entity types and link types. The most significant part of defining the database requirements is to identify the best entity and link types for the data given the type of analysis that is required.

In iBase, each entity and link is represented by a database record.

Gathering database requirements

Consider how the data is entered into the database, as this might influence the database design. There are many ways of adding data to an iBase database.

It is important to consider all the different types of user to find out what tasks they are responsible for, and how they complete these tasks. For example, are the users who enter and check the data different from the users who analyze that data? Do all analysts perform the same types of analysis? Designers can make allowance for the different types of user by designing data sheets. A data sheet is a custom form that is tailored to suit the task performed by the user.

If the database is to hold confidential data, you might need to identify specific user types (defined in iBase as user groups) in order to determine what sort of controls are required to protect the data.

Designers need to know what questions the users want to ask of the data, and also what they expect to produce from the database. Users might need to produce:

- Queries
- Reports
- Charts
- Maps
- Data for export

Knowing how the data is used allows the designer to adapt the schema of the database so that it is possible to import and export data to third-party applications.

Selecting entity and link types

After a designer gathers information about the data to be held in the database, its users, and the types of analysis they perform, the designer is ready to define the entity and link types to represent the data. It is important to understand that there is no right or wrong way to do this.

The best design is the one that allows all the relevant data to be entered as quickly as possible, while also allowing users to complete the tasks they specified during the consultation phase of the requirement gathering process.

Selecting field types

Before you define the entity and link types, the designer needs to examine the format of the data in detail, and decide which field types are most suitable. It is a good idea to get this right before data is entered into the database as there are some limitations on changing between field types.

© N. Harris Computer Corporation

The total number of fields that you can define for an entity or link is 244. System-generated fields, such as Creation Date, account for some of the total. This limits you to a maximum of around 235 fields.

There is a wide range of field types, covering:

- Free text (text typed by the user in whatever format they choose)
- Fixed text (that is selected from the lists of various types)
- Numbers and currencies
- Dates and times
- Documents on your network, on websites and intranets
- Documents and pictures for inclusion in the database
- User information (contact details)
- Fields for use on charts (such as icons)
- Geographical coordinates
- Security information (such as Security Classification codes and cases)

Creating an Entity Type

An entity type defines the name and the default icon of the entity records created in iBase. It usually contains fields to hold information about the entity.

About this task

You can create a new entity type or edit an existing entity type using the New Entity and Entity dialogs.

When you have created an entity type, you may want to add fields specific to the new entity type by creating new fields or copying existing ones. See Creating a field for details.

Procedure

1. Select **New** > **Entity Type**.
2. Specify an icon for the entity type. Either click the name of an icon in the Icon list or enter the full name of a known icon.

   Tip: Double-click on an icon name if you want the entity to have the same name as the icon. Alternatively, edit the text in the **Name** box if you want the entity to have a name different from the icon.
3. Enter a description in the Description box. This can help others to understand what kind of data this particular entity type should be used for, and therefore ensure that a suitable entity type is used when adding information to the database.
4. Deselect **Select in 'Expand' list**.

   Note: Select in 'Expand' list determines whether an item type is included in expand operations on Analyst's Notebook charts. Deselecting **Select in 'Expand' list**, prevents the item type from being added to the chart as the result of a default expand. If you would like to expand an item type that isn't added to the default expand operation, you must select the item type in an Analyst's Notebook **Expand with settings** operation.
5. Assign a semantic type to the entity type. For further information, see Assigning a semantic type.
6. Click **OK** to save your choice.

© N. Harris Computer Corporation

Creating a Link Type

A link type defines the name and the default color of the link records created in iBase. It might also restrict the entity types allowed at the end of the link.

About this task

A link type usually contains fields to hold information about the link. For example, you might want to create a link type named Telephone Owner, with color Blue, and restricted to use with one of the link end entities being Telephone. If, for example, you chart iBase data in Analyst's Notebook you might also want to assign a suitable semantic type for it.

You can create a new link type or edit an existing entity type using the New Link and Link dialogs.

After creating a link type, you can add fields specific to the new link type by creating new fields or copying existing ones. See Field Types for details.

Procedure

1. Select **New > Link Type**.
2. Click the **Details** tab to display the **Details** page.
3. Enter the name for the link type.
4. Enter a description. This can help to indicate the type of data this particular link type should be used for.
5. Select a link **Color**.
6. In most cases, you should leave the Select in 'Expand' list checkbox turned on. As for entity types, this checkbox controls an initial setting when viewing charts sent from iBase to Analyst's Notebook, which users can change easily.
7. Modify the information about the ends that can be used with the link type by clicking the **End Types** tab to display the End Types page and turning off the checkboxes for any entity types that you want to make invalid.

   Note: You can leave all settings on the End Types page unchanged. If you do make changes to the settings on the End Types page, you are restricting the entities that the link type will allow to be connected at either end of the link type. It is not meaningful to turn off all checkboxes in a list, that makes it impossible to create one end of the link. If you try to do this, iBase allows all entity types at that end.
8. If required, assign a semantic type to the link type. For further information, see Assigning Semantic Types to your data.
9. Click **OK** to save the link type.

Results

When you click **OK** you are returned to the Database window where the new link type is listed in the Database Explorer under Link Types. The right pane of the window displays the color and name of the link type and a list of any standard fields that already exist for the link.

© N. Harris Computer Corporation

Creating a field

You can create or edit a field that is specific to an entity or link type, or a standard field that is common to all entity and link types. The options that are available are specific to the type of field that is being created.

Before you begin

Examine the format of your data in detail, and decide which field types are most suitable. The total number of fields that you can define for an entity or link is 244. System-generated fields, such as Creation Date, account for some of the total. This limits you to a maximum of around 235 fields.

Procedure

1. Select whether to create a field that is specific to an entity or link type, or a standard field that is common to all entity and link types:
   a. To create a field for a specific entity or link type, in the left pane, right-click the entity or link type and select **New Field**.
   b. To create a standard field, in the left pane, double-click **Standard Fields**.
2. Enter a name for the field. The maximum length of a name is 50 characters. The field names within each entity or link type must be unique, and if the field is a standard field, the name must be unique to the database.
3. Enter a description in the Description box. This can help others to understand what kind of data this particular field should be used for, and therefore ensure that the right sort of data is added to the database.
4. From the **Type** list, select the type of field. See Field Types for information on the different field types.
5. If appropriate:
   a. Select the maximum number of characters you want to allow in an Text field. You can set any value in the range 1 through 255.
   b. Enter a display format. Select from the **Display Format** list, or click in the box and enter a format. For details of all the formats, see Field Types.
   c. Select a **Default Value**. This is an initial suggested value for the field. You can enter a different value, either as a fixed value or as a code for data such as the creation date of the record.
   d. For a calculated field type (which is a field based on the content of another field), click **Define** to open a dialog for defining the calculation, or selection of data to show in this field. For example, you can choose to display the day of the week corresponding to a date field.
   e. Specify a Chart Attribute for the field. You can use any attributes you have already defined for this database or define a new chart attribute, by clicking New to display the Creating chart attributes.
6. Where appropriate, use these options to control how iBase will use this field:

Option	Description
Indexed	Turn on this checkbox to create an index on the field. This can also increase the speed of searching in this field. Do not index Yes/No fields or fields with less than five allowed values. (**Note:** Not all field types can be indexed.

© N. Harris Computer Corporation

Option	Description
	These include calculated fields, hyperlink fields, Multi-Line Text fields and system fields such as Create Date. A maximum of 31 fields can be indexed per field type).
Mandatory	Turn on this checkbox to force users to enter a value in this field when creating or editing the relevant entity or link. In iBase, mandatory fields are displayed with a blue label.
Discriminator	Turn on this checkbox to mark this field as one that marks a record as unique, as an aid to avoiding record duplication. If there are several discriminator fields, it is the combination of their values that must be unique.
Characteristic	Turn on this checkbox to make the field a characteristic field. For example, the color and style of a vehicle may be a characteristic that is useful when finding vehicles in the Matching Records dialog.

7. Enter a short description of the field. The description appears as a tooltip for the field.
8. If required, assign a semantic type to the field. This displays the Select Semantic Type dialog. For further information, see Assigning Semantic Types to your data.

   Note: You must assign each standard field to a semantic type that is unique to the database. Other fields can be assigned to any semantic type provided that it is unique to the entity or link type.
9. Click **OK** to save the field.

Results

The field is created.

If you later edit a field, there are some limitations on what you can change. For example, you are unable to change the type of a Currency field, and you can change a Text field only to a field of a related type, including Multi-Line Text and Suggested from Code List.

If you make an existing text field mandatory and there are any existing records with this field blank, iBase Designer fills those record fields with a single hyphen (or minus) character, that is, the value "-".

In this context, text fields are:

- Hyperlink
- Selected/Suggested from Code List
- Text

Note: To prevent data loss, you cannot reduce the size of a field. The only way to do this is export the data to be truncated along with the record ID, then delete the field and create a new one of the correct size. Before importing the data using record ID matching, you must ensure that the data is the correct size for the target field. For further information, see Importing and exporting data.

© N. Harris Computer Corporation

Assigning a semantic type

To benefit from the visualization and advanced analysis capabilities of other i2 applications, such as Analyst's Notebook, you can assign relevant data with a semantic type that identifies the real-world content of the data.

About this task

There are two ways of assigning semantic types. You can either assign a single semantic type when adding or editing entity types, link types, fields or standard fields by using the Select Semantic Type For dialogs (as described below), or you can assign semantic types to all the items in the database schema by using the Semantic Types dialog. See Assigning Semantic Types to your data for more information.

You must log on as a database administrator in order to assign semantic types.

Note: When you are assigning semantic types to fields, you cannot assign the same semantic type to more than one field in the same entity or link type.

Note: You cannot unassign the semantic type of an item in this dialog. You can only assign a different semantic type. To unassign a semantic type, use the Semantic Types dialog. Select **Tools > Database Design > Semantic Types**.

Procedure

1. If this is the first time you have assigned semantic types in this database, you should load the Semantic Type Library for your organization. See Loading the semantic type library for details.
2. Select an Entity Type, Link Type, Field, or Standard Field,
3. In the Semantic Type area, open **Select Semantic Type For**. The **Search Available Semantic Types** box displays the name of the current entity type, link type, or field. The Ordered Results area suggests some semantic types that may be suitable for assigning to it, based on a comparison of its name with the name of the semantic type and any synonyms set up for it. You can review the suggested semantic types by clicking on a result to display information on the right.
4. If none of the semantic types in the Ordered Results area are suitable, you can search the library. There are two ways of doing this:
   - In the **Search Available Semantic Types** box, enter the word or phrase that you want to search on. As you type, possible matches are displayed in the Ordered Results area.
   - Browse the semantic types displayed in the tree view.

   For detailed information, see Searching for semantic types.

   Note: If none of the semantic types are suitable, and you are working in the database that contains the central semantic type library for your organization, you can create a custom semantic type. For important information on creating custom semantic types, see Defining custom semantic types and Maintaining the semantic type library.
5. When you have located the correct semantic type:
   a. Select it and click **OK** to return to the Entity Type, Link Type, or Field dialog, which displays the selected semantic type.
   b. Click **OK** to save your changes. The assignment is not completed until you click **OK**. To cancel the assignment, click **Cancel**.

© N. Harris Computer Corporation

Creating labeling schemes

Labeling schemes determine how the label that identifies and represents a record is derived from the record's fields. For example, person records may have a label comprising the 'Last Name' field together with the 'First Name' field, while for vehicle records the label may comprise the 'Manufacturer', 'Model' and 'Registered Number' fields

About this task

You can also include 'free text' in the label. This is text that does not vary between labels.

There are two separate label definitions in each labeling scheme; one for the label to be used within iBase, and one to be used when a record is added to an Analyst's Notebook chart.

Each labeling scheme covers all the entity and link types in the database. If there is more than one labeling scheme it is because you might want different label formats at different times. You specify which scheme is in use by making it the 'default'.

Note: For certain entity types, Smart Matching in Analyst's Notebook assumes a property semantic type for the label value. Consider that for some entity types, the assumed property semantic type is a Details kind. The label is then parsed during Smart Matching to locate the various parts of the textual value. Do not assign a Details kind of semantic type to a property that will be used for a label. Instead, specify a property that is a part of a Details type, and assign the semantic type with the correct specific meaning. For example, for a Credit Card, specify the label to be the Card Number property, which is a part that is located in a Bank Card Details property.

Procedure

1. Select **New** > **Labeling Scheme**.
2. Select an Entity or Link Type from the list.
3. Display the page for the type of label you want to define; the Standard page for the label to be used with iBase records, the Chart page for the label to be used for chart items.
4. In the Fields list click to select fields you want to include in the label. Then click **Add** to add them to the Label Parts list.
5. In the **Free Text** box, type any text that you want to appear in all the labels (without variation). Then click **Add** to move it to the Label Parts list.
6. If required, in the Label Parts list, click to select an item then:
   - Repeat the previous steps. The new items are inserted below the selected one.
   - Click **Space** to add a space between label parts.
   - Use the up or down arrow key to change the selected item's position.
   - Click **Delete** to remove the selected item.
7. Preview your label in the Sample box. You can change which record the preview is based on by clicking **Next** and **Previous**. Repeat these steps until you have defined standard and chart labels for all the required entity and link types.

What to do next

You might now want to set your labeling scheme as the default:

1. Find the scheme in the Database Explorer.
2. Right-click the scheme and select **Set as Default**.

© N. Harris Computer Corporation

Note: A charting scheme can override the default labeling scheme.

Setting valid end types

The entities allowed at each end of a link can be restricted to one or more entity types. This helps users to enter data consistently, and improves the effectiveness of searches and data analysis.

Restricting the end types that are valid for each link type forces users to enter data consistently. This can bring many advantages for users when searching and analyzing the data. For example, users can:

- Set the end types of the links they want to browse.
- Build queries using links with specific combinations of end type.
- Report on links with specific combinations of end type.
- Import and export link data with selected end types.

Some examples of links with restricted end types, in the example User Guide Database, are:

- A Relationship link type may have a Person entity type for each end of the link.
- An Owner link type may have Telephone and Vehicle entity types as possible End 1 types and Organization and Person entity types as possible End 2 types.
- A Location link type may have only one entity type at one end (Address) but several possible entity types at the other end, such as Organization, Person, Vehicle.

There are two ways of reviewing and setting the entities used as end types on links:

- For single link types, use the Link Type dialog.
- For all link types in the current database, use the Valid End Types dialog (described below). This also enables you to find out how link end types are used in the database and to apply this usage to the database design. If you change the valid end types for a database that already contains data, you can identify which link types have invalid end types by using an Exceptions report. See below Viewing exceptions and correcting links with invalid end types.

Displaying valid end types for a link type

In the Valid End Types dialog, choose a link type from the Link Types list. The end types currently defined in the database design are shown.

The end types are sorted in alphabetical order. To sort a list with the valid end types at the top, turn on the Sort by checked checkbox.

Finding out how link end types are used in the database

To see a list of the available end types for the selected link type as used in the database, click Usage. To compare this with the end types defined in the database design, move the Used End Types dialog so that you can see the end type lists in the Valid End Types dialog.

You have two choices:

- To update the design based on usage, click **Replace**. This replaces all the end type combinations, for the selected link, in the database design with those used in the database
- To retain the existing design, click **Close**. This retains the existing database design, and you can then identify any link types with invalid end types by generating an Exceptions report (click **Exceptions** in the Valid End Types dialog).

© N. Harris Computer Corporation

Defining valid end types

In the Valid End Types dialog, the End 1 Types and End 2 Types lists show all the entity types defined in the database.

Note: Although these are called End 1 and End 2 this does not imply direction.

There are two ways of defining valid end types. You can:

- Click **Usage** to display the Used End Types dialog, review how the end types are used in the database, and then click **Replace** to replace the end types in the database design with the end types as used in the database. You can then adjust the end type combinations further if required.
- In the Valid End Types dialog, turn on or off the checkboxes against the entity types that you want to include as valid or exclude as invalid.

When you have set the end types that you require at each end of the link type, click **Apply** to save the changes, or click **Restore** to cancel.

Restoring or applying changes

You can cancel the changes you have made to a link type by clicking **Restore**. You will lose any changes that you made since you last clicked **Apply**.

After you click **Apply**, you cannot restore the original end types.

Viewing exceptions and correcting links with invalid end types

Any end type combinations that are not in the database design are invalid, and might limit how users can analyze and search links. For example, in a query, users will be forced to search using the link type { Any} rather than on specific end types.

You can identify which link types in the database are used with invalid end types by producing an Exceptions report: in the Valid End Types dialog, click **Exceptions**.

To add the exceptions to a set, right-click any of the branches in the tree view, for example on **Exceptions** or on one of the link types, and from the shortcut menu, select **Add to Set**. This can be useful to users who will need to correct their data.

Note: The Exceptions totals include soft deleted records, and these will be included in any sets which you create. Soft deleted records are only seen by SYSADMIN users. You may want to correct soft deleted records in order to prevent a problem from occurring where uncorrected soft deleted records are restored with invalid end types.

Reserved words

Words that are used to define iBase structures such as column names within the database tables are reserved. This means that in order to prevent potential issues when commands are run, these words cannot be used to define other objects such as item types or fields.

Each object type in the database has a different list of words that are reserved. For example, the list of reserved words for entity types, include: unique id, create date, and record status.

© N. Harris Computer Corporation

Creating chart attributes

A chart attribute is attached to a field of an entity or link type and can be used to provide a value from the field in a form suitable for use in a chart. You can create a new chart attribute or edit an existing one using the Chart Attribute dialog.

About this task

A chart attribute can include, in this order, any combination of:

- a symbol
- prefix text
- the value of the field
- suffix text

You can also set other configuration options that allow you to specify how attributes behave during merge or paste operations on a chart.

Once you have created a chart attribute, you must use that attribute in the definition of one or more fields before the attribute becomes available to Analyst's Notebook.

The use of chart attributes also depends on the user's settings in iBase, including the Charting page of the Options dialog and the definition of charting schemes.

The final choice of how iBase data appears in a chart depends on user settings in Analyst's Notebook.

Procedure

1. Select **New > Chart Attribute**.
2. Enter a name for the chart attribute in the **Name** box. You must name each chart attribute. The maximum length of a name is 50 characters.
3. From the **Type** list, select the type of chart attribute. The available types are: Flag, Numeric, Text, and Time.
4. In the Display area of the dialog, use the controls to construct and format the attribute. The box at lower right of this area displays a preview of the attribute using example field values.
5. Turn on the **Symbol** checkbox to include a symbol as part of the attribute, click **Browse** and select a symbol from the palette.
6. Turn on the **Prefix** checkbox to place a short piece of text after any symbol and in front of the attribute value. Type a short description in the text box. End the description with a space character if you want to separate the description from following parts of the attribute.
7. Turn on the **Suffix** checkbox to place a short piece of text after all other parts of the attribute. Type a short description in the text box. Start with a space character if you want to separate the description from earlier parts of the attribute.
8. Turn on the **Show On Chart** checkbox to have Analyst's Notebook display the attribute by default. The user can choose to override this setting on a case by case basis within iBase or Analyst's Notebook.
9. Turn on the **Value** checkbox to include the value of the associated field as part of the attribute.

The remainder of the Display area depends on the type of attribute being defined. It may be empty or contain formatting controls such as the displayed number of decimal places for Numeric attributes, or the date or time parts of Time attributes.

© N. Harris Computer Corporation

1. In the Behavior area, select options from the lists to define the attribute value that is the result of pasting or merging attributes when working in Analyst's Notebook. The operations are:
   - **On Paste** select what happens when users paste the entity or link.
   - **On Merge** select what happens when users merge the entity or link.
2. Click **OK** to save the chart attribute.

Adding Security Classification Codes

Security Classification Code lists (SCC Lists) can be used in conjunction with Data Access Control groups to manage the access levels of records. In addition to the standard lists that are available within iBase, you can create your own lists and values that match the organization that you work for.

About this task

Security Classification Code lists:

- Determine the values available when users select from an Security Classification Code field type.
- If Data Access Control groups are used with SQL Server databases, determine the access to records that are assigned a specific security classification code.

SCC lists and SCC fields do not on their own enforce security. To enforce security based on the value in an SCC field, you need to add Data Access Control groups, assign users to those groups and then deny access to the SCC values using the Data Access Control command. See Setting up Data Access Control groups for details

Procedure

1. Select **New** > **Code List** > **SCC List**
2. Enter a name for the list. This is just the name of the list and should be different from the name of the field that will use the list.
3. Enter a description. Use this to give a useful hint on where the list is used, for example.
4. For the items in the list:
   - Click to select an item in the Items list. Click again to start editing the text. Do not edit items that are used in data records.
   - Delete a selected item by right-clicking and from the shortcut menu, selecting **Delete Item.** Do not delete items that are used in data records.
   - Add a new item using the blank item at the bottom of the list.
5. Click **OK** to confirm your changes and close the dialog.

Viewing the number of records in a database

The number of records of each of the entity and link types are recorded in the Database Statistics. You can use this view to report on the structure of your data.

Procedure

1. Click **File** > **Database Statistics**.
2. Click **Print** to print the contents.

© N. Harris Computer Corporation

Viewing the database design report

You can use the database design report to view information about your database design. The database design report includes details about the design of the database. For example, database statistics, entity and link type fields, code lists, and semantic types.

About this task

The database design report summarizes the design of a database that you have access to. To gain a full view of the database design, you must ensure that you run the report with the correct access permissions.

Procedure

1. To generate a report, click **File** > **Database Design Report**.
2. With the report open, you can:
   - Browse the pages.
   - Refresh to ensure that the latest information is captured.
   - Print the report.
   - Export the report to Microsoft™ Excel, PDF, or Microsoft Word.

Code lists

Code lists are the collective term for the pick lists, icon lists, and Security Classification Code lists that are stored within the database. These lists contain specific values that increase the consistency of information that is entered into the system.

Pick lists

Pick lists determine the values available for selection when using a Suggested From Code List type field or a Selected From Code List type field. You can restrict who is allowed to edit pick lists.

There are two versions of the Pick List dialog, depending on whether the list is filtered or not. Filtered lists allow you to assign a group of values to one or more values in a parent list. See Editing filtered pick lists.

Pick lists, and the fields they apply to, are defined in the database design. Filtered lists, where one list is assigned as the parent of another, are also set up in iBase Designer.

Note: Some similar fields may use different code lists. The name of a list may indicate what field(s) it is used by. For example, a Hair Color list might be used for a Hair Color field in Person entity type records, and a Vehicle Color list might be used for the Vehicle Color field of a Vehicle entity type. Although both lists contain lists of colors they are separate pick lists.

Creating pick lists

Create a new empty list and add the values one at a time. This is suitable for lists that will contain only a small number of items, or where you do not have a file containing the values you want to use.

Procedure

1. Select **New** > **Code List** > **Pick List**.
2. In the **Name** box, type the name for the pick list. Optionally, type some text to provide more information about the pick list in the **Description** box.

© N. Harris Computer Corporation

3. To add items to the list, enter an item value and, optionally, a description in the first row. As soon as you type (or paste some text) another blank row is automatically created. To add further items, do one of the following:
   - Click on the row below where you want the new item, and then click **Insert Row**. Enter the name and description in the new blank row.
   - Click the blank item at the bottom of the list. This adds a new item at the end of the list. You can then use the **Move** buttons to move the item to where you want it in the list.
4. Continue to add items, then sort the list as required, by moving items up or down the list, or by sorting the entire list alphabetically. Item values must be unique.
5. When you are happy with the contents and the order of the list values, click **OK**.
6. Assign the pick list to the required field in an entity type or link type. See Creating a field.

Editing pick lists

You can edit item names and items descriptions, add and remove items and change the order in which the items are displayed. You can also edit the name and description of the pick list itself. The pick list description is only visible when editing a pick list.

Procedure

1. In the **Items** list, click to select an item.
2. To edit the item name, click in the **Value** column and enter the name.
3. To edit the item description, click in the **Description** column and enter the description. Item descriptions are optional but can help you to choose the right value when entering data in a record.
4. To add a new item, do one of the following:
   - Click on the row below where you want the new item, and then click **Insert Row**. Enter the name and description in the new blank row.
   - Click the blank item at the bottom of the list. This adds a new item at the end of the list. You can then use the **Move** buttons to move the item to where you want it in the list.
5. To delete an item, right-click and select **Delete**.
6. Click **OK**.

   Note: Other users who are logged on to the database will continue to see the old list until they close and reopen the database, or display the Pick List dialog and then click **OK** to close it (which updates the contents of all the lists).

Importing pick lists

If you have a large list of values that you would like to use in multiple lists, or a list that has been exported from an external source, you can import pick list values. Importing pick list values can reduce both entry time and errors.

About this task

An import file uses the following format, where the description is optional:

```
[child list name] <tab> description <tab> [parent list name]
child item name <tab> description <tab> parent item name
```

Note: To see an example of a text file containing valid pick list data, you can export a pick list in the database and view the file which is created. You could then use this file as a template for other pick lists

© N. Harris Computer Corporation

to be imported. See Exporting code lists. The number of tab-separated values must be the same for each row in the file. Two tab characters are required if there is no description.

Procedure

1. In the left Database window, select a pick list, right-click, and select **Import**.

   The Import Code List dialog is displayed, in which you select a file containing the pick list data. Only text files (with a `.txt` extension) can be selected. If your file has a different extension, you must change it to `.txt` before it can be imported.
2. Select the required file and click **Open**.

   After choosing a suitable text file, you need to define how you import the list. There can be zero, one, or more pick lists defined in a text file, and the dialog shows you what is defined in the file and, if there are valid lists, offers you options for importing each list.

   Information about the number of pick lists in the selected text file and the name of each pick list to be imported is displayed in the Import Code List dialog.

   Each pick list is imported in the order in which it is found in the text file. For each list, you have the option to import it or skip the import and move to the next list.

Pick list example files

Simple example

Each pick list is identified in the text file by a header row, which contains the pick list name in square brackets. Each item in the pick list is separated from the next by a new line:

```
[Vehicle Type]
Airplane
Bus
Car
Container ship
Ferry
Hovercraft
Minibus
Motorcycle
```

In the simple example above, a pick list called Vehicle Type will be imported. The list ends when another header row is found, or at the end of the text file.

Simple example with description

If the pick list has a description, it follows the name as a tab-separated value in the same row:

```
[Vehicle Type] Basic category of vehicle
Airplane
Bus
Car
Container ship
Ferry
Hovercraft
Minibus
Motorcycle
```

Note: If the pick list has no description, then the date it was imported will be used as the description.

© N. Harris Computer Corporation

Simple example with list item descriptions

If the items in the pick list have a description, the description follows the item name as a tab-separated value in the same row:

```
[Vehicle Type] Basic category of vehicle
Airplane Comercial or freight airliner
Bus Public passenger vehicle
Car
Container ship Goods carrying vessel
Ferry Passenger vessel
Hovercraft
Minibus Max. 20 passenger occupancy
Motorcycle Two or three wheeled vehicle
```

Note: All the items do not have to have a description. You can always add a description or edit the description later.

Example with parent list assignments

To import a filtered pick list (that is, a pick list which is assigned a parent list), the header row must also contain the name of the parent list in square brackets:

```
[Vehicle Style] Current vehicle type [Vehicle Type]
Business jet Seats a maximum of 9 people Airplane
Single-engine piston Small one or two person carriers Airplane
Amphibian Supports both aquatic and land based landing Airplane
Commercial Standard passenger plane Airplane
Coach a single decker Bus
Double-Decker a double decker Bus
```

In the above example [Vehicle Type] is the name of the parent list, and the first four items are assigned to the Airplane parent list item.

Filtered pick lists

Filtered pick lists aid data entry by creating a parent-child relationship between two consecutive pick list fields in a form or datasheet in iBase so that the selection of a value in the parent list limits the values that can be selected by the user in the child list to only those that are relevant or suitable. This can speed up data entry and ensure consistency in the database.

You define the parent-child relationship between two pick lists in iBase Designer and then assign groups of items in a child list to each item in the parent list. See Editing filtered pick lists.

Note: The assignment of items can also be created and edited in iBase by any user with sufficient permissions.

The child list can be thought of as an amalgamation of several sublists, each one relevant for a single item in the parent list.

You can arrange filtered pick lists into a hierarchy of any number of levels, such that the value selected in the first list filters the available values in the second list, and the selected value in the second list then filters the values available in the third, and so on. It is most common for filtered pick lists to consist of just a parent and a child.

Note: You can use a child pick list in an entity type, link type or datasheet without its parent list. The pick list behaves as an ordinary pick list, with all its values available for selection, sorted alphabetically. In addition two consecutive pick lists in a form or datasheet will only function as parent and child, if you explicitly set them up to behave in that way in iBase Designer.

© N. Harris Computer Corporation

Creating filtered pick lists

A filtered pick list allows you to set the values in the child pick list based on the selection in the parent list. You can create filtered pick lists manually, or import formatted lists.

About this task

Filtered pick lists can either be created manually or imported from a text file. See Importing pick lists.

Procedure

1. Create the list that will become the parent list.
2. Create the list that will be the child list:
   a. Right-click on the parent list, and select **Create Child**. A blank Pick List dialog is displayed with the parent list already assigned.
   b. Enter the name and each child item. See Creating a child pick list.
3. Assign the child pick list to its parent list. See Assigning values to a parent pick list.
4. Open the child pick list and assign each group of values to an item in the parent list. You can at this stage add and remove items.
5. Assign both pick lists to the required fields. You need to decide whether the pick lists are Suggested From Code Lists (which allow values other than those in the list to be entered) or Selected From Code Lists (in which only the values in the list are valid for selection).
6. Make sure that the fields are arranged in the entity type, link type or datasheet so that the field using the parent pick list is directly above the field using the child list.

What to do next

Once you have set up your parent-child pick lists, you should test them in iBase.

To do this, close the database in iBase Designer and open the same database in iBase. Select the entity type or link type to which you have added the parent and child pick lists and create a new record.

Editing filtered pick lists

Filtered pick lists improve data entry by allowing available values in one pick list to be filtered by the selection in a pick list above it. After you have set up filtered pick lists, you can manage the contents to match your needs.

About this task

If you would like to modify the contents of a filtered pick list, remember:

- Changing the name or description of any item in a parent pick list will change it in the Pick List dialog when editing its child pick list.
- Deleting an item in a parent list to which items in a child list have been assigned will result in those items being unassigned.
- Duplicates are not allowed in parent lists

Note: Pick lists can also be edited by users in iBase but you can restrict who is allowed to edit pick lists.

Procedure

1. Select **Code Lists** > **Pick Lists**

© N. Harris Computer Corporation

2. Right-click on the pick list you want to edit, and from the shortcut menu, select **Edit**. You can also double-click on the pick list. The Pick List dialog is displayed.

Creating a child pick list

You can create a pick list to offer the user a set of related or more detailed values for each item in an existing list.

About this task

You can create a pick list to offer the user a set of related or more detailed values for each item in an existing list. For example, the Vehicle Manufacturer pick list may have a child list of Vehicle Model, which allows the user to record a more refined level of information about a vehicle - not just the make of a car, but the particular model. The Account Type pick list may have a child list of currency values, to control which particular currencies are valid for a selected account type.

In both cases, the child list values available for selection are filtered by the selection of a value in the parent list. The two lists need to be assigned to consecutive fields in the entity/link type field list or datasheet for this behavior. For more information, see Filtered pick lists.

You cannot create a child list for a pick list which itself has duplicate items. A pick list with duplicate items must be a child list itself.

You can enter all the values for a single parent item, and assign them all as a group. To do this, when you have entered all the values, click on the square at the top left of the Items area. This selects all the items in the list.

You can then assign all the items to the required parent item.

Alternatively, select the parent item in the list on the left and then enter all the items on the right side of the dialog. Items entered will automatically be assigned to the selected parent item.

Procedure

1. Display the list of pick lists by clicking the plus symbol next to **Code Lists** then select **Pick Lists**.
2. In the right side of the dialog, right-click on the list for which you want to create a child list, and from the shortcut menu, select **Create Child**.
3. In the right side of the dialog, enter the first item. Add a description if required. Press `Enter` to enter the item and move to the next row.
4. Click **Assign** to assign this value to one or more values in the parent list. The Parent List area is updated to show the number of items assigned to each parent as you continue to create the list.
5. When you have finished creating the list, click **OK**.

Assigning values to a parent pick list

If the list you are editing has a parent list, you can assign each entry to one or more values in the parent list. Each assigned entry shows the name of the parent list, and also the number of assigned items for each item within the parent list.

About this task

When you open a pick list, you can see how many items have been assigned, and to which parent items, in the panel on the left.

- To view all the items in the list, click <All>.
- To view only those items that have not been assigned to any parent item, click <Unassigned>.

© N. Harris Computer Corporation

Procedure

1. Click **Code Lists > Pick Lists**, select a pick list, right click and select **Edit**.
2. To assign values to an item in the parent list: Select an item in the list on the right. Click anywhere in the row to select it. It is usually easier to find the items to assign when the list is filtered to show only Unassigned items.

   Tip: To select more than one item, click on the square on the left of the row, and hold down `Ctrl` while selecting the items you want.
3. Click **Assign**.
4. The Assign Parent Items dialog is displayed. Turn on the checkboxes for the parent items to which you want to assign the selected values. You can select more than one parent item. Click each parent item checkbox to assign the items to the selection.
5. Click **OK**.

   Note: When you assign an item to more than one parent, a duplicate item is created in the list. If you assign an item to 10 parent items, 9 duplicates will be created. The total number of items, displayed in the Pick List dialog, will be updated accordingly.

   The description for all the duplicate items will be the same. When you assign an item to more than one parent item, the original description is inherited by all the other items. Changing the description for any one item will update that description for all the duplicate items.

Editing icon Lists

For certain types of entity, you might want to allow the icon to be selected from a list rather than providing a single icon. You can edit icon lists to determine which icons you can select when using an Icon type field.

About this task

Not all icon fields use the same list. The name of a list should indicate the intended use. For example, the 'Crime Icon' list might be the list for the 'Icon' field of 'Crime' entity type records.

The icon lists, and the fields they apply to, are defined in the database design. Your system administrator may restrict who is allowed to edit icon lists. The Name can only be changed in iBase Designer.

In the Icon List dialog, the left list contains all the available icons from the Icon List file as set up by the system administrator. The right list contains the icons in this icon list.

Procedure

1. Select **New** > **Code List** > **Icon List**.
2. Enter or edit the description.
3. Click to select an icon in the left Items list and note its preview to the right of this list. If you want it to be available in the icon list click **Add** to move it to the right list.
4. Click to select an icon in the right list and note its preview to the left of the list.

   Tip: It you want to remove it from this icon list click **Remove** to move it to the left list.
5. Click **OK**.

© N. Harris Computer Corporation

Exporting code lists

Code lists can be exported for use either in other iBase databases, or other applications. You can export all code list types, whether a pick list, icon list or Security Classification Code (SCC) list.

Procedure

1. To export a code list, right-click on it in the left or right pane of the Database window, and click **Export**.
2. In the Export Code List dialog, choose a name and location for the file. Exported code lists are saved as text files. The default for the file name will match the name of the list being exported.

   Note: When you export a pick list which is part of a hierarchy, you are given the option to export all the pick lists in the hierarchy, or just the pick list itself.

Viewing changes to code lists

A record of the changes to code lists can be kept.

With a pick list, icon list, or SCC list open, click **History** to view the changes to the list. These changes include:

- Values
- Descriptions
- Parent lists, for filtered pick lists.

All the changes that are made in the same session are grouped by username, date, and time. Current® updates to the list are not shown - you need to confirm your changes by clicking **OK** before the changes are logged as part of the audit history.

You can print the list or save it as a Microsoft™ Excel spreadsheet or PDF file.

Note:

To view the audit history, the database must be an SQL Server database and the audit history must be turned on. To find out whether the database logs audit history, select **File** > **Database Properties** and check the setting of the **Audit History**.

Viewing changes to code lists requires that the Report Viewer is installed.

Designing datasheets

You can set up datasheets so that the fields for a particular entity are in a specific format. You can create or edit datasheets that contain just one entity (the main entity) or with a main entity and links to one or more linked entities.

About this task

Datasheets are designed and managed in iBase Designer.

When opened in iBase Designer, all datasheets contain controls that are always displayed:

Area	Description
Name	The name of the datasheet.

© N. Harris Computer Corporation

Area	Description
Page Style	Determines whether fields are displayed on a single page (Standard), or multiple pages that group information accessed on tabs or hyperlinks.
Form type	If you turn on **Use this form in place of the standard 'Show' form**, the datasheet is used as the default display for items of this type.

The available fields for the datasheet are found in lists:

- **Fields** - fields that are available but haven't been assigned to a page of the datasheet.
- **Selected Fields** - the fields that are displayed on the selected page of the datasheet.

Procedure

1. Select **Tools** > **Datasheet Manager**.
2. Click **New** to display the **Datasheet Designer** with a blank sheet.
3. Enter a **Name** to identify the datasheet.
4. Select the **Page style**:
   - **Standard** - Fields for the main entry are displayed on a single page, with a separate page for the details of each link.
   - **Tabbed** - Selected fields can be placed on tabs to help group information.
   - **Hyperlink** - Selected fields can be placed on hyperlinks to help group information.
5. Define the main entity:
   a. From the **Entity Type** list, select the entity type. When you first select an entity, the **Fields list** contains all non-mandatory fields that are associated with the entity type and the **Selected Fields** list contains all the mandatory fields.

      Note: If you select the Tabbed or Hyperlink style, you can split the Selected Fields list; each sublist is displayed in iBase as a separate tabbed page of the datasheet.

   b. In the **Selected Field** list, select the fields to display and the order in which to display them.
   c. Click **OK** to save the datasheet or click the Links tab to add one or more links to the datasheet.
6. You can split fields into pages, which appear to the user as tabbed or hyperlinked pages. For more information, see Creating datasheet pages.
7. You can add the details of a linked entity to your datasheet. For more information, see Adding a link in a datasheet.
8. Click **OK** to save the datasheet.

What to do next

To verify that the datasheet is correct when used in iBase:

1. Close the database in iBase Designer. You do not need to log off.
2. Start iBase and open the database.
3. Select **New** > **Datasheet** > **name**(where name is the name of your datasheet).

© N. Harris Computer Corporation

4. Test that the datasheet looks and works as expected.

Creating datasheet pages

Datasheets can contain different pages to help group related fields. If you would like to add pages to your datasheet, you can select whether the pages are accessed using tabs or hyperlinks.

Procedure

To add fields to tabbed or hyperlinked page:

1. From the **Page Style** list, select **Tabbed** or **Hyperlink**.

   Note: The Standard page style uses a single page.
2. Accept the default page name, or change it to be more meaningful by editing the **Tab Name**.
3. Click **Add Tab** or **Add Hyperlink** to create a new page.
4. For each page, select the fields to display and the order in which to display them.

Adding a link in a datasheet

You can create a new link for use in a datasheet provided that you have an appropriate link type defined in the database. You need to specify the main entity, the link type (and direction) and the entity type to link to.

Procedure

1. Enter a name in the **Link Tab Description** box. Make this a short meaningful name, ideally combining the names of the link type and the linked entity.
2. From the **Link Type** list, select a link type.
3. From the **Linked Entity** list, select an entity type.
4. If the link type has a restricted set of entity types for its ends, you might see only one or a small number of entity types in the **Linked Entity** list.
5. If the link is to have a direction, choose the required option from the selection of plain and arrowhead lines listed. This choice sets the direction of the link created when you use the datasheet as a way of creating links.
6. If you want the datasheet to retrieve only links of the direction you have just specified, turn off **Ignore direction when retrieving links**. In most cases, you should leave this option turned on so that the datasheet retrieves all links, regardless of direction.
7. Click **OK** to save the link and return to the Links page in the Datasheet Designer.

Field Types

Each type of field stores a particular type of information and has a range of typical uses. Many types have options for display formats and default values.

User Fields

User fields are the fields that record values supplied by users or by data import from other sources. In a data entry form, a quick way to identify user fields is to look for boxes with a white background showing that they allow user data entry.

The following user field types are available:

© N. Harris Computer Corporation

Coordinate

You can add fields to entity types or link types to store coordinate data so that the geographic location of an entity or link can be plotted on a map using a GIS package or i2 Analyst's Notebook.

When you add a coordinate type field with two real number fields to make the combination for automatic conversion, the field order is important. The coordinate field *must* be followed by the latitude field and then the longitude field. The two Real Number fields are used to store the converted coordinate values.

When the user enters geographic data into the Coordinate field, a Coordinate Conversion utility converts the coordinates into decimal degrees, using the WGS 1984 datum (a global standard for plotting geographic locations).

Counting Number

Counting Number fields are used for whole numbers. For example, 656 and -100001 are valid entries.

Note: If a number has a fractional element, use Real Number or Currency field types.

Currency

A Currency field is used for financial values. A Currency field accepts either numbers or numbers with the Currency symbol from the current Regional Settings. The number can include a fractional part, expressed as decimals. This type of field stores numbers in a way that minimizes any errors during calculation and storage. If you analyze records that have different currencies, do one of the following:

- Include the currency in the field name, such as Amount in $, Amount in £ and do not enter the symbol when entering a currency amount.
- Hold currency information in two fields. A Currency field, called Amount to hold the number and a Text field called Currency to hold the currency symbol.

Note:

- You can set the currency symbol to one that is not the default for your current regional settings, for example, to Yen if you are using US English. When you do this, iBase no longer accepts any other currency symbols (including the default for your Regional Settings).
- There is no link to the currency in use when the data was entered. If you have users that work in different locales, you might want to convert all monetary values to a single currency before you enter them into the database record.

A Currency field can be up to 19 digits in length, 15 digits before the decimal point and 4 digits after it.

Note: Analyst's Notebook does not support values in decimal formats. When currency values are added to Analyst's Notebook charts, values will be converted to doubles, that is 15 digits in total, which may result in lower precision for your values.

Date

Date fields are used when entering dates.

You can specify a default value, or use a special value 'Today' which is the creation date of the record.

There are special field types for the system fields **Create Date** and **Update Date**.

Document

Document fields are used to embed documents in a record, such as charts, and files with extensions of: .doc, .docx, .xls, .xlsx, .ppt, .pptx, .txt, and .htm. The advantage of embedding a document is that

© N. Harris Computer Corporation

users are no longer dependent on an external file system or a web server. An embedded document cannot be updated by any changes made to the external file, or by editing in iBase.

Embedded documents can be:

- Viewed in the Show or data entry dialogs
- Searched if Search 360 is available. (Searching documents is optional and requires that the appropriate iFilters are installed on the server.)
- Edited, provided that users have a suitable application for the file type. Users edit a temporary copy of the file; they cannot edit the document embedded in the record. To update the embedded document, users need to reload the edited document into the record.

Hyperlink (including owner hyperlinks)

Hyperlink fields are used to specify one or more locations to link to. A Hyperlink field consists of a series of text strings, with a maximum length of 65000 characters. Hyperlink fields cannot be indexed.

A special type of hyperlink field displays the name of the user who owns the record, and clicking on this type of field displays their contact details.

Hyperlink fields can contain:

A file

For example `C:\My Documents\Person Report.doc`, for a file on your computer or `\\server/Shared/Person Report.doc` for a file on a network.

A URL to the Internet or an intranet

The user can enter any valid URL and double-click on it to display it in their Web browser.

An entity or link record

This can be another entity or link record in the database. Each record is identified by a # character and the record identifier which is displayed in the Properties dialog of the other record.

For example, the record #PER156\GEN. This is record 156 of a PERSON entity type that has a database identifier of GEN. The user clicks the Browse button to display the Select dialog.

A user name

This is the name of the user who owns the record: either the user who created the record (their name is automatically inserted when the record is saved) or the user selected as the owner. To define the hyperlink field as an owner field rather than a general hyperlink field, use the dollar symbol ($) as the default value. Unformatted text, suitable for notes that require editing

Multi-Line Text

Multi-line Text fields are used for unformatted text, such as long unstructured notes that might require subsequent editing. These fields are displayed over several lines; the limit is 65535 characters (in English, this is typically between 10000 and 12000 words).

Note:

- Do not use Multi-line Text fields for information that you might want to sort in record lists.
- If you want to protect fields such that users cannot edit or delete them, use Multi-Line Text (Append only) fields.
- If users need to use large amounts of text, tables or graphics, or character formatting such as bold or italic, use Document or Hyperlink fields.

© N. Harris Computer Corporation

Multi-Line Text (Append Only)

Multi-line (Append Only) Text fields are used for unformatted text. They are similar to Multi-line Text fields except that users cannot edit or delete existing text; they can only append to it. This is useful if you need to store notes as an investigation progresses and you do not want to lose any of the existing information.

If you define a default value for this field, it always precedes any text that a user enters.

OLE Object

OLE Object fields are similar to Picture fields. This is a legacy type; they typically appear in iBase 3 databases that have been converted to a newer version of iBase. OLE Object fields are not supported in iBase 4 or later. They are read-only fields which you need to convert to appropriate field types.

To convert to appropriate field types:

1. Export all the data in the OLE fields that you want to keep.
2. Delete the OLE field type.
3. Import the data back in to fields that use either the Document or Picture types.

Picture

Picture fields are used for pictures in a record. If a Picture file contains a graphics file, this is displayed on Analyst's Notebook charts instead of the icon.

Note:

- There can be more than one picture field in a record, however only the first one that is defined can be displayed on an Analyst's Notebook chart instead of the icon.
- Picture fields cannot be indexed for searching in Full-Text search.

Real Number

Real Number fields are used for numbers that can include a fractional part, expressed as decimals. For example, 1234.56 and 0.005.

A Real Number field can hold a larger range of values than a Counting Number or Currency field. Currency fields offer greater accuracy within a smaller range.

Selected from Code list

Selected from Code List fields make a user select a value from a code list (drop-down list). For example, a grading system might use grades A to E, the code list therefore, would contain only the values A, B, C, D and E from which the user can select.

When editing a field, you can change a Selected from Code List field to a Suggested from Code List field and vice versa. This does not affect any existing records.

Note: This field type is only suitable for use with complete code lists. If you have an incomplete code list, you should consider using a Suggested from Code List field.

Strength

A Strength field defines the line style to be used for event frames in Analyst's Notebook charts.

When defining a Strength field, for a record in iBase you can choose one of the following strengths:

- Confirmed - solid line

© N. Harris Computer Corporation

- Unconfirmed - dashed line
- Tentative - dotted line

Note: You cannot add a Strength field to a link type, or as a standard field.

Suggested from Code List

Suggested from Code List fields allows a user to select a value, either suggested from a code list (drop-down list) or to enter any value. For example, you might supply a list of common car models, but allow the user to type in other models.

When editing a field, you can change a Selected from Code List field to a Suggested from Code List field and vice versa. This does not affect any existing records.

Note: Use this field type when the code list is not complete or definitive.

Text

Text fields are used for small amounts of plain text; these fields are limited to 255 characters. The default field size is 50 characters. You do not save space in a database by using a small field length, however the layout of the fields in dialogs or datasheets is best if limited to 100 characters.

Time

Time fields are used when entering times. You can specify a default value, or use a special value 'Now' which is the creation time of the record.

Time Zone

Time Zone fields can be used to specify the time zone of a record. They are used to provide a reference to dates and times when working with iBase data in Analyst's Notebook.

Note: If you want to import records containing time zones from external data sources, then you need to represent each time zone by the appropriate code. For example, in the import file, the time zone (GMT +00:00) Greenwich Mean Time: Edinburgh, London must be represented by 32. Also, when exporting data that contains time zones, the time zone will also be represented by a code.

The time zones and their codes are listed in Time zones in import and export.

Yes or No (Boolean)

Any data with two values: Yes or No, True or False

System Fields

System fields display and use values that iBase provides from system information maintained for all records. Users can see most of the information for a particular record by displaying the Properties dialog for that record.

Optionally, you can include any of the system information as system fields in the definition of an entity type or link type, either for specific entity types or link types or for all types. Including a system field in an entity type does the following:

- Makes the field available for use in a query.
- Makes the information available in Browse dialogs and other lists.
- Makes the date fields available for use in calculated fields.
- Makes it possible for users to select custom icons and security classification codes for given records.

You can add only one system field of each type to an entity or link type.

© N. Harris Computer Corporation

The following system field types are available:

Case

The system field Case records the case name to which each record belongs, but unless specifically added to an entity or link type it is not exposed to the user. Its value cannot be edited.

Adding a Case field simply maps the value of the system Case field, and is intended for use in multi-case analysis mode, when you want users to be able to identify which case a record came from when running a query, browsing or finding data across several cases (that is, when they are logged on in multi-case mode). If you want to expose the case value in this way, then you should add a Case field to every entity type and link type in the database.

Note: This field type is only available in case controlled databases.

Create Date

The creation date and time for each record. It is automatically added by iBase whenever a record is entered in the database.

The format matches the current date and time settings specified in your Windows environment.

Create User

The user identifier of the person who created the record. This is automatically added by iBase when the record is first saved in the database.

The Create User field appears as a hyperlink in the Properties dialog for a record. When clicked it displays contact details for the user who created the record.

Icon

Provides the representation for an entity record in place of the default icon for the entity type. For example, a Location Icon code list might include icons to represent houses, offices, bars, and so on. If you do not provide an icon field, then the default icon for the entity type is used.

Record ID

The system field Record ID provides a unique record identifier, which is automatically added by iBase whenever a record is created.

A record ID, for example PER130 or PER130\GEN, consists of the following:

Element	**Description**
Table prefix code	This is unique for each entity type or link type, for example PER for a Person entity, or PE0 if there is a naming clash with another prefix. This is three characters long.
Record number	This is unique for each record within the table and automatically assigned by iBase. For this example, the record number is 130.
Separating character	A separating character is present only if there is a following database identifier. It is always the backslash (\ ) character.

© N. Harris Computer Corporation

Element	Description
Database identifier code	This is blank by default. When you create and name a new database, you can specify a code of your choice up to five characters long, in the Advanced page of the Create New Database dialog.

Security Classification Code

The system field Security Classification Code forces a user to select a value from an SCC list (a special code list) that you provide, or to leave the field blank. You can only add one field of this type to an entity type or link type.

Once defined, this field is linked to the chosen code list. It is still possible to edit the code list, which is listed in the iBase Designer Explorer pane under the heading Code Lists \ SCC Lists.

Note: This value is not shown in the Properties dialog. The only way to see or affect its value is to create a field of type Security Classification Code.

With SQL Server databases only, and when Extended Access Control is enabled, a security administrator can make various records of various entity types or link types inaccessible according to the security classification code (SCC) given to each record.

Update Date

The date and time at which the record was last updated. This is automatically added by iBase whenever the record is edited in the database. The format matches the current date and time settings specified in your Windows environment.

The field has no value until an edit has been performed after initial creation of the record.

Update User

The user identifier of the person who last updated the record. This is automatically added by iBase whenever the record is edited in the database.

The field has no value until an edit has been performed after initial creation of the record.

The Update User field appears as a hyperlink in the Properties dialog for a record, and when clicked, displays contact details for the user who last updated the record.

Calculated Fields

Calculated fields allow you to manipulate existing fields by arithmetic operations on numbers and dates, or extraction of portions of dates and times.

The following calculated field types are available:

Calculated Date

Use a Calculated Date field to derive a new date, produced by calculation from another date field of the same entity or the current date and time.

The result is always a pure date. That is, any portion of a day is discarded from the starting field if it had a non-zero time component, and all calculations can only add or subtract whole days. This allows simple comparison with other dates.

© N. Harris Computer Corporation

For example, given a field containing the date of birth, you can calculate the current age of a person or, given the age, calculate the approximate year of birth. Calculated fields also help in identifying records for purging, based on dates of data entry or changes of status of the represented entities and links.

Calculated Date Part

Use a Calculated Date Part field to derive a part of a date or time, selected from another date field of the same entity or from the current date and time. For example, given a date you can display just the day, month, or year.

Calculated Number

Use a Calculated Number field to derive a number from another numeric or date field or the current date and time.

The permitted operations are addition, subtraction, multiplication and division, shown in the drop-down list as +, -, *, and / .

You can mix fields of type Real Number, Counting Number, Currency, Date, and Time (but not Time Zone) with fixed values, both real numbers and integers.

You cannot affect the order of calculation by using brackets.

Formatting fields

For many fields, you can choose a format in which the field is displayed from the **Format** list. The list of formats varies depending on the type of field selected. In some cases, you can also leave the Format choice blank or use code letters to enter a custom format.

If you have existing records in the database, you can preview the effects of most display formats by right-clicking the relevant entity, and selecting **Records**. Due to the potential scale of format conversion, Multi-Line Text formats cannot be previewed.

You can choose a format for fields of the following types:

Text

Formats: Upper Case, Lower Case, Upper First, or left blank to show the text as entered

Examples:

- Upper case - EXAMPLE
- Lower Case - example
- Upper first - Example

Calculated Date

Formats: Long Date, Medium Date, Short Date, Custom (with m d y)

Examples:

- mmmm dd, yyyy - August 23, 2022
- yy mmm dd - 22 Aug 23
- d-m-yy 23-8-22

Calculated Number

Formats: Custom (with #.0)

Examples:

© N. Harris Computer Corporation

- #.000 - 34.567
- #.00 - 34.56

Real Number

Formats: Custom (with #.0)

Examples:

- #.000 - 34.567
- #.00 - 34.56

Date

Formats: Long Date, Medium Date, Short Date, Custom

Examples:

- mmmm dd, yyyy - August 23, 2022
- yy mmm dd - 22 Aug 23
- d-m-yy 23-8-22

Hyperlink

Formats: Upper Case, Lower Case, Upper First, or left blank to show the text as entered

Examples:

- Upper case - EXAMPLE
- Lower Case - example
- Upper first - Example

Time

Formats: Long Time, Medium Time, Short Time, Custom (with h, m or n, s am/pm and a separator)

Examples:

- hh:mm:ss am/pm - 12:20:08 am
- nn:ss - 20:00

Currency

Formats: Currency (using Windows currency symbol), Custom (with #.0)

Examples:

- #.000 - 34.567
- #.00 - 34.56

Multi-Line Text, Multi-Line Text (Append Only)

Formats: Upper Case, Lower Case, Upper First, or left blank to show the text as entered

Examples:

- Upper case - EXAMPLE
- Lower Case - example
- Upper first - Example

© N. Harris Computer Corporation

You can define a custom format by entering code letters and extra text, or display the field unformatted by leaving the entry blank. If you specify a custom format, the field either contains the code that you enter or is blank; it never contains the word Custom.

Note: Be aware that when formats are applied to numerical field types (Currency, Calculated Number, and Real Number fields), the displayed accuracy or number of decimal places might be limited. Add a hint to the Description of such fields that queries should test for a difference of plus or minus the least significant digit rather than absolute equality.

Coordinates in iBase

To plot an entity or link on a map, you need to enter coordinate values in two fields that have been set up for this purpose. Your GIS package is configured to interpret the values in these fields so that the data can be plotted in the correct location.

In iBase, you will also be able to store geographic data in a number of formats, which are then converted, either manually when you enter the record or automatically after an import or using a bulk conversion. You can also run coordinate queries.

Note: When you add a coordinate type field with two real number fields to make the combination for automatic conversion, the field order is important. The coordinate field must be followed by the latitude field and then the longitude field.

Types of field

The fields used to contain the coordinate data must be defined as Real Number type fields. They might contain the following types of coordinates:

- Latitude and Longitude values, entered in decimal degrees
- Easting/Northing data, entered in meters

These fields will typically be called Latitude and Longitude or X and Y. If you are not sure which fields you need to use, move the pointer over the field name to see its tooltip, or speak to your database administrator.

About converting coordinates to a standard format

When you convert coordinates, they are always converted to decimal degrees of latitude and longitude, using the WGS 1984 datum (a global standard for plotting geographic locations).

When you add a coordinate type field with two real number fields to make the combination for automatic conversion, the field order is important. The coordinate field *must* be followed by the latitude field and then the longitude field.

The original coordinate value is stored so that it can be searched for, and for audit purposes.

Note: The conversion process will change longitude values greater than 180 to their equivalent negative value in order that they can be plotted correctly.

© N. Harris Computer Corporation

Supported coordinate systems

There is a wide range of formats in which you can enter coordinate data. Be aware of the following points when you are using any of the following coordinate systems.

Military Grid Reference System

MGRS coordinates with less that the prescribed five-digit northing and easting values are accepted by iBase, but these low-precision values represent a large square surface area. For conversion purposes, the upper left corner of their effective area is used.

For example, 40UCE11 and BCE11 are interpreted, for conversion purposes, as being identical to 40UCE1000010000 and BCE1000010000.

British National Grid

BNG coordinates with better than 1-meter accuracy are not supported when automatically converting coordinates in bulk. You can choose one of the following options:

- Treat as conversion failure: the conversion is skipped so that you can review the record and update the coordinates as required.
- Round to nearest meter: this conversion automatically rounds the coordinate down to the nearest meter.

Due to the potential for overlap with Degrees, BNG coordinates that fall within 0, 0 and 360, 360 are not recognized. If you want to enter coordinates in this area, use a zone letter. For example, SV0030000300.

Decimal Degrees

Latitude and longitude must be within the range 90 - 90 and 180 - 360.

The flags N, S, E, and W can be replaced by words (North, South, East, and West) when this format is used: 01.00°X, and 02.00°Y. These values are not case-sensitive.

Decimal Minutes

Decimal minutes is not a natively supported system, so all decimal minutes formats are converted to decimal degrees and stored in the decimal degrees format.

Latitude and longitude must be within the range 90°S to 90°N and 180°W to 360°E.

The flags N, S, E, W can be replaced by words (North, South, East, and West) for the following formats (these values are not case-sensitive):

- 01° 02.00'X, and 03° 04.00'Y
- 01°02.00'X, and 03°04.00'Y

The characters that are assigned as the degree and minutes representations must remain constant for a single set of coordinates. For example:

- 56°45'N 32°14'W is valid.
- 56°45'N 32D14MW is not valid.

As a minimum, there must be a single character between the degrees and minutes if you omit the degree representation for the following formats:

- -01°02.00', and -03°04.00'

© N. Harris Computer Corporation

- 01°02.00'X, and 03°04.00'Y
- X01°02.00', and Y03°04.00'

If not using the degree representation, use a space instead. For example:

- -1234 8221.4 is not valid.
- -12 34 82 21.4 is valid.

Important: Decimal minutes formatted as minutes are not supported. For example, it is not valid to format 03° 04.00'Y as 184.00'Y.

Degrees Minutes Seconds

Latitude and longitude must be within the range 90°S to 90°N, and 180°W to 360°E.

The flags N, S, E, and W can be replaced by the words (North, South, East, and West) for the following formats (these values are not case-sensitive):

- 01° 02' 03.00"X, and 04° 05' 06.00"Y
- 01°02'03.00"X, and 04°05'06.00"Y

As a minimum, there must be a single character between the degrees and minutes if you omit the degree or minute representations for the following formats:

- -01°02'03.00", and -04°05'06.00"
- 01°02'03.00"X, and 04°05'06.00"Y
- X01°02'03.00", and Y04°05'06.00"

If not using the degree or minute representation, use a space instead. For example:

- 123443.6 822113.8 is not valid.
- -12 34 43.6 82 21 13.8 is valid.

Universal Polar Stereographic

The easting and northing values are interchangeable if E or N is used. If neither E or N is used, then the first number is assumed to be the easting value. For example, the following are all valid and represent the same point:

- 2,500,000mE 1,850,000mN
- 1,850,000 2,500,000mE
- 2500000 1850000

If easting and northing are swapped over, the final character on E must be "E" and the final character on N must be "N". Neither of these values are case-sensitive.

Defining coordinate queries

Coordinate queries can find entities or links of a particular type within a defined geographic area or close to a location.

About this task

You can use coordinate queries to search entities or link types that have coordinate fields. You can only run coordinate queries on entity or link types that have a Coordinate type field, followed by two real number fields that contain coordinate data.

© N. Harris Computer Corporation

Procedure

1. Select **Analysis** > **Coordinate Query Builder**.
2. Select the entity or link type that contains the coordinate data.
3. Select the Coordinate type field. This will typically be selected automatically, as there is usually only a single Coordinate field for each entity or link type.
4. In the Source area, specify which records you want to include in the coordinate query.
5. In the Query Operator area, select the required operator:
   - Is near - finds records with coordinate data that is within a specified distance of a location you enter. Enter the location using coordinates and then specify the required distance and units.
   - Is between - finds records with coordinate data that falls within a rectangle whose corners you define. Enter the two sets of coordinates to form the corners of the search area.
6. In the Coordinates area, enter the coordinates

   If you selected 'Is near', enter one set of coordinates, then enter a value in the Tolerance box and the units for that value, such as Feet, Kilometers, Miles, Seconds, or Yards. This value is used to calculate the distance from the entered coordinates. The use of *seconds* as a unit is a way to specify small distances. Each degree is divided into 60 minutes, and each minute is divided into 60 seconds. If you, for example, set a tolerance of 30 seconds on a latitude of 38 degrees, the tolerance will be 30/3600 = 0.0083333 degrees.

   The Tolerance is calculated by adding the specified distance to both the horizontal (longitude) and vertical (latitude) part of the coordinates to create a square with the original coordinates in the middle. Any record with coordinates that are located within this square is found.

   If you selected 'Is between', enter two sets of coordinates. These coordinates form two corners of a rectangle. Records with coordinates that fall within the defined square are found.
7. Click **Next** to create the query.
8. Click **Results** to run the query.

Converting Coordinates in bulk

To ensure that all the records that store coordinate data have a complete set of coordinates in a consistent format, you use the Bulk Coordinate Converter dialog.

All coordinates are converted to decimal degrees against the WGS 1984 standard. Converting your coordinates enables you to:

- Use the data in the Coordinate field to insert or update latitudes and longitudes (as decimal degrees). This allows records to be plotted on maps or for records to be included in coordinate queries.
- Use the data in the latitude and longitude fields to insert or update the coordinate field value. You may want to do this to ensure that your data is complete.

You can specify how the update is applied. You can also save a list of records that fail to update, for example because of insufficient data, in a set for review later.

Note: You can only convert coordinates if there is a Coordinate type field followed directly by two other fields for the latitude and longitude (Real Number fields). These two fields will contain the converted coordinate values. The first field, stores the original coordinate before it is converted.

See Coordinates in iBase for further background information.

© N. Harris Computer Corporation

Note: See Supported coordinate systems for further information on converting data to the coordinate system standard to your organization. This topic also describes the rules that are applied when converting coordinates.

To convert coordinates

1. Decide on the scope of the bulk conversion. For example, you can convert all the records in the database or you can restrict the conversion to the records in a query or set. You will convert only the records that you have access to.
2. To track which records converted successfully, you can create sets that you can review later:
   - To obtain a list of failures, turn on the **Add records that failed to update** checkbox, and enter the details of the set.
   - To obtain a list of successes, turn on the **Add successfully updated records** checkbox, and enter the details of the set.
3. Click **Next** to continue.
4. In the Update area, specify whether you are updating latitudes and longitudes or the coordinate field value as explained in detail below.
5. Select the datum of the original coordinates if they do not use WGS 1984. The datum you select will be remembered for the next time you use the Bulk Coordinate Converter dialog.
6. Click **Next** to continue.
7. Depending on the coordinate system:
   a. For UTM or UPS coordinates where the hemisphere is not specified, select **North** or **South**.
   b. For BNG coordinates, select the precision that you want to use (see Supported coordinate systems for details).
8. Click **Convert** to apply the conversion.

Updating latitudes and longitudes

You can automatically update latitudes and longitudes if the records have a value in the Coordinate type field, which will enable you to plot these records on maps.

Select the Update latitude and longitude field values option and then decide on the scope of the update:

Option	Update scope
Only if both the fields are blank	Select this option to update only those records that are missing both the longitude and latitude.
Only if either one or both the fields are blank	Select this option to update any records with a missing latitude, missing longitude or both.
Always	Select this option to update all records, including records that already have a latitude and longitude.

Updating the coordinate field value

You can automatically update the Coordinate type field if the records have latitudes and longitudes, which were entered either manually or by importing.

© N. Harris Computer Corporation

Select the **Update coordinate field values** option and then decide on the scope of the update:

Option	Update scope
Only if the field is blank	Select this option to update only records without a value in the Coordinate type field.
Always	Select this option to update all records, including those that already have a value in the Coordinate type field. For example, you might want to do this after editing a series of latitudes and longitudes.

Setting up Semantic Types

A semantic type is a category of data that defines how iBase interprets that data. For example, the Person entity semantic type could be applied to entity types such as Male, Victim and Witness. The semantic type allows iBase to understand that each of those entity types are a different way of depicting people in the real world.

All i2 products at your site should use the same Semantic Type Library. To achieve this, assign semantic types to the database schema, and define new ones, in one database only and then distribute them to any other related databases in your organization.

To make use of semantic types, you can assign a semantic type to each relevant entity type, link type, field, standard field, and icon. You do not need to assign a semantic type to everything in your database schema.

Semantic types can then be saved to a file for distribution to others in your organization. Semantic types are also saved in any templates that you create from the database.

How to use semantic types in iBase

Although the Semantic Type dialog is displayed in various locations in iBase Designer, it is only displayed in iBase when a user runs a query that includes semantic types, to allow the selection of entity, link or semantic property types to search.

At this release, users can use semantic types within iBase itself for running queries with semantic conditions. Semantic types are also used when iBase data is charted on Analyst's Notebook charts.

Note: Certain entity types can have Smart Matching behavior in Analyst's Notebook if they have a field that is assigned an identifying property semantic type.

About the Semantic Type library

We provide the i2® Semantic Type Library, which contains semantic types that you assign to data in your data sources. These semantic types identify the meaning of the data they represent, and are used by applications such as Analyst's Notebook to properly interpret and align the data from different data sources.

The library includes three different kinds of semantic type definition:

- Entity semantic types (for entity types and icons)
- Link semantic types
- Property semantic types (for entity and link type fields, including standard fields)

© N. Harris Computer Corporation

You must decide which kinds of semantic type best represents your data.

Each semantic type consists of the following elements:

- Name
- Data type, such as text or number
- Optional synonyms--- alternative names that are used when searching for suitable semantic types
- Description that provides guidance on how the type should be used
- Additional notes

Depending on its location in the hierarchy of semantic types, the function of a semantic type will be general or specific. For example, Motor Vehicle is a specialized type of Transport, and Bus is a specialized type of Motor Vehicle. In the event that Bus is not specific enough, you could create a custom semantic type. However, you should not add any custom types without the agreement of others at your site and, once you have added them, you must share the updated Semantic Type Library with all users of i2 products at your site. For details, see Maintaining the semantic type library.

Assigning semantic types in iBase Designer

There are two ways of assigning semantic types in iBase. You can:

- Work with single entity types, link types, and field types. See Assigning a semantic type for details.
- Work with all the objects in the database schema. See Assigning Semantic Types to your data for details.

Restrictions on how you assign semantic types

There are a few restrictions on how you assign semantic types:

Entities and icons

You can use any entity semantic type that is suitable for the data. Assigning Semantic Types to your data for details.

Links

You can use any link semantic type that is suitable for the data. See Assigning Semantic Types to your data for details.

Fields

You can use any property semantic type. However, consider the underlying data type when making your choice:

Data Type	**Possible Semantic Type**
Number	Any of the numerical semantic types found by expanding **Abstract Number**
Text	Any of the text semantic types found by expanding **Abstract Text**
Yes or No (Boolean)	Any of the flag semantic types found by expanding **Abstract Flag**

© N. Harris Computer Corporation

Data Type	Possible Semantic Type
Date and time	Any of the numerical semantic types found by expanding **Abstract Date & Time**
Binary	Any of the numerical semantic types found by expanding **Abstract Binary**

When assigning semantic types to fields, you cannot assign the same semantic type to two or more fields in the same entity or link type. See Assigning Semantic Types to your data for details.

Standard fields

You can use any property semantic type as explained above for Fields. When assigning semantic types to standard fields, you cannot assign the same semantic type to two or more standard fields in the same database.

Note: You cannot assign abstract semantic types to database objects--- you can only create custom semantic types from them.

Loading the semantic type library

The first time you display the Semantic Types dialog or the Select Semantic Type For dialog, they display the i2 Semantic Type Library only. If there are any semantic types specific to your organization (custom semantic types), you need to load these before assigning semantic types to your data. The Semantic Type Library for your organization is saved in a file with a `. mtc` file extension.

Procedure

1. Select **Tools** > **Database Design** > **Semantic Types**.
2. In the Semantic Types dialog, click **Load**.
3. Select the required custom semantic type file (MTC) file, and click **Open**. The tree view is updated to show all the semantic types in the library.

   Note: If you see any names ending 001, 002, and so on, there are duplicate names for the semantic types in use in your organization. You need to remove the duplicates created in either this database or another database. How you do this will depend on which database holds the central Semantic Type Library for your organization. For details, see Maintaining the semantic type library.
4. If you load the wrong MTC file, click **Cancel** to remove the library, otherwise when you click **OK** you will add the custom semantic types to the current database.

Searching for semantic types

Procedure

1. Select **Tools** > **Database Design** > **Semantic Types**.
2. In the Semantic Types dialog, click the **Entity Types**, **Link Types** or **Standard Fields** tabs to go to the appropriate page
3. Enter the semantic type that you want to search for. As you type, possible matches are displayed in the Ordered Results area.
4. You can widen your search by trying the following on the text displayed in the **Search Available Semantic Types** box:

© N. Harris Computer Corporation

Tip	Example
Shorten the displayed text	"Documents" to "Document"
Simplify the displayed text	"End date" to "date" or "end"
Consider alternative spellings	"tire" to " tyre"

5. If none of the semantic types in the Ordered Results area are suitable, you can browse the semantic types displayed in the tree view. You may find it easier to browse the semantic types if you first familiarize yourself with the top-level semantic types and their contents. Click on each semantic type to display a brief description of how each one is used.

Assigning Semantic Types to your data

To benefit from the visualization and advanced analysis capabilities, for example, of Analyst's Notebook when charting iBase data, you can assign relevant data with a semantic type that identifies the real world content of the data.

Do I have to assign semantic types to all data fields?

To construct a Semantic Type Library that accurately models your database schema, you can add a semantic type for relevant entity types, link types, fields, standard fields, and, optionally, icons in your database. Doing so ensures that your users can take full advantage of other i2 applications that use semantic types, such as Analyst's Notebook.

Are different semantic types available?

There are three semantic types that are supported: entity semantic types (for entities and icons), link semantic types, and property semantic types (for fields and standard fields). For more information, see Setting up Semantic Types.

What if I cannot find a suitable semantic type?

If you cannot locate a suitable semantic type in the Semantic Type Library, you can derive your own custom semantic type from the appropriate generalized semantic type. iBase, Analyst's Notebook and other i2 applications will treat a custom semantic type as a specialization of its recognized parent semantic type.

It is important to select the correct parent because the custom semantic type will inherit its behavior, and this will determine how the custom semantic type is used during, for example, matching operations on Analyst's Notebook charts. For details, see Defining custom semantic types.

Note: You must log on as a database administrator in order to assign semantic types.

Unassigning semantic types

To remove a semantic type from an entity type, link type, field, standard field, or icon:

1. In the Database area of the Semantic Types dialog, select the item that you want to unassign. The semantic type is highlighted in the tree view.
2. Click **Remove**.
3. When you have finished, click **OK** to save your changes.

© N. Harris Computer Corporation

Assigning semantic types to entity and link types

Procedure

1. In the Semantic Types dialog, load any custom semantic types specific to your organization. See Loading the semantic type library.
2. In the Database area, click the appropriate tab to display the Entity Types or Link Types page. Depending on the settings in the lower left of the dialog, the page displays all the entity or link types defined for the current database.
3. Select the item to which you want to assign a semantic type. The Search Available Semantic Types box displays its name. The Ordered Results area suggests some semantic types that may be suitable for assigning to the item, based on a comparison of its name with the name of the semantic type and any synonyms set up for it. For further details, see Searching for semantic types.
4. You can review the suggested semantic types by clicking on a result to display additional information below. In particular, the description may provide some guidance on when to use the selected semantic type. The icon shown in the tree view indicates whether the semantic type is a standard type or a custom type specific to your organization.
5. If none of the semantic types in the Ordered Results area are suitable, you can search the Semantic Type Library. SeeSearching for semantic types.
6. When you have located the correct semantic type, you can do one of the following:
   - Select it and click Assign. (The button is unavailable if you have already assigned a semantic type to this item.)
   - Drag the semantic type from the Semantic Types area on to the appropriate item in the Database area. When dragging and dropping is not allowed, the cursor changes.

   The name of the assigned semantic type is displayed in the Database area.

   Note: If you see the message, `Cannot assign abstract library types`, you need to select a different semantic type. You cannot assign any of the top-level semantic types, such as Entity, Link, Abstract Text, or Abstract Number.
7. Click **OK**.

Assigning semantic types to fields

You assign property semantic types to fields and standard fields in the same way as entity and link types. The possible semantic types are listed on the Property page in the Semantic Types area.

About this task

You may find it easiest to assign the semantic types to standard fields first. For some general information, see Setting up Semantic Types.

Procedure

To assign a semantic type to a field or standard field:

1. Select **Tools > Database Design > Semantic Types**.
2. Turn on the Show Fields checkbox in order to display fields in the Database area of the Semantic Types dialog. See below Showing and hiding assigned and unassigned items.
3. On the Property page of the tree view, locate the semantic type.

© N. Harris Computer Corporation

You cannot assign the same semantic type to more than one field in the same entity or link type. Similarly, you cannot assign the same semantic type to more than one standard field. If you try to assign a semantic type that you have already used, you will see the message:

```
This semantic type has already been assigned. Do you want to create a
custom semantic type derived from this semantic type?
```

Click:

- **OK** to create a custom semantic type based on the name of the selected type followed by a number, for example Transaction Date 1.
- **Cancel** to cancel this assignment so that you can select a different semantic type or create your own.

Important: You should only create a custom semantic type if you are working in the database that contains the central Semantic Type Library for your organization. For important information on creating custom semantic types, see Defining custom semantic types and Maintaining the semantic type library.

Assigning semantic types to icons

You can also assign entity semantic types to icons in order to override the default semantic type for the entity type. The semantic type for the icon will be used for records where the default icon type is overridden.

Procedure

1. Select **Tools** > **Database Design** > **Semantic Types**.
2. Click the **Icons** tab to display the Icons page.
3. Locate the icon that you want to assign.
4. If there is no direct match between the name of the icon and the names (or synonyms) of the semantic types, search for the semantic type in the usual way. See Searching for semantic types.
5. Select the required semantic type and click **Assign**. The name of the semantic type is then displayed on the Icons page.

Defining custom semantic types

You might find that the semantic types supplied do not contain a semantic type that is appropriate for your data. In this case, you can define custom entity, link, and property semantic types.

About this task

When you define a semantic type, it inherits some of the properties of the parent, but not its name or synonyms.

Note: Be sure to carefully search for an available semantic type before you define your own custom semantic types. Before you can do this, you might need to load all the custom semantic types available at your site. See Loading the semantic type library.

Never define a custom semantic type when the Semantic Type Library in use at your site already contains an appropriate semantic type. Duplicate types, for example Football Match, Football Match_001, Football Match_002, mean that the information retrieved from your database cannot be aligned with information retrieved from other data sources that have the correct semantic type assigned. This limits your users' ability to analyze data from different sources.

© N. Harris Computer Corporation

A custom semantic type has a globally unique, internal identifier which is derived from the database in which it is created. Therefore an entity semantic type called Football Match, created in one database is distinct from an entity semantic type of the same name created in a different database. To avoid the problems that this causes, make sure to share the custom semantic types with other users in your organization.

In order to avoid the creation of duplicates, you should do only create custom semantic types in the database that holds the Semantic Type Library for your organization.

When to define custom semantic types

You might decide to define custom semantic types for a variety of reasons.

For example, consider if your data contains different kinds of sporting events. The Semantic Type Library contains an Event entity semantic type, but it does not contain entity semantic types for distinguishing between different kinds of sporting events. To ensure that appropriate semantic types for sporting events are added to your Semantic Type Library, you must define custom semantic types that are derived from the Event entity semantic type.

If it is not necessary to distinguish between different specializations of an entity, then you can simply assign the appropriate generalized entity semantic type to your data. For example, suppose your data contains a list of people who have attended an annual convention. The library does not contain a Convention entity semantic type, but you can assign the Event entity semantic type to your Convention data field because your data contains records for only one kind of event.

You might want to define a custom property semantic type if you want to assign multiple values for the same property to a single entity or link. For example, suppose your database contains a list of a person's bank account numbers, and you have decided to represent each bank account as a field on the entity type, rather than use bank account entities with links to the person that owns them. Since a property semantic type can only be added once to each entity semantic type or link semantic type in a Semantic Type Library, you can create specializations of the Account Number property semantic type so that each occurrence has a unique property semantic type assignment.

Deriving the custom semantic type from the correct parent

Choosing the correct semantic type to derive your new custom semantic type from is a critical decision because the custom semantic type inherits characteristics and behaviors from its parent. In the sporting event example given earlier, it would be inappropriate to derive the custom semantic types from the Document entity semantic type, for example, because a sporting event is not a special type of document.

Sharing and reusing custom semantic types

If others in your organization are also assigning semantic types to data, you should share your custom semantic types so that all databases use the same Semantic Type Library. If two people define custom semantic types of the same name, they are not identical because the semantic type name does not uniquely identify the semantic type. Its unique identity is determined by the database in which it is created.

For more information about duplicate names and sharing your custom semantic types with others, see Maintaining the semantic type library.

Backing up the Semantic Type Library

After adding custom semantic types to your library, save them to file so that you can:

- Distribute the new semantic types to others in your organization.

© N. Harris Computer Corporation

- Restore deleted custom types (you cannot recreate custom semantic types by adding a new one of the same name).

To do this, click **Save** in the Semantic Types dialog. The Semantic Type Library is saved in a file with an .mtc file extension. For further information, see Maintaining the semantic type library.

Procedure

1. Select **Tools** > **Database Design** > **Semantic Types**.
2. Load any custom semantic types specific to your organization. See Loading the semantic type library for details.
3. Locate the semantic type that is a generalization of the special type that you require. You can do this by searching for semantic types that have a generalized name. For example, if you require additional entity semantic types to represent different stolen property articles, you should derive these custom semantic types from the Property entity semantic type.
4. On the appropriate page, select the generalized type, right-click, and select **New**.
5. Change the name of the custom semantic type to a name that reflects your usage.
6. In the **Synonyms** box, enter some other words that have the same meaning, and that you want to group together under the same semantic type. For example, synonyms for Location might be Area, Map Reference, Region, and Situation. Enter these like this (with no space after the commas):

   ```
   Area,Map Reference,Region,Situation
   ```
7. In the **Description** box, enter some notes on how to use the custom semantic type.
8. Click **OK** to add the new semantic type as a child of the generalized semantic type. Notice that the icon changes slightly to indicate a custom semantic type. This allows you to see which are standard semantic types and which are specific to your organization.
9. Assign the custom semantic type to an item in your database schema in the usual way.
10. When you have finished, click **OK**.

Maintaining the semantic type library

All i2 products and databases at your site should use the same Semantic Type Library. The best way to achieve this is to define any custom semantic types centrally in one database, and treat this library as the central Semantic Type Library for your organization. You can then distribute them to other iBase databases by using a custom semantic type (MTC) file. See below Saving the Semantic Type Library to file for details.

You can edit and delete custom semantic types but not ones from the standard Semantic Type Library. You should always do this in the database that holds the Semantic Type Library for your organization. All work on custom semantic types should be done in one central place because a semantic type is uniquely identified by the database in which it was created rather than by its name.

It is important to control how custom semantic types are created and edited. Lack of control can result in duplicate names for semantic types in one or more of your databases. One possible method of resolving duplicate semantic types when there are several iBase databases involved is described below.

Saving the Semantic Type Library to file

Custom semantic type files store details of the semantic types defined in the database from which they are saved. They do not store any details of how the semantic types are assigned; you need to use the Database Design report to obtain this information.

© N. Harris Computer Corporation

You should save your Semantic Type Library to file whenever you add, edit, or delete semantic types to the database that holds the central Semantic Type Library for your organization:

- In the Semantic Types dialog, click **Save** and select a folder for the Semantic Type Library file. The semantic types are saved in a file with a .mtc file extension.

Editing custom semantic types

You can edit the name, description, and synonyms of a custom semantic type, but not of a standard type from the Semantic Type Library. You cannot add additional notes to custom semantic types.

Note: Do not alter the name or description for a custom semantic type in a manner that changes the original meaning of its usage. Different instances of the same custom semantic type are aligned (matched) regardless of the name or description of the custom semantic type.

To edit a custom semantic type:

1. Select **Tools** > **Database Design** > **Semantic Types**.
2. Right-click the semantic type and select **Edit**. The Edit Custom Semantic Type dialog is displayed.
3. Click **Save** and save a new custom semantic type file to record your changes.
4. Click **OK** to save your changes.

Deleting custom semantic types

To delete an unassigned custom semantic type, and any children that it has:

1. In the Semantic Types dialog, unassign the custom semantic type if required. For details, see Assigning Semantic Types to your data.
2. Right-click on the semantic type and from the shortcut menu, select **Delete**. The custom semantic type is deleted immediately.
3. Click **Save** and save a new custom semantic type file to record your changes.
4. Click **OK** to save your changes.

Note: If you inadvertently delete the wrong custom semantic type, reload the semantic types from file. Do not recreate it.

How semantic types with duplicate names can occur

Duplicate names for semantic types may occur when you:

- Copy and paste entity types, link types, and fields between databases that define their own Semantic Type Libraries rather than make use of one centrally-defined library.
- When you load a Semantic Type Library into a database where similarly named semantic types already exist.

Duplicate semantic types are renamed so that they can be displayed in the tree view of the Semantic Types dialog.

© N. Harris Computer Corporation

Resolving duplicate names

If you have duplicate names in different Semantic Type Libraries, you must remove these duplicates before you can carry out analysis that involves these sources. To resolve duplicate names, you must combine the libraries before removing the duplicates.

Procedure

To resolve a situation where duplicate names exist, which you do not want to keep:

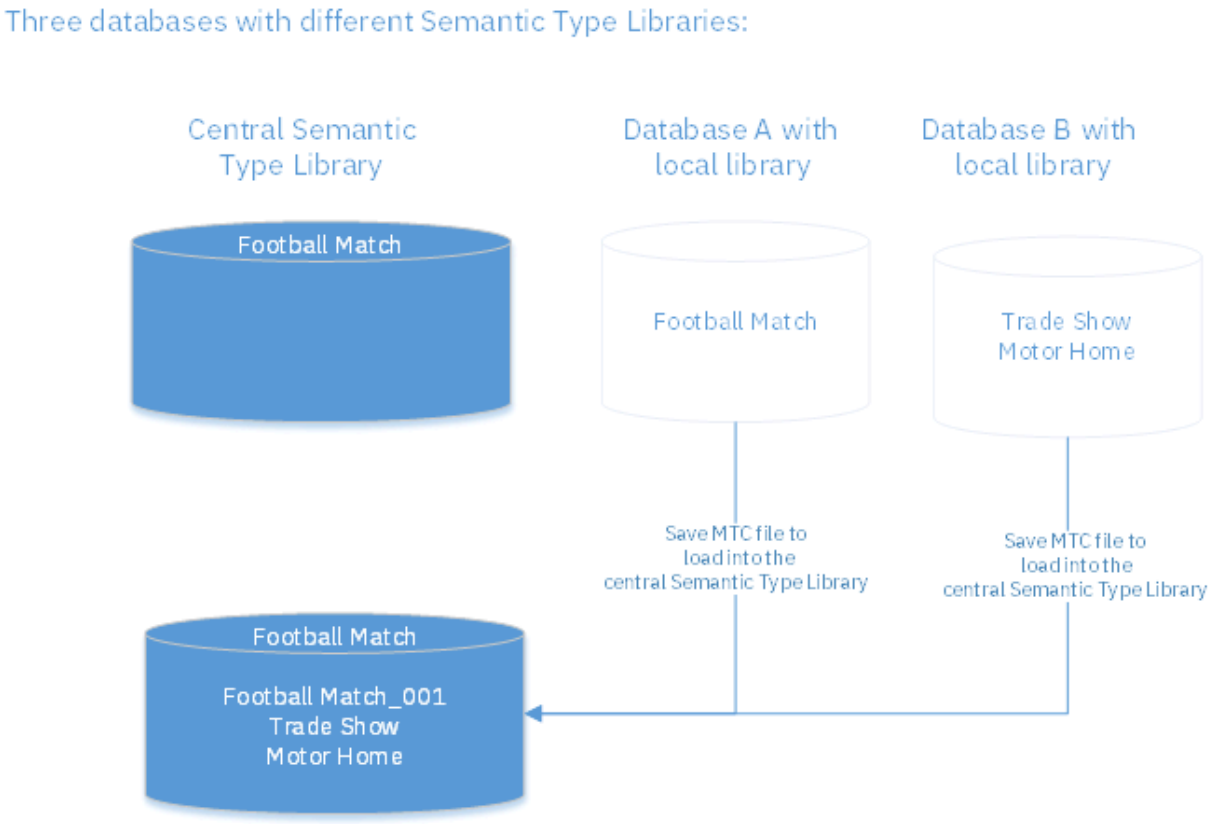
1. Print Database Design reports to record how the semantic types are assigned in each database. This information is not saved in custom semantic type (MTC) files.
2. Decide which database contains the central Semantic Type Library.
3. In the databases that do not contain the central Semantic Type Library, save the semantic types to an MTC file.
4. Load the MTC file(s) into the database that you have designated as holding the central Semantic Type Library. In this database:
   a. Make a note of the duplicate semantic types--- these will have names ending _001, _002, and so on.
   b. Delete all duplicate semantic types.
   c. Save an MTC file, which should now contain an updated, and clean, Semantic Type Library.
5. In the other databases, delete the duplicate semantic types. You may need to unassign them first. You may find it useful to refer to your list of the duplicate custom semantic types deleted from the central Semantic Type Library.
6. In the other databases, load the MTC file from the database that holds the central Semantic Type Library. The MTC file should load correctly without creating any duplicates.
7. If, when you load the MTC file, duplicate semantic types are displayed, then the correct semantic types from the central Semantic Type Library were renamed with a numeric suffix (because this database treats them as duplicates). In this situation:
   a. Unassign and then delete the duplicate semantic types. The duplicate will not have a numeric suffix; it is the centrally-created custom semantic type that has the numeric suffix.
   b. Edit the name of the centrally-created semantic type (the one with the numeric suffix) to remove the suffix.
   c. Reassign the semantic types if required.

Example of resolving duplicate names

If semantic type libraries are not managed centrally, you may encounter conflicts if you try to resolve custom types at a later time. The following example shows a method of resolving these conflicts.

In this example, custom semantic types are set up and assigned in three databases:

© N. Harris Computer Corporation

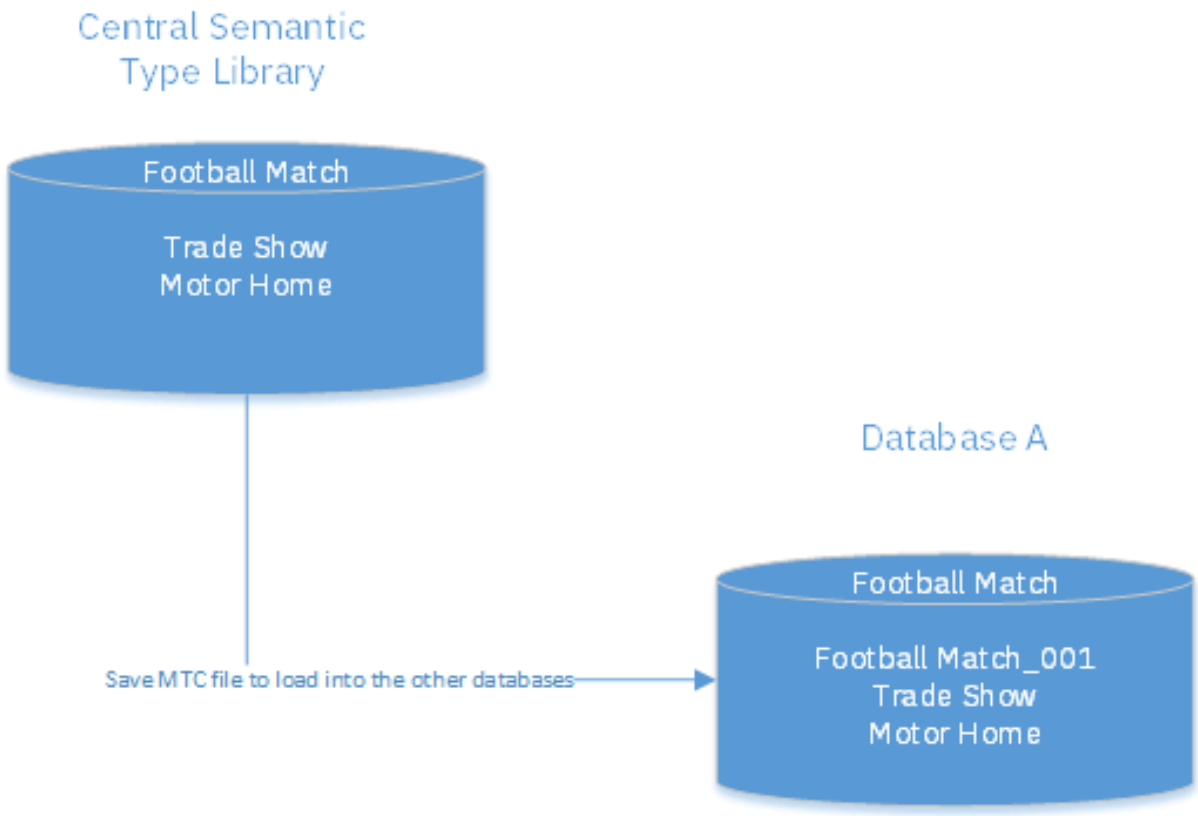


You update the central Semantic Type Library by loading the custom semantic types from Databases A and B. The Trade Show and Motor Home semantic types are unique and are therefore loaded without any problems but the name Football Match already exists in the central Semantic Type Library, and loading the MTC file creates a custom semantic type with a duplicate name (Football Match_001) this can be deleted.

You then save an MTC file in order to distribute the new custom semantic types to the other databases. However, before loading it into Database A, you should delete the Football Match semantic type in Database A. After loading the MTC file, there will be no semantic types with duplicate names.

If the following situation occurs on loading the MTC file into Database A:

© N. Harris Computer Corporation



You need to remove the real duplicate in Database A. To do this, unassign and then delete Football Match, and then rename Football Match_001 to Football Match and reassign it.

Note: Loading the MTC file from the central Semantic Type Library into Database A, created a custom semantic type with a duplicate name. It is important to understand that in Database A, the real 'duplicate' is Football Match and that the one from the central Semantic Type Library was renamed on loading (Football Match_001).

Managing security

You can define a security policy and create new users and security groups using the Security Manager. All groups have users as members.

A particular user can be a member of any number of groups, of any types. The user gains the properties defined for all the groups in which they are a member.

You can also set the other properties of database management groups, and change users' passwords or active status.

Creating a security policy

The security policy specifies rules for adding and changing passwords that apply only to user accounts with iBase user names - they do not apply to users that can log on with single sign-on. For further details, see Creating a security policy.

© N. Harris Computer Corporation

Types of security group

There are four different types of security group:

Database Management

A database management group controls read, write, update, and delete permissions to, for example, entities, links, and folder objects. The properties are set in the Group dialog.

See Creating security groups for details.

System Command Access Control

A system command access control group denies access to specific iBase commands. This provides finer control over the actions a user can perform. Denied commands are typically hidden from the user. The properties are set in the System Commands Access Control dialog.

See Setting up System Commands Access Control groups for details.

Data Access Control

A Data Access Control (DAC) group controls permissions related to entities, links, and fields in each database. This allows a very fine control of how individual pieces of data are made visible to, or modifiable by, groups of users. The properties are set in the Data Access Control dialog.

See Setting up Data Access Control groups for details.

Folder Object Control

This has no management properties set in iBase Designer. Users define the usage for groups of this type, using the Categorize dialog and settings made in the Options dialog.

See Working with categories for details.

Creating users and groups

To create a new user:

1. Select **Security** > **Security Manager**.
2. Click the **Users** tab. The Users page is displayed listing any existing users.
3. Click **New** to display the User dialog where you can enter the user details. For further information, see Creating users.

To create a group:

1. Select **Security** > **Security Manager**.
2. Click the **Groups** tab. The Groups page is displayed listing any existing groups.
3. Click **New** to display the Group dialog where you can choose the type of group and define its properties. For further details, see Creating security groups.

Inspecting users and groups

To view the:

- Database management permissions for a user: on the Users page, right-click a user name, and from the shortcut menu, select User Permissions. See Checking user permissions for details.
- Groups a user belongs to: on the Users page, double-click the user name to list the groups. The user is inactive if there is no plus sign next to it.

© N. Harris Computer Corporation

- Users belonging to a group: on the Groups page, double-click the security group type, and then double-click the particular group.

Editing and deleting users

You can edit and delete users on the Users page of the Security Manager dialog.

To make a user a member of additional groups, edit their database management permissions, or make them inactive:

1. Select **Security** > **Security Manager**.
2. On the Users page, select the user name.
3. Click **Edit**. See Creating users for details.

To remove a user's membership of one or more groups:

1. Select **Security** > **Security Manager**.
2. On the Users page, select the group.
3. Right-click, and select **Remove**.

Note: A user must belong to at least one group otherwise they will not be able to log on.

You can also delete a user and remove any record of this user from the database. For details of the consequences of deleting user accounts, see Creating users; you may prefer to make the account inactive instead.

Editing and deleting groups

You can do the following on the Groups page of the Security Manager dialog.

To add users to a group:

1. Select **Security** > **Security Manager**.
2. On the Groups page, locate the group by double-clicking the appropriate type of security group and then select the group.
3. Click **Edit** to display the Group dialog. See Creating users for further details.

To remove users from a group:

1. Select **Security** > **Security Manager**.
2. On the Groups page, locate the group by double-clicking the appropriate type of security group and then double-click the group to lists its members.
3. Right-click a user, and from the shortcut menu, select Remove.

To delete a group:

1. Remove all the users from the group as described above.
2. Select the group and click **Delete**.

© N. Harris Computer Corporation

Creating a security policy

A security policy sets restrictions on the user accounts that are set up to access iBase. The security policy specifies rules for adding and changing passwords that apply only to user accounts with iBase usernames and passwords.

About this task

New security files do not have a security policy because by default none of the settings on the **Security Policy** page of the Security Manager are turned on.

The absence of a security policy means that:

- Minimum password length is four-characters.
- No restriction on the characters that are used to make up passwords.
- Passwords never expire.
- No limit to the number of attempts to log on.
- Last used username is displayed at the next logon.
- No password history (although a new password cannot be the same as the current password).

Note: Although a security policy is part of the security file, it is not replicated even if you choose to replicate the security file. Enabling each site that is involved in iBase Database Replication to maintain their own security policy. However, the password history is replicated as it is possible that users might need to log on and change their account details at any of the sites.

Procedure

To view a security policy or change its settings:

1. In iBase Designer, Select **Security** > **Security Manager** > **Security Policy**.
2. Enter the requirements for new iBase passwords.
   - **Minimum password length**

     Enforce a minimum number of characters for the password, 1 - 20 characters.
   - **Minimum password age**

     Prevent the user from changing their password for a specified number of days.

     Note: This restriction can be overridden by turning on **Reset password at next logon**.
   - **Maximum password age**

     Force the user to change their password after a specified number of days has passed. By default, passwords never expire.
   - **Show password expiry reminder**

     Remind the user to change their password for a specified number of days before the expiry date.
   - **Enforce password history**

     Prevent the user from changing their password back to one used previously. The new password is compared to all previous passwords. Set the passwords remembered option to limit the number of passwords that are used in validating the new password.
   - **Lock out user after**

© N. Harris Computer Corporation

Control the number of times the user can enter an incorrect password before their account is disabled.

Note: You can unlock the account in the User settings by turning on **Account is active**.

- **Reset account lock-out after**

  Automatically unlock an account that has been disabled as a result of too many failed logon attempts.

  Note: Administrative accounts are automatically reset after thirty minutes.

- **Enforce complex passwords**

  Force the user to select a password of a suitable complexity.

- **Hide last username when logging on**

  Hide the name of the last user to use iBase. By default, last used username is displayed at the next logon.

- **Enforce SQL Server certificate validation for connections**

  This option is unchecked by default, and iBase connects without completing certificate validation. Check this option to enforce SQL Server certificate validation for users logging into iBase.

  Note: If you use this option you must configure the certificates on the SQL Server in order to connect to your SQL Server databases.

Note: The changes that you make do not affect existing passwords unless you require users to change their passwords when they next log-on.

3. Click **Apply** to save your changes. The changes come into effect when you log off.
4. If you are editing an existing policy, and change the password settings, select whether you want to force users to change their password when they next log-on.

Creating security groups

You can create security groups and edit group membership. You can also set the properties of database management groups, that is the database permissions that users gain through membership of one or more database management groups.

About this task

All groups have users as members. A particular user can be a member of any number of groups, of any types. The user gains the properties defined for all the groups of which they are a member.

For details of the group types, see Managing security.

Note: Data Access Control and Folder Object Control groups have a part of their definitions in a database. These parts and their relationships to security groups are not preserved when you create a template from a database. However, the groups are still available in the security file, so you can recreate any settings required in a newly created database, based on that template.

Procedure

To create a group:

1. Select **Security > Security Manager**.
2. Click the **Groups** tab, and do one of the following:

© N. Harris Computer Corporation

- Click **New**.
- Select an existing group and click **Edit**.

3. In the Group dialog, select a type from the **Group Type** list. For details of these groups, see Managing security.

   Note: The Data Access Control type is only available if Extended Access Control is enabled.
4. Enter a **Name** for the group, up to 50 characters.
5. If you are defining a Database Management group, set the permissions by selecting the checkboxes for the desired permissions. See Checking user permissions for a description of the permissions.
6. If you want to set the membership for the group, click the **Users** tab and select the checkboxes for users you want to add. If there are a large number of users, you might find it useful to:
   - Display the users who do not belong to the group by turning on the **Show Unselected Items Only** checkbox.
   - Add all users to the group by clicking **Select All**.
   - Remove all users from the group by clicking **Clear All**.
7. Click **OK** to create the new group.
8. If you have created a System Commands Access Control group or a Data Access Control group, define the security for the group. For details, see Setting up System Commands Access Control groups or Setting up Data Access Control groups.

Creating users

You can create and edit user accounts. Managing the access that users have to an iBase database allows you to secure your data.

You can:

- Create new users.
- Add contact details for users (that are used by the Created By and Updated By fields in the properties of a record, and if you assign owners to records).
- Change a user's password (only for users with iBase user names and passwords).
- Add and remove group memberships to affect a user's permissions.
- Remove a user's access, and prevent them from logging on to a security file.
- Make users inactive or delete them.

How users acquire permissions

Users gain the database management permissions accumulated from all database management groups of which they are a member. There is a similar combination of permissions or restrictions for the user's membership of each other type of group.

If there are Data Access Control groups, then a new user is automatically made a member of all these groups. This gives them the lowest possible level of data access, which is safe from a security perspective but might prevent the user doing useful work. You can change this default group membership, whenever you want, to give the user meaningful access to data.

To see the current database management permissions, click Show User Permissions. For further details, see Checking user permissions.

© N. Harris Computer Corporation

Creating an iBase logon account

Users in iBase can either be managed exclusively in iBase Designer or by connecting to a Windows Active Directory instance to allow single sign-on. If you would like to manage all user information using iBase Designer, you can set up all the information directly.

Procedure

To add a new user who will log on by entering an iBase user name and password:

1. Select **Security > Security Manager**.
2. Click the **Users** tab to display the Users page.
3. Either:
   - Click **New**.
   - Select an existing user and click **Edit**.
4. Enter a **Name**.

   Important: Do not enter any foreign characters unless your database supports these. For details, see Viewing the Properties of the Security File.
5. Select **iBase Security** and in the **Password** box, enter a password that conforms to the security policy for your organization.
6. Enter the details of the account:

Option	Description
Account is active	By default, an account is active when created but you can disable it by turning off the checkbox in order to prevent them from logging on. (**Note:** All the details of the account, including their permissions, are retained.)
User cannot change password	Turn on this checkbox if you want to prevent the user from changing the password that you give them. For example, use this option when creating accounts for use with iBase Scheduler.
Restricted audit log	This setting is applicable only if you use Audit History and Audit Viewer. It is used to restrict other users from viewing the audit logs of the user you are creating.
Account expires after	Turn on this checkbox if you want to add an account that becomes inactive after the specified date.
Reset password at next logon	Turn on this checkbox to force the user to change their password when they next log on.
Password never expires	Turn on this checkbox to create an account with a password that never expires even though your

© N. Harris Computer Corporation

Option	Description
	security policy may specify a standard duration for passwords. For example, use this option when creating accounts for use with iBase Scheduler.

7. In the **Default Category** box, enter the name of the category in which all folder objects will be saved by default for this user. Leave this box blank to use the General folder.
8. Click the **Permissions** tab to display the Permissions page.
9. Assign permissions to this user. You can do one of the following:
   a. Click **Copy Permissions** and then select an active user with the required permissions.
   b. In the list of groups, turn on the checkbox for each of the required groups. To display just the groups of which the user is not a member, turn on the **Show Unselected Items Only** checkbox. You can also:
      - Add the user to all the groups by clicking **Select All**.
      - Remove the user from all the groups by clicking **Clear All**.
10. Click **Show User Permissions** if you want to inspect the user's permissions after turning on one or more database management groups. For information on these permissions, see Checking user permissions. After inspection, click **Close** to return to the User dialog.
11. Click the **Information** tab to display the Information page where you can add user contact details. See Adding user information for details.

    Note: Users cannot edit user information in iBase where the iBase user account represents a Windows user group.
12. Click **OK** to save the details of the new user.

    Note: If you failed to enter the same password in both boxes, iBase Designer asks you to enter the password again.
13. In iBase, users can change their own passwords by selecting **File > Change Password**.

Creating a single sign-on account

Users in iBase can either be managed exclusively in iBase Designer or by connecting to a Windows Active Directory instance to allow single sign-on.

Before you begin

If you would like to set up your users to allow them to connect using their Windows account, you need to configure the users in iBase Designer to determine the access levels.

Procedure

To add a user who can automatically log on via single sign-on:

1. Select **Security > Security Manager**.
2. Click the **Users** tab to display the Users page.
3. Either:
   - Click **New**.

© N. Harris Computer Corporation

- Select an existing user and click **Edit**.

4. Select **Windows User / Group**.
5. Enter the Windows user name, in any of these formats:
   - DisplayName (example: FirstName LastName)
   - ObjectName (example: Group1)
   - UserName (example: Username1)
   - ObjectName@DomainName (example: Username1@Domain1)
   - DomainName\ObjectName (example: Domain1\Username1)

   Tip: If you do not know the user name, click **Browse** to search the network domain.
6. Click **Check Name** to verify the name. The name is converted to the format `<domain name>\<account name>`. Successfully verified user names are displayed underlined.
7. Enter the basic details of the user account:

Details	**Notes**
Account is active	By default, an account is active when created but you can disable it by turning off the checkbox in order to prevent them from logging on.
Restricted audit log	This setting is applicable only if you use Audit History and Audit Viewer. It is used to restrict other users from viewing the audit history of the current user.
Account expires after	The user can log on up to and including the specified date.
Default category	Optionally, enter the name of the default folder in which the user will save their folder objects. Leave this box blank to use the General folder.

8. Assign permissions to this user. You can do one of the following:
   a. Click **Copy Permissions** and then select an active user with the required permissions.
   b. In the list of groups, turn on the for each of the required groups. To display just the groups of which the user is not a member, turn on the **Show Unselected Items Only** checkbox. You can also:
      - Add the user to all the groups by clicking **Select All**.
      - Remove the user from all the groups by clicking **Clear All**.
9. Click **Show User Permissions** if you want to inspect the user's permissions after turning on one or more database management groups. For information on these permissions, see Checking user permissions.
10. Enter contact details on the Information page. Click on the **Information** tab to display this page. See Adding user information for details.
11. Click **OK** to save the details of the new user.

© N. Harris Computer Corporation

Checking user permissions

Each user's permissions are displayed in the User permission dialog, you use this dialog to check what actions can be performed in *iBase*. You can perform an action if there is a check mark in the box to the left of each action. These permissions are part of the database design; they cannot be changed in this dialog.

The following objects are folder objects, and are subject to the folder object permissions set for the user account.

- Browse definitions
- Queries and Scored matching (definitions)
- Sets
- Report definitions
- Import and export specifications
- Import and export batch specifications
- Charting schemes

Note: Labeling schemes and alert definitions are not folder objects.

The user permissions are described below.

Permission	**When turned on**	**When turned off**
Add Entity/Link Records	You can add new records to the database.	You can find, browse, and show the records in the database but you cannot add any new ones, either individually or by importing them.
Update Entity/Link Records	You can edit records that you have added.	Once you have added a new record, you cannot change it in any way. This includes batch editing, assigning new icons, and merging. (**Note:** Users who can apply icon shading will also be able to assign icons.)
Delete Entity/Link Records	You can delete records that you have added.	Once you have added a new record, you cannot delete it, either individually or by using batch delete.
Update/Delete Entity/Link Records created by other users	You can edit and delete any record in the database.	You cannot edit or delete records created by other users.
Add Folder Objects	You can add new sets, and save queries, report definitions, import specifications, and so on that you add yourself.	You can run queries, reports, and so on, either by using definitions created by other users or by using new definitions

© N. Harris Computer Corporation

Permission	When turned on	When turned off
		of your own. You cannot save your definitions.
Update Folder Objects	For folder objects created by you, you can edit existing queries, report definitions, import specifications, and so on. You can also edit the contents of existing sets, including appending records to existing sets.	Once you have added a new folder object, you cannot edit it.
Delete Folder Objects	You can delete folder objects that you added yourself.	Once you have added a new folder object, you cannot delete it.
Update/Delete Restricted Folder Objects created by other users	You can update and delete restricted folder objects created by other users.	You cannot update or delete restricted folder objects created by other users.
Update/Delete Public Folder Objects created by other users	You can update and delete public folder objects created by other users.	You cannot update or delete restricted folder objects created by other users.
Database Creator, Database Administrator, Security Administrator	A system role that is only relevant when using iBase Designer. See below for details.	
Audit Administrator	The Audit Administrator role is not administrative. Instead, it allows a user with this permission to view the records displayed and modified by other users who are defined as having a restricted audit log.	

Note: The folder objects actions (as in Add Folder Objects for example) apply to folder objects in general. There is also access control on individual folder objects based on the membership of Folder Object Control Groups.

There are three system roles:

- Database Creator
- Database Administrator
- Security Administrator

Note: Audit Administrator is not a system role.

These roles are not modified in any way by the other types of iBase security groups. As supplied, iBase gives all these roles to members of the System Administrators group, which is suitable where

© N. Harris Computer Corporation

you intend a small number of people to be able to perform all roles including database design, security administration, and maintenance of data integrity in operational databases.

It is possible to create groups that partition the overall administration capability. For example, you can create:

- Database Designers able to create database designs but not access data.
- Security Administrators able to create groups, manage users, and monitor audit logs, but not access data.
- Database Managers, able to change data and folder objects for the purposes of resolving conflicts, weeding or archiving old data, and generally maintaining the operational efficiency and relevance of a live database, but not manage users.

Setting up System Commands Access Control groups

System Commands are types of actions that are carried out on the database. For example, adding records, performing types of search, or accessing database statistics. You can restrict access to types of actions, to members of specified security groups.

Before you begin

You must have the security groups available to assign the access control permissions. See Creating security groups.

About this task

For users in each security group, you can:

Deny use of iBase commands

Users can be denied access to iBase commands. This provides finer control over the actions a user can perform. It may also simplify the user interface for certain tasks, even if the commands are not denied by the user's database management permissions.

Note: Commands that are denied are typically hidden; they are not displayed as unavailable. However, some denied commands may be displayed, should a user attempt to use them a message is displayed that they do not have the correct permissions.

Request the user to record the reason for use

Request a reason for using the command, then record the reason and action in the audit log.

Audit command groups

You can set iBase to audit specific types actions for members of a security group:

- Search 360
- Data Exposure
- Charting
- Analysis

Note: If no types are specified all actions will be audited following the audit level of the database.

Procedure

To set up System Commands Access Control:

1. In iBase Designer, open the security file and login with administrator privileges.

© N. Harris Computer Corporation

2. Select **Tools** > **System Commands Access Control**.
3. Choose the **Security Group** to set the access controls for and set the permissions in the three access control type lists:
   - Access Denied - prevent members of the security group from accessing the specified action.
   - Reason for Action - require members of the security group to provide a reason for carrying out the action. The action, and the reason are subsequently stored in the audit log. If you turn on a command group in the Reason For Action page, there is no need to turn on the same command in the Audit page.
   - Audit - logs information about the types of action in the audit log at all audit levels, and to all databases accessed through the same security file.

Setting up Data Access Control groups

A Data Access Control (DAC) group controls permissions related to entities, links, and fields in each database. This allows a very fine control of how individual pieces of data are made visible to, or modifiable by, groups of users.

About this task

Data Access Control Group Permissions control:

- Denying access or modification to all records for a particular entity type or link type.
- Hiding administrative fields in records or making administrative fields read-only to certain groups of users.
- With SQL Server databases only, making selected records of various entity types or link types inaccessible according to the security classification code (SCC) given to each record.

Data Access Control is specific to each database in which it is defined. Consider carefully how you might want to use a scheme using this type of conditional access.

Important: After making changes to a Data Access Control group in a database that uses alerting, log off and then reopen the database as soon as possible, in either iBase or iBase Designer. This applies the security changes to any existing alert definitions.

Procedure

1. Open a database.
2. Select **Security > Data Access Control**.
3. Use the Security Manager dialog to create one or more Data Access Control groups, and assign users as members of those groups.
4. Open the Data Access Control dialog. The dialog has two main areas, a list of security groups on the left and a tabbed area on the right, with tabs for:
   - **Tables**

     List of checkboxes and names of all the entity types and link types in the database. Each name is of the form Type: Name, to show which type it represents. For example, the names might include Entity: Account.

     If a checkbox is turned on then the named table (all records of that named entity or link type) or field is denied to members of the selected security group.
   - **Fields**

© N. Harris Computer Corporation

List of checkboxes and names for all the fields of all the entity types and link types in the database. Each name is of the form TypeName: FieldName, to show which entity type or link type contains the field. For example, the names might include Account: Account Type. In these pages, standard fields appear separately for each entity or link type and you can control the appearance of each standard field independently.

Important: You are warned if you deny access to a mandatory field (or if you make a denied field mandatory). If you choose to deny access to this field (or make a denied field mandatory), you prevent members of the group from adding records of the entity or link type.

If a checkbox is turned on then the named field is denied to members of the selected security group.

- **Read-Only Tables**

  If a checkbox is turned on then the named table (all records of that named entity or link type) or field is made protected from change by members of the selected security group.

- **Read-Only Fields**

  If a checkbox is turned on then the named field is made protected from change by members of the selected security group.

- **Security Classification Codes**

  List of checkboxes and names for all classification entries in all SCC code lists defined in the database.

  If a checkbox is turned on then all records with that classification are denied to members of the selected security group. (If any classification name appears in more than one SCC list, the denial of records applies to all records with that classification regardless of the list in which it appears.)

Note: If you have opened an Access database, the dialog does not display the Security Classification Codes tab. This is because iBase does not support this form of control for Access databases. For this reason, there is some duplication of contents in these tabbed pages.

5. To view the current configuration or to configure a group, first select the group in the Security Groups list. Then click each tab to see the entries where the checkboxes are turned on and, if you want to, turn on or off various entries.
6. Save the changes.

Results

The specified access will be applied.

Note: The relationship to database contents means that the full definition of a Data Access Control group is stored in two parts. The name and membership of each group is stored in the security file. The restrictions on members of each group are stored in the database.

To apply the same control to another database controlled by the same security file, open that database and with the window of that database active, enter the Data Access Control dialog. Your security groups will already exist so you need only turn on the same checkboxes to apply the same security.

© N. Harris Computer Corporation

Adding user information

Depending on your organization, you may need to record additional information about users. This information is used by other iBase users who may have queries about the data added to this database, or who may need to discuss the record before editing, deleting, or merging records owned by that user.

About this task

You can add information for all users, including those who use single sign-on even though their user accounts may be based on a Windows group rather than individual Windows users. Alternatively, you can ask the user to add their own contact details when they first open the database in iBase. However, users cannot do this if Alerting is in use or if they do not have their own iBase user account.

As part of the contact details, you can enter a location which could be a geographical area, a division, or an area of responsibility. The location can be used in the Audit Viewer to filter the audit data. You will need to predefine the locations by entering each location for at least one user on the Information page. In iBase, users can only select a location, they cannot type it in.

Procedure

1. Select **Security** > **Security Manager**
2. You can then do one of the following:
   - Click **New** to create a new user.
   - Select an existing user and click **Edit**.
3. In the User dialog, click the **Information** tab.
4. Enter the full name, location, telephone number, e-mail address, and any notes.

   The location will appear on a **Location** list that can be used by other users entering their own contact details in iBase.
5. Confirm that the details are correct by clicking **OK**.

Reporting on database security

The security design report provides details about the security groups, users, and their consequent permissions or restrictions that have been applied to the database. You can select the items you want to include in the report.

Before you begin

To view the details of the database security, you must be assigned the `SecurityAdministrator` role.

Procedure

1. In iBase User, Select **File** > **Properties** > **Security Design Report**.
2. Select the types of details to include in the report:
   - Groups - For each security group that has been defined for the database, you can list:
     - Permissions and restrictions - What members of the group can access, and what they are explicitly prevented from doing (for example, editing read-only items).
     - Users - The users that are currently members of the group.

© N. Harris Computer Corporation

    - Denied SCC items - The types of entities or link that have restrictions in place, to prevent members of this group from accessing particular records.
  - Users - For each user that can access a database, you can list:
    - User Information - The information entered centrally about the specified user.

      Note: This report only uses information added in iBase Designer.
    - Groups - The groups the user is a current member of.
    - Permissions - The specific permissions for the user.

3. Once you have generate the report you can:
   - Browse through the pages
   - Refresh the information it contains
   - Print the report
   - Export the report as a spreadsheet, a PDF, or a Microsoft Word document.

Configuring Auditing

You can set up iBase to log each time that a user modifies or accesses a record and to log virtually all user actions with or without user-supplied reasons for performing the actions.

What is recorded?

iBase starts auditing at the lowest possible level of detail when you create a database. You cannot stop this level of auditing but you can choose to start at a higher level, and to modify all auditing options for existing databases.

Attention: The option to record user accesses to records, without change of data, creates large volumes of log data so it is available only with iBase SQL Server databases. Use this option only when strictly required. Your SQL Server administrator can configure the disks to improve performance in this area; for details, see Server machines.

Independently of the audit level of the database (SQL Server format only), you can audit changes to data. The iBase field types that you can audit depend on the SQL Server version.

Where is it recorded?

There are separate audit logs for security files and databases.

Security file logs track the opening of databases, failed logon attempts, and a range of administrative actions such as creating templates, and managing users and groups. They do not record logons and logoffs.

Database logs track the opening and closing of databases, historical data (if logged), and all the requested actions within databases. Actions are recorded regardless of origin: users can request database actions from iBase Designer, iBase, Analyst's Notebook, or third-party mapping applications.

The physical form and location of logs is different for security files, Access databases, and SQL Server databases. The audit viewer handles these differences and can produce archive files in a standard form.

Viewing audit logs

To use the Audit Viewer, a user needs to be a system administrator, a database administrator or an audit administrator.

© N. Harris Computer Corporation

The Audit Viewer, if installed, is available from the Windows start menu, under, for example, **i2 iBase** > **iBase Audit Viewer**. It allows you to view and manage audit logs for databases and security files. You can open multiple windows to inspect logs for several databases provided that those databases are managed through the same security file.

Audit Viewer does not display all the entries in the audit log:

- Some users generate restricted audit log entries and you need the Audit Administrator role to view these
- Some audit log entries are hidden if SC codes are used - you can only view the entries for records relating to your security classification

The level of detail in the audit log is determined by the audit level set for the database.

Audit log databases

If you are using iBase and an Access database, the database log is held in the `.idl` file stored in the same folder as the database file.

In an SQL Server installation of iBase, an audit log database is created alongside the main SQL Server database. The name of the database is the same as the main database name with the suffix `_log`. For instance, the database `User_Guide` has an audit log database `User_Guide_log`.

Your SQL Server administrator must ensure that iBase users can access this audit log database. If a user has no access to the audit log database, iBase attempts to create a new one and fails with a message that says it could not do so successfully

Note: For information on backing up audit log databases, see backing up iBase.

Setting Up Search

There are two main ways to search for text in an iBase database, Search 360 (in SQL Server databases only) or Word Search. An index must exist before users can use either search method.

Search indexes allow users to search multiple fields in multiple entity and link types, and can include information that is stored in attached documents. Database designers can define the rules used to build and search an index.

Using Search 360

Search 360 can be used to search any type of text field in SQL Server databases, including documents and charts, using a range of techniques that allow for:

- Any number of words in a phrase
- Typing mistakes (for example typing ROBETRSON instead of ROBERTSON, or Danielsmith instead of Daniel Smith)
- Variations in spellings and variations based on how words are pronounced (for example typing PETERSON instead of PEDERSON because, the two names sound the same)
- Synonym matching (for example, Mike and Michael are synonym matches)
- Variations in word order (for example "Joyce Gallagher" and "Gallagher, Joyce" are exact matches)
- Allowances are made for punctuation and accents (for example Francoise and Françoise are exact matches)
- Records directly related to the main search (that is, just one link away from a record found by the main search)

© N. Harris Computer Corporation

Using Word Search

Word Search can be used to search a database of any format using:

- Exact words
- Wildcards
- Synonym matching
- Soundex

When matched text is found, it is highlighted in context in the document. Word Search also allows users to browse the index and find text based on the number of times they occur or by their leading characters.

What is not searched?

The search methods do not search:

- The targets (records, files or URLs) of Hyperlink fields, only hyperlink text is searched.
- In Search 360, document types for which there is no installed iFilter on the machine that performs the indexing.
- Number type fields, including numbers, dates, time zones, system and calculated fields.

Setting up Search 360

Search 360 allows entity types, link types, and text-based field types to be indexed. The Search 360 index is updated whenever data is added, changed, or removed in any of the entity types and link types in the database schema, or in embedded documents.

The index is also updated following changes to the database schema itself. Indexing occurs at scheduled times that are determined by an SQL Server administrator although an iBase system administrator can request a rebuild outside of scheduled times, for example after bulk imports. The service is suspended while you have the database open in iBase Designer.

There might be a delay between changes that are made to the iBase records and the time when the changes are reflected in the index, depending on how indexing is scheduled. Therefore, it might be possible for users to obtain results that are no longer an accurate reflection of the database contents.

The Search 360 Administration window, and the Search 360 window in iBase, provide information on when the index was last updated.

By default, all entity types, link types, and text-based field types are indexed. This indexing allows users to search all fields for information, but increases the database size. If required, the index can be restricted, reducing the index size.

Note: If the index is restricted, users might not get back the search results that they expect. Clearly communicate any change to the scope of the index to users of the system.

Required permissions

The Search 360 index is created in the same SQL Server instance as the iBase database. The indexing tools can either be run directly, or using the command line tooling, elsewhere on the network.

To use the indexing tools, you require a Microsoft SQL Server login with suitable administrative rights to allow databases and jobs to be created:

© N. Harris Computer Corporation

Task	Role
To configure the index service	`dbcreator` server role
To schedule an index service job on the database itself	`SELECT`, `INSERT`, `UPDATE` on `_Configuration_Text`; `SELECT`, `INSERT`, `UPDATE` on `_Configuration_Def`
To schedule an index service job on the `msdb` database	Member of the `SQLAgentUser` role; `SELECT` on `sysjobs`

After the index service is configured, you require an account to run the index service. In SQL Server, this can be one of the following:

- a Windows account that is a member of the sysadmin server role
- a proxy account that is a member of the "public database role" of the database to be indexed

Process Overview

The following steps summarize the processes that are required to enable Search 360 on your system.

About this task

Note: Steps 3 - 5 must to be repeated for each iBase database.

Note: If you use iBase database replication, then you must to repeat steps 1 - 5 for each replicated database because the search indexes are not replicated.

Installing the Index Service Configuration tools

The iBase Server components contain the tools that you use to configure the Search 360 Index. These tools must be installed to run the indexing.

iBase Index Service Configuration tool

The iBase Index Service Configuration tool is a Windows application that manages databases in a local instance of SQL Server. You must install the Index Service Configuration tool on a machine that has a local instance of SQL Server. You can only index databases in the local instance, and the index database is created in this instance.

iBase Indexer

The iBase Indexer (`i2.iBase.SearchIndexerExe.exe`) is a command-line tool that allows you to specify the server that holds the database you would like to index. The index is stored as a separate database in the same SQL Server instance as the database that is being indexed.

Checking iFilters

Suitable iFilters might already be available depending on the version of the applications you are using.

Search 360 automatically indexes all documents with the suffixes ANB, CSV, LNB, and TCV. Your Windows Operating System also has iFilters built in that index many other document types such as DOC, DOCX, HTM, PDF, RTF, TXT, XLS, and XML. The Microsoft Office 2010 Filter Pack is required to index XMLX documents. The easiest way to determine whether the correct iFilter is installed is to import a document of the required type, update the index, and then search for a term that you know is in the document.

© N. Harris Computer Corporation

Any other document types might require that an appropriate iFilter is installed on the machine, on which indexing is performed.

iFilters are third-party plug-ins that enable index services, such as the Search 360 Index Service, to scan the different document types that can be embedded in entity and link records and extract index terms.

Attention: Documents larger than 100 MB might not be included in the index.

If your document does not appear in the search results, you must install the related iFilter. See Installing iFilters for further details.

Installing iFilters

On the machine that is running the indexing tools, you can install additional iFilters to allow documents of a set type to be searched using Search 360. These third-party iFilters can support additional document types or metadata from the standard iFilters that are supplied with the operating system.

About this task

iFilters are used to access documents and store the details in the index. You can see the iFilters that you currently have installed and associated with specific files using the **Indexing Options** in the control panel for your operating system all the files are listed in **Indexing Options** > **Advanced** > **File Types**.

Note: If an iFilter is installed after the index is built, the index will need to be rebuilt for the document type fields before the document will be indexed correctly.

Procedure

1. Install a supported iFilter on the Search Indexer workstation, that is the computer running the index.

Note: Sometimes non-standard iFilters can cause issues because they are not supplied as part of iBase or your supported Operating System. If you have any questions, contact i2 Support.

Configuring a database for Search 360

To allow Search 360 capabilities such as spelled-like and sounds-like, you need to set up additional indexing of the data that is stored in the database. By default, Search 360 indexes all fields, but you can reduce the size of the index and improve performance by excluding any that are not relevant to your search requirements.

Before you begin

It is useful to understand the types of fields defined in your iBase schema, because this can help you decide what to include in the Search 360 index. You can generate a **Database Design Report** from the iBase Designer **File** menu, which shows all the field names and field types associated with Entity and Link records.

Fields with types of 'Selected from Code List', 'Suggested from Code List', or 'Yes/No' are often good candidates for exclusion from the Search 360 index. This is because they contain information from pick lists, which is more suited to standard iBase queries. For example, a Person entity might have fields such as 'gender', 'nationality', or even 'wears glasses'. There is little benefit in running 360 searches on fields of this type.

Similarly, if you are indexing bank accounts, it is inefficient to index different kinds of accounts, such as 'savings' or 'current'. If you are searching documents for information about a bank account, you are more likely to be interested in the specific details of the account rather than what sort of account it is.

© N. Harris Computer Corporation

Procedure

To initialize the database for Search 360, in iBase Designer:

1. Select **Tools > Search > Search 360 Administration**.
2. In the Search 360 Administration window, turn on **Enable Search 360**.

   Note: Turning off **Enable Search 360** prevents the index service from running.
3. When Search 360 is enabled for the first time, the **Build Index** button is available. If you want indexing to occur when the index service next runs, click **Build Index**. The SQL Server administrator needs to set up the index service. On subsequent occasions, the available option button changes to **Rebuild Index**. You might not want to rebuild the index every time you make a change to the database, as this can be time-consuming and have an impact on performance. Changes to the database schema, such as adding new fields, require a rebuild of the index.
4. To build the index you must select at least one iBase record type from the **Entities** and **Links** lists. If you select the checkbox at the top level, all record types are automatically included, along with their fields. iBase can index all fields, but this takes longer to build, longer to search, and can return results of little value. You can click to select record types individually, and then click to deselect fields within each record type.
5. **Optional:** To include metadata from any Document field types, select **Index document metadata**. The metadata can provide useful context about the document, such as the author, date created, or title.
6. Click **OK** and then close the iBase database. The index service cannot connect to the database while you are logged in.

What to do next

System administrators should use the Index Service Configuration tool to set up the index service and its schedule. Alternatively it is possible to configure the index service manually. During the index service setup, the server name and database name must be correctly specified in the Configure Database dialog. This ensures:

- the SQL Server Agent job is created in SQL Server.
- The 360 search field is available to iBase users when they log in to the database.

To exclude specific *words* from the index, you can connect to SQL Server Management Studio with an account with the required permissions on the iBase SQL Server database. For more information, see this iBase Support article.

Setting up synonyms in Search 360

Users can search for groups of words known as synonyms, that is words that have the same or very similar meanings. When you have set up synonyms in Search 360, users can choose whether or not to use them each time their search term includes one of the synonyms.

About this task

Search 360 is provided with synonyms suitable for the US and the UK. These synonyms are not necessarily suitable for other regions, although your supplier might provide localized alternatives. Synonyms are provided for the following types of text:

- Given names, including variant spellings
- Parts of full addresses

© N. Harris Computer Corporation

- Common parts of organization names
- Vehicle makes

Synonyms may be useful in these types of situation:

Searched text	**Example**
Is from different sources	Text can use a mixture of spellings from US and UK English and users might want to treat 'color' and 'colour', or 'fender' and 'bumper' as synonyms.
May have known variations or errors in data entry	Many drugs have a variety of names in both formal and casual usage. Also, a name using accented characters might be presented in a different order or be spelled in different ways when converted to a form without accented characters: Müller and Mueller.
Is too precise for the purpose of the search	A database of vehicles might hold color names specified by the paint manufacturer's names, such as crimson, scarlet, flame, and so on, where the search term derived from a witness statement is simply 'red'.

You must add synonyms in groups, even if you create only one group. Each group should contain words that users consider as being the same for the purposes of their search. For example, a synonym group named Firearm might have the values: Firearm, Shotgun, Rifle, Handgun, Revolver, Pistol.

If you create many groups and synonyms, you may prefer to export the synonyms to a file and edit that file in a text editing application.

To import a synonym list, you require a text file of the correct format in a known location on an accessible disk volume. This example defines a synonym group named Firearm with the values: Firearm, Shotgun, Rifle, Handgun, Revolver, Pistol:

```
[Firearm]
Firearm
Shotgun
Rifle
Handgun
Revolver
Pistol
```

You can export synonyms defined in one database for external editing or transfer to other databases. You can also transfer synonyms between Full Text Search and Search 360.

Procedure

1. Select **Tools > Search > Search 360 Administration**.
2. In the synonyms page of the Search 360 Administration dialog, in the Groups area, click **Add**.
3. Type a name for the group.
4. To add a semantic type, click the Browse button, select the appropriate semantic type, and click **OK**.

© N. Harris Computer Corporation

5. Click **OK**.
6. In the Synonyms area, click **Add**.
7. Type a synonym for addition to the group and click **OK**. If the name you chose for the group is one of the words to be considered as synonyms, you must explicitly add that word to the group. Any synonym you add must:
   - not be a blank value
   - must not exceed 20 characters in length
   - must be considered a valid word, that is it must not contain large numbers of symbols, for example !"£$%^&*
8. Repeat this procedure from step 6 to add more words to the same group.
9. Repeat this procedure from step 2 to add another group and its member synonyms.
10. To export your synonyms to a text file:
    a. In the synonyms page, click **Export**.
    b. In the Export dialog, do one of the following:
       - In the File box, type the path and file name of the file you want to create.
       - Browse to the location of an existing text file, which you must be willing to replace with the exported synonyms. Select the existing file and click **Save**.
    c. In the Export dialog, click **Export**.
    d. When you see a message that the export was successful, click **OK** to dismiss the message and then click **Close**.
11. To import synonyms from a text file:
    a. In the Synonyms page of the Search 360 Administration dialog, click **Import**.
    b. In the Import dialog, do one of the following:
       - Type the path and file name of the file you want to import. Finish by pressing `Enter`.
       - Browse to the location of an existing text file. Select the existing file and click **Open**.
    c. Inspect the **Preview** window carefully to see that the file contains the expected data.
    d. To keep existing synonyms, turn on **Merge with existing data**. Use this option only if you are certain that all existing synonyms are as you want them.
    e. In the Import dialog:
       - Click **Import**. Click **OK** to dismiss the message saying that the import was successful.
       - Click **Close** to dismiss the Import dialog. You should now be able to see the groups and synonyms in the Search 360 Administration dialog.
12. Click **OK** to save the synonyms and close the Search 360 Administration dialog.

Results

When you have created several groups, be careful to select the correct group in the Name list, before editing the group or its member synonyms. You can delete or rename existing groups and synonyms or add new ones.

If you create many groups and synonyms, you might prefer to export the synonyms to a file and edit that file in a text editing application.

© N. Harris Computer Corporation

Excluding words in Search 360

Search 360 supports excluding a user-defined list of words from its index, so that those words make no contribution to search results. You can customize the list before you create the search index for the first time.

You might want to exclude words from the index because:

- They are short words that are almost never useful as search terms, such as "and", "as", "but", "by", "is", and "not".
- They are words that appear in so many records that searching for them doesn't return useful results.

To view the default list of excluded words and make changes to it, you can open and edit the text file named **SearchStopWords.txt**.

Procedure

1. Go to the folder `C:\Program Files\i2 iBase 10\Searching\` and open the **SearchStopWords.txt** file.
2. Add or remove words from the file, following the format of one word per line.
3. Save the changes to the file.
4. Configure the index service with your changes in place.

What to do next

When the index is up and running, you can test your changes. Create a sample record that uses an excluded word that you added to the **SearchStopWords.txt** file. If you search for that word, Search 360 does not find the sample record.

Configuring the index service

iBase Index Service Configuration is an application that indexes one or more iBase databases for Search 360.

About this task

For each database, you can set up an index service and a job schedule. Each time the job runs, it starts the index service, which obtains the location of the iBase database to index from the index database (IBaseIndexDB). No sensitive information is stored in this database as the index service connects to the SQL Server database using Windows™ authentication.

Note: Set up any synonyms and exclusions that you want to use in Search 360 before you configure the index service for the first time.

Procedure

To set up index services for your iBase database:

1. Make sure the database that you want to configure is closed, and that you are logged in using a suitable Windows or SQL Server account.
2. Select **All Programs** > **i2 iBase** > **iBase Index Service Configuration**.

   Running the Index Service Configuration tool for the first time on the local machine creates a database called IBaseIndexDB and a file named `Searching Config.xml` in the `Users`

© N. Harris Computer Corporation

application data folder on the local machine, specifically: `C:\ProgramData\i2\i2 iBase \<language>\Searching`.

If necessary, the name of the database is appended with a number to make it unique.

Note: The index database and configuration file should be included in your backup schedule. Losing either of these files requires you to reconfigure your iBase databases for indexing.

3. Click **Add** to set up configuration for this database. The **Configure Database** dialog opens.
4. In the **Configure Database** dialog you must populate the server name and database fields. This ensures that the Search 360 search field is available to iBase users when they log in to the database you have set up the index services for.
5. Advise the iBase database administrator of the date and time of the first scheduled job as this is not visible in iBase Designer.

Configuring the index service manually

You must run the iBase indexer on a computer with a network connection to the SQL Server instance that contains the iBase database to index.

To set up a search 360 index for a database stored remotely:

1. In iBase Designer, enable Search 360.
2. Open a command prompt and run the iBase indexer. For example:

```
"i2.iBase.SearchIndexerExe.exe" /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "password" /FullReindex /
ShowSummary
```

The following arguments are valid, but can't all be used at the same time.

Note: Newly upsized databases in iBase 10 have a randomly generated Database Access Token that must be supplied to Search 360 in order to connect to SQL Server based databases. You can locate your DAT string in iBase Designer. Go to the Database Access Token tab on the Database Properties page, and click **Copy**. An example DAT argument is provided below.

iBaseDBName

The name of the iBase database as stored in SQL Server.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password"
```

ServerName

The server name and SQL instance to access.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password"
```

Username

The username used to access the database.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password"
```

Password

The password used to access the database.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password"
```

© N. Harris Computer Corporation

FullReindex

This argument can be added to reindex the database.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password" /FullReindex
```

Database Access Token

This argument is required to provide the Database Access Token for iBase 10 databases.

```
"i2.iBase.SearchIndexerExe.exe" /iBaseDBName "YourDatabaseName" /
ServerName "YourServerName" /Fullreindex /DAT <DAT>"
```

Here is an argument with an example Database Access Token (DAT) value. Note each database has its own unique DAT.

```
"i2.iBase.SearchIndexerExe.exe" /iBaseDBName "YourDatabaseName" /
ServerName "(local)\SQL2019" /Fullreindex /DAT CE
+Q8EtNRZZG1NWTNUCQW8ZFcv6T9gQxLneC8Eiy48754Roij0cWpxWqa4Nog6
```

ShowSummary

This argument can be added to display a summary of the state of the index after the process finishes.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password" /ShowSummary
```

Pause

This argument can be added to pause the indexing at the end of the task that is being processed.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password" /Pause
```

DisplayRecordId

This argument can be added to display the Record ID as each record is processed.

Note: Depending on the number of records that are being processed, displaying this information might have performance implications.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password" /
DisplayRecordId
```

DisplayCallstack

This argument can be added to display the SQL command call stack if an error occurs.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password" /
DisplayCallstack
```

TrustServerCertificate

This argument can be added to trust the server certificate when connecting to the SQL Server instance.

```
SearchIndexerExe /iBaseDBName "User_Guide" /ServerName
 "(local)\SQL2019" /UserName "Admin" /Password "Password" /DAT "DAT
 token"/TrustServerCertificate
```

© N. Harris Computer Corporation

Troubleshooting indexing

Confirmation that the index service is configured successfully for a specific database is available in several places.

To detect if your search index is configured correctly, search for information in your database that you know is present, and check the results that are returned. If the results are not as you expect:

Check that the index was run since you added the record

The index only contains information about records that were available when the last index was created. The data and time that the index took place is displayed at the top of the following screens:

- **i2 iBase Designer** > **Tools** > **Search** > **Search 360 Administration**
- **i2 iBase** > **Tools** > **Analysis** > **Search 360**

If your record was added after this time, run the index to refresh the information it has stored.

Check the SQL Server indexing job for information about the indexing job

In SQL Server Management Studio, SQL Server Agent lists an i2_Search_Indexing job for the database, which reports success or failure in the SQL Server Agent job history. The messages this contains can be used to determine any problems such as issues with the installed iFilters for including different types of document.

Note: In addition, the SQL Server Agent must be running for the indexing to take place.

Check in the Windows™ Event Viewer for details about scheduled indexing

In the Windows Event Viewer, started and completed events are listed for the database. You can use the service logs for more information.

Note: Depending on the frequency with which the index service runs, you might want to change the properties of the Windows Event Viewer log file. For example, change its size or control what happens when the maximum log size is reached.

Run the indexing service from the command line

You can use the iBase Indexer tool via the command line to see the progress of an index in real time. For more information on running the index service, see Configuring the index service manually.

Managing the Index Service

The index service runs at the scheduled frequency if Search 360 is enabled in iBase Designer. Unlike most other SQL Server job properties, the enabled flag enables both the job and the schedule.

Attention: Jobs run if Search 360 is turned off in iBase Designer but exit without performing any indexing.

Use Index Service Configuration to manage the Search 360 index service:

- To modify the schedule for a Search 360 index service, click **Configure**. Do not use SQL Server to change the schedule.
- To stop or suspend the index service, click **Configure** and turn off the **Enabled** checkbox. This allows iBase users to continue to use Search 360 but without any updates for new information.
- To remove the index service, for example before uninstalling the Index Service Configuration tool or moving an iBase database, click **Delete**. This deletes the job and prevents the index tool from running.

© N. Harris Computer Corporation

Include the `IBaseIndexDB` database and Searching `Config.xml` configuration file in your backup schedule. Losing either of these files require you to reconfigure your iBase databases for indexing. Do not schedule index service backups for the same time as iBase backups.

Note: If you accidentally delete the iBase index database, you do not delete the jobs but they are not visible in the Index Service Configuration dialog. To resolve this, you need to either restore the database from backup or reschedule the jobs (which deletes and then recreates the jobs).

As a part of the regular maintenance of your index service you should monitor the transaction log and clear this down when it becomes too large. This is especially important after you complete a full index as many entries are added.

Setting up Word Search

There are two ways to search for text in an iBase database, using Word Search or Search 360 (in SQL Server databases only). Word Search allows you to index and search specific field types.

Word Search can search fields of the following types:

- Hyperlink
- Multi-Line Text and Multi-Line Text (Append Only)
- Security Classification Code
- Selected from Code List
- Suggested from Code List
- Text

To provide users with Word Search you must:

- Define how to create the index, by specifying which fields you want indexed.
- Create the index.
- Ensure that maintenance of the database includes rebuilding of the index at appropriate times. You must start each rebuild of the index.

More about Word Search indexes

Word Search indexing is performed within the database system. The indexes are stored in database tables. The location of the tables varies for different types of databases:

- In SQL Server databases, the Word Search index is stored in the main database. Backing up the database is sufficient to back up the index.
- In Microsoft Access databases, the Word Search index is stored in a separate database, created in the same folder as the database file. If the database is called `dbname.idb`, the index file is called `dbname.idx`. Add the `dbname.idx` file to your backup lists.

Note: The .idx file is a password-protected database file, which you can open in Microsoft Access if you want to. The password is the database password that you can see in the Advanced page of the Options dialog.

Building a Word Search index

The build process is used to generate an index of words in the parts of the database chosen for indexing and the records in which the words occur.

A full build scans all chosen records and generates a new index.

© N. Harris Computer Corporation

An incremental build scans only those records that have changed since the last full build was performed, and modifies an existing index.

Any change to the build conditions, for example the indexing of an additional field, means that you must delete the index first, and then perform a full build.

Quick Start

Here is an overview of the procedure for building a Word Search index.

1. Select **Tools > Search Word Search Indexing**.
2. If there is an existing index, click **Delete** Index.
3. Make any configuration changes. It can take several minutes to create a Word Search index for a large database. If you want to make an index for testing, it is enough to select just one entity type or link type in the **Fields** page of the dialog.
4. Click **Full Build**.

   iBase Designer displays the Word Search Index Build dialog to help you monitor the progress of the build.

   While the build is progressing, the dialog contains two progress bars. The upper bar shows progress through each table, corresponding to each selected entity or link type. The lower bar shows the total word count for the index.

   Note: You can click **Stop** to abandon the build if it is proceeding slowly, but this leaves an incomplete index. If you click **Stop**, delete the index to avoid anybody using the incomplete index.
5. When the build is complete, the dialog changes to display a summary of the words and the time taken. Click **OK** to close the dialog.

At this point, you can inspect the index using the Index Browser page of the Word Search Administration dialog or exit iBase Designer. Any permitted user can start iBase and use the word search index, by selecting **Analysis > Word Search**. Users can also see a read-only version of the dialog that you have used in iBase Designer, enabling them to see which fields are indexed.

Word search

You can find the records anywhere in the database that contain specified text by using a word search. You do not need to know which field the text might occur in.

In addition to finding specific words you can also use the following to broaden your search:

Synonyms - words that have the same meaning

Synonyms are lists of words such that whenever you specify a particular word to search for, all the relevant words on the synonyms list are also searched for. The synonyms list might contain all the words that have the same meaning; for example, synonyms for Firearm might be: Firearm, Shotgun, Rifle, Hand gun, Revolver, Pistol. The lists are pre-defined in the database design, so you cannot change them here. However, you can see which synonyms are searched for.

Soundex - words that sound the same

Soundex means that words that sound the same as your specified word are also searched for. For example, using Soundex you might specify 'check' and find 'cheque'.

Any words added to the database since the database administrator last generated the Word Search index will not be found. The date when the index was last updated is shown. If you need to see details of how the index is defined, click **Index** on the Enter Words page.

© N. Harris Computer Corporation

Use the:

- Enter Words page if you are interested in specific words (or synonyms or similar sounding words) and where they appear.
- Word Index page if you are more interested in how frequently words occur in the database.

Finding records containing specific words

Use the Enter Words page if you are interested in specific words (or synonyms or similar sounding words) and where they appear:

1. Select **Analysis > Word Search**.
2. In the Word Search dialog, click the Enter Words tab to display the Enter words page.
3. Click the Search for box and enter one or more words to search for, separating words with spaces. You can use wildcards to broaden the search. The search ignores the lettercase. It might exclude certain other things, such as entirely numeric values. See 'What you can and cannot search' below for details.
4. In the Combine area, select one of the following:
   - And - the record must contain all your specified words or synonyms of those words if the User Defined checkbox is turned on in the Synonyms area.
   - Or - the record must contain at least one of your words, or one of the synonyms if the User Defined checkbox in the Synonyms area is turned on.
5. In the Type area, select either **Normal** or **Soundex** (includes similar-sounding words).

   Note: A list of words appears whenever any member of the list is specified in the Search for box. All of these words are searched for, in addition to the specified words.
6. If you want the search words to be highlighted in any records found by the word search, turn on the Highlight Words Found checkbox.
7. Click **Search**. Any records that contain the search words are then displayed. The records are identified by their label as defined in the current labeling scheme.

Finding by word frequency

Follow these steps if you are interested in how frequently words occur in the database:

1. In the Word Search dialog, click the Word Index tab to display the Word Index page.
2. Select Occurrences and then either Most or Least (frequent).
3. Specify how many words to list; use the upper button next to the number of words to increase it; use the lower button to decrease it. Alternatively, just click the box and type into it.
4. Click **Find**. The word list will show the most or least frequently occurring words. No records are found as a result of this step.

   Note: It is possible to exclude unwanted words such as "of", "for", and "from" from your search results. Ask your database administrator to set up an exclusion list.
5. In the list of words, select one of the words and then click Search for highlighted word to find the records that contain the selected word. The number of records that will be found is shown in the Count column.

Finding using the beginnings of words

Follow these steps to find records containing words that start with specific characters:

© N. Harris Computer Corporation

1. In the Word Search dialog, click the **Word Index**.
2. Select Beginning with if you want to find words that start with your specified characters.
3. Click the box and type in the starting characters. As you type, the word list shows the matching words and their frequency. No records are found as a result of this step.
4. In the list of words, select an entry and click Search for highlighted word to find the records that contain the selected word. The number of records that will be found is shown in the Count column.

What you can and cannot search

Using a Word Search, you can search fields of the following types:

- Hyperlink
- Multi-line Text and Multi-line Text (Append Only)
- Security Classification Code
- Selected from Code List
- Suggested from Code List
- Text

You cannot search:

- Document type fields - to search the text of embedded documents, use a fuzzy search (if available, SQL Server databases only).
- For punctuation, because punctuation is treated as a word break.
- For special characters, such as €, ~, <, +.
- For words over a certain length (the maximum length is set by the database administrator)
- For purely numeric values (unless your database administrator has chosen to use this option)

To find out the maximum word length or whether you can search for purely numeric values:

1. In the Word Search dialog, click Index. The Word Search Index Build dialog is displayed.
2. Click the Advanced tab to display the Advanced page. The page displays the maximum word length.
3. If you can search for purely numeric values then the Exclude numerics option will be turned off.

Note: By default, entirely numeric values are excluded from the Word Search index. Consider these examples:

Example	**Result if numerics are excluded**
BMW 320	320 is a numeric value, therefore BMW is indexed but 320 is excluded
BMW 320i	320i is not a completely numeric value, therefore both BMW and 320i are indexed
0012-3963	0012-3963 is indexed as a single non-numeric word
-3	Excluded because it is a numeric value

© N. Harris Computer Corporation

Example	Result if numerics are excluded
+3	Excluded because it is a numeric value

Building the Word Search index

The Word Search Index is a list of the words that can be searched when using Word Search in iBase. It contains the words extracted from specific (text-type) fields for selected entity and link types.

About this task

Configuration consists of:

Selecting the fields which can be indexed.

You can specifically exclude or include the values of specific fields.

Excluding specific words from the index, such as words that appear in almost all records.

You might want to exclude words because:

- The words occur in all or nearly all records and contribute little to the searchability of the database, while adding greatly to the size of the search index.
- The words relate to administrative or other uses that you do not want to be visible to all users of text search.

The excluded word list must be an unformatted text file containing one word per line with no blank lines. You cannot exclude words that are longer than 20 characters.

Double quotation marks are optional, but if they appear they must do so in pairs, one each at the start and end of a line. Paired double quotation marks are ignored and only the words between them are imported.

The words can be in any order, use uppercase, lowercase or any mixture of case, and may be duplicated within the file. Duplicates are removed, and the list is sorted when importing; the case remains unchanged.

If you see the message ERRORS OCCURRED AT, check the line number in the list to locate the errors. The most likely causes are:

- A space character anywhere in the file, possibly before or after one of the words.
- A completely blank line.
- A mismatched or badly placed double quotation mark.

Note: iBase is supplied with an example file of common words. The file is formatted for importing into the Word Search system. Inspect the file to see if it is suitable for your database: `C:\ProgramData\i2\i2 iBase\en-US\Configuration`

You can import this file to provide a starting point, which you can then modify.

Set a limit on the length of the words that you can search for, in order to create a smaller index that returns search results more quickly.

Allows you to set the maximum length of entries in the index. The default setting of 10 is usually satisfactory. Longer words are trimmed to this length when placed in the index. For example, if the first 10 characters are indexed and the word `shoplifting` is indexed, this list shows only the

© N. Harris Computer Corporation

first 10 characters, `shopliftin`. A user can search for and find this word with either of the terms `shoplifting` or `shopliftin`.

If you set a limit of less than 10, a smaller index is created and search results are returned more quickly. However, the search results might contain unexpected results. For example, it will find words that are different but which have the same first few characters. You can check for false results by inspecting the found records.

Deciding whether to index words that contain only numerical characters.

By default, entirely numeric words are excluded from the index, but you can opt to include these entries.

For example, the text 320 in the BMW 320 is not indexed if numerics are excluded. The text 320i in the entry BMW 320i is indexed because 320i is not a completely numeric value.

A more complex example concerns the hyphen or minus character (-), where the character's position affects the interpretation. Numerals surrounding a hyphen character are considered non-numeric. For example, a banking reference for an account in the format 0012-3963 is treated as a single non-numeric word, not two numerics. Conversely, a number written with a leading minus character, in the style -3, is treated as a numeric (as is the positive equivalent, +3).

If you need to change the configuration of the index, then you must first delete the current index by clicking **Delete**. When you have built the index (by clicking Full Build), update it on a regular basis in order to include data from new records. You can use an incremental build for this (click Increment).

Note: iBase users can review how the index was built, in order to understand what can and cannot be searched for, but they cannot update it themselves.

Note: Words need not necessarily contain only letter characters; they can contain, or consist entirely of, numbers for example. There is however the option of excluding words consisting entirely of numbers. In addition, punctuation characters are not included in indexed words. This is because these characters are used to determine where words start and end. For example the space in 'first word' or the underscore in 'first_word' means the index would contain the words 'first' and 'word'.

Procedure

1. Select **Tools > Search > Word Search Indexing**.
2. In the Word Search Index Build dialog, click the **Fields** tab to display the Fields page.
3. Click **Delete** to delete the current index.
4. To configure the indexed fields, turn on or off the checkbox next to the entity type, link type, or field. If you include an entity or link type it will initially include all its fields, you can then turn off the checkbox next to the fields that you want to exclude.
5. In the Excluded Words page of the Word Search Index Build dialog, you can exclude words from the index. You can then review which words you have excluded, either on screen or by exporting those words to a text file.
   - **Exclude Word**

     Enter a word in the upper text box of the dialog, then click this button to transfer the word to the list in the lower list box, where it appears in alphabetic order.
   - **Remove Selected**

     Select one or more words in the list box, then click this button to remove all selected words from the list.
   - **Import Words**

© N. Harris Computer Corporation

Click this button to import a text file of the words you have chosen to exclude. In the Import dialog:

- Enter a file name or find a file using the **Browse** button and **Select Import File** dialog.
- Inspect the preview in the Import dialog to check that the files contains the word you want to exclude and check that the message ERRORS OCCURRED AT does not appear.

  Note: If you see this message, edit the file so that it meets the required format as described next, save the file, and click **Refresh** to reread the file.
- Turn on the **Merge with existing data** checkbox if you want to keep the existing list.
- When you have identified a suitable file, click **Import** to read the list from this file.

- **Export Words**

  Click this button to create a text file of the words you have chosen to exclude. In the **Export** dialog, enter a file name or find an existing file using the **Browse** button and **Specify Export File** dialog. When you have identified a file, click **Export** to write the list into this file.

You cannot exclude words that are longer than 20 characters.

When you have finished modifying the list of excluded words, click **Apply** to confirm your changes.

6. In the advanced index options, select the **Number of Characters to index** and whether to **Exclude Numerics**.
7. Click **Full Build** to generate a new index.

   Note: When the build is completed, you can use the Fields tab to view the information that has been included or excluded from the index.

Setting up synonyms in Word Search

Users can search for groups of words known as synonyms, that is words that have the same or very similar meanings. You can define synonyms for use with Word Search in iBase.

About this task

Synonyms are defined in the Synonyms page of the Word Search Administration dialog. For details of the Index Browser page, see Modifying existing indexes.

You must add synonyms in groups, even if you create only one group. Each group should contain words that users consider as being the same for the purposes of their search. For example, a synonym group named Firearm might have the values: Firearm, Shotgun, Rifle, Handgun, Revolver, Pistol. Groups can also contain colloquialisms, for example a group named 'Cannabis' might have defined values of Marijuana, Weed, Hash, Smoke, and Skunk.

If you create many groups and synonyms, you may prefer to export the synonyms to a file and edit that file in a text editing application.

To import a synonym list, you require a text file of the correct format in a known location on an accessible disk volume. This example defines a synonym group named Firearm with the values: Firearm, Shotgun, Rifle, Handgun, Revolver, Pistol:

```
[Firearm]
Firearm
Shotgun
Rifle
Handgun
Revolver
```

© N. Harris Computer Corporation

```
Pistol
```

When the user enters a search term into the Enter Word page, the list in the Synonyms area remains empty until they enter a word that has been defined as a member of a synonym group. iBase then displays the full list of synonyms and the user can turn on the User Defined option if they want to find records holding the synonyms.

Note: Name variants are not available in Word Search.

Procedure

1. Select **Tools > Search > Word Search Administration**.
2. To add groups and synonyms:
   a. Click the Synonyms tab to display the Synonyms page.
   b. In the Groups area, click **Add**.
   c. Enter a name for the group and click **OK**.
   d. In the Synonyms area, click **Add**.
   e. Enter a synonym to add to the group and click **OK**.

      Note: If the name you chose for the group is one of the words to be considered as synonyms, you must explicitly add that word to the group.

      When you have several groups, be careful to select the correct group in the Name list, before editing the group or its member synonyms. You can delete or rename existing groups and synonyms or add new ones.

      If you create many groups and synonyms, you may prefer to export the synonyms to a file and edit that file in a text editing application.
   f. Repeat this procedure from step 3 to add more words to the same group.
   g. Repeat this procedure from step 1 to add another group and its member synonyms.
   h. Click **OK** to save the synonyms and close the Word Search Administration dialog.

   When you have several groups, be careful to select the correct group in the Name list, before editing the group or its member synonyms. You can delete or rename existing groups and synonyms or add new ones.

   If you create many groups and synonyms, you may prefer to export the synonyms to a file and edit that file in a text editing application.
3. To export synonyms for editing or for import into another database:
   a. In the Word Search Administration dialog, click the **Synonyms** tab.
   b. Click **Export**.
   c. In the Export dialog, do one of the following:
      - In the File box, enter the path and file name of the file you want to create.
      - Click the Browse button and browse to the location of an existing text file, which you want to replace with the exported synonyms. Select the existing file and click **Save**.
   d. In the Export dialog, click **Export**. A message is displayed indicating that the export was successful.
   e. When you have seen a message that the export was successful, click **OK** to dismiss the message and click **Close** to dismiss the Export dialog.

© N. Harris Computer Corporation

You can inspect the exported file with Notepad or any other text editing application.

4. To import synonyms from a text file:
   a. In the Word Search Administration dialog, click the **Synonyms** tab.
   b. Click **Import**.
   c. In the Import dialog, do one of the following:
      - In the File box, enter the path and file name of the file you want to import. Finish by pressing the `Enter` key or clicking **Refresh**.
      - Click the **Browse** button to locate an existing text file. Select the file and click **Open**.
   d. Inspect the **Preview** window carefully to see that the file contains the expected data. If a message is displayed reporting that the file is missing or contains an error, you can choose another file or edit the file to remove the error. Click **Refresh** to preview the new contents if you change the file.
   e. To keep existing synonyms, turn on **Merge with existing data**. Use this option only if you are certain that all existing synonyms are as you want them.
   f. In the Import dialog, click **Import**. Click **OK** to dismiss the message saying that the import was successful.
   g. Click **Close** to dismiss the Import dialog. You should now be able to see the groups and synonyms in the Word Search Administration dialog.
   h. Click **OK** to save the synonyms and close the Word Search Administration dialog.

Modifying existing indexes

The index browser displays words in the index, allowing you to choose words for display by their frequency in the index or by their initial characters. If required, you can exclude some of these displayed words from the index.

About this task

The index browser has two key areas. The upper part allows you to search the index for specific words. The lower part displays the words you have found, and allows you to exclude words from the index.

Note: If the index has been built for a number of leading characters (set in the **Advanced** page of the Word Search Index Build dialog) then only the specified number of characters appear in the list. For example, if the first 10 characters are indexed and the word "shoplifting" is indexed, this list shows only the first 10 characters, shopliftin. Each index entry shows the number of times it appears in the index.

Procedure

1. **Tools > Search > Word Search Administration**.
2. To display the index browser, in the Word Search Administration dialog, click **Index Browser**. The upper part of the dialog helps you find words to display. The lower part displays the words you have found, and allows you to exclude any found words from the index.
3. To find and list words:
   - Select **Occurrences** to find common (**Most**) or rarely (**Least**) occurring words.
   - Select **Beginning with** to find words beginning with the character or characters that you enter in the box.

   Click **Find** when you have set up your condition.
4. To work with the listed words, select the word and:

© N. Harris Computer Corporation

- Select **Exclude** to exclude a word from the index.
- Select **Relist** to return the selected word to the index. The returned word appears at the bottom of the Words Found list, without a count.

5. Click **OK** to confirm your choice of excluded words and close the dialog.

Setting up Full-Text search

You can find records that contain specified text anywhere in the database by using a Full-Text Search. Full-Text Search requires an SQL Server database, which must be provided by a server that is running Microsoft™ SQL Server with the Microsoft Search service.

To provide Full-Text Searching within a database, you must:

- Define how to create the index.
- Create the index.
- Set up rebuilding or updating of the index at appropriate times.

Full-Text Search works in collaboration with SQL Server but it is separate from SQL Server and is installed only as part of a custom installation.

The Microsoft Search service is used to search, catalog, and index the database. This service maintains catalogs and indexes outside SQL Server, on the server computer's file system.

Creating an index

Full-Text searching needs the database to be indexed before terms can be found.

Procedure

To create a new Full-Text Search index:

1. Select **Tools** > **Full-Text Search Indexing**.
2. In the Fields page, select the fields to index.

   Note: The list of entity types and link types might be shorter than you expect. Only the types with fields that Full-Text Search can index are available.
3. Click **Apply**.

Results

iBase Designer starts building the index for your selection of fields, with a default option for updating the index as the database changes. If you want, you can click the Index Maintenance tab to display the Index Maintenance page then click **Index Status** to check the progress of building the index.

Any permitted user can now start iBase and use the full-text search index, by selecting **Analysis** > **Full-Text Search**. Users can also see a read-only version of the iBase Designer index setup, enabling them to see which fields are indexed.

Note: Person name variants are automatically imported when you build a Full-Text Search index. If Name Variants are imported, the SMB library version is displayed in the lower-left corner.

There is no explicit command to delete a Full-Text Search index. You can remove the index and free the server resources required to provide and maintain the index by deselecting all fields in the Fields page.

© N. Harris Computer Corporation

Selecting fields for indexing

You can exclude some fields from the index because including those fields add little to the value of the index or add too much to the size of the index. For example, you can exclude fields that hold geocoding data or any field that holds a unique identifier for the record that holds it.

Procedure

1. Select **Tools** > **Full-Text Search Indexing**.
2. On the Fields page:

Option	Description
Expand	Expands the tree view to show all fields of all entity types and link types.
Collapse	Collapses the tree view to show only entity types and link types.
Included	Show the indexed fields, which are listed in columns for type and field.
Excluded	Show the non-indexed fields, which are listed in columns for type and field.

Excluding words from the index

You can select words for exclusion from the index.

You might want to exclude words because:

- The words occur in all, or nearly all, records and contribute little to the search-ability of the database.
- The words relate to administrative or other uses that you do not want to be visible to all users of text search.

You might also want to restore to the index some common words that SQL Server excludes by default, as being common words in the relevant language, but that have some special meaning in the databases.

These excluded words are for reference only. The words that are excluded by the server affect the results of Full-Text Search in iBase.

Select **Tools** > **Search** > **Full-Text Search Indexing** > **Excluded Words** to display the Excluded Words page:

Exclude Word

Add words to the list to exclude. Words are listed in alphabetical order.

Remove Selected

Removes all selected words from the list.

Import Words

Import a text file of the words that you have chosen to exclude:

© N. Harris Computer Corporation

1. Enter a file name or browse for a file.
2. Inspect the preview to check that the file contains the word you want to exclude and check that the message ERRORS OCCURRED AT does not appear.
3. If you see this message, edit the file so that it meets the required format, save the file, and click **Refresh**.
4. Turn on **Merge with existing data** if you want to keep any existing terms.
5. Click **Import**.

Export Words

Creates a text file of the words that you have chosen to exclude.

When you have finished modifying the list of excluded words, click **Apply** to confirm your changes.

Format of the excluded word list

The excluded word list must be an unformatted text file containing one word per line with no blank lines. You cannot exclude words that are longer than 20 characters.

Double quotation marks are optional, but if they appear they must do so in pairs, one each at the start and end of a line. iBase Designer imports the word between the marks, ignoring the paired double quotation marks.

The words can be in any order or case (uppercase, lowercase or any mixture of the two), and may be duplicated within the file. iBase Designer sorts the list when importing, removes duplicates, and leaves the case unchanged.

If you see the message ERRORS OCCURRED AT, check the line number in the list to locate the errors. The most likely causes are:

- A space character anywhere in the file, possibly before or after one of the words.
- A completely blank line.
- A mismatched or badly placed double quotation mark.

Setting up synonyms

Users can search for groups of words known as synonyms, that is words that have the same or very similar meanings. You can define synonyms to help users with their analysis.

Synonyms can be useful in these types of situation:

Searched text	**Example**
Is from different sources	Text might use a mixture of spellings from US and UK English and users might want to treat 'color' and 'colour' or 'fender' and 'bumper' as synonyms.
Might have variations or errors in data entry	Many drugs have various names in formal and casual usage. Also, a name that uses accented characters might be presented in a different order or be spelled in different ways when converted to a form without accented characters: Müller and Mueller.

© N. Harris Computer Corporation

Searched text	Example
Is too precise for the search	Color names specified by the paint manufacturer's names, such as crimson, scarlet, and flame, where the search term that is derived from a witness statement is red.

When you set up synonyms, users can choose whether to use those synonyms each time their search term includes one of the synonyms.

You must add synonyms in groups, even if you create only one group. Each group contains words that users can consider as being the same for the purposes of their search.

You can export synonyms that are defined in one database for external editing or transfer to other databases.

Editing synonyms

To edit synonyms:

1. Select **Tools > Search > Full-Text Search Administration**.
2. In the Groups area, click **Add**.
3. Type a name for the group and click **OK**.
4. In the Synonyms area, click **Add**.
5. Type a synonym for addition to the group and click **OK**.

   Note: If the name you chose for the group is one of the words to be considered as synonyms, you must explicitly add that word to the group.
6. Repeat this procedure from step 4 to add more words to the same group.
7. Repeat this procedure from step 2 to add another group and its member synonyms.
8. Click **OK** to save the synonyms.

Exporting synonyms

To export all synonyms to a text file:

1. Select **Tools > Search > Full-Text Search Administration**.
2. Click **Export**.
3. Either:
   - In the **File** box, type the path and file name.
   - Browse to the location of a file, which you must be willing to replace with the exported synonyms. Select the file and click **Save**.
4. Click **Export**.

   A message is displayed indicating whether the export was successful.

You can inspect the exported file with Notepad or any other text editing application.

© N. Harris Computer Corporation

Importing synonyms

To import a synonym list, you require a text file of the correct format. This example defines a synonym list with one group that is named Firearm and the values: Firearm, Shotgun, Rifle, Hand gun, Revolver, Pistol.

```
[Firearm]
Firearm
Shotgun
Rifle
Hand gun
Revolver
Pistol
```

To import synonyms from a text file:

1. Select **Tools > Search > Full-Text Search Administration**.
2. Either:
   - In the **File** box, type the path and file name.
   - Browse to the file, and click **Open**.
3. Inspect the Preview window carefully to see that the file contains the expected data. If iBase Designer reports that the file is missing or contains an error, you can choose another file or edit the file to remove the error. Click Refresh to preview the new contents if you change the file. Click Close if you want to abandon the import.
4. Decide whether you want to keep any synonyms already defined for this database.
   - To keep existing synonyms, turn on the option Merge with existing data. Use this option only if you are certain that all existing synonyms are as you want them.
   - To clear all synonyms and use only the synonyms defined in the file you are importing, turn off the option Merge with existing data.
5. Click **Import**, and then click **OK** to dismiss the message saying that the import was successful.
6. Click **Close**. You should now be able to see the groups and synonyms.
7. Click **OK**.

Person name variants

Person name variants are provided by i2 and imported automatically when you first build a Full-Text Search index. If a more recent version of the SMB library is installed, then you are prompted to upgrade the name variants file when you next open **Tools** > **Search** > **Full-Text Search Indexing**.

If you install a new version of the SMB library, you can upgrade to the person name variants file provided as part of the new SMB library by selecting **Tools** > **Database Design** > **Update Person Name Variants**.

You can view the version of the SMB library that is currently installed by looking in the lower left corner of Full-Text Index Maintenance.

The Update Person Name Variants command is unavailable when:

- The Full-Text Search plug-in is not loaded.
- You do not have both Security Administrator and Database Administrator permissions.

© N. Harris Computer Corporation

Updating an index

You can control how iBase Designer updates the Full-Text Search index. In routine use, the search service updates the index without you needing to request updates. However, you can use various settings to balance the need for an up-to-date index with the performance of the server. For example, you might choose to suspend index updates while you import or edit large volumes.

About this task

You can set up a maintenance schedule for rebuilding the index. The choices that you can make allow various combinations of full rebuilding or incremental updates.

The schedule can depend on several factors:

- The size of the database
- The amount of change in the database
- The timing of the changes, particularly whether changes are typically made by near-continuous data entry or made by less frequent but larger batch operations.
- The available memory, disk, and processor resource
- The load on the server that is presented by interactive or batch tasks such as data entry, analysis, and data maintenance.

Your aim is to keep the search index as up-to-date as possible without noticeably slowing down the server's performance in other areas.

Given these factors, the best way to update an index can vary over time as the size and uses of the database change.

Guidelines for choosing an update method:

- Use change tracking and continuous incremental updates when enough processor and memory resources are available, the value of an up-to-date index is high, and records are being added or changed at slow or moderate rates, slow enough for the indexing to keep up with changes.
- Use change tracking with scheduled incremental updates when processor and memory can be used at scheduled times, there is enough disk space for storing changes, and changes between the scheduled times are small enough for the incremental updates to take less time than full rebuilds. This is typical of a database with large numbers of interactive users.
- Use a full rebuild when a large percentage of records changes or are added at particular times, especially if the change is concentrated at one time each day or week. This is typical of a database in which most records are added by import. If imports take place in small batches over long periods of time, consider change tracking with scheduled or continuous incremental updates.
- Use incremental updates when a large number, but not a large percentage, of records change at one time. If many records change over an extended period, consider change tracking with scheduled or continuous incremental updates.

Monitor the effects of your choice. Where possible, try more than one method, for example so that you know the time taken and the server loading, both for a typical incremental update and for a full rebuild.

The Index Maintenance page is divided into areas for tracking changes and updating; performing full builds manually or with a schedule; and checking the status of the index.

- Track Database Changes

  Update the index in response to changes in the data. Updates are changes to an existing index.
- Update Index

© N. Harris Computer Corporation

Start an immediate update of the index.

- Incremental Updates

  Enable automatic incremental updates, and choose either continuous or scheduled updates.

- Continuous automatic update of the index

  Keeps the index as up-to-date as possible.

  Note: There is some continuous processing load on the server when you choose this option and you might want to suspend automatic updates for large batch imports or similar operations that change large amounts of data. After the operation, perform a manual update using one of these methods. For moderate amounts of change, click Update Index; this method always works but might not be the fastest. For large changes, it is quicker to turn off **Track Database Changes** and click **Rebuild** in the Full Index Build area; after the rebuild, turn on **Track Database Changes** to resume automatic updates.

- Scheduled automatic update of the index

  Make updates at regular times on particular days, based on changes to the data since the last update. You must also click Change to choose an appropriate schedule.

- Change

  Set up a new schedule for regular incremental updates of the index. For example:

  - If data is added interactively you might choose a schedule of Every 2 weeks at 00:00, on Sun, Mon, Tue, Wed, Thu, Fri, Sat.
  - If you use a database for read-only data that is added weekly, you might reasonably update the index on the day of updates soon after the completion of the import.

Procedure

To rebuild the index:

1. Select **Tools** > **Search** > **Full-Text Search Indexing** and select the **Index Maintenance** tab.
2. Turn off **Track Database Changes**.
3. To build a full index, use the options in the Full Index Build area:

Option	**Description**
Rebuild	Start an immediate rebuild of the index.
Schedule Full Rebuilds of the Index	Specify full rebuilds on a scheduled basis.
Change	Set up a new schedule for regular full rebuilds of the index.

Tip: Click **Index Status** to display the status.

© N. Harris Computer Corporation

Installing iFilters

On the machine that is running the indexing tools, you can install additional iFilters to allow documents of a set type to be searched using Full-Text search. These third-party iFilters can support additional document types or metadata from the standard iFilters that are supplied with the operating system.

About this task

iFilters are used to access documents and store the details in the index. You can see the iFilters that you currently have installed and associated with specific files using the **Indexing Options** in the control panel for your operating system all the files are listed in **Indexing Options** > **Advanced** > **File Types**.

Note: If an iFilter is installed after the index is built, the index will need to be rebuilt for the document type fields before the document will be indexed correctly.

Procedure

1. Install a supported iFilter on the SQL Server.
2. The following steps depend on which iFilter you installed:
   a. If you have installed any PDF iFilter (or other iFilter that does not have a signed binary) run the following command in SQL Server Management Studio for your instance:

      ```
      exec sp_fulltext_service 'verify_signature', 0;
      ```

   b. If you have installed any new iFilter, run the following command in SQL Server Management Studio for your instance:

      ```
      exec sp_fulltext_service 'load_os_resources', 1;
      ```

   c. Stop and restart the SQL Server service.
   d. Rebuild the Full Text Search index in iBase Designer to include the documents.

   Note: To determine which file extensions the iBase database currently has iFilters for, you can run the following SQL Script:

   ```
   select document_type, path from sys.fulltext_document_types;
   ```

Configuring alerting

Alerting monitors records to detect when an item of interest changes or is viewed by someone. To monitor items of interest, alert definitions are added to records (single or multiple) and to queries. When a change is detected, an alert is raised.

Alerting is available in SQL Server databases only. When you initialize an SQL Server database for alerting, you turn on Audit History and create SQL Server triggers and jobs to raise and process alerts. To correctly process results, the audit generates queries that are known as alert definitions based on each user's requests. These queries are run on a schedule.

Every time the alerting jobs run, they generate one alert for each alert definition where actions are detected, batching up all detected actions so that users receive a single alert or email.

Note: The date and time of an alert is the date and time that the job is processed at the server. Alerting requires that the server on which SQL Server is running has the same date and time as the iBase client. A significant time difference between an iBase server and its clients might prevent the server from detecting the events that trigger iBase alerts.

There are four types of alert:

Record Viewed alerts

© N. Harris Computer Corporation

A Record Viewed alert is raised whenever the record is:

- Displayed in a record list, for example as a result of finding, browsing, or opening a set
- Displayed in Show or on a datasheet
- Displayed when soft deleted records are purged or restored
- Listed or viewed in Audit Viewer or the Audit History (but not when you are setting up alert definitions or viewing alerts)
- Listed as a link end record
- Viewed on an iBase link chart
- Exported or sent to an Analyst's Notebook chart

The alert is raised when the record is first shown or listed.

Record Changed alerts

A Record Changed alert on entities is raised when:

- Any entity fields are changed
- New links are added to the entity
- The strength or direction of any link to the entity is changed
- The entity is deleted
- Any links to the entity are deleted
- Entities or links are soft deleted or restored

Note: Changing a field on any links to the entity is not a change to the entity itself.

A Record Changed alert on links is raised when:

- Any link fields are changed
- The strength or direction is changed
- A link end entity is replaced by a different link end entity
- The link is deleted
- A link end entity is deleted causing the link to be deleted
- Link end entities or links are soft deleted or restored

Note: Changing a field on any link end entities is not a change to the link itself.

Records Added alerts

A Records Added alert is raised whenever an extra record is found that matches the selection criteria for the Query. This might be for the following reasons:

- New record added that matches the Query
- Changed so that it now matches the Query
- Restored (having previously been soft deleted)
- Changes to your permissions, which mean that you can now see more records

Records Removed alerts

A Records Removed alert is raised whenever a record that previously matched the selection criteria for the Query is no longer found. This removal can be for the following reasons:

- Changed so that it no longer matches the Query

© N. Harris Computer Corporation

- Deleted
- Changes to your permissions, which mean that you can now see fewer records

Note:

- Alerting is available in SQL Server databases only
- You can only add alert definitions if you are permitted to do so
- Email alerts can only be sent if your system administrator has enabled this feature

Users use the alert details to determine the exact nature of the changes. Users who are denied access to the audit history receive alerts and can see the alert summaries but are unable to see the details of the alert. The alert details are taken from the audit history and the same details are displayed regardless of the audit level of the database.

Note: Contact i2 Support for details about:

- Setting up alerting on SQL Server Express (you must to use Windows scheduling).
- Alerting in a replicated environment.

Permissions configure alerting

The user who configures alerting must be an iBase system administrator. They must also have an SQL Server login for the msdb system database that is a member of both the public and SQLAgentUserRole database roles.

Alerting security

Alerting complies with the security that you have set up for your iBase system and case-control, if it is used.

Users cannot be added as subscribers to alert definitions that would give them access to denied records. Specifically, when a user adds an alert definition, the only users eligible to subscribe are users who can see the same (or a wider range of) records, entity types, link types, and fields as the user adding the alert definition.

Alert definitions are updated whenever group permissions are changed in the security file. This update occurs after exiting from iBase Designer and when any user next opens the database in iBase or iBase Designer. For this reason, after making security changes and logging off, reopen the database as soon as possible to apply the changes.

Changes to user permissions can mean two things:

- If the user is a subscriber, then they are automatically unsubscribed from any alerts that monitor the denied item. This also means that the user can no longer view the details of any alerts already in their alerting Inbox.
- If the user is the owner of the alert definition, then the alert definition is automatically deleted. This also means that subscribers can no longer view the details of any alerts already in their alerting Inbox.

No sensitive information is included in an email alert. The information is restricted to the name of the alert and the emails are recorded in the audit log.

You can also deny users the right to add alert definitions or the right to add alert definitions that send emails. In iBase Designer, select **Security > System Commands Access Control command**, and on the Access Denied page, turn on:

- **Alerting** to deny permission to add alert definitions

© N. Harris Computer Corporation

- **Alert Email Notifications** checkbox to deny the permission to send email alerts.

Note: Alerting applies only to records in the main database and not to any database subsets. However, when the database subset is synchronized with the main database, alerts are raised in the usual way on any records that are modified as a result of the synchronization.

Attention: Exporting data as XML does not raise alerts because of the large volume of data that might be exported, potentially all the records in the database.

Initializing alerting

Alerting needs to be initialized in iBase Designer before alerting can be started.

Procedure

To initialize alerting:

1. In iBase Designer, select **Tools > Feature Availability > Alerting Configuration**.
2. **Optional:** Select the database if you plan to run the alerting jobs in a replicated environment.
3. Enter a login name, and password, with appropriate permissions for configuring alerting.

   Note: The login must also be a member of the SQLAgentUserRole database role.
4. Enter the number of alerting jobs that will run on the server.
5. Click **Initialize Alerting**.

What to do next

After initializing alerting you need to back up the security file if it is in SQL Server format. See Backing Up iBase Databases.

In addition, if you would like to send email alerts you need to set up the SQL Server instance for email. Alerting uses database mail with the default public profile. Refer to the SQL Server documentation for information.

Note: Users must have valid email addresses as iBase cannot verify these.

Opening the database in iBase Designer after alerting is configured

Before you can open a database in iBase Designer, you must have exclusive access to that database. This means that no users can be connected to that database in iBase, Analyst's Notebook, and no services for that database should be running.

You will be asked whether you want to shut down the database services. After waiting a few minutes for the services to be shut down, you can attempt to open the database. If you do not succeed, for example because users are still connected to the database, you are asked whether you want to retry or cancel. If you choose to cancel, the alerting jobs will automatically resume and the database will not be opened.

On exiting iBase Designer, the alerting jobs are automatically resumed.

© N. Harris Computer Corporation

Scheduling alerting jobs

By default, a single alerting job runs every 10 minutes on the server. However, you can change the number of alerting jobs and their frequency.

About this task

- You can configure alerting to use up to 10 alerting jobs to distribute alert processing more efficiently. Although the jobs run to the same schedule, the start times are offset slightly.
- The time taken to run an alert depends on the hardware setup, system load, and the complexity of the data and any queries in the alert. To ensure that alerting is working as expected, account for the performance implications of running your alerts before setting your alerting schedule.
- For performance reasons, do not schedule these jobs to run at the same time as the routine backup or large imports (or the alerting jobs should be temporarily disabled while these tasks are running).
- There is one job for processing email alerts. No setup is required on the SQL Server machine beyond setting up the SQL Server instance for email.
- The number of actions detected by the alerting job, and whether all actions are detected, will depend on the frequency with which the job runs. In particular, editing a record after the job completes and then undoing that edit before the next job runs will not be detected. However, the two edits will always be recorded in the audit log (depending on how you have chosen to configure auditing).

Procedure

1. In iBase Designer, select **Tools** > **Feature Availiabilty** > **Alerting Configuration** and click **Schedule**.
2. Select the schedule type from the list (all the jobs will use the same schedule type):
   - **Start automatically when SQL Server Agent starts** - alerting jobs will run to the same schedule as the SQL Server Agent.
   - **Start whenever the CPUs become idle** - see your SQL Server documentation for information on CPU Idle Schedules.
   - **Recurring** - alerting jobs will run on a specified schedule.
   - **Once** - alerting jobs will run once on the date and time you specify.

   Note: Turn off **Enable** if you do not want the schedule to take effect immediately. In SQL Server, the job will be listed as not enabled and not scheduled.
3. Specify the frequency at which the job runs. This can be changed at any time without needing to pause alerting.

   Note: Do not schedule alerting jobs to run at the same time as any routine backups or large imports.
4. Click **OK**.

Results

The job will run at the next scheduled date and time if the job is enabled, and the SQL Server Agent is started.

© N. Harris Computer Corporation

Managing alerting jobs

You can suspend alerting if you need to suspend the jobs, change the scheduling or change the number of jobs. When you resume alerting, all changes since the last time the job ran will be detected. This may result in users receiving alerts with a larger than usual number of changes detected.

Procedure

1. In iBase Designer, select **Tools** > **Feature Availiabilty** > **Alerting Configuration**.
2. Select the action for your alerting jobs:
   - To view the status of your alerting jobs, select the **Status** tab. For each alerting job, the page displays the job number, the date and time that the job last ran and the time it took to complete in seconds.
   - To disable your alerting jobs, click **Schedule** and then turn off **Enable**. This will not remove the alerting jobs, it just prevents them from running until you enable them again.
   - To suspend alerting jobs, click **Suspend Alerting**. This will remove the alerting jobs so that you can change the number of jobs. Any alert definitions and alerts in the database are hidden. When you are ready to start alerting, click **Resume Alerting**. This creates new alerting jobs that you will need to schedule. Any alert definitions and alerts in the database become visible.
   - To remove alerting, click **Remove Alerting**.

     Note: When you remove alerting, you delete the alerting jobs as well as all alert definitions and alerts in the database. If Audit History was already turned on when alerting was initialized then it will be left on.

Administering iBase

There are a number of options that you can use to affect the way that iBase and iBase Designer operate on the local machine that are not database specific. These options can be selected without logging into a specific database.

Editing the Most Recently Used list (MRU)

The Most Recently Used list is the list of databases that have been recently accessed. You can edit the most recently used list of databases (MRU list) to change the order, or to remove a database you do not want to be displayed.

About this task

The MRU list is the list of databases at the end of the File menu; you can select a database from the list to reopen it. Each time a different database is opened, an entry for it is placed at the top of the list.

It is possible that some of the databases no longer exist, for example, if a user deletes or moves the database or connection file. In addition, some databases might be temporarily unavailable while a server is out of service. Such databases still appear as entries in the **MRU List Manager**, but not in the File menu. To remove such databases from the list, you can delete entries. In addition, you can change the order of entries. You might want the most used databases at the top of the list, for example.

Note: The database is opened using the security file that is in the same folder as the database file.

Procedure

1. Select **Tools** > **Database Adminstration** > **MRU List Manager**.

© N. Harris Computer Corporation

2. In the MRU list, click a database to select it.
3. Use the up or down arrows to change the position of a database.
4. **Optional:** If required, delete a database from the list by clicking **Delete**.

Setting user options

To create a workflow that matches the way that you would like to work, there are a number of settings that you can specify that only affect your user account. These settings are divided into three main categories: general, charting, and advanced.

About this task

Procedure

1. Select **Tools** > **Feature Availability** > **Options**.
2. Click the:
   - General tab to specify some basic settings for using iBase, for example, how you use categories.
   - Charting tab to set defaults that will be used when charting in Analyst's Notebook unless specified otherwise in a charting scheme or charting settings.
   - Advanced tab to set options that affect just you (the User Settings area of the dialog) as well as all users of this computer (the options in the Local Machine Settings area). For example, you may specify the location of the templates folders.
3. Click **OK** to save your changes.

General Settings

Basic settings for using iBase, for example, how you use categories. These settings do not affect any other user.

Default Category Name

Choose the default category that you want to use when you save a new folder object (such as a set). By default, you use the folder name General if you leave this blank.

Prompt for Category when Saving Folder Objects

If turned on, a prompt for a category and access type is displayed when you save a new folder object (such as a set).

If turned off, you automatically save folder objects in the default category with the default access type wherever possible. However, you are always prompted for a category if you belong to more than one folder object control group.

For more information, see Working with categories.

Default to 'Public' access

Determines whether access to a folder object is public, private to the user who flagged it as private, or restricted to members of a folder object control group. Private folder objects are only listed and viewed by the user who flagged it as private and the system administrator.

For more information, see Working with categories.

Maximum number of most recently opened databases to show in the file menu

© N. Harris Computer Corporation

The Most Recently Used list is the list of databases at the end of the options on the File menu.

Each time a new database is opened, an entry for it is placed in the list. Selecting the entry is a quick means of reopening the database.

This setting determines the maximum number of entries there can be in the list. When this number is reached, new entries at the top push the bottom entries off the list.

Number of rows to be displayed in a multi-line text box

This determines the size of the box when entering or editing data in multi-line text type fields, in terms of the number of lines it can display.

Open last used database on start-up

Turn this on to quickly reopen the database you opened last, whenever you start iBase (not iBase Designer).

Check for matching records whenever a discriminator field value changes (datasheets only)

This only applies to datasheets and displays a warning about potential duplicate records when you enter data in a discriminator field which results in a match with one or more existing records.

This allows you to check your data at an earlier stage than the Prompt to confirm creation of matching records option which only warns you when you attempt to save the record.

Prompt to confirm creation of matching records

You are always warned when you attempt to save a record that will create a potential duplicate. However, you can display an additional prompt that appears when you click Yes to create the record.

Remember user for Windows single sign-on

Users can use their Windows credentials to automatically log on to iBase if their Windows credentials match an iBase account for either their Windows user name or the name of a Windows group to which they belong. However, a user cannot log on automatically if they belong to two or more Windows groups and there is an iBase user account for each group. In this situation, the user is prompted to select the user to log on as, and there is a Remember my selection option in the Logon dialog so that users do not need to repeat the selection each time.

To reset this option, users turn off the Remember user for Windows single sign-on checkbox. The option is only available when logged on to a security file.

Display dialogs in tabs

Choose whether new dialogs appear as tabbed or independent windows. By default, dialogs appear as tabbed windows.

Use legacy icons

Change the appearance of the entity icons to display earlier icon set. By default, the latest icon set is used

Activate global shortcut keys

Choose whether to use and potentially customize global shortcut keys. By default, global shortcut keys are activated.

© N. Harris Computer Corporation

Charting Settings

Basic options for charting in Analyst's Notebook. These settings can be changed in Analyst's Notebook.

- **Chart entity/link attributes**

  When a record is added to a chart to become a chart item, determines whether chart attributes are added or not.

  This setting applies to particular entity or link types:

  - If chart attributes are defined for the entity or link type in the database design.
  - If it is not overridden in the charting scheme by the Chart Attributes option settings (for the entity or link type or 'Defaults').

  As an alternative to using attributes, you can use cards or data records.
- **Chart pictures to represent entities instead of their icons**

  When an entity is added to a chart and the entity has a Picture type field, determines whether this picture field value is used to represent the chart item.

  This setting applies if there are no applicable Chart Picture options settings in the charting scheme (for the entity type or 'Default') that have a non-'Blank' setting.

  If there is more than one picture type field, the top field when you open the entity is the one used. If you are in doubt, contact your system administrator.
- **Rearrange new items added to a chart (not the whole chart)**

  Determines how much a chart is rearranged to accommodate newly added items.
- **Show toolbar in Analyst's Notebook**

  Determines whether the iBase toolbar is displayed when charting iBase data.
- **Default Link Label**

  Determines the labels for chart links added from iBase. The selected option becomes the default selection for Charting Settings.
- **Multiple Link Style**

  Determines how iBase links are represented on the chart. The selected option is the default selection in the Charting Settings.

Setting the link style

You can specify the type of labeling on links and how multiple links between entities are displayed.

1. In the Charting Settings dialog, click the Link Style tab to display the Link Style page.
2. In the Link label area, decide how you want to label the links on the chart:
   - **From database**

     The label for the chart link is the chart label as specified in the default labeling scheme.
   - **Type name**

     The label for the chart link is the source iBase record link type.
   - **Occurrences**

     The label for the chart link is the number of iBase links that it represents.

© N. Harris Computer Corporation

This is only relevant when **Multiple Link Style** is set to **Single** or **Directed**, when a chart link might represent several iBase links.

- **Sum numeric**

  If the iBase label for a link has a numerical component, then the values of this component, for all the iBase links represented by the chart link, are summed. The resulting number is used as the chart link label. The direction of links is taken into account; values are added if the links are in the same direction, or subtracted if they are in opposite directions.

  For example, Financial Transaction links might have an iBase label containing the value of the transaction, and there may be two links of this type between two bank account entities, one of $2000 and one of $1000. On the chart both of these links might be represented by one link, which will have a label of either $3000 (if both links are in the same direction), or $1000 if the links are in opposite directions.

  This option becomes relevant when **Multiple Link Style** is set to **Single** or **Directed**, when a chart link might represent several iBase links.

3. In the **Multiple Links** area, decide how you want to handle multiple links between entities:

   - **Single**

     One link on the chart between two entities can represent many iBase links between the entities. It represents all the iBase links of a particular type.

     If the chart link has an arrowhead, it indicates that at least one of the represented iBase links is in that direction (it may have two arrowheads showing there is a bidirectional link, or two links; one in each direction).

   - **Directed**

     One link on the chart between two entities can represent many iBase links between the entities. It represents all the iBase links of a particular type that are in the chart link's direction.

     The possible directions are: no direction, an arrowhead at one end, and an arrowhead at both ends.

   - **Multiple**

     Each chart link only ever represents one iBase link.

Note: There are two limitations when charting multiple links that use the Single or Directed options. For example, on association charts, the grades, dates and times, and source references given to the compressed link can be taken from an arbitrary constituent link. Also, these options do not display attributes for link fields that are defined as chart attributes.

Advanced Settings

You can set advanced user options, that not only affect your user, but also other users that are using iBase locally. To change settings that affect all users of this computer, you require write access to the Settings.xml file.

User Templates Folder

Path name of the folder that contains templates for creating new databases.

Temporary Files Folder

Path name of the folder for temporary files.

© N. Harris Computer Corporation

These files are created when, for example, you use View to edit a document specified in a document type field.

Command Group File

Path name of the Access database that organizes the iBase command access control.

Icon List File

Path name of the file that lists all the available icons when, for example, you are editing an Icon list.

Note: For more information about editing an icon list, and add custom icons, see Customizing your icon set.

Workgroup Templates Folder

Path name of the folder that contains database templates that are available to all local users.

Graphics Folder

Path name of the folder that contains the icons that are used in, for example, the **Database Explorer** and menu items (but not the entity, or entity type, icons).

Number of records to be displayed before auto-pausing

The number of records to be displayed before an automatic pause. This setting is only used in situations where the number of records need to be restricted.

Specify '0' to disable auto-pausing.

Note: Because this setting can impact performance, contact your system administrator before you modify the default.

Database Password

The password required to open the iBase Microsoft Access database.

Note: A 20-character password is generated for you when the database is created. You should keep a record of this password as it will be required if, for example, that you need to repair the database using Microsoft Access utilities.

Custom terms

If your organization has terms that you would like to use, you can create a file that replaces terms in iBase and iBase Designer. Your custom term file can be used locally to change terms on your machine, shared with other users to import onto their installations, or distributed to all machines that connect to a given database.

Creating a file of custom terms

You can create a custom term file that uses the list of terms that are available in iBase and iBase Designer and specifies custom terms to replace them. This file changes the terms that are available on your local machine.

Procedure

1. Select **Tools** > **Feature Availability** > **Configure Custom Terms** > **Configure**.
2. Select the terms that you would like to modify from the **Avaliable Terms**, and use the down arrow to move the terms into the **Custom Terms** list to be customized.
3. For each row, enter the revised term in the modified column.

© N. Harris Computer Corporation

4. Click **Save**.

Results

The terms are saved in files in the following location: `%ProgramData%\i2\i2 iBase \``language``\Trans`. For example: `C:\ProgramData\i2\i2 iBase\en-US\Trans`.

Important: Any changes that you make to your local custom terms file will be overridden if a custom terms file is distributed via a database that you connect to. You can create a backup of the files in this location if you would like to keep your versions.

The new terms will be loaded and used in iBase, Analyst's Notebook, and iBase Designer.

If you would like to revert to the original terminology at any time, reopen the Custom Term Editor and use **Reset**.

What to do next

If you would like to share your custom terms with other users, you can either provide a copy of your file, or associate your terms with one or more databases. For more information, see:

- Importing a file of custom terms
- Distributing custom terms to machines that access a database

Importing a file of custom terms

If you have been provided with a file of custom terms, you can import these terms into your local environment. The terms that are available in the file will replace the original terms, and any local custom terms.

Procedure

1. Select **Tools** > **Feature Availability** > **Configure Custom Terms** > **Configure**.
2. Select **Import**, and confirm that you are happy to overwrite the current configuration.
3. Locate the file, and click **Open**. The custom terms are listed in the Custom Terms list.
4. Review, modify and save any changes you would like to make.

Distributing custom terms to machines that access a database

If you would like to ensure that a file of custom terms is used on all the machines that access a given database, you can store the custom terms file in the database. When a machine connects to the database that has not already been provided with the custom terms file, the file will be downloaded and imported.

Procedure

1. Open iBase Designer, and logon to the database.
2. Select **Tools** > **Feature Availability** > **Configure Custom Terms** > **Save Custom Terms to Database**.
3. Select the file, and click **Open**.
4. Restart any open iBase applications to see the changes.

© N. Harris Computer Corporation

Modifying the user-defined dictionary

Spell checking for fields on records, can be customized to include custom terms that are specific to your organization. Adding terms to the user defined dictionary prevents these terms from being flagged when the spell check is used on a record.

Procedure

1. Open `iBase.lex` in a text editor. This file is stored in the roaming application data for the logged on user. For example: `C:\Users\User1\AppData\Roaming\i2\i2 Shared\Spelling 8\en-US`.
2. Add or remove the terms that you would like to change, ensuring that each term is entered on a separate line.
3. Save your changes and restart any open applications.

iBase tools

iBase tools are provided to help you to carry out advanced database administration. Most tools are installed separately to the standard iBase installation by selecting the Extended Features option in the InstallShield Wizard (Search Index configuration tool is installed by selecting the Server option).

The tools, if installed, are accessible from the Windows Start menu in **i2 iBase** > **Tools**.

Database Configuration (SQL Server databases only)

You can use the Database Configuration utility to manage connections to SQL Server databases, specifically to set the server name, server login name and the use of Windows security.

iBase Index Service Configuration

You can use the Index Search Configuration tool to prepare the database for Search 360 and to configure the index service. See Setting up Search 360 for details.

Repair Compact

You can use the Repair Compact utility to repair a damaged iBase Microsoft Access database. This utility can also be run from the **Tools** > **Database Administration** menu in iBase Designer.

Scheduler Configuration

The Scheduler Configuration dialog allows administrators to set up Scheduler for running batch imports and exports.

Audit Viewer

You can use Audit Viewer to view and manage audit logs for database and security files.

Administering a database

You can change options on how iBase functions. These settings can affect the user account that you are currently using, the machine on which iBase is currently installed, or the entire database.

Managing database templates

The Template Manager displays any existing database templates. You can view individual templates, delete an existing one, or create a new one.

The templates must be files in the Templates folder, as named in the Advanced page of the Options dialog.

© N. Harris Computer Corporation

For information on workgroup and user templates, see the Administration Center.

Creating database templates

To create a new template from a database, select **File** > **New Database Template**. Name the template and select a database on which to base the template. For detailed information, see Creating a database template.

Viewing a database template

Click **View** to display the Database Template dialog where you can inspect the entity types, link types, and fields defined in the template.

Deleting an existing database template

Click **Delete** to remove the selected template.

Applying a template to a database to update the schema

Use the Update Database Schema dialog to apply a template generated from one database to update the schema of another. This is useful when you have a group of databases which you need to keep consistent in their design and content.

About this task

The template must be compatible with the current database. This typically requires that the target database will have been created from a template generated by the source database, or by the same template as the source database.

Note: To update a database schema you will need permissions to:

- create a template file in the iBaseTemplates folder -- as specified in the Advanced page of the Options dialog
- create a database in the same folder as the main database .idb file

The schema update process

To create a group of databases with a fully consistent design, you should do the following:

1. In iBase, define a group of folder objects in the Common Folder Objects dialog that you want as a core set of folder objects across a number of databases. You can, if required, view or edit an object (such as a query or browse definition) to test that it is the one required or update it, before making it common.

   Sets cannot be defined as common folder objects as they refer to specific records in the database, which will not exist in other databases.
2. In iBase Designer, make any other changes to the schema that you want to apply to the databases, such as adding or modifying entity and link types, or editing fields.
3. Generate the template.
4. Use Schema Update to apply the template to each database in turn. The common folder objects will be synchronized between the source database and target database, that is, the result of the update will be that the same set of folder objects exists in both databases.

What is updated?

The following elements are updated in the schema:

- Entity types, link types, fields and standard fields

© N. Harris Computer Corporation

- Datasheets
- Pick lists, icon lists and SCC lists
- Common folder objects, such as report definitions, queries, charting schemes and so on (but not sets or labeling schemes). Common folder objects are defined in iBase User. For more help, see the iBase help topic Defining Common Folder Objects.

Updating a schema can result in data being removed from the target database. Removing entity or link types will result in the loss of any data stored using these types. Data will also be lost when you remove fields. Ensure that you check the Deletions page of the Update Report carefully before proceeding with the update.

Procedure

To select a template:

1. From the **Tools** > **Database Design** > **Update Database Schema**.
2. Click **Browse** and select the template file. If the selected template is incompatible for any reason with the database, then an error message is displayed.
3. If the template is compatible, then the two schemas are compared and the differences between them are displayed in the Update Report area of the dialog.

   Before updating the schema, you should review any differences between the schema and the template displayed in the Update Report area. The changes are organized into two pages: the Additions and Modifications and Deletions pages.

   Note: Once a template is selected, it can be viewed by clicking **Open**. The database entity and link types and fields defined for each type are displayed. You cannot edit the types or fields in this dialog.
4. On the Additions and Modifications page and the Deletions page, review the changes that are listed. Any changes, additions or deletions to the following are listed:

   - Entity types and their fields
   - Link types and their fields
   - Standard fields
   - Datasheets
   - Pick lists, icon lists and SCC lists
   - Common folder objects (listed separately for each type of folder object)
   - Semantic Type Library (but specific changes are not listed)

   Note: If required, click **Export** to save a list of the schema changes in a text file that you can print later.
5. When you are ready to apply the changes, click **Update**. When this is finished, you are warned if any folder objects were renamed because they have the same name as a common folder object in the template. Objects which are renamed will have an underscore added as a prefix. For more help on common folder objects, refer to the iBase help topic Defining Common Folder Objects.
6. A confirmation message is displayed. Click **Yes** to proceed.

Results

When the update is completed, a message is displayed and the Update Database Schema dialog will close automatically.

© N. Harris Computer Corporation

Setting up composite indexes

iBase uses relational database indexes to search for records according to the values in specific database columns. In some circumstances, you can increase the speed of record retrieval by creating a composite index that combines the values of fields that contain related information.

About this task

When iBase is using a SQL Server database, indexes can improve the retrieval times of records by allowing data to be retrieved without the need to open the record table. For operations involving large amounts of data, you can sometimes improve performance further by creating composite indexes.

For example, importing a large number of records can be slow when data contains multiple discriminators, because iBase has to interrogate the index for each discriminator to determine whether an incoming record matches an existing one. You can speed up the process by creating a composite index that covers all the discriminators you're using.

To create a successful composite index, you need to pay attention both to the types of fields that you add, and to the ordering of those fields.

In iBase, *key fields* form the main part of an index and are important for index structure. They determine the order that the data is stored, and are used to filter and sort data. *Included fields* are used to cover queries that are not covered by the key fields. They are not used to sort or filter data, but are used to retrieve data more quickly.

You should create index entries for the most common terms to be associated, in the order that they are most likely to be run. For example, if you would like to add composite indexes to improve the performance of running queries align the fields in your index with the discriminators that are used in the import specifications.

Note: You can create a maximum of 31 indexes per field type.

Procedure

1. Select **Tools** > **Database Design** > **Composite Indexes**.
2. Choose the **Record Type** that you would like to create a composite index for.
3. Select **Add**.
4. For each index combination that you would like to add for this record type:
   a. Select **Add**.
   b. On the **Key Fields** tab, choose the **Key Fields** to use and click **OK**. To remove Key Fields, select one or more fields and click **Remove**.
   c. On the **Included Fields** tab, choose the **Included Fields** to use and click **OK**. To remove Included Fields, select one or more fields and click **Remove**.
   d. Click **OK**.

   The index is created but can be deleted.
5. Continue to create indexes for other record types by repeating steps 2 thru 4.
6. Click **Close**.

© N. Harris Computer Corporation

Setting up suggested indexes

Suggestions for composite indexes are made by scanning import specifications for discriminators. Given the performance impact, you should only create indexes for types of items that you intend to import regularly.

About this task

Suggestions scan all import specifications for index combinations that haven't already been implemented. As import specifications can import links, these suggestions may be made for multiple item types, and subsequently multiple indexes.

Procedure

1. Select **Tools** > **Database Design** > **Composite Indexes**.
2. Click **Suggestions** to list possible composite indexes that are suggested based on the discriminators selected in available import specifications.
3. Click the **Add Composite Index** checkbox to select any import specifications that you want to use the suggested indexed fields.
4. Click **Add Indexes**.

Additional field features

In SQL Server databases, you can enable three different features on fields that help provide information about the values they hold. As this additional information will have an impact on performance and database size, these features should only be enabled on databases where they are required.

Enabling field security

Field level security works in addition to standard SCC control to secure records based on the information that they store. You can enable or disable field security at a database level.

Before you begin

In order to use field security, you must have data access control enabled that uses SCC control. For more information, see Setting up Data Access Control groups.

About this task

When field security is enabled, each record is given an additional option that specifies the security classification for the record value. Field security can be used to restrict access to records based on the values of the fields that they contain.

Procedure

To turn on field level security:

Select **Tools** > **Feature Availability** > **Enable Field Security**.

© N. Harris Computer Corporation

Enabling field confidence

You can indicate the level of confidence you have in a specific field value. The confidence level can be viewed by other members of the team and can be used to decide how that information is treated.

Procedure

To turn on field confidence:

Select **Tools** > **Feature Availability** > **Enable Field Confidence**.

Enabling field attachments

Field attachments are documents and images that are linked to specific fields. You can use field attachments to highlight information on charts, or to provide additional information about a specific field value.

About this task

Field attachments allow you to associate additional information with specific fields. For example, a scan of identifying documents that verify a given name.

Note:

- Field Attachments are enabled at a database level, you cannot restrict the types of records or fields that allow attachments
- Documents and images that are attached to fields are not indexed.

If you would like the document or image information to be searchable, add a field of the correct type to store your document or image. For more information on creating fields, see Creating a field.

Procedure

To turn on field attachments:

Select **Tools** > **Feature Availability** > **Enable Field Attachments**.

The database is now ready to accept field attachments.

Managing Plug-ins

A plug-in is a software component that extends the basic functionality of iBase.

You need to activate plug-ins before they can be used. You might want to deactivate any plug-ins not in use to simplify the user interface, and use less memory. Plug-ins that can be installed on your system include:

Plug-in	Description
Alerting Configuration	Used in iBase Designer to set up alerting.
Alerting Inbox	Used in iBase Designer to set up alerting.
Audit History Viewer	In SQL Server databases, you can store the history of the changes that are made to records in the database.

© N. Harris Computer Corporation

Plug-in	Description
Bulk Coordinate Validator	Ensures that coordinate formats are valid when converting coordinates in bulk.
Chart Importer	Allows the contents of ANB charts to be imported into an iBase database.
Composite Indexes	Allows the creation and management of SQL Server composite indexes from within iBase Designer.
Custom Terminology Builder	Provides the ability to customize the text displayed within iBase applications.
Database Subsetting	Manages the creation and synchronization of database subsets.
Duplicate Checker	Used to identify records that contain similar information.
Excel Interface	Used to export data to Microsoft Excel.
Field Attachments	Enables attaching of files to individual fields.
Field Security	Enables security classification codes to be applied to individual fields.
Full-Text Search	A method of searching SQL Server databases. In later versions, the method of searching SQL Server databases is Search 360.
Import and Export	Provides data importing and exporting.
Merge Links	Allows links of the same type to be merged into a single link.
Schema Update	Used in iBase Designer to manage changes to the schema.
Search 360	The search mechanism for Microsoft SQL Server databases.
Valid End Types Utility	Used in iBase Designer to set the valid record types to add to the ends of a link type.
Word Search	The search mechanism for Microsoft Access databases.

© N. Harris Computer Corporation

Plug-in	Description
XML Import and Export	Used in both iBase and iBase Designer to import and export data as XML.

After making changes, you need to restart iBase or iBase Designer for the changes to take effect.

To manage plug-ins:

1. In the Plug-in Manager dialog, turn on a checkbox to activate a plug-in or turn off a checkbox to deactivate a plug-in.
2. Click **OK** to confirm your changes and close the dialog.
3. Restart iBase or iBase Designer for the changes to take effect.

Setting the plug-ins to be loaded

Both the standard plug-ins and any custom plug-ins that have been applied to your installation can be enabled or disabled according to your needs. Plug-ins are enabled or disabled on a per machine basis.

Procedure

To enable or disable plug-ins:

1. Select **Tools** > **Feature Availability** > **Plug-in Manager**.
2. Choose the plug-ins to be loaded.
3. Click **OK**.
4. **Optional:** Deny the related commands to some or all users by using Setting up System Commands Access Control groups groups.
5. Restart iBase Designer for the changes to take effect.

Customizing your icon set

An icon set describes a group of icons that can be used within an iBase database to differentiate entities. Each icon set consists of a group of icons that are described within an icon list.

About this task

iBase is currently shipped with four different icon lists:

Icon list

The default list of icons that are shipped with iBase.

Military Icon list

A list of icons that have been designed for use within a military environment.

Combined Icon list

A list that combines the contents of both the Icon list and the Military Icon list.

ANB Standard Icon list

A list of icons that matches the icons available using Analyst's Notebook.

© N. Harris Computer Corporation

Procedure

To select an icon list:

1. Select **Tools** > **Feature Availability** > **Options**
2. Select the **Advanced** tab.
3. Find the **Icon List file** option within the **Local Machine Settings** section, and browse to the desired icon list file.

Setting custom icons

You can create a custom icon list if the standard icon lists that are included with iBase do not match your needs. Custom icon lists can include icons that you have created yourself but can also reference icons from the standard icon set.

Procedure

1. Create a new icon with any graphics application that allows you to save the file in the correct format. Alternatively, you can copy and edit an existing icon.
2. Make sure that the image is the correct size (32x32 for a screen icon), has the correct color depth, background color, and so on.
3. Save the file in the correct icon folder: `<My Documents>``\i2\i2 Shared\Custom Images\`

   Where:

   `<My Documents>` is the location of your 'My Documents' folder

   Note: If you are creating the image for use as the default icon for a specific entity type, give the image the same name as the entity type. This simplifies future maintenance.
4. Create the image for the corresponding printer icon.
5. To add the icons to a new icon list:

   a. Create a new text file using the text editor of your choice.

   b. Add the details for the icon into the icon list that you want to use, in the following format:

   `<icon name> <icon file name>`

   Where `<icon name>` is the name you want to display within iBase, and `<icon file name>` is the exact name of the icon image without the file extension. These should be separated using a tab character. For example:

   `Humvee HMMWV`
6. Save your file as a text file (*.txt).
7. Select **Tools > Feature Availability > Options**
8. Select the **Advanced** tab.
9. Find the **Icon List file** option within the **Local Machine Settings** section, and browse to the desired icon list file.

What to do next

Ensure that all i2 applications are closed and restarted. The new icons are recognized when the applications restart.

© N. Harris Computer Corporation

Working with cases

You can partition data in your database into different cases. Each case contains records belonging to a particular investigation.

You can then assign access to groups of users to one or more cases. To create and manage cases, you need both Database Administrator and Security Administrator permissions.

Note: Before you can create a case, you need to activate the database for case control. Case control is only supported on SQL Server. You cannot use case control if i2 iBase Database Replication is installed on your machine. You cannot use cases with Scheduler. For more information see Activating case control.

You can assign users to several cases, but to add or modify data in a case, the user must select only that case when opening the database. Users can view records across all the cases to which they have access, but they will not be able to modify the data.

Each case has the following properties:

Property	Explanation
Name	The name given to a case when it is created. Case names must be unique across the entire database.
Date Created	Automatically captured when the case is first created.
Date Closed	Automatically captured when the status is set to closed.
Description	Used to provide more information about the case. Can be updated when required.

Note: Word search is unavailable if you have a case-controlled database.

Opening and closing cases

Data can only be added to an open case. Closed cases can be selected by users when opening the database, but only in read-only mode. Closed cases are included in multi-case analysis mode.

You can close and re-open a case multiple times. Each time you close a case, the Date Closed column in the Select Case dialog is automatically updated.

To close or re-open a case:

1. In the left pane of the Database window, select **Cases**.
2. Double-click the case whose status you want to change.
3. In the Case dialog, select **Open** or **Closed** in the General page.

Assigning users access to cases

Users assigned to a single case will be connected to that case automatically when they log in, without being prompted to choose a case in the Select Case dialog. When working in a single case, users can create new records as well as viewing existing data.

© N. Harris Computer Corporation

Users authorized to access several cases can open a single case or open all cases at once in multi-case analysis mode. When opening all cases in multi-case analysis mode, new records cannot be created.

To assign users to a case:

1. In the left pane of the Database window, select **Cases**.
2. Double-click the case to which you want to add or remove users.
3. Select the **Users** tab to add or remove individual users. To assign a user to this case, double-click their name, or click once to select them and click **Add**.

   Added users appear in the list on the right. To remove a user, double-click their name in the Users that can access this case box or click once and then click **Remove**.
4. Select the **Groups** tab to add or remove Data Access Control groups of users. To assign a group to this case, double-click the group name, or click once to select it and click **Add**. To remove a group double-click their name in the **Groups that can access this case** box or click once and then click **Remove**.

Note: Users who are not authorized to access any cases will be unable to open the database.

For help on setting up users and groups in the database, see Creating users and Creating security groups.

Removing users from cases

When you remove a user from a case, you deny the user access to any of the records in the case.

If alerting is in use, then:

- the user is removed from any alert definitions that they own (and only a system administrator can change the alert definition)
- the user's alert definitions remain active for other users
- no alerts are removed from the user's alerting Inbox (but the alert details can no longer be viewed)

Adding data to a case

If you have data that you want to add to a particular case, the quickest way is to import that data into the case. See Importing and exporting data.

If you have a large amount of data, you can use Bulk Import to import it into the required case more quickly than by using a standard import. See Bulk importing.

You can also add records manually, one at a time, to the case in iBase User. Select the required case

Records in a case

When a single case is selected by a user, any queries that are run will return results based only on the records in the current case. Similarly, sets and reports will only include records in the current case.

Whenever a user selects "All records" when logged into a single case, this refers to all the records in that case only.

When several cases are selected in multi-case analysis mode, then "All records" applies to the records in all of the cases to which you have access.

© N. Harris Computer Corporation

Note: In contrast, the alerting Inbox always shows all the user's alerts regardless of the current case. However, the user can only view the details for an alert when they are logged into the case that contains the alert definition.

Multi-case analysis mode

Multi-case analysis mode is useful for querying, browsing or reporting on data across several cases. In multi-case analysis mode, users can view records in all the cases (open and closed) to which they have been given access, but they cannot add, modify or delete any records in the database.

Deleting a case

To delete a case, right-click on the case name in the left or right pane in the Explorer view and select **Delete**.

Important: Deleting a case purges (hard-deletes) all records in the case, the audit history for those records, all alert definitions and any alerts remaining in the alerting inboxes of the subscribers.

Activating case control

Case control is used to partition the records in your database into a number of cases, so that groups of users can be given access only to certain cases. This is useful when you want to authorize users to work only on records relating to particular investigations.

About this task

Note:

- Case control options are only available if database replication is not installed.
- Case control can only be applied to SQL Server databases.
- You cannot use both Standard (SCC) Control and Case Control in a database. You need to decide which of these security methods is the most suitable for your requirements.
- You cannot initialize a database for replication when you have activated case control. You cannot activate case control in a replicated database.
- You cannot use cases with Scheduler.

Activating case control in a new database

When you create a new database, you can set up case control in that database before you add data to it.

Procedure

1. Select **File** > **Database Properties**.
2. On the **Advanced** page, select the **Case Control** option.

   Note: Case control options are only available if database replication is not installed.

   If you select a template other than <Blank>, then the case control option is read from the template. It is not possible to change the case control or Standard (SCC) control option in this situation.

   Before you finish designing the database, you need to create at least one case. If you do not create at least one case, then the database cannot be opened in iBase User except in read only mode (that is, no data can be added to it). Nor can you import data into the database in iBase Designer. For more information, see Creating a case.

© N. Harris Computer Corporation

Activating case control in an existing database

You can create cases in any non-replicated database. In addition to enabling the case control option, you also need to update existing features such as alert definitions to take account of new cases.

Procedure

To activate case control on an existing database:

1. Select **File** > **Database Properties**.
2. In the Database Properties dialog, click the **Advanced** tab.
3. Select the **Case Control** option.

   Note: Case control options are only available if database replication is not installed.
4. Click **OK**. You will be prompted if you want to go ahead with the conversion to a case controlled database.
5. Click **Yes** to proceed. You will now be prompted for the name of a default case. A case-controlled database requires at least one case to be created.
6. Enter the name for the case and click **OK**.
7. If alerting is in use, and before closing the database, assign all users who have created alert definitions to the case you have just created.

   Note: Alert definitions without owners will be deleted when you close the database

Additional information about converting an SCC database to a case-controlled database

A database that is set up to use SCC control can be converted to be case controlled. A case-controlled database cannot be set to use SCC control.

When you convert a database with SCC control to case control, be aware:

- All the records in the database are added initially to a single case. You cannot convert security classification codes into different cases based on the SCC value. To move some records into another case, you need to export those records and then re-import them into a new case.
- All previous security that was set up for particular entity types or link types, based on the SCC code, is removed. Users can either see all the records in a case (if they are given access to that case), or none of the records (if they are not given access).
- Any SCC field is converted automatically to a case field. See Field Types.
- You cannot revert to the previous SCC security settings in the database after you have specified case control.

Note: You may want to backup your database before performing this action.

Creating a case

Cases allow you to store the information used in an investigation together. The access to a case can be controlled by assigning users and groups.

About this task

Note: Before you can create a case, you need to activate the database for case control. See Activating case control

Use the Case dialog to create a new case or edit the properties of an existing case.

© N. Harris Computer Corporation

The Case dialog is organized into three pages:

- General - enter a description for the case and set its status as open or closed
- Users - assign users to the case
- Groups - assign Data Access Control groups to the case

Procedure

1. Select **New** > **Case**.
2. Set up the general properties for the case:
   a. Enter or edit a description for the case. This description is visible to the user when they select a case in the Select Case dialog, when logging on to the database.
   b. Specify whether the case is **Open** or **Closed**. Users will be able to add data to an open case when they select only that case when logging on. Data in a closed case cannot be added, modified or deleted by users.
3. Assign users to the case:
   a. Click **Users**. A list of all the users of this database is displayed on the left.
   b. To assign a database user to this case, double-click on their name, or click once to select them and click **Add**. Added users appear in the list on the right. To remove a user, double-click on their name in the **Users that can access this case** box or click once and then click **Remove**.
4. Assign Data Access Control groups to the case:
   a. Click **Groups**. A list of all the Data Access Control groups related to this database is displayed on the left.
   b. To assign a group to this case, double-click on the group name, or click once to select the group and click **Add**. Added groups appear in the list on the right. To remove a group, double-click on the group name in the Groups that can access this case box or click once and then click **Remove**.

Results

The case is created and ready for use by the assigned users. When the investigation is complete, you can delete a case to remove it from the database.

Before deleting a case, you need to:

- Archive all the data in the case.
- Archive the audit log for the period covered by the case.

To delete a case, right-click on the case name in the left or right pane and select **Delete**.

Note: Deleting a case purges (hard-deletes) all records in the case, and deletes all entries in the audit log for those records.

Importing and exporting data in cases

In a case-controlled database, you can only import data into a single, open case. Duplicate checking, for example, is only carried out against records in the current case.

About this task

Note: You cannot use iBase Scheduler to import data into a case-controlled database.

© N. Harris Computer Corporation

Procedure

To import data into a case or series of cases:

1. In the import wizard, click **Run** to start the import in the usual way. You are then prompted to select a case.
2. In the Import Progress dialog, when the import completes for the first case, click **Back** to redisplay the last page of the import wizard.
3. If required, change the name of import statistics summary file and the error file, or select the **Append a timestamp** option (so that you do not overwrite the files generated by the first import).
4. Click **Run** to start the second import and, when prompted, select the next case.

In a case-controlled database, you can only export data from the selected case or from all the cases in the database:

1. In the export wizard, click **Run** to start the export in the usual way.
2. You are then prompted to select a case - you can select any open case in the database or, to export data from all the cases in the database, turn on **All Cases**.

Importing and exporting data

This documentation provides information to help you import and export data.

Importing Data

You might already have data that you want to analyze in electronic form, for example in another iBase database, a spreadsheet, some other database, or as a text file. You can use an import specification to define how the source data is interpreted during the import.

Planning imports

Before you create an import specification, it is always worthwhile to compare the data with the database to see what type of entity will be imported, which fields are mandatory, and which fields will be identifiers (used to decide if records match those already in the database). You need an import specification for each entity and link type in the source data. You might also find it useful to print out a database design report and a sample of the data that you want to import.

Note: If you are importing entities or links that use multi-line text (append only) fields, you may want to test the import first using a copy of the database. This is because you can only add text to the end of fields of this type; you cannot delete or edit existing text.

Validating and protecting the data

During the import you can check that values imported into Selected from Code List type fields are valid; any invalid values are reported as errors during the import, and you can then correct them before re-importing the data.

You can protect existing data by turning on **Do not update existing field values with blank values** in order to prevent existing data from being overwritten by blank or empty values in the source data.

Comprehensive record matching also enables you to control how records are created or updated. For details, see the iBase help.

© N. Harris Computer Corporation

Manipulating the data

When importing data, you can transform field values in source records prior to assigning them to *iBase* fields in order to change the record structure and discard unwanted data. You can:

- Copy a source field in order to assign it to more than one iBase field.
- Merge two or more source fields in order to assign them to a single iBase field.
- Split a source field in order to assign parts of the field value to several iBase fields.
- Update specified values with new ones (for example where the source data uses a different code list) by creating a substitution file.
- Trim unwanted space from the start or end of a source field.

Recording the results of the import

You can record the results of the import by saving new and modified records in a set. If required, you can also log the numbers of new and modified records to a file.

Note: If you use auditing with an Access database and the audit level is set to 4, the audit log records only the start and end of the import. It does not log the individual records.

Handling errors

You can save the errors that can occur during the import to a file. You can then fix the problems with the source data by editing the error file, and re-import the remainder of the data using the error file as the source for the import. For further details, see the iBase help.

Importing into a database with case control

If your database is case enabled, you have to specify the case into which you want to import the data.

You can only import data into one case.

When you run the import, the Select Case dialog is displayed. Select a single case to which all the imported records will be added.

Matching during the import will only be carried out against records in the specified case.

Bulk imports and XML imports

A bulk import allows significantly faster importing of large quantities of data, including XML data, without user intervention. You set up a bulk import in the same way as any other import, using an import specification, although there are a few minor differences between a standard and a bulk import.

For further details, see:

- Bulk importing
- Running a bulk import
- Using XML import and export

Note: After importing large numbers of records, you might want to compact your database.

© N. Harris Computer Corporation

Importing data using an existing specification

You might already have some data in electronic form that you want to analyze. Rather than entering the records one at a time, you can import them into your database.

Typically, you use iBase rather than iBase Designer for this. However, you can only run bulk imports and XML imports in iBase Designer (or iBase Scheduler).

To display the Import Wizard, select **File > Data > Import**.

Setting field values for the session

You can set a session default value for any standard field; this adds this same value to a specified field for each record imported during a particular session. For example, the source document reference or weed date could be added automatically as each record is imported.

Procedure

To set defaults for the session:

1. On the last page of the Import Wizard, click **Session Defaults**.
2. In the Session Defaults dialog, click in the **Value** column for the row of the field you want to set.
3. Enter or edit the default value, or select a value from the list.

   Note: If you leave iBase Designer and restart, session default values are reset to blank.

Transforming source data

When importing data, you can transform values in source records prior to assigning them to iBase fields.

There are two main types of transformation that you can make:

- You can create a new value by combining data from other fields in the source record.

  For example, you could combine a first name with a last name to create a full name. You do this by creating a new field, choosing which fields to combine and specifying how they should be combined. When you do this, the data is copied from the source data, so you can continue to use the original data for other fields that you want to import.
- You can apply actions to the data in individual fields to change its format or value.

  For example, you could prefix the text in a specific field with fixed text, suffix it with text from another column, replace it with a value retrieved from a substitution file such as Female for F, or remove ordinal suffixes such as st, nd, rd, and th from (English-language) dates. These transformations are referred to as Actions, and any number of them can be applied to individual fields.

The actions you specify are saved as part of the import specification.

For more information, see iBase user help.

Upgrading iBase 5 import specifications

If you are modifying an iBase 5 import specification that does not have any transforms set up, then you will use the steps above.

However, if the existing import specification has one or more transforms then you can:

- Continue to use the import specification as before. You can modify the existing transforms or add new ones by using the old Transforms dialog.

© N. Harris Computer Corporation

- Upgrade the import specification by removing the existing transforms and reassigning the source fields. When you next click the Field Actions button the new Field Actions dialog will be displayed. Re-enter the transforms as described above.

Bulk importing

Bulk imports enable you to import data more quickly, and should be considered if you have large volumes of data to import or if you find the standard importer too slow.

You can only run bulk import on an SQL Server database. Bulk imports from XML files additionally require that the database supports Unicode. In addition, you can only run a bulk import from iBase Designer or the Scheduler utility. Use the Scheduler to run bulk imports at times when the database is not being used.

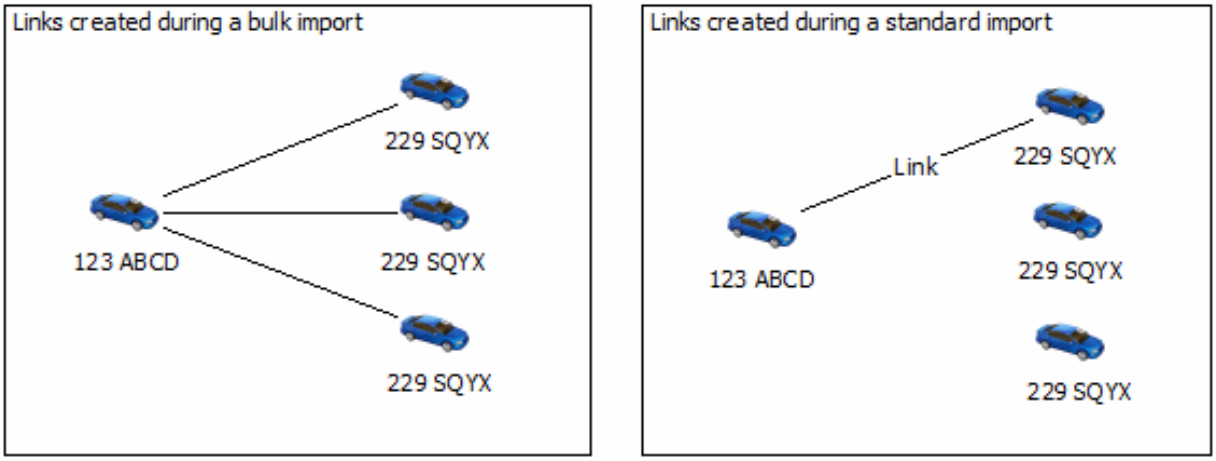
What is a bulk import?

A bulk import allows significantly faster importing, and is useful for importing large quantities of data without user intervention. You set up a bulk import in the same way as any other import, using an import specification, although there are a few minor differences between a standard and a bulk import (see the next section).

To define a bulk import specification:

- You need to be logged on as a database administrator.

Note that bulk importing has the potential to create more links than a standard import. In bulk importing, all specified links between matching link ends are created, in contrast, for standard imports only the first link between specified ends is created, see the example below:



A bulk import specification is the same as any other import specification, with the following limitations:

- You cannot import picture and document fields.
- There is no user action during the import to confirm matching records.

Differences between bulk imports and standard imports

Bulk imports have the following features:

© N. Harris Computer Corporation

- Bulk imports are not sensitive to trailing spaces.
- The order of importing elements can be different. When importing link with ends of the same type, bulk import will import all end 1 records before all end 2 records. If records are updated by both end 1 and end 2 data, end 2 updates will take precedence.
- Bulk imports are case sensitive when comparing the contents of Append Only fields.
- String comparisons take account of the locale.
- If no records are imported, an empty import set will be created, to identify the fact that the import took place.

Bulk import is incompatible with Audit Levels 4 and 5. At audit level 4 or 5 changes to individual records are audited, but when running a Bulk Import the creation or update of individual records is not audited.

Defining a bulk import specification

Bulk import specifications are defined, edited and saved in the same way as any other import specification. You can create a new specification from scratch, typically in iBase rather than iBase Designer, or load an existing one. For more information, see creating import specifications.

To mark the import specification as a bulk import, turn on the Bulk Import checkbox on Page 3 of the Import Wizard.

Note: The Bulk Import checkbox is unavailable if the database has not been activated to allow bulk import, or you are using an Access database.

Importing into a database with case control

If your database is case enabled, you have to specify the case into which you want to import the data when running the import.

When you run the bulk import, the Select Case dialog is displayed. Select a single case to which all the imported records will be added.

Note: You cannot run a bulk import into a case-enabled database using the Scheduler utility.

Running a bulk import

Bulk import is a useful way to manage large volumes of data, and can only be run on a SQL Server database.

To mark an import specification as a bulk import, turn on the Bulk Import checkbox on Step 3 of the import wizard. The rest of the import specification is the same as for a standard import, with some minor exceptions that are described below. For more information, see creating import specifications.

Import specifications that are not suitable for bulk import

If the current import specification is not suitable for bulk importing, a message is displayed when you select the Bulk Import checkbox to warn you that you need to make some changes in your specification first. This might be for the following reasons:

- Picture or document fields are assigned in the Assign Fields page of the import wizard. You must reverse the assignment of these fields before you can use the specification for a bulk import.
- The Confirm action option is selected in the Check for Existing Records page of the import wizard. You must turn off this option before you can use the specification for a bulk import.

© N. Harris Computer Corporation

Using all available identifiers

When you are setting up a bulk import, you should try to ensure that the specification imports as much identifying detail as possible for each record. If the data you import is too vague, it might result in large numbers of matches with records that are already in your database.

In a bulk import, you cannot choose what action to perform for each of these possible matches. This could result in a lot of records being incorrectly updated. In the case of link imports, it might result in a lot of records being incorrectly added.

For this reason, use as many as possible of the available identifier fields when setting up the specification. This minimizes the possibility of records being matched that are not the same, as in the following examples:

- Using only the Last Name field to match existing Person entities in the databases ("Smith") would result in many matches and a lot of records being updated.
- Using the Last Name field and the First Name field (for example, "Smith, John") would result in fewer matches, but still a number of records for several different 'John Smiths' might all be updated.
- Using the Last Name field, the First Name field, and the Date of Birth field would result in matches where there already happened to be a John Smith with the same date of birth in the database. This is less likely, but still possible in a large database.
- Using the Last Name, First Name, Date of Birth, Place of Birth, and Social Security Number field for each record would almost certainly guarantee that any record that matched one already in the database relates to the same real-world person, and therefore should be updated.

If you have not specified all of the available identifier fields in a specification for bulk import, then a message is displayed:

```
WARNING: This specification does not use all available discriminators as
identifiers for record matching. Usually all discriminators should be used
to ensure that the correct target data is updated.
```

An import that would result in a high proportion of records being updated might indicate that there is insufficient identifying detail for each record. This concern will be reported as part of the safety check.

Running the bulk import specification

On the last page of the import wizard, click **Run**. If you are importing into a case-enabled database, at this point you will need to specify the case into which the records will be imported.

The import will run and all errors will be reported as they are encountered. Unlike a normal import, the process is not interactive.

Note: A bulk import will stop if the number of errors reaches a threshold that is set to 10,000 errors by default. The following error types count against this threshold:

- Value not defined in list for field (for 'selected from code list' fields)
- Mandatory field value not supplied
- Text field value too long
- Invalid direction
- Invalid strength
- Invalid date
- Invalid time
- Invalid numbers

© N. Harris Computer Corporation

- Incorrect number of columns

Verifying a bulk import specification

On the last page of the import wizard, click **Verify**. No data will be imported, but any errors that would result from importing the data are listed. This allows you to check the import source data and correct any errors in the source file or the import specification before importing the data.

Performing a safety check

As a safety check, the source file is analyzed to determine the number of records currently in the database that would be updated if the data is imported. Errors are raised based on the ratio of the number of records to be updated to the number of records to be created.

Safety checking is performed automatically when verifying the import specification, unless the **Do not perform safety checks before importing** checkbox is selected.

Bulk import details

Bulk import is a faster method of importing large quantities of data, for example in the region of hundreds of thousands of records. You can also import from XML.

Bulk import:

1. Uses a bulk insertion technique to import data into temporary tables.

   Note: You do not need to change the database recovery model to BULK LOGGED because bulk import uses bulk operations on temporary tables, and bulk operations on objects in `tempdb` are already BULK LOGGED.
2. Transfers the data into the iBase database.

Bulk import can be used with:

- Import specifications run manually from iBase Designer (they can be set up and verified in iBase)
- Import batch specifications run manually from iBase Designer (they can also be set up in iBase)
- Import batch specifications are run on a scheduled basis by iBase Scheduler

For each entity or link type imported, bulk import runs through four phases:

1. It copies the import data to a temporary table.

   The import fails if any errors are encountered while transferring the data. However, carrying out a preliminary verification in iBase would identify these errors in advance of the import. The possible errors include:

   - Data that is too long for the iBase field
   - Mismatch between locales (for example dates that are in the wrong format for the locale)
   - Too many matches because the identifiers are not sufficiently selective

   Before running a bulk import that uses a new import specification, you can verify it with some representative sample data. To verify an import, click **Verify** on the last page of the Import wizard.
2. For each entity and link type, it reports any records with any of the following errors:

   - Value not defined in list for field (for 'selected from code list' fields)
   - Mandatory field value not supplied
   - Field value too long

© N. Harris Computer Corporation

- Invalid direction
- Invalid strength
- Invalid date
- Invalid time
- Invalid numbers
- Incorrect number of columns

Bulk import then maps data to records in the database to obtain statistics on the numbers of records added, updated, and unchanged when the data is imported. All work takes place in temporary tables; no changes are made to the iBase database at this stage.

Any errors that occur at this stage prevent individual records for this entity or link type from being created or updated.

Note: Users can decide on the threshold for the number of errors that will cause an import to stop completely in phases 3 or 4. If unspecified, this is set to a default threshold of 10,000 errors. Records imported into the database before the threshold is reached will remain in the database.

For example, suppose that you are importing links, and the 'entity 1' records are already imported. During the import of the 'entity 2' records, the error threshold is reached. This means that the 'entity 1' records will remain in the database, but no records for 'entity 2' or the links would be imported.

(Standard imports will continue to run even if 10,000 errors occur.)

3. Finally, it transfers the data into the database and updates the corresponding import set. Because the data transfer and the update of the import set occur in the same step, the import set will accurately reflect the data that was imported (even after a catastrophic server failure, such as a power cut).

   If you are using audit level 4 or 5, the start and end of the import is recorded in the audit log.

Permissions for running bulk import

When iBase is run with an SQL Server database, the security context for all database operations is set at the database level and defined through the iBase application role.

Backing up databases that use bulk import

You need to ensure that the iBase database is always backed up before bulk import is run, and then again after the import has completed.

SQL Server database statistics and bulk import

A SQL Server administrator must update the SQL Server database statistics on the tables that are being loaded (including the _LinkEnd table when importing links) if the following error is displayed but you consider the identifiers to be sufficiently selective:

```
Bulk Import failed the safety check because there are too many matches; the identifiers are not sufficiently selective.
```

© N. Harris Computer Corporation

Exporting data

In iBase Designer, you can export entity and link records to text files; to export to CSV or XML files, use iBase. To export to text files in iBase Designer, you need to select the data to export and define how the exported data is formatted.

For example, you select the entity or link type to export, specify how dates and numbers are formatted and whether you want to base the export on a query or set. If you regularly export data, you can save the settings in an export specification.

You can also run a series of export specifications together using an export batch specification, which allows you to export a number of records, perhaps of differing entity or link types. The Scheduler utility can be used to export data on a regular basis.

You can also export smaller amounts of data, by using **Copy to Clipboard** when listing records in the Records dialog.

Note: In iBase, you can also consider exporting data by using report definitions to produce reports in these formats: HTML (Hypertext Markup Language), Rich Text Format (RTF), Microsoft Word, Microsoft Access.

Exporting data using an existing specification

You may need to share data from your iBase database with others. You do this by exporting the data to a file that can then be imported into another application or into a different iBase database.

The type of data to export is defined in an export specification. You can either set this up yourself or you can use an export specification defined by someone else. There will be one export specification for each entity and link type involved in the export.

For information on creating export specifications, see the iBase help.

Using XML import and export

You can import data from an XML file, using an XML schema specific to iBase and your database. XML import is a type of bulk import and so can only be run in iBase Designer, although import specifications can be created in iBase provided the user logs on as a system administrator.

You can export data to XML, although you might also want to consider exporting Microsoft Excel as an alternative. The export can contain any number of entity and link types. In iBase, you select the data to export using queries and sets, then save the selection as a database subset definition. You base the export on this definition.

There are some restrictions on how the import and export is run, as explained below. In particular, you cannot import pictures or documents.

Because XML exports can be used to export large amounts of data (potentially all the records in a database), XML exports are not audited and no alerts are raised.

Setting up the database for XML import and export

XML import and export is available in SQL Server databases that support Unicode. To allows export to XML files, Initialize the database for database subsets. You need to do this in iBase rather than iBase Designer, when you are logged on as a database administrator. Any iBase user can run XML export.

Note: You can deny access to XML export using System Command Access Control groups.

© N. Harris Computer Corporation

Generating an iBase schema file

You can only import data from XML files that use a schema that is compatible with your iBase database. To generate a schema that you can use to validate XML files for import:

- Select **Tools** > **Database Design** > **Generate XML Schema File**.

This will generate two files: `<name>.xsd` file and `iBaseTypeLibrary.xsd`.

The type library contains definitions for all iBase field types apart from Picture and Document (some field types share the same definition).

Note: All the details of the database schema will be saved in the XSD file. You might want to consider this before sharing XSD files with third parties.

Refer to the XSD files for information about the supported data types and the required file structure.

Importing data as XML

1. Select **File** > **Data** > **Import**.
2. In the Import Wizard, turn on the **Bulk Import** checkbox and select the **XML (iBase Schema) File** option.
3. Set up and run the import specification in the usual way. Notice that the following features are not available when importing from XML:
   - The **Source Field** column displays the XML element names, which should correspond to the iBase field names as you cannot reassign source fields or skip unwanted fields.
   - You cannot preview the source records.
   - You cannot transform the data in a source field by applying actions to it.

Batch import or export

If you regularly receive data from other databases, or you regularly prepare data for others, you might want to run a series of import or export sessions. You set up the specifications that you want to use in an Import Batch Specification or an Export Batch Specification.

This is simply a list of pre-existing import or export specifications that can be run with no further intervention required.

The specifications are run sequentially in the same order as the list.

For information on setting import and export specifications, see the iBase help.

Scheduling batch imports or exports

If you are importing or exporting large volumes of data, you might want to run batch specifications when your system is less busy.

To schedule this type of import or export, you can use iBase Scheduler, which might be available from the Windows Start menu: **Programsi2 iBase>Tools > iBase Scheduler Configuration**.

© N. Harris Computer Corporation

Exporting and importing externally edited iBase Data

You can export data from iBase for editing in a different application, and then re-import it. This allows you to manipulate the data in a set of records without accessing each record independently.

About this task

There are two ways of doing this. You can:

- Export the records in the usual way and then, when importing, carefully match the records so that the edited data updates the correct records.
- Export the data along with their record IDs in order to avoid the need for record matching during the import.

This topic describes how to export and import data with visible record IDs. For details of how to perform a standard export and import, see the iBase help:

Procedure

1. Before you can export data that has a visible record ID, you must ensure that you have the Record ID field type associated with each of your entity and link types.
2. Export the records that you want to edit in the usual way. You should export the data to a text file.

   Tip: You may prefer to make editing easier by including the field names in the first row.
3. Edit the data. When editing the data:
   - Do not change the values in the record ID field.
   - Do not add new records - only existing records will be updated using this method.
   - Take care not to edit or delete characters such as text qualifiers and field delimiters.
4. Reimport the edited data:
   a. In Step 1 of the Import Wizard, in the Source area, turn on **The import source contains 'Record ID's that originated from this database**. This option is unavailable if the entity type does not have visible record IDs. In this case, you will need to match the records carefully when you import them.
   b. Click **Next** to display the Step 2 page.
   c. Enter the file name. You do not need to set any of other options on this page. Click **Next**.
   d. Click **Auto Assign**, to automatically assign fields in the source data to the iBase fields.
   e. Check that the fields are assigned correctly. You must map the iBase Record ID type field to the import source field that contains the record ID. Click **Next**.
   f. You do not need to decide whether to check for matching records because iBase will automatically check against the record ID and only update existing records. Any new record IDs will not be accepted; new records will not be created. Click **OK** to continue.
   g. Complete the import in the usual way.

© N. Harris Computer Corporation

Time zones in import and export

If you want to import records containing time zones from external data sources, then you need to represent each time zone by the appropriate code. For example, in the import file, the time zone (GMT +00:00) Greenwich Mean Time: Edinburgh, London must be represented by 32.

If an entity or link type has at least one time zone field defined, then a Default Time Zone field is displayed on the final page of the import wizard. If the import file does not contain time zone values, then whatever value you select for the default will be used.

When exporting data that contains time zones, the time zone will also be represented by an iBase code.

You can include a value for the field in the imported data or provide a default time.

The time zones and their codes are listed in the following tables.

List of time zones by iBase code

The time zones in the following table are sorted in numerical order by their iBase code.

Code	Time difference	Name
1	(GMT+00:00)	Coordinated Universal Time
2	(GMT+04:30)	Kabul
3	(GMT-09:00)	Alaska
4	(GMT+03:00)	Kuwait, Riyadh
5	(GMT+04:00)	Abu Dhabi, Muscat
6	(GMT+03:00)	Baghdad
7	(GMT-04:00)	Atlantic Time (Canada)
8	(GMT+09:30)	Darwin
9	(GMT+10:00)	Canberra, Melbourne, Sydney
10	(GMT-01:00)	Azores
11	(GMT-06:00)	Saskatchewan
12	(GMT-01:00)	Cape Verde Is.
13	(GMT+04:00)	Baku, Yerevan
14	(GMT+09:30)	Adelaide
15	(GMT-06:00)	Central America
16	(GMT+06:00)	Astana, Dhaka

© N. Harris Computer Corporation

Code	Time difference	Name
17	(GMT+01:00)	Belgrade, Bratislava, Budapest, Ljubljana, Prague
18	(GMT+01:00)	Sarajevo, Skopje, Sofija, Warsaw, Zagreb
19	(GMT+11:00)	Magadan, Solomon Is., New Caledonia
20	(GMT-06:00)	Central Time (N America)
21	(GMT+08:00)	Beijing, Chongqing, Urumqi
22	(GMT-12:00)	Eniwetok, Kwajalein
23	(GMT+03:00)	Nairobi
24	(GMT+10:00)	Brisbane
26	(GMT-03:00)	Brasilia
27	(GMT-05:00)	Eastern Time (N America)
28	(GMT+02:00)	Cairo
29	(GMT+05:00)	Ekaterinburg
30	(GMT+12:00)	Fiji, Kamchatka, Marshall Is.
31	(GMT+02:00)	Helsinki, Riga, Tallinn, Vilnius
32	(GMT+00:00)	Greenwich Mean Time: Edinburgh, London
33	(GMT-03:00)	Greenland
34	(GMT+00:00)	Casablanca, Monrovia
35	(GMT+02:00)	Athens, Istanbul, Bucharest, Minsk
36	(GMT-10:00)	Hawaii
37	(GMT+05:30)	Calcutta, Chennai, Mumbai, New Delhi
38	(GMT+03:30)	Tehran

© N. Harris Computer Corporation

Code	Time difference	Name
39	(GMT+02:00)	Jerusalem
40	(GMT+09:00)	Seoul
41	(GMT-06:00)	Guadalajara, Mexico City, Monterrey
42	(GMT-02:00)	Mid-Atlantic
43	(GMT-07:00)	Mountain Time (N America)
44	(GMT+06:30)	Yangon, Rangoon
45	(GMT+06:00)	Novosibirsk
46	(GMT+05:45)	Kathmandu
47	(GMT+12:00)	Auckland, Wellington
48	(GMT-03:30)	Newfoundland
49	(GMT+08:00)	Irkutsk, Ulaan Bataar
50	(GMT+07:00)	Krasnoyarsk
51	(GMT-04:00)	Santiago
52	(GMT-08:00)	Pacific Time (N America)
53	(GMT+01:00)	Brussels, Copenhagen, Madrid, Paris
54	(GMT+03:00)	Moscow, St. Petersburg, Volgograd
55	(GMT-03:00)	Buenos Aires
56	(GMT-05:00)	Bogota, Lima, Quito
57	(GMT-04:00)	Caracas, La Paz
58	(GMT-11:00)	Midway Island, Samoa
59	(GMT+07:00)	Bangkok, Hanoi, Jakarta
60	(GMT+08:00)	Kuala Lumpur, Singapore

© N. Harris Computer Corporation

Code	Time difference	Name
61	(GMT+02:00)	Harare, Pretoria
62	(GMT+06:00)	Sri Jayawardenepura
63	(GMT+08:00)	Taipei
64	(GMT+10:00)	Hobart
65	(GMT+09:00)	Osaka, Sapporo, Tokyo
66	(GMT+13:00)	Nuku'alofa
67	(GMT-05:00)	Indiana (East)
68	(GMT-07:00)	Arizona
69	(GMT+10:00)	Vladivostok
70	(GMT+08:00)	Perth
71	(GMT+01:00)	West Central Africa
72	(GMT+01:00)	Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna
73	(GMT+05:00)	Islamabad, Karachi
74	(GMT+10:00)	Guam, Port Moresby
75	(GMT+09:00)	Yakutsk
76	(GMT-12:00)	Yankee (military)
77	(GMT-11:00)	X-ray (military)
78	(GMT-10:00)	Whiskey (military)
79	(GMT-09:00)	Victor (military)
80	(GMT-08:00)	Uniform (military)
81	(GMT-07:00)	Tango (military)
82	(GMT-06:00)	Sierra (military)
83	(GMT-05:00)	Romeo (military)

© N. Harris Computer Corporation

Code	Time difference	Name
84	(GMT-04:00)	Quebec (military)
85	(GMT-03:00)	Papa (military)
86	(GMT-02:00)	Oscar (military)
87	(GMT-01:00)	November (military)
88	(GMT+00:00)	Zulu (military)
89	(GMT+01:00)	Alpha (military)
90	(GMT+02:00)	Bravo (military)
91	(GMT+03:00)	Charlie (military)
92	(GMT+04:00)	Delta (military)
93	(GMT+05:00)	Echo (military)
94	(GMT+06:00)	Foxtrot (military)
95	(GMT+07:00)	Golf (military)
96	(GMT+08:00)	Hotel (military)
97	(GMT+09:00)	India (military)
98	(GMT+10:00)	Kilo (military)
99	(GMT+11:00)	Lima (military)
100	(GMT+12:00)	Mike (military)
101	(GMT+02:00)	Amman
102	(GMT+00:00)	Dublin
103	(GMT+00:00)	Lisbon
104	(GMT-06:00)	Galapagos
105	(GMT-06:00)	Easter Island (Chile)
106	(GMT-05:00)	Cuba
107	(GMT-04:00)	Falkland Islands

© N. Harris Computer Corporation

Code	Time difference	Name
108	(GMT-04:00)	Paraguay
110	(GMT-03:00)	Eastern Brazil
111	(GMT-03:00)	Uruguay
112	(GMT-03:00)	French Guiana
113	(GMT-02:00)	Fernando de Noronha (Brazil)
114	(GMT+05:00)	Tashkent
115	(GMT+01:00)	Windhoek
116	(GMT-01:00)	Tunis
117	(GMT-04:00)	Tbilisi
118	(GMT-02:00)	Beirut
119	(GMT-06:00)	Almaty
120	(GMT+07:00)	Chihuahua, La Paz, Mazatlan
121	(GMT-08:00)	Tijuana, Baja California
122	(GMT+04:00)	Manaus

Working with categories

You can manage your folder objects (such as import specifications) by storing them in categories. In this way you can keep folder objects together by user or by case.

Categories appear as folders, with similar behavior to Windows Explorer folders, and you navigate the folders in a similar way.

Notice that categories:

- Cannot be renamed.
- Disappear when you delete the last item in them and then close the dialog.

Access to the contents of some categories may only be available if you are a member of a particular user group.

When you create a new folder object, you may be prompted to specify a category and access restrictions. This depends on the setting of Prompt for category when saving folder objects in the Options dialog (available from **Tools** > **Feature Availability** in iBase). If you are not prompted to specify a category when you save an item, then it is saved by default in either a General category or in the default category defined in the Options dialog. By default, the access to a folder object may be public, private or restricted to a folder object control group.

© N. Harris Computer Corporation

Note: System administrators can restrict access to folder objects by users according to their membership of Folder Object Control groups. Alternatively, you can make useful definitions available for general use. Folder Object Control groups and their members are specified in the database design.

Access control is set on individual folder objects, not on categories.

To move items between categories, you need to use iBase rather than iBase Designer.

Category defaults

The default category to use, access level and whether to select the default automatically or prompt are store in the iBase options. You define the default behavior for categories by selecting **Tools** > **Feature Availability** > **Options**.

Default Category Name

To avoid the need to reselect a different category each time, you can set a default category. A category with this name does not have to exist. In the **Default Category Name** box, enter the default name to use.

Default to 'Public' access

Access to a folder object in a category can be public, private to the user who flagged it as private, or restricted to members of a folder object control group. Private folder objects cannot be listed or viewed by any other user (apart from the system administrator).

To set the default access type for any new folder objects that are created:

- Turn on - new folder objects are public by default (overrides any membership of folder object control groups to which you belong).
- Turn off - new folder objects are private by default, or default to the folder object control group to which you belong.

  Note: If you belong to several folder object control groups, you are prompted to select an access type.

Prompt for category when saving folder objects

Folder objects are always saved in a category. However, you can choose whether to use the defaults, or select the category and set the access type before saving the object:

- Turn on to select the category and access type each time a folder object is saved.
- Turn off to automatically save folder objects in the default category with the default access type where possible.

Saving into a category

Folder objects can be stored in existing categories, or into categories that you define when the items is saved.

Procedure

To save an item to a category:

1. When editing a folder object, click **Save**.

   You are prompted for the item name.

   If you are not prompted to select a category after clicking **Save**, then you cannot change the category. You can, however, change it later.

© N. Harris Computer Corporation

2. Enter the name for the item, which must be unique to the database, and then click **OK**. The default category is displayed in the **Selected Category** box. If the default category is empty, (does not exist yet) is displayed.
3. **Optional:** Select a different category in which to save the item. The category name is displayed in the **Selected Category** box.

   Tip: To create a new category, simply enter the names of the categories you want, separated with a backslash (\. For example: `Operation Crest\Unit B\Vehicle Owners`. In this example, the category Unit B will be created for you if it does not yet exist.
4. **Optional:** Restrict who can access the item by clicking **Restricted to groups** and then selecting the groups who can access the item.
5. Click **OK**.

Setting folder object access

Folder objects can be set so that only a restricted set of users can access them. You can set the access level on individual objects, or a selection.

Procedure

1. When you modify multiple folder objects that currently have different access restrictions, you need to select **Change Access** before you can modify the access restrictions.
2. Select the type of access to apply:
   - **Public**

     Any user can access the folder objects.
   - **Private**

     Only the person who flagged the folder object as private and the system administrator can see it. For example, unless you are logged on as a system administrator, private import specifications belonging to others will not appear in the All Import Specifications folder.

     Note: If you are a member of a folder object control group, membership of this group may prevent you from setting the access on folder objects to private. Access to the object will always be set to the folder object group. For detailed information, see below About folder object control groups.
   - **Restricted to groups**

     Only users who are in the specified groups can see these folder objects.

     With **Restricted to groups** selected, check the Folder Object Control groups that you want to have access. You only see the Folder Object Control groups of which you are a member.

Folder object control groups

A folder object control group is the most restrictive of the three types of folder access: public, private, and group. Folder object control groups are defined in iBase Designer using the Security Manager but their usage is defined by the users who belong to the group, and the category defaults.

Folder object control groups

The following category defaults are set in **Tools** > **Feature Availability** > **Options**:

- **Default to 'Public' Access**
- **Prompt for category when saving**

© N. Harris Computer Corporation

How these settings affect the use of folder object control groups is summarized below.

Category prompt	Default access type	Result when you save a folder object
ON	Private	You are prompted to select the access type for the folder object. The default access type is the folder object control group to which you belong. If you belong to several groups, the access type defaults to private.
ON	Public	You are prompted to select the access type for the folder object - the default access type is public.
OFF	Private	The folder object is automatically saved in the folder object control group to which you belong. If you belong to several groups, you are prompted to select one or more groups, or to change the access type.
OFF	Public	The folder object is automatically saved in a public folder - overriding the membership of the Folder Object Control group. If you belong to several groups, you are prompted to select one or more groups, or to change the access type.

Creating a record of actions for your database

You can set up iBase to log virtually all user actions with or without user-supplied reasons the actions. Different auditing levels can be set depending on the requirements of your environment.

What is recorded?

iBase starts auditing at the lowest possible level of detail when you create a database. You cannot stop this level of auditing but you can choose to start at a higher level, and to modify all auditing options for existing databases. See Controlling What is Audited for further details.

Attention: The option to record user accesses to records, without change of data, creates large volumes of log data so it is available only with iBase SQL Server databases. Use this option only when strictly required. Your SQL Server administrator can configure the disks to improve performance in this area. For more information, see SQL Server clients, servers, and networks.

© N. Harris Computer Corporation

Independently of the audit level of the database (SQL Server format only), you can audit changes to data. The iBase field types that you can audit depend on the SQL Server version. See Audit History.

Note: For more information about auditing in a replicated database, see Replicating and synchronizing databases.

Where is it recorded?

Separate audit logs are created for security files and databases.

Security file logs track the opening of databases, failed logon attempts, and a range of administrative actions such as creating templates, and managing users and groups.

Database logs track the opening and closing of databases, historical data (if logged), and all the requested actions within databases. Actions are recorded regardless of origin: users can request database actions from iBase Designer, iBase, Analyst's Notebook, or third-party mapping applications.

The physical form and location of logs is different for security files, Access databases, and SQL Server databases. The audit viewer handles these differences and can produce archive files in a standard form.

Viewing audit logs

To use the Audit Viewer, a user needs to be a system administrator, a database administrator, or an audit administrator.

The Audit Viewer, if installed, is available from the **i2 iBase** section of the Windows start menu. You can view and manage audit logs for databases and security files. You can open multiple windows to inspect logs for several databases if those databases are managed through the same security file.

Audit Viewer might not display all the entries in the audit log:

- Some users generate restricted audit log entries that you need the Audit Administrator role to view.
- Some audit log entries are hidden if SC codes are used (you can only view the entries for records that match your security classification).

The level of detail in the audit log is determined by the audit level set for the database.

Audit log databases

If you are using iBase and an Access database, the database log is held in the `.idl` file that is stored in the same folder as the database file.

In an SQL Server installation of iBase, an audit log database is created alongside the main SQL Server database. The name of the database is the same as the main database name with the suffix `_log`. For instance, the database `User_Guide` has an audit log database `User_Guide_log`.

Your SQL Server administrator must ensure that iBase users can access this audit log database. For more information, see Access control.

For information on backing up audit log databases, see Archiving audit logs.

Controlling what is audited

iBase starts auditing at the lowest possible level of detail when you create a database. You cannot stop this level of auditing but you can choose to start at a higher level, and to modify all auditing options for

© N. Harris Computer Corporation

existing databases. The audit level applies to all users equally, and only to the database in which you specify it.

Each of the available auditing options and the circumstances when you might want to use them are described in the following information.

Note: Independently of setting the audit level, you can configure the database to log commands that are run by users, case control, and audit history. For more information, see:

- System Commands Access Control Groups
- Working with cases
- Audit History

Audit levels 1 - 5

Level 1 records the least detail and level 5 records the most detail. The level of auditing is cumulative, each level records the information for all lower numbered levels. For example, level 3 records queries and all information specified by levels 1 and 2.

Level 1

Logs each time that a user logs in, a database is opened or closed, or when an email alert is sent.

Note: If the database is configured to audit the use of commands, or to request a reason for use of a command, those commands, and reasons appear at this level. If your SQL Server database is set up for Audit History, extra logging occurs at all levels. Also, if an SQL Server database is case-controlled, the log always records when cases are added, modified, deleted, renamed, closed, or reopened.

Level 2

Also logs when entity types, link types, and fields are added, changed, or deleted. In other words, this level logs a change of database design.

Level 3

Also logs each time that a query is run on the database. The query can be direct, for example by using Find, Browse, Query, or Search 360; or indirect, for example by using a browse definition based on a query. Search 360 search criteria are audited at level 3 and upwards.

Note: The log does not include work on sets or how the data was retrieved.

Level 4

Also logs when entity and link records are added, changed, or deleted. In other words, this level logs a change of database data content. The log includes when records are soft-deleted, or purged and when a conflict is detected, or restored, or solved.

Note: The log does not include individual records that are affected by a Bulk Import, only the start and end of the import is recorded.

Level 5

Also logs when entity and link records are accessed or viewed, without change to the data. This logging produces large volumes of audit data and for this reason, is available only for SQL Server databases.

Note: The log does not include individual records that are affected by a Bulk Import, only the start and end of the import is recorded.

© N. Harris Computer Corporation

Attention: Because XML exports can be used to export large amounts of data (potentially all the records in a database), XML exports are not audited.

More about audit level 5

Audit level 5 produces high volumes of audit data. For this reason, it is available only with iBase SQL Server databases. Use this option only when strictly required.

As a way of controlling the volume of audit data, you can set **Number of records to be displayed before auto-pausing** to a low number. When the audit level is 5, this option pauses the listing of records, returned by a query or browse, at the specified number.

The useful consequence for auditing is that the audit log records only the number of records that the user views. For example, if the user cancels after a pause that shows 50 records, only those first 50 records are shown in the audit log. If the user continues to list the other records, those records are audited as normal.

Audit level 5 can be used with Reason for Action entries. See System Commands Access Control Groups for details.

Audit history

In SQL Server database, changes to the data in entity and link records, and code lists, can be recorded if the **Audit History** is turned on. For audit levels 1 - 4, changes to the data are recorded and additionally, at audit level 5 record accesses (views) are logged. A reason for an update can also be recorded as part of the audit log of a record. See Audit History for details.

Note: If you initialize a database for alerting, audit history is automatically turned on. Alerting must be turned off before audit history can be turned off. The audit history provides the details that enable users to understand the edits and views that raised the alerts. The same details are displayed regardless of the audit level of the database. A user who is denied access to the Audit History cannot see alert details.

Audit log options

Depending on the type of database and your logging requirements, you can define how log data is written to the Audit Log database with the following options.

- Choose the initial level of auditing detail for a new database.

  In iBase, select **File > New Database > Details > Audit Level**.
- Change the audit level for an existing database.

  In iBase Designer, select **File > Database Properties > Audit Level**.
- Audit the usage of selected commands.

  In iBase Designer, select **Security > System Commands Access Control**.

  - Selecting any command groups on the **Reason for Action** page will prompt the user for a reason for running the command. After the user supplies a reason, iBase adds the text to the audit log (as Detail). This reason will subsequently be used as a default for all subsequent reasons within the same session of work.
  - Command groups selected on the **Audit** page record the action without prompting for a reason or otherwise notifying the user.

© N. Harris Computer Corporation

Auditing that is configured in this window applies to particular groups of users, at all audit levels, and to all databases accessed through the same security file. For more information, see System Commands Access Control Groups.

- Record the history of changes to individual records in SQL Server databases.

  From the **File** menu in iBase Designer, select **Database Properties**. Use the **Audit History** checkbox in the Database Properties dialog box. You can also configure Audit History to disable the guest account and replace it with an existing SQL Server account for audit history logging.

  For more information, see Audit History and Changing account used to log audit history.

- Activate case control in a new SQL Server database.

  In the iBase window, select **Create New Database** and click **OK**. Use the **Case Control** option on the Advanced tab of the Advanced page to set up case control in a new database before any data has been added to it.

- Activate case control in an existing SQL Server database.

  From the **File** menu in iBase Designer, select **Database Properties**. Use the **Case Control** option on the Advanced tab.

Audit history

In SQL Server databases, changes to the data in iBase entity and link records can be recorded.

Changes are recorded following these iBase operations:

- Entering and editing records
- Deleting records (including soft-deleted records)
- Batch editing
- Merging entities
- Assigning icons
- Importing data, including bulk import
- Editing code lists

Audit History is independent of the audit level of the database and, if used, the following actions become available at all audit levels in Audit Viewer:

- Record Added
- Record Modified
- Record Deleted (not including soft-deleted records)
- Code List Modified
- Bulk Import

However, in a database with audit level 5, you can also find out who viewed specific records.

Note: Audit History is automatically turned on if you initialize a database for alerting and you cannot turn it off when alerting is active. The audit history provides the details that enable users to understand the edits and views that raised the alerts. The same details are displayed regardless of the audit level of the database. A user who is denied access to the Audit History cannot view the details.

© N. Harris Computer Corporation

Audited field types

Aside from data associated with calculated fields (that is not directly stored, but depend on values held in other fields) all field types can be audited. In the audit log, all data is converted to text apart from Document and Picture fields which are stored in their original format. You can view this historical data in Audit Viewer or in iBase itself when showing a record or link, unless permission to do so is denied.

What is recorded

The following are recorded when a record is updated in iBase:

- Original value
- iBase user who made a change
- Date and time the change was made
- Machine name of the editing user
- OS user name (name of the Windows user)
- Reason for the change
- SCC -- needed to ensure that the user only sees the data they should if SCC values are altered during records history
- Location of user -- from iBase user location
- Reason for the update (optional)
- Whether the update was made using an i2 product
- Data in the extra field (if this feature is used)

The following is recorded if an iBase record is updated directly in SQL Server:

- The name of the account used to connect to SQL Server

If a single record was changed, the audit log records a Record Modified action and the record ID is displayed in Audit Viewer. This is not possible for a bulk import when the audit log records a bulk import action.

Note: Changes to code lists are also audited, that is old and new values, descriptions and parent pick lists.

Setting up audit history

To enable and set up audit history, in the iBase Designer Database Properties, turn on **Audit History**. An Audit History action is added to the audit log to record when, and who, enabled this feature.

You can require users who modify records in iBase to enter the reason for the edit before they can save their changes:

- In iBase Designer, select **System Commands Access Control >Reason For Action** and turn on or off **Data Auditing**.

Note: You might need to run the `Tools Update Command Groups` command first.

By default all users are able to view the audit history. To deny users access, edit the appropriate user group:

- In iBase Designer in the System Commands Access Control dialog, display the Access Denied page and turn on or off the **View History** checkbox.

© N. Harris Computer Corporation

You can also configure audit history to disable the guest account and replace it with an existing SQL Server login for audit history logging. For further details, see Changing account used to log audit history.

Maintaining auditing stored procedures

When a user logs into a database with **Audit History** turned on, checks are made on the SQL Server database and, if any problem is detected with auditing, the user is denied access to the database. To fix the problem, reopen the database in iBase Designer.

Changing account used to log audit history

By default Audit History enables the guest account in the iBase log database. However, you can disable the guest account, and use an alternative SQL Server user for logging Audit History information to the log database.

Note: If you are using Azure Managed Instance, you cannot use an alternative SQL Server user to log audit history. You must use the guest account.

Before you begin

Attention: These steps are not reversible. After the guest account is disabled, you cannot enable it again.

The SQL user that you use to replace the guest account must be associated with a login that also has a corresponding account in the iBase log database. It needs appropriate permissions in each database.

To create an account to replace the guest account, create a login that is associated with a user in the iBase main database and the iBase log database.

The following database roles are needed:

- The iBase main database user must be a `db_datareader`.
- The iBase log database user must a `db_datawriter`.

Procedure

To disable the guest account and use an alternative SQL Server user account:

1. Display the Configure Audit History by clicking **Database Properties >Configure**.
2. Connect to the SQL Server as a user with system administrative permissions. Select **Use specific SQL Server account**.
3. Select an Authentication type:
   - Windows authentication. Your Windows user account must have the system administrator permissions.
   - SQL Server authentication. You must enter an SQL Server login, and password, that has system administrator permissions.

   Note: You can use either method of connecting to the server, regardless of the security method that is used in the iBase connection file.
4. Click **Connect**.
5. In Audit history logging account, select an SQL Server user to replace the guest account.
6. In Authentication with iBase Log Database, enter a certificate password to be used by the SQL Server to:

© N. Harris Computer Corporation

- Create the certificate.
- Back up the certificate (the password is required to restore the certificate from the backup).
- Provide the security context for logging audit history.

7. Click **OK**.

Restricted audit logs for sensitive data

You can restrict access to the audit logs of users who work on sensitive data. This requires changes to the user accounts of those who work on sensitive data and to the user accounts of those authorized to view restricted audit logs.

Before you begin

Users who do not have access to restricted audit logs can still view the audit history of any record accessible to them but they cannot see any of the changes made by users with restricted audit logs. Users who generate restricted audit logs cannot view the restricted audit logs of other users.

Note: You can also prevent users from viewing the history of entity and link records by denying access to the History button. To do this, you turn on the View history checkbox in the System Commands Access Control dialog.

For each user who works on sensitive data:

1. In iBase Designer in the Security Manager, edit the account of the user whose audit log you want to restrict.
2. On the Account page of the User dialog, turn on **Restricted Audit Log**.

This will prevent unauthorized users from filtering the records displayed in Audit Viewer or the iBase Audit History dialog to view just the records touched by that user.

Note: By default, system administrators do not have permission to view restricted audit logs. They must be granted this permission like other types of user.

Procedure

To grant permission to view restricted audit logs, you need to create a new Database Management group that includes the Audit Administrator role or edit an existing group. For example:

1. In iBase Designer in the Security Manager, edit the group to which you want to add the Audit Administrator role.
2. On the Permissions page of the Group dialog, turn on **Audit Administrator**.
3. On the Users page of the Group dialog, review the list of users in this group as the existing members of the group will acquire the new permission. For more information, see Creating Groups and Adding Members.

What to do next

A user who has the Audit Administrator role can use the Audit Viewer or iBase Audit History, and they can view the records touched by any iBase user.

© N. Harris Computer Corporation

Auditing records from external databases

In a database with an audit level of 4 or 5, the audit log can record data from a named field whenever it logs an audited action specific to an entity or link record. The value is displayed in the **Extra Detail** column of the audit log.

Before you begin

A field needs to be specified provides the extra detail:

- In the **Create New Database** or **Database Properties**, enter the field name in **Extra Detail for Audit Log**.

The **Extra Detail for Audit Log** has a specific application that requires some detailed work when designing and creating databases.

The specific application is to import data from an external database and maintain a way of associating records in the external database with records in the iBase database. The imported records must have a unique reference from the external database.

Procedure

1. In iBase Designer, create a field or standard field to hold a unique reference from the external database. You need to add this to all entity and link types that might be linked to external records. This field must have the same name in all those entity and link types. Using a standard field is a convenient way to ensure this consistency.

   Note: The standard field is also present in entity and link types that are not imported, but you can set the default value so that in these non-imported types the field is blank or has a non-conflicting value.
2. In the Advanced page of the Database Properties dialog:
   a. Set the **Extra Detail for Audit Log** field to the name of the iBase field holding the unique reference.
   b. Select either audit level 4 or 5 so that the audit log records actions on specific records. This means potentially large audit logs.
3. When you import data from an external source, import the unique identifier used in the external database or system into the field that you have created to hold a unique reference.

Results

After you have done all this, the audit log will record the value of the named field for all actions affecting imported records, in the **Extra Detail** column. The audit log will also record the iBase unique identifier, in the **Record ID** column.

Auditing options

The different options available in iBase to control what is viewed and who can view it.

For example, to be able view restricted audit logs (history) in iBase, a user requires the following permissions:

- **View History**, set in the System Commands Access Control dialog
- Audit Administrator role, set in the User dialog

© N. Harris Computer Corporation

Option	**Required permissions**		
	View History	**Audit Admin role**	**Database Admin role**
View history in iBase (Details of the changes to the record data are only available in databases with the Audit History property turned on.)	Yes	-	-
View restricted history in iBase (Only restricts what is viewed in the Audit History and Audit Viewer dialogs. Users will not know who worked on the record and when but they may be able to work out what some of the changes were.)	Yes	Yes	-
View Audit Viewer	-	Either	Or
View history in Audit Viewer (Applies to the Record Modified and Code List Modified actions, at all audit levels.)	Yes	-	Yes
View restricted entries in Audit Viewer	-	Yes	Yes
View restricted entries & history in Audit Viewer	Yes	Yes	Yes

Note: The facility to restrict audit logs is set for individual users as part of their user account in the User dialog.

Viewing audit logs

You can use the Audit Viewer to view the audit entries for a database that has already been configured for auditing. The physical form and location of logs is different for security files and database files. In addition different options are available for Microsoft Access databases, and SQL Server databases.

To view audit entries, you must be a system administrator, a database administrator, or an audit administrator. However, it is important to note that not all entries might be accessible:

© N. Harris Computer Corporation

- Some users generate restricted audit log entries that you need the Audit Administrator role to view.
- Some audit log entries are hidden if SC codes are used (you can only view the entries for records that match your security classification).

The level of detail in the audit log is determined by the audit level set for the database within iBase Designer, and any changes to that audit level will affect the creation of future entries in the log, not previous actions that have already occurred. You can open multiple windows to inspect logs for several databases if those databases are managed through the same security file.

You can view the audit logs for security files and databases:

- Security file logs record the opening of databases, failed logon attempts, and a range of administrative actions such as creating templates, and managing users and groups.
- Database logs record all the requested actions within databases, and the closing of databases. Actions are recorded regardless of origin: users can request database actions from i2 iBase Designer, i2 iBase, i2 Analyst's Notebook, or third party mapping applications. You can inspect logs for several databases provided that those databases are managed through the same security file.

Viewing the audit logs allows you to monitor usage of iBase databases and commands. For example, you can find:

- Failed logons.
- Microsoft Access to databases by unexpected users or at unusual times.
- Use of commands that send data outside of iBase: to a printer, to a file, or to an external application such as Analyst's Notebook or a mapping application.
- History of changes to single iBase entity or link records and who made them (if you log historical information).
- Journal Entries detailing the specific comments of an individual user.

As audit logs are potentially very large files, which the viewer displays as a grid of rows and columns, where each row represents an action on a database or security file and each column provides a different piece of information about an action. Much of Audit Viewer is designed to provide ways to identify and arrange actions (the rows) that are interesting or related in some way.

You can:

- Print the displayed actions
- Export the actions to a file for further analysis using a spreadsheet, database, or other visualization tool
- Archive them to a standalone database file

Opening an audit log

When you log on to iBase Audit Viewer, you open a security file that defines the permissions for the user account to which you are logged on. When your credentials are recognized, you can select the type of log you would like to view.

Procedure

1. When you have started Audit Viewer, select **Logon** to log on to a security file without opening a log or archive.
2. Select one of the following commands from the **File** menu:

© N. Harris Computer Corporation

Option	Description
Open Database Log File	Displays the database log for viewing.
View Security Log	Displays the security log for viewing.
Open Archive File	Displays an archive of an audit log.
Open SQL Server Archive	Opens the SQL Server archive database. (**Note:** You must specify a valid archive database and not a standard iBase database.)

Results

The viewer opens each log or archive in a separate window with a title bar matching the title of the log or archive. You can maximize the window within the application window. On opening each log or archive, the viewer displays all of the logged actions today, meaning the day of viewing the log.

Note: Slight differences in the contents of the window occur depending on whether database logs, security file logs, or archives are viewed.

The grid displays the audit log data, one logged action on each row. For database logs, the columns in the grid are:

Date

The date and time of the action.

User

The iBase logon name of the user for the action.

Action

The action, such as Database Opened or Record Added. Double-click the action to display the detail.

Record ID

The iBase record identifier, if the action referred to an individual record, or blank if the action refers to multiple records as the result of a bulk import. If Audit History is turned on, double-clicking the record ID displays the history of that record.

Extra Detail

The contents of a field chosen by the database designer if the action referred to a specific record.

Location

The location of the user performing the action.

Note: The location for the user is recorded at the time the log entry is made. Subsequently, the location may have been edited, but the log reflects the correct location at the time of the action.

Filtering by location will not identify the records which a user owns (if you are using owner hyperlink fields), only those records that they create or update.

Network Login

© N. Harris Computer Corporation

The machine and user identifier of the user performing the action, as identified by Microsoft™ Windows™. This uniquely identifies users who log on using single sign-on rather than an iBase user account.

Note: The network login is displayed in both the **User** and **Network Login** columns for changes made outside of iBase as the machine name cannot be determined for this type of change.

Detail

Information, typically the item affected or setting changed by the action.

What to do next

You can change the rules used to display the log entries on the Selection Criteria and Actions pages. To apply your changes, click **Refresh**.

On the Selection Criteria page, for example:

- You can extend the date range if there is no data shown for today.
- You can view subsets of log data based on various criteria. For example, actions made at a time of day specified by start and end times, on particular days of the week, or by a specific user.

On the Actions page:

- You can filter the types of actions displayed. For example, you might want to know when a database is opened.
- You can change the type of actions that are available for display by selecting an audit level in **Display actions**

You can use wildcards to include or exclude specific log entries. For example:

- Entering [!user1] in Detail (contains) excludes log entries containing user1 in their Details field.
- Entering [user1] includes these log entries.

Logging on

To log on to the audit viewer:

Procedure

1. From the **File** menu, select **Logon** or **Logon As**. Select **Logon As** if you usually log on using your Windows user name and password but on this occasion want to log on using an iBase user name and password.
2. In the Security File dialog, browse for the security file to open. The file name will end with .ids.
3. Click **Open**. The Logon dialog may be displayed if you have an iBase user name and password. If you use your Windows user name and password, then the dialog is only displayed if you are able to log on as one of several iBase users.
4. If the Logon dialog is displayed:
   - either, enter your iBase user name and password
   - or, select the iBase user from the list

   Tip: to avoid this step in the future, turn on the Remember my selection checkbox
5. Click **OK** to open the security file.

© N. Harris Computer Corporation

Logging on as a different Windows user

Depending how Windows security is set up at your site, you may be prompted to select the user to log on as. To avoid repeating this step each time you log on, you may have turned on the **Remember my selection** checkbox in the Logon dialog.

Procedure

To cancel this selection:

1. Start iBase, not Audit Viewer.
2. Log on in the usual way (you do not need to open the database).
3. Select **Tools** > **Options**.
4. On the General page, turn off the **Remember user for Windows single sign-on** checkbox.
5. Log off.
6. Log on to Audit Viewer and you will then be prompted to select the iBase user log on as.

Audit log entries

The audit log can be sorted, searched, and filtered to make locating specific entries easier.

To navigate through the log entries displayed in the grid, you can:

- Use the scroll bars to scroll horizontally or vertically.
- Click any cell in the grid, then use the `Page Up` and `Page Down` keys.

To sort the log entries displayed in the grid, select the appropriate sort column from the **Sort Order** list.

Double-click anywhere on the row of the log entry to display details. For most entries, this provides the details of the action performed in a separate form. If you are accessing a log for a SQL Server database, a number of extra options are available:

- Double-clicking rows for actions that changed database records, shows the history of the changed record.
- Text can be copied from rows.
- Information can be appended to the details section of a row (like a Journal Entry, this is in the format of an additional string with the username and timestamp).

Exporting the log entries

You can select **Action > Export** to export the displayed log entries to a comma separated value text file (.csv file). This file can be read into a spreadsheet or other visualization tool.

Note: The audit history is not exported.

Printing the log entries

You can select **Action > Report** and click the printer icon from the toolbar (or select the printer icon from the Report toolbar), to print the log entries displayed in the grid.

Note:

- The audit history is not printed.
- Log entries are printed using Landscape layout.
- You might find that the contents of the Detail column wrap around to the beginning of the next line.

© N. Harris Computer Corporation

Deleting and archiving the log entries

The only way to delete selected data from the audit log is to create an archive. If you no longer require the data you placed in the archive, you can delete the archive file.

Note: For SQL Server databases, you can back up the whole archive database and then delete it. An empty archive database is created the next time a user performs an auditable action.

Changing the selection criteria

To reduce the scope of information that you are searching within an audit log, you can add selection criteria. This allows data to be manually checked, but also allows reports to be generated about specific date ranges, users or records.

Procedure

1. Open the audit log you are interested in.
2. On the Selection Criteria page, refine the audit entries based on criteria:
   - **Between these dates**

     A date range to list activity between.

     Note: The date range must be valid, with the From date occurring before the To date.
   - **Between these times**

     A time period within the date range to list activity. For example, activity that happened after specified office hours, or at lunch time, on days in the date range.
   - **On these days**

     Select activity that occurs on a particular day of the week in your specified date and time range.
     - <All Days> - any activity
     - Weekdays - any activity that occurs Monday to Friday
     - Weekends - any activity that occurs Saturday or Sunday
     - Mondays - Sundays - any activity on the day specified
   - **User**

     Activity logged for a specified User account within the date and time range.
   - **Location**

     The registered geographic area that the activity happened within.
   - **Network Login**

     The registered machine and logged in user that logged the activity.
   - **Record ID**

     Activity for a specific database record.
   - **Extra Detail**

     Searches for information added to the audit log from the field set up in the advanced database settings to add extra auditing information. For more information about extra details, see Adding extra details for auditing.
   - **Detail (contains)**

© N. Harris Computer Corporation

Allows you to search for text in the details of audit entries.

For information on how to refine the log based on types of auditable action, see Changing the types of actions.

3. Click **Refresh** to update the audit log view.

Changing the types of actions

When you start Audit Viewer, it lists entries for all possible actions at the current audit level. You can change the listed actions to match a different audit level, or select specific types of actions to list.

Procedure

To change the type of activity that is listed in the log:

1. Click the **Actions** tab to display the Actions page.
2. Select the actions that you would like to list.
   - Change the list of audit-able actions by choosing an audit level in the **Display actions** list.
   - Select the actions in the report by selecting them in the displayed list.

   Note: If you select a higher level of audit than the current level set in the database, only actions that were carried out in a period when the database was also set to that higher level will display in the log.
3. Click **Refresh**.

Viewing audit histories

In SQL Server databases, you might be able to view the history of the changes made to records in the database provided your database is set up to use Audit History. How far back the history goes depends on how frequently your system administrator archives this data.

You can access the audit history of a record from:

Audit Viewer

To display the audit history of a record, double-click an audit log entry with a Record Modified action.

iBase User

To view the history of a record:

- From a list of records, right-click and select **Show History**.
- With a record open directly (not using a datasheet), select **History**.

Analyst's Notebook

With an item selected on the chart surface, in the Data Sources task pane, select **Show > History**.

With the Audit History open, you can now filter the records that are displayed by user, entity and link type, and by time, or you can extend the selection to include other records of interest.

Note: In a database set to audit level 5, the number of times the records have been viewed, but not edited, is displayed in the Views column.

Selecting records of interest

To select further records of interest, click **Select** in the top right of the Audit History.

To display all records touched by a specific user:

© N. Harris Computer Corporation

- In the Records to display area, select **All records used by** and select the user name from the list. The selected user name will be displayed as a reminder in the top left of the Audit History.

To filter by entity and link types:

- In the Types to display area, turn the entity or link type checkboxes 'on' or 'off'.

  Note: Only types with records in your selection are listed, and the records for a selected type are displayed only when the appropriate checkbox is turned on.

To filter by time:

- In the Time period to display area, select a time period.

Changing what's shown in the edit history area

The entries in the edit history area can be displayed in a variety of ways:

1. Make sure that `by all users` is selected in **Show Edits**.
2. Turn on or off the following options:
   - **Show Headers** to hide or show the shaded line that displays either the date/time/user name or the field name. You cannot expand and collapse when this option is turned off.
   - **Expand** to show the complete history.
   - **Collapse** the complete history to just display the headers.
   - **Audit** groups the entries by the name of the user who worked on the record and when they were created, updated or deleted.
   - **Field** groups the entries by the data that has been added, updated or deleted. Click again to sort in ascending or descending order by date edited.
   - **Edits** displays a history of the changes to the record (only available if the database is set to audit level 5).
   - **Views** displays a history of who viewed the record and when (only available if the database is set to audit level 5).

In the edit history area, you can filter the edits to those made by a specific user. Filtering by user makes the other display options unavailable:

- Select **Show Edits > `user_name`**. The users on this list are selected by clicking **Select** in the top right of the Audit History dialog.

Copying the edit history

Click **Copy** to copy all the information shown in the History of edits area to the Windows clipboard.

Note: You cannot copy image and document fields.

Description of the columns in the history of edits area

The history of edits area shows information on the changes made to the selected record:

Information shown	Description
Field Name	The old and current values.

© N. Harris Computer Corporation

Information shown	Description
Edited by	The logon name of the user who made the change.
Date Edited	The date and time of the change.
Reason	If required by the database, the reason given by the user for making the change.
OS User	The Windows name of the user made the change.
Machine Name	The machine that the user was working on.
Location	The location as entered in the User Information dialog.
iBase Change	When this checkbox is turned off, the update was made outside iBase (and Audit Viewer might be unable to determine the machine name).
Extra Detail	You might see an additional Extra Detail column that displays additional information for the current record. (For information on how an administrator can set up the audit log to record extra detail, see Adding extra details for auditing) .

Additional historical data

Additional data may be shown for each record. This can include:

- the name of the icon if an alternate icon is assigned to the record.
- the icon color (which will be blank if the standard icon color is used).
- the record status (applicable only if Soft Delete is used). The record status might be Soft Deleted, Normal (because the record has been soft deleted or restored), and Purged.
- Security Classification, the old and new SC code (if this feature is used and if you have authority to view this information).

Some information might be displayed that you do not usually see, such as the date the record was created and the record ID.

Code list histories

Changes to pick lists, icon lists, or SCC lists are classed as **Code List Modified** actions in the Actions page.

Details could include:

- old and new values
- old and new descriptions
- old and new parent pick lists, for filtered pick lists

© N. Harris Computer Corporation

All the changes made in the same session are grouped together by user name, date and time. As there may be several pages of changes, you can print the list or save it as a Microsoft Excel spreadsheet or PDF file. How far back the history goes depends on how frequently your system administrator archives this data.

Note: Changes to code lists are only logged if the database is an SQL Server database and audit history is turned on. To find out whether the database logs audit history, check the Audit History setting in the Database Properties.

Filtering using sets

If you are only interested in events that affect a given set of records, you can filter the audit log entries based on set membership. This feature is only available for SQL Server databases.

Procedure

1. Click the **Set Membership** tab to display the Set Membership page. Any
2. Choose the sets of records used to filter audit log entries:
   - To add a set - Click **Add**, and select the set to use.
   - To remove a set - with the set selected in the list, click **Remove**.
3. Click **Refresh** to update the audit log with your selection.

   Note: Log entries that are not associated with a records, are not filtered by this feature.

Saving filters

If you have changed the types of entries listed in the audit log, you can save the criteria as a filter. This filter can then be reapplied should you want to investigate the same types of activity in the future.

Procedure

To set up a filter:

1. Choose the types of audit log entries to display. For more information on changing the types of audit log entries, see:
   - Changing the selection criteria
   - Changing the types of actions
2. On the Filters page, select **Save**.
3. Enter a name for your filter, and press **OK**.
4. Choose the category and access level used to store the filter, and press **OK**.

Results

You filter is saved. You can **Apply** your filter to use this set of criteria on the audit log in the future.

© N. Harris Computer Corporation

Creating an audit log report

You can take the audit log that you have open, and convert it into a report. The database audit log can be printed, or exported into a file format for your records.

About this task

Depending on the type of audit log that you have open, and the information that you are interested in, the audit log report will display different information.

Procedure

To create an audit report:

1. In the Audit Viewer, open an audit log and filter the log using the **Selection Criteria** and **Actions** to determine the contents of interest. If you regularly filter your audit information in the same way, you can save the filtering options as a **Filter**, that can be applied in later sessions.
2. To generate the report, click **Action** > **Report**.
3. With the report open, you can use the options in the toolbar to:
   - Refresh - check for new actions.
   - Print - send the report to a printer.
   - Export - save the report as a file compatible with Microsoft Excel (*.xlsx), Microsoft Word (*.docx), or a PDF document.

Example of using the audit viewer

You might want to discover if any user opens a database outside of normal working hours, here taken to mean between midnight and 09:00.

Procedure

1. Click the **Selection Criteria** tab to display the Selection Criteria page.
2. Set the date and weekday ranges, time of day, and User ID:
   - In the **Between these dates:** box, select an appropriate start date. For example, to set a date two months before today, click to highlight the month part of the date and press the down arrow key on your keyboard. In the to: box, leave the end date at today's date
   - In the **Between these times:** box, enter the time that you want logging to start. For example, the earliest time you can enter is midnight, which you must enter using the 24-hour system as 00:00:00.
   - From the **On these days:** list, select the appropriate days of the week. For example, to look at all days, leave the setting at its initial value of <All Days>.
   - In the **to:** time picker to the right, enter the time that you want logging to stop. For example, normal working hours may start at 9 am, which you must enter using the 24-hour system as 09:00:00.
   - You want to see all users who have opened databases outside of normal working hours so you can leave the value in the User box as <Any User>, which is the default. You could use wildcards in this and other text boxes.

   This set of rules will find all the users who have performed a logged action in the date and time ranges you have set. The grid does not change automatically.

© N. Harris Computer Corporation

3. Click **Refresh** to find the log entries matching these rules.
4. You can filter the log entries by selecting one or more actions. For example, you can only view the actions when a database is opened:
   - Click the **Actions** tab to display the Actions page. When you first open a log, this page lists all possible actions for the current audit level.
   - Click **Clear All** to turn off all actions.
   - Turn on only the **Database Opened** checkbox.

   This sets all the necessary rules. The grid does not change automatically.
5. Click **Refresh** to display actions based on all the rules that you have specified. The matching results are displayed in the grid.

Working with security logs

The security log lists the transactions of interest. To open a security log, select **File** > **View Security Log**.

There are two possibilities:

- The log is displayed immediately. (This is the log for the security file to which you are currently logged on.)
- A Security File browser is displayed where you can locate and open a security file. Audit Viewer opens a security log once you have successfully logged on to the security file.

Compared to working with a database log, there are some minor differences:

- The grid does not contain columns for record IDs or extra detail because these columns are relevant only to specific records, for which the security log does not record actions.
- The Selection Criteria page has fewer controls. The unavailable controls are those relevant only to specific records.
- The Actions page lists different actions and you cannot change the audit level. The extra actions are those relevant to database and security operations: compacting, converting, and upsizing databases; creating databases and templates; managing users and groups; and failed logon attempts.
- You can only sort by date (in the **Sort Order** list).

Archiving audit logs

You can reduce the size of the audit logs and security logs for a database by archiving some of the records in them.

Archiving logs using Audit Viewer

When you use iBase Audit Viewer you can choose to archive the audit log to a file or to a SQL Server database.

If the **Audit History** option in **Database Properties** is unchecked, the audit log is archived to a file. Open Audit Viewer and connect to the iBase database. Select the log entries you want to archive, and then click **Action > Archive**. If you click **Yes** to the prompt, Audit Viewer creates an archive file with the extension `.idla` in the same folder as the database. The archive file is password protected.

If the **Audit History** option is checked, the audit log is archived to a SQL Server database. Open Audit Viewer and connect to the iBase database. Click **Action > Archive**, and select a SQL Server database to archive the log to. Audit Viewer creates a new archive database in the same instance as the SQL

© N. Harris Computer Corporation

Server database. The archive database has the same name, but with `_log` appended. For example if the database is called `User_Guide`, the archive database is called `User_Guide_log`.

Archiving logs using external tools

If you want to archive logs using tools external to iBase and Audit Viewer, you need to know where the logs are stored. Different locations are used for different types of databases and logs.

For Access databases, the database log is held in the _AuditLog table of the `.idl` file that is stored in the same folder.

For SQL Server databases, the database audit log is created as a separate SQL Server database with a name constructed from the database name with the added text _log, for example `dbname_log`. This database can get large, so you can either:

- Use Audit Viewer to extract sections to store as archives. For more information, see Archiving SQL Server databases.
- Use a Microsoft SQL Server backup tool to archive the whole log, and then delete the audit log database. After you delete the audit log database, open the main database in iBase designer to create a blank audit log database.

Opening an archive in Microsoft Access

For iBase databases in Microsoft Access format, the audit log is held as an archive file, `.idla`, a password-protected database file.

The password for Access databases is the database password, which is listed in **iBase Designer** > **Tools** > **Feature Availability** > **Options** > **Advanced**.

The database password is the same for all databases that are accessed through one security file.

Archiving SQL Server databases

You can save the audit log for an iBase SQL Server database to a new SQL Server database on a different server machine (a linked server), or on the same SQL Server as your iBase database.

About this task

When you archive data, it is no longer accessible from within Audit Viewer. You will need the assistance of a SQL Server administrator to retrieve specific log entries or run reports.

Procedure

To save part of the audit log to a SQL Server database (and then delete those records from the audit log):

1. Open the database audit log.
2. Select **Action > Archive**.
3. From the **Server Name** list, select the machine on which you want to create the new database. This can be your local SQL Server.
4. Enter the name of the new database.
5. Enter the name and password for a user who has permission to create databases on this machine.
6. Enter the cutoff date for the archive. All audit log entries before this date will be deleted from the audit log. You can inspect archived audit logs in Audit Viewer. Log on to Audit Viewer and select **Open SQL Server Archive** from the **File** menu.

© N. Harris Computer Corporation

Archiving Microsoft Access databases

To save data in an iBase Microsoft Access format database, or security file, to an archive file (and then delete those records from the audit log).

Procedure

1. Open the database audit log or the security log.
2. Select the data that you want to archive.

   Important: All the records shown in the grid will be deleted from the database log after archiving.
3. Select **Action** > **Archive**.
4. Click **Yes** to save the displayed records to a new archive file and, when prompted, enter a descriptive name for the database audit log or security log.

To inspect archived audit logs in Audit Viewer in the same way that you can inspect the original database or security logs:

1. Select **File** > **Open Archive File** and browse for the archive.

To find out when the archive was created and the name of database or security log used to create the archive:

1. Open the archive as described above.
2. Click on the **Properties** tab to display the Properties page.

Replicating and synchronizing databases

iBase database replication is the process of automatically distributing copies of iBase data and database objects between SQL Server instances in different locations and keeping this data synchronized. The data is copied by use of SQL Server merge replication, using the standard tools provided in SQL Server. iBase database replication provides more tools to manage the iBase database. All servers that are involved in replication must use the same SQL Server version.

In iBase database replication, one of the iBase database servers is configured as the Publisher, and empty iBase databases are created at the other locations. To start replication, the SQL Server administrator either configures a subscription that downloads a snapshot of the data over the communications link or transfers the data on removable media using a backup file.

Owing to the complexity of SQL Server replication, the users who configure and maintain the underlying SQL Server databases require appropriate SQL Server training. Analysts do not require any additional skills to operate iBase within a replicated environment. Senior analysts with responsibility for operations such as merging, batch editing and deleting, restoring, and purging, and reviewing conflicts require an understanding of the replication environment.

What does iBase database replication contain?

iBase database replication is installed as part of iBase. It contains the following functionality to enable iBase administrators to manage replication:

- Conflict Viewer dialog - for reviewing the data conflicts that might occur when two users change the same record within the same replication cycle. Conflicts are reviewed on a record by record basis at the publisher site.
- File Manager dialog for uploading files into one database for replication to the databases at other sites. Files might be a database template, audit archive files, and so on.

© N. Harris Computer Corporation

- Update Database Schema dialog for applying changes to the database design.
- Status report to show whether replication is configured in SQL Server.

Project management

The following section introduces the different steps that are involved in deploying iBase database replication. It brings together the preparation that is required for the iBase security file, the database, and the audit log with the configuration required in SQL Server.

Users can work in the database that becomes the publication database, even if it is not yet configured for replication. Users at the subscriber sites can only start work after replication is fully configured.

Version 5 security files and databases require upgrading to version 8, and after these files are upgraded, they cannot be used with iBase 5. You must also convert (upsize) the security file to SQL Server format (and the database if it is in Microsoft Access format). You can only replicate SQL Server databases.

The following flow charts show the sequence in which the configuration tasks can be completed, who performs these tasks (whether the iBase or the SQL Server administrator) and what the dependencies are.

For more background information on managing the deployment of iBase database replication, see:

- Overview of Supported SQL Server Replication Features
- Overview of Preparing iBase for Database Replication
- Overview of Setting Up iBase Database Replication in SQL Server
- Overview for Large Databases

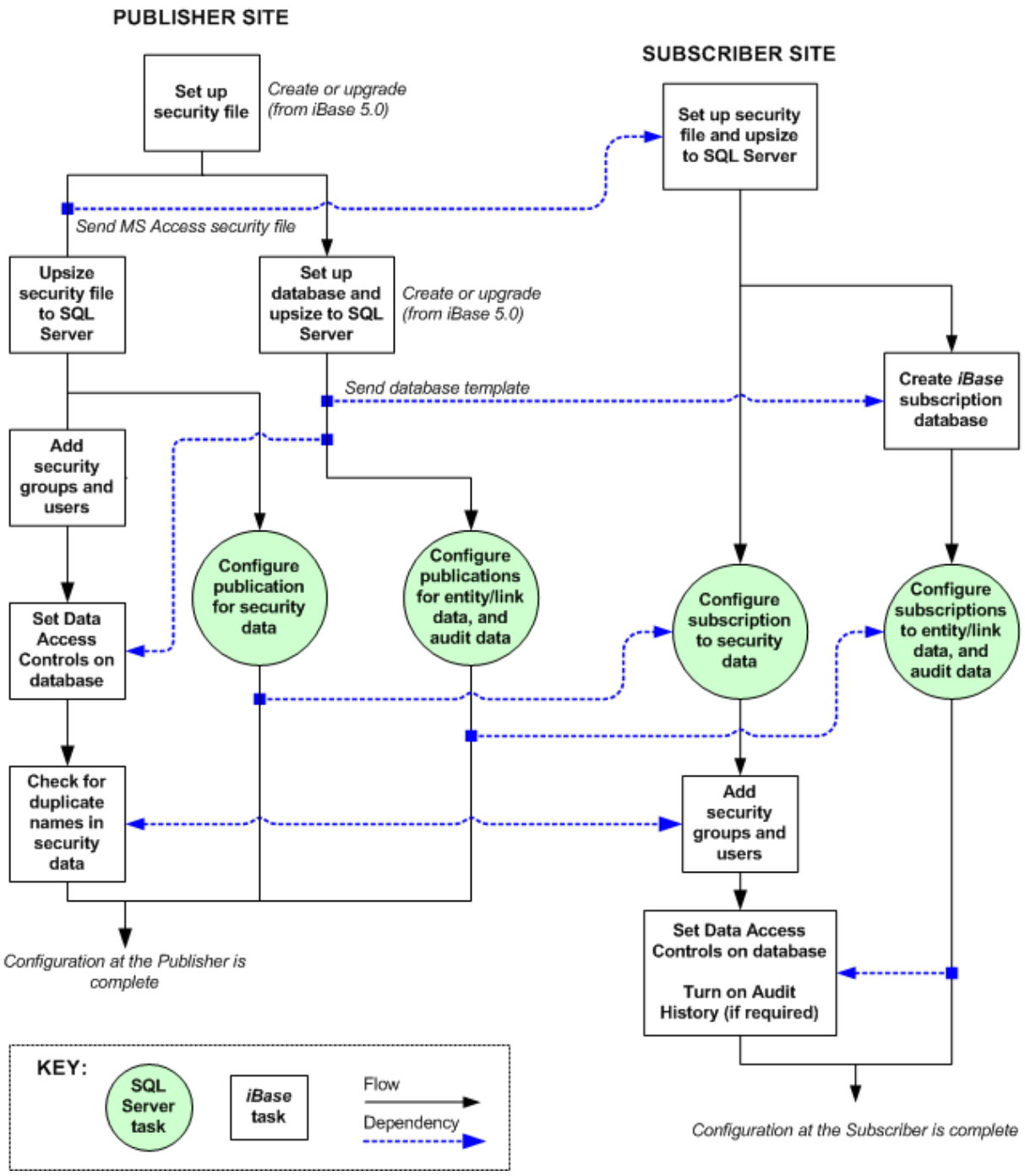
New systems or systems with small amounts of data

The following flow chart shows the sequence of tasks and the dependencies when you are configuring a new system or a system in which the initial snapshot of data, for the iBase database containing the entity and link data, is small enough to transfer over a communications link.

The following flow chart shows, for example, that an SQL Server administrator might configure the publications and subscriptions at the same time if the preparation of the iBase security file and database is complete.

The work flow is identical for all supported versions of SQL Server.

© N. Harris Computer Corporation



Further information

Publisher site: iBase administrators

- Creating New Security Files and a Publication Database
- Preparing Existing Security Files

© N. Harris Computer Corporation

- Preparing Existing Databases
- Updating the Database Design
- Managing Security
- Updating User Data

Publisher site: SQL Server administrators

- Setting the Time on the Servers
- Configuring the Distributor
- Publishing Security Data
- Publishing Entity and Link Data
- Publishing Audit Data

Subscriber site: iBase administrators

- Converting the Security File to SQL Server
- Protecting the Security Connection File
- Creating Subscription Databases
- Managing Security
- Updating User Data

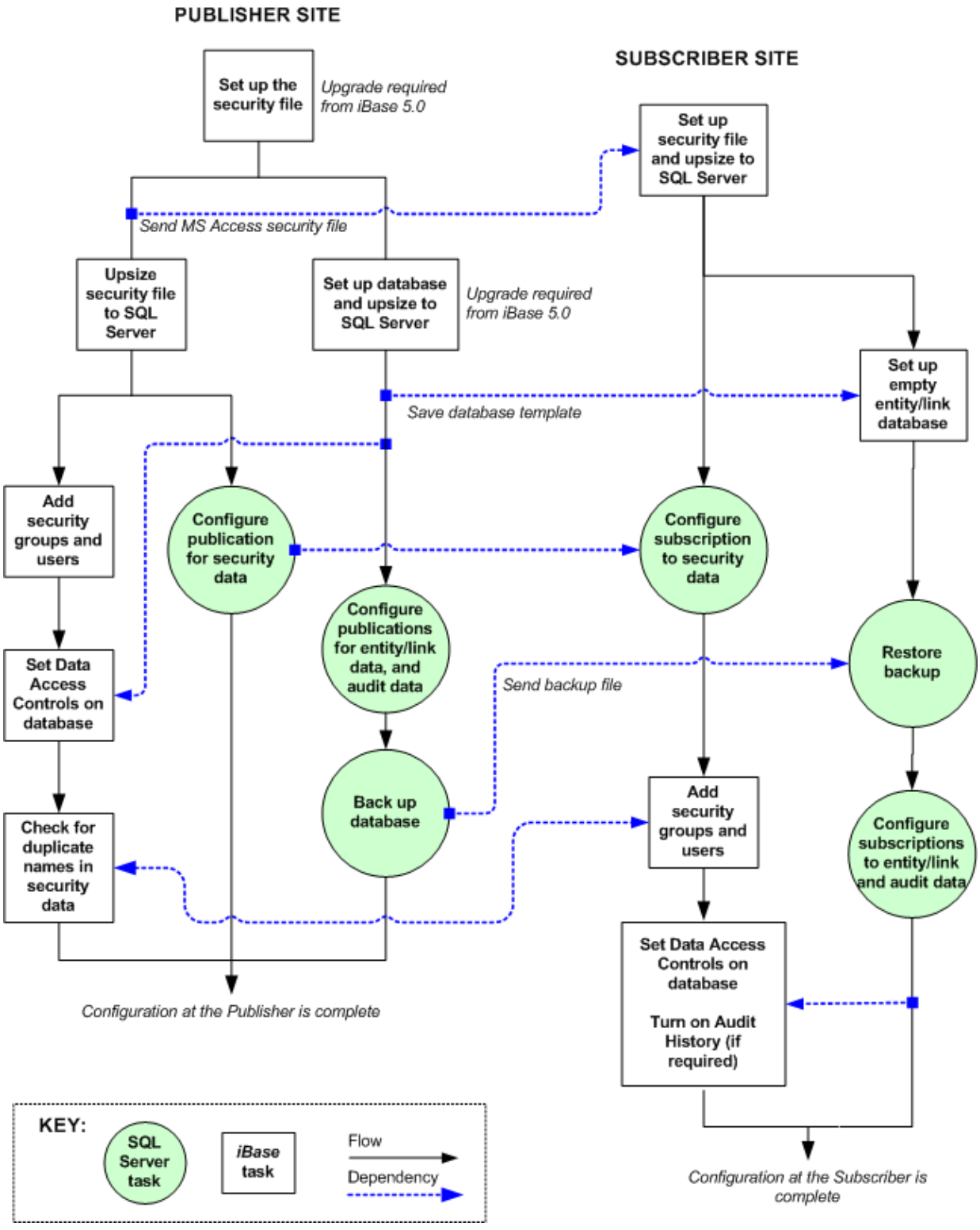
Subscriber sites: SQL Server administrators

- Setting the Time on the Servers
- Subscribing to Security Publications
- Subscribing to Entity and Link Publications
- Subscribing to Audit Publications
- Setting Up the Merge Agents

Systems with large amounts of data

The following flow chart shows the sequence of tasks and the dependencies when you are configuring a system in which the initial snapshot of data, for the database containing the entity and link data, is too large to transfer over a communications link.

© N. Harris Computer Corporation



Further information

© N. Harris Computer Corporation

Publisher site: iBase administrators

- Preparing Existing Security Files
- Preparing Existing Databases
- Updating the Database Design
- Managing Security
- Updating User Data

Publisher site: SQL Server administrators

- Setting the Time on the Servers
- Configuring the Distributor
- Publishing Security Data
- Publishing Entity and Link Data
- Publishing Audit Data

Subscriber site: iBase administrators

- Preparing Existing Security Files
- Creating Subscription Databases
- Managing Security
- Updating User Data
- Testing Replicated Databases

Subscriber sites: SQL Server administrators

- Setting the Time on the Servers
- Subscribing to Security Publications
- Subscriptions for Restored Databases
- Subscribing to Audit Publications
- Setting Up the Merge Agents

For iBase administrators

How to prepare iBase databases for replication, and then how to maintain them after replication is configured in SQL Server.

Note: Unless stated otherwise, the information in the following section applies to all supported versions of SQL Server.

Introducing iBase database replication

The following section introduces the principles of iBase database replication, and describes how working in a replicated database is different from a non-replicated database. It is intended for iBase administrators and senior analysts with some administrative responsibilities rather than SQL Server administrators. It also provides background information on database replication.

For an overview more relevant to SQL Server administrators, see the topics in For SQL Server administrators.

© N. Harris Computer Corporation

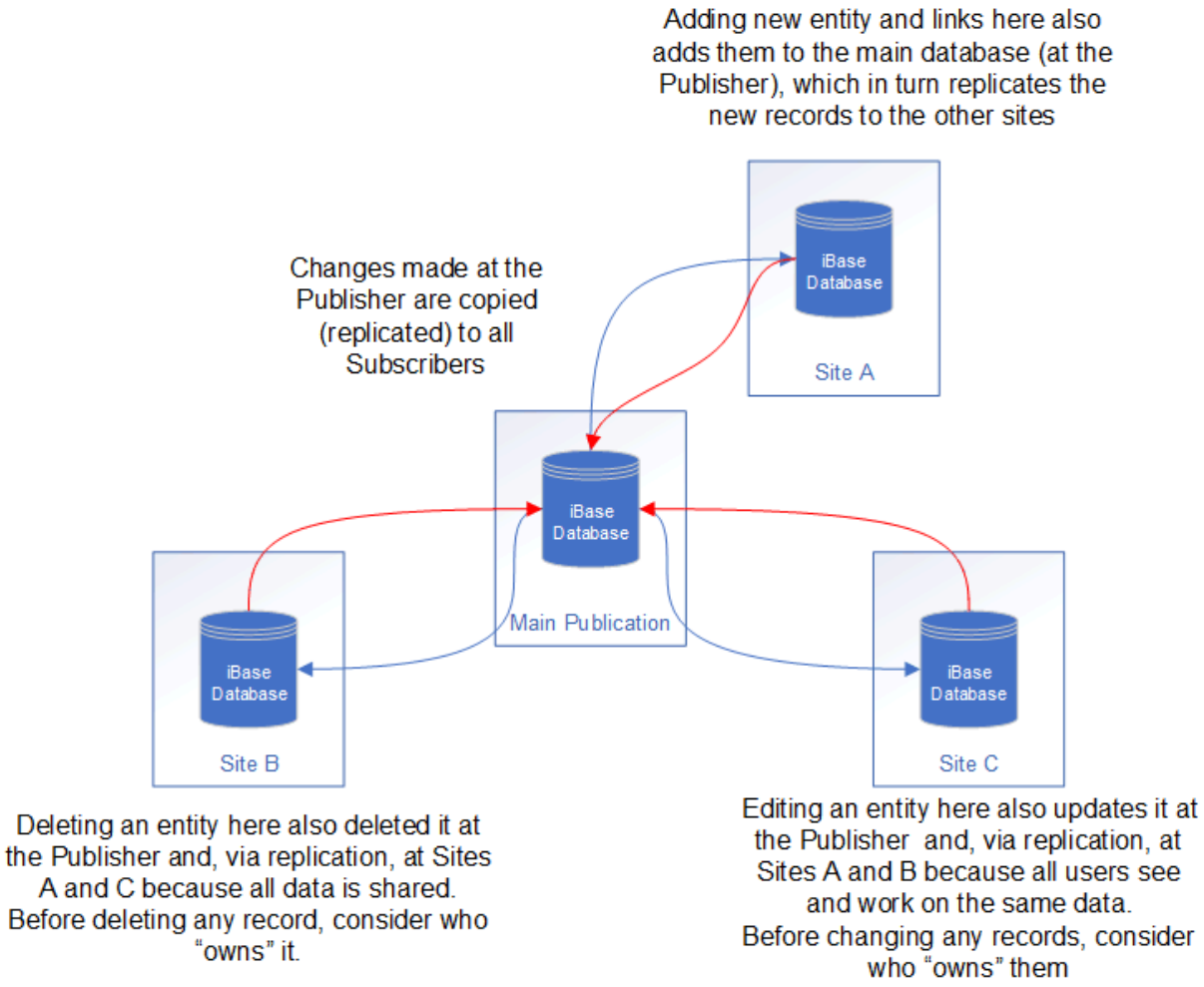
What is database replication?

Replication is the process of copying data from a central database to one or more databases.

The central database is called the publication database because it provides the data for users at other sites. The data in the publication database is copied (replicated) to subscription databases at other locations. All users whether connected to the publisher server (the Publisher) or to a server at one of the remote sites (a Subscriber) see the same data and work on the same records. A Subscriber can be a Subscriber such as a file server or a disconnected Subscriber such as a laptop.

A change that is made in one subscription database is copied to the Publisher (merged), and the Publisher then replicates that change to the other Subscribers. On fixed Subscribers, this is a continuous process and you are not aware of it happening, synchronization usually occurs in a matter of seconds. The effect is that all users on fixed Subscribers have an almost identical view of the data even though they are working on different databases.

This figure shows the main database that is configured as the publication database with three subscription databases on servers that have permanent (fixed) connection to the Publisher:



© N. Harris Computer Corporation

The situation is slightly different if you are using a disconnected Subscriber, such as a laptop, and the Subscriber is only periodically connected to the Publisher. Your changes can only be merged with the Publisher, and you only receive updates to the data, when the laptop is on the network and connected to the Publisher. For this reason, your view of the data can become progressively outdated, and there is also a risk that your changes might conflict with changes made by others.

Note: Many of the terms that are used in iBase database replication are derived from Microsoft™ SQL Server merge replication on which iBase database replication is based.

Editing and deleting records in a replicated database

When you edit or delete a record, you affect the data in use at other sites. You can, for example:

- Edit a record that is "owned" by a user at a different site.
- Delete a link that is associated with an entity that is part of another user's area of investigation.
- Delete an entity that a user at another site is about to edit.

None of this is new but it is important to be aware of who owns a particular record.

You can find out who last worked on a record by looking at the record properties (in iBase, right-click on the record, for example in a record list, and from the menu, select **Properties**).

Record identifier

The record identifier is composed of `<table><record number>\<database identifier>`. Therefore, SUB in this example identifies the database in which the record was created.

Note: The record identifier for records entered using iBase 3 has the database identifier before the `<table><record number>`.

Created by

The name of the user who entered the data. The optional number is part of the username and can be used to show where the user is located. You can also assign users to locations as part of their contact details as explained in the subsequent information.

Updated by

The name of the user who last entered the data.

Note: Because updates are made independently by users at each Subscriber, the same data might be updated at the Publisher or by users at more than one Subscriber. Therefore, conflicts can occur when data modifications are merged. However, this should be a rare event, and the iBase administrator can resolve any conflicts that occur.

For information on who created or last modified the record, or who owns the record, you might be able to display their contact details.

Restricted operations in a replicated database

In a replicated database, particular care needs to be taken with:

- Batch editing
- Batch deleting
- Merging entities
- Imports

© N. Harris Computer Corporation

With the first three types of operation, unexpected consequences can occur if you do not carefully analyze the records involved in the operation. For this reason, these operations are restricted to specific users.

Soft delete is always used in a replicated database. This means that deleting a record, marks it as deleted but leaves it in the database so it can be restored if necessary. Soft deleted records can only be restored and purged in the publication database. Soft deleted records are only visible to system administrators. They do not appear in search results, when listing and browsing records or when charting but they do appear, for example, in the Links dialog.

Soft deleted records cannot be purged or restored if there are any conflicts.

Merging entities is a restricted and time-consuming operation in a replicated database (requiring cooperation from all the different sites). For this reason, steps need to be taken to reduce the number of unnecessary duplicates in your data, including duplicates created when you import. See Merging Entities for details.

Note: Bulk Import is not supported with iBase database replication.

Setting up iBase database replication

When you and your SQL Server administrator set up replication, you replicate the iBase:

- Entity and link data (including the contents of code lists).
- Security data, and the users and groups that it contains.
- Audit log and its entries, so that changes to the audit trail can be analyzed for a single site or for the organization as a whole.

Different types of data are held in different SQL Server databases and require separate publications and subscriptions.

For an iBase database, you do not replicate the database schema. The database at each site must have the complete database schema before replication is configured. For more information on what is and is not replicated, see Updating the Database Design.

When to use iBase replication?

Use iBase replication to copy and distribute data across sites and servers.

Replication is a solution if your organization needs to:

- Copy and distribute data to one or more sites.
- Distribute data changes to other servers.
- Allow multiple users and sites to change and then merge the data modifications together, identifying and resolving conflicts if they occur.

Organizations that can benefit from replication include:

- Organizations with geographically distinct groups of users who need to share common data.
- Organizations that operate separate iBase databases to overcome practical restrictions that are imposed by iBase when large numbers of users simultaneously access the database.

Replication can also be used to store data redundantly so that part of the organization can continue operating if one or more of the servers or communication links fail. To take advantage of this feature, users need to be able to log on to a remote server.

© N. Harris Computer Corporation

When to replicate the security data

Replicating the security data means that all sites have the same user groups and users, and adding a user at one site adds that user to the security files at all sites. This can be administered centrally by one security administrator by remote logon, or locally. For more information, see How Security Works in a Replicated Database.

When to replicate the audit log

If you currently use the audit log, you probably want to replicate it. Replicating the audit log allows both analysts and administrators to extend their analysis of the log entries to include changes that are made at any site. For more information, see How the Audit Log Works in a Replicated Database.

If you do not replicate the audit log:

- Users are only able to analyze log entries made at the local site.
- Users who need to analyze changes at other sites need to be able to log on to remote servers. This increases the workload for the network administrator (who must grant the users permission to access servers in other domains).
- Users at the subscriber sites do not see any entries that show how conflicts between entity or link records were resolved.
- Security administrators might not be able to see how a group or username conflict was resolved if the conflict was detected at a different site. The outcome is recorded in the audit log for the site that detected the conflict. See Handling duplicate group and user names for details of naming conflicts.

Note: Conflict resolution and detection are only logged if the database is set to audit level 4 or 5.

When to use read-only subscription databases

You might want to distribute data and data changes to other servers in your organization but prevent users of that data from changing it. Use iBase database replication to create subscription databases, which are read-only but still receive updates from the Publisher.

How iBase uses replication

Replication is not a feature of iBase itself. To replicate an iBase database, your SQL Server administrator uses functionality within Microsoft SQL Server. However, in iBase you can prepare the databases for replication, and manage the databases after replication is configured in SQL Server.

Note: iBase uses Microsoft SQL Server merge replication in its simplest form. For information on supported configurations, see Overview of Supported SQL Replication Features.

Database configuration

iBase entity and link records, security records, and audit log are held in separate SQL Server databases, and each database is replicated separately. Replicating the audit log is optional. You cannot replicate Microsoft Access databases.

An iBase database that is set up for replication requires the following properties:

- A database identifier that is unique across all the replicated databases. The database and the security file for a site can have the same database identifier (site identifier).
- Soft delete, which is required by the iBase Conflict Viewer. Using the Conflict Viewer, users can review the way in which the conflicts between entities and links are resolved and can undo the change in order to select a different outcome.

© N. Harris Computer Corporation

The databases at the subscriber sites can be read-only. They receive updates from the Publisher but users are not able to make any changes.

Note: The maximum length of database and security file names is 119 characters. For more information, see Before creating any iBase databases.

Publishers and subscribers in an iBase system

One server must be selected as the Publisher, and the iBase database on this server is the publication database - the central database that provides the initial data for the other sites, and receives and sends updates from the databases at these sites.

The database that you select as the publication database:

- Is used to detect and resolve conflicts.
- Is the only database in which records can be restored, purged, and merged.
- Provides the security file that will be used at all the sites involved in replication.

The subscription databases are identical to the publication database in terms of the entity and link data. Filtering of data (for example, by SQL Server) is not supported. However, each database can have its own report definitions, queries, and import specifications.

Subscription databases can be administered centrally if you are able to log on remotely to the servers on which replication is running. Conflict resolution is always done centrally at the Publisher.

Note: Only one site should change the schema of a replicated database in iBase Designer, however the site might be either a publisher or a subscriber site. Regardless of which site the changes are made, there is a procedure that must be followed.

The security file in a replicated database

With a replicated security file, any iBase security administrator can add, delete, or update user groups and users, and the new user data is merged with the security file at the Publisher and then replicated to the security files at the Subscribers. Any duplicate user or group names are detected and a security administrator, working at any site, can correct any duplicates that might occur.

You should replicate the security file. You can only replicate a security file that you upsized to SQL Server.

Attention: If you use iBase with Extended Access Control, then it is essential to replicate the security file and failure to do so undermines the additional security provided by Extended Access Control.

For more information, see How Security Works in a Replicated Database.

Working on entity and link records

Any user given the correct permissions can log on to their local database, whether at the publisher or a subscriber site, and add, modify, or delete records. Changes that are made at fixed Subscribers are quickly merged with the data held in the publication database (if the user is working at one of the subscriber sites) and the changes are quickly replicated to the other sites so that all sites see the same data. In SQL Server, a separate merge process for each Subscriber uploads local changes to the Publisher and downloads all the changes made at or received by the Publisher. For a disconnected Subscriber, such as a laptop, merging, and replication can only occur when the Subscriber is on the network and connected to the Publisher.

For fixed Subscribers, merging and replication should be quick because the merge processes run continuously, and the frequency with which the processes run generally prevent any conflicting changes

© N. Harris Computer Corporation

from occurring. In the unlikely event that a conflict does occur, SQL Server automatically resolves the conflict in favor of the Publisher and then the Subscriber who merged first. There is an iBase Conflict Viewer available in the publication database to review these conflicts and, if required, change how they were resolved.

For fixed Subscribers, conflicts are most likely to occur after a problem with a communication link between the publisher and one of the subscriber sites is rectified (there are no conflicts while the link is down). For disconnected Subscribers, conflicts can occur after connecting with the Publisher and merging their changes into the publication databases.

For more information, see Conflicts and How They Occur.

The audit log in a replicated database

You can choose to replicate the audit log to maintain an audit trail that covers all the replicated databases and security files. You can filter the audit log to view activity at just one site or for the whole organization.

Note: The facility to control who can view the audit log generated by users that work on sensitive data requires you to replicate the security file.

For more information, see How the Audit Log Works in a Replicated Database.

How security works in a replicated database

Replicate the database, and then replicate the appropriate security file.

There are two file formats for iBase security files:

- Microsoft™ Access (which cannot be replicated)
- Microsoft SQL Server (a more secure format that can be replicated)

When you select SQL Server format for the security file, you create an SQL Server database that contains the data, and a connection file that stores the connection information. Keep the security connection file in the iBase database folder alongside the connection files for the iBase databases that use it.

Replicating a security file

Replicating the security data enables your organization to implement a global security system whereby the following are identical:

- Security policy
- User groups
- User accounts
- Access rights derived from membership of the user groups

The advantages of replicating the security data are:

- Reduced administration for the security administrator as replication synchronizes the security data at the different sites.
- Any security administrator at any site can maintain the security data. You control access to the local copies of the security file in the usual way.
- Usernames and contact details are always consistent and up-to-date, which assists analysts who use the audit log, record properties, or owner details in iBase.

© N. Harris Computer Corporation

Non-replicated databases can continue to use a replicated security file, provided the databases are using the same version of iBase.

Why you must replicate the security file

Using a replicated iBase database with an unreplicated security file reduces the security of your system - if you are replicating the database, you must implement a unified and global security system. This is important if users can log on remotely to any server in the iBase system. Local security systems can potentially give users access to different commands and data depending on the server they log on to.

The facility to control who can view the audit log generated by users that work on sensitive data requires you to replicate the security file.

Attention: If you use iBase with Extended Access Control, then it is essential to replicate the security file and failure to do so undermines the additional security provided by Extended Access Control.

Importance of the security file at the publisher site

All sites must have a copy of the security file or database used at the publisher site because this is the security file associated with the database that is replicated. You can only open this database, or the subscription databases created from it, using the user accounts defined in its replicated security file.

It is not possible for the subscriber sites to create their own security file for use with a replicated database. Each site must either start with a copy of the publisher security file and convert this from Microsoft Access format to SQL Server, or start with a duplicate of the publisher security database created by restoring from an SQL Server backup provided by the publisher site.

Attention: You should keep a copy of the security file in Microsoft Access format in case you need to add new subscriber sites in the future.

How the audit log works in a replicated database

If required you can choose to replicate the audit log to maintain an audit trail that covers all sites. Typically, the audit level is set to the same value in all the replicated databases. You must set the audit level to 4 or 5 if you want to include actions that relate to the Conflict Viewer.

The main consideration for the audit log when you set up iBase database replication is the method of identifying the site at which a specific change was made. Depending on your organization, you might obtain this information from:

- User names that contain a site identifier
- Locations that are associated with the user
- Network logins

The origin of a record that is entered after replication was configured in SQL Server (but not of any updates to it) can be determined from its record ID, which always contains a database identifier.

Using the audit log

When you view the records in the audit log, you see the actions taken in each replicated database. To restrict the actions to a specific site, you need to filter the audit log.

There are actions specific to iBase database replication to cover resolving conflicts in the database, and in the security file. These actions are used when the audit level is set to 4 or above.

Archived audit log files are not replicated.

© N. Harris Computer Corporation

Note: The dates and times that are recorded in the audit log are for the server on which the user was working.

Note: If Audit History is turned on, then the audit history will also log changes made by the Conflict Viewer if there is a data conflict.

Filtering the audit log

There are various ways of filtering the audit log to view activity for a specific site.

Filter by user name

In a replicated iBase system, each user must have a unique user name (so there can be no confusion about who worked on which record). This allows you to list the records worked on by a single user regardless of the server they log on to.

Optionally, each user name might contain an element to identify the site at which the user works, and this can be a useful way of filtering the audit log by site.

Filter by network Login

The audit log records the server and network login name of the user who performed the action. This may be a useful way of determining the location of the user if users can only connect to a database on one server.

Filter by location

Locations are optional. A location is associated with a user name and records something extra about the user, such as:

- The site they work at
- The division or business unit they work in
- Their area of responsibility

If you want the capability to filter by location, then all user accounts should have a location and the location names must be consistent.

Choosing not to replicate the audit log

If you choose not to replicate the audit log, then actions are still logged as usual but the audit log only records actions local to the replicated database or security file. Other sites might not be able to view this information or easily compare it with actions recorded in other audit logs.

For more information, see When to replicate the audit log.

Differences between working in a replicated and non-replicated database

Analysts see little difference in how they work in a replicated database. The main differences are for senior analysts who check for and review conflicts, and merge records, and for database designers who use iBase Designer.

Analysts

When a user edits or deletes a record, the updated or deleted record is merged with the publication database, and the Publisher replicates that change to the other subscription databases. The user is not aware of this process.

If another user is working on the same record when the change is replicated to their database, they are prevented from saving their changes to the record. They must click **Cancel** to close the dialog and then

© N. Harris Computer Corporation

re-enter their changes. This is identical behavior to a non-replicated database, for example, if another user makes a change first.

Users see little difference when they work in a replicated database. From their point of view, the main difference is the need to consider who owns a record before they edit or delete it. To make this process easier, individual users should add their own contact details. See Checking the Ownership of Records for further details.

Note: If an analyst makes a change that results in a conflict, their change is overwritten if the change made by the other site merged with the Publisher first. The user going back to their records does not see their change and might want to re-enter it. However, before they continue they should review both the data and the record properties to find who changed it and when.

During data entry, users should always pay attention to warnings that concern duplicate values, and should always investigate the circumstances that produced the warning.

Database designers

Attention: Unless you are a security administrator, do not open a replicated database in iBase Designer while replication is configured in SQL Server.

After a database is being replicated, you are restricted to what you can do in iBase Designer with the publication and subscription databases. In particular, you cannot make any changes that affect the database schema, for example change the entity or link types, add new code lists, or assign semantic types. See What is a Schema Change? for further details.

However, you can do the following (although you might use iBase for most of these tasks):

- Import and export data.
- Update search indexes (these are not replicated).
- Create database templates (these are not replicated unless you specifically upload them using the File Manager).
- Define labeling schemes (these are not replicated and this can be done in iBase).
- Add values to code lists (this can be done in iBase).

To make changes to the database schema, the SQL Server administrator must disable replication at all sites. This requires all fixed and disconnected Subscribers to be connected to the Publisher. Changing the schema of a replicated database is not a trivial task for either the iBase or SQL Server administrators. For more information, see Changing the Schema of a Replicated Database.

Note: Changes made to the database while replication is not configured are not replicated to the other sites when replication is reconfigured in SQL Server.

iBase database administrators

There are a number of differences between replicated and non-replicated databases that might affect day-to-day working, for example how you merge records, and restore and purge soft deleted records. However, the principal administrative task in a replicated database is to check regularly for conflicting changes to the entity and link data.

When there is a conflict:

- All sites see the same data.
- Users are not aware that an entity or link is in conflict, and work with the winning record until the outcome of the conflict is changed in the Conflict Viewer.

© N. Harris Computer Corporation

- Users can modify and delete the "winning" record in the usual way.
- Users in the publication database cannot remove the conflict by restoring a record using the Restore Deleted Records dialog (because this dialog cannot be displayed while any conflicts exist).

If a user at a subscriber site is working on a record while it is in conflict, for example editing it, and the record is changed and replicated as a result of reviewing it in the Conflict Viewer, that user is prevented from saving their work. However, this is identical behavior to a non-replicated database when another user saves their changes first.

There are a number of ways in which you might be able to improve how you use iBase to minimize the risk of conflicts occurring. This is important if you use the `Merge Entities` command as any conflicts that arise from merges cannot be handled by the Conflict Viewer.

iBase security administrators

When the security data is replicated, security administrators must implement a procedure to prevent the creation of duplicate user accounts. They are also responsible for checking that no duplicate groups or user accounts exist. Allowing users to log on with a duplicated user account and password can give one of those users access rights to which they are not entitled.

Replication and other i2 applications

Ensure that you understand the impact that replication has on other i2 Applications.

Replication has no effect on how you use these i2 products with iBase:

- i2® Analyst's Notebook®
- i2® iBase Geographic Information System Interfaces
- Extended Access Control

For Analyst's Notebook, you can use the File Manager on the iBase **Replication** menu to replicate Analyst's Notebook charts and templates.

iBase GIS Interfaces

Using iBase GIS Interfaces to provide an interface in iBase to a mapping application has no effect on how you set up iBase database replication. Each site that uses a mapping application must install iBase GIS Interfaces and set up its own mapping configurations. Mapping configurations are not replicated.

If you choose to hold geocoding data in the iBase records, then all sites see this data, even if they do not use any mapping applications.

iBase Scheduler

It is only necessary to install iBase Scheduler and configure the Scheduler service to run at one of the sites. This can be any one of the publisher or subscriber sites.

Note: The Scheduler service should only run while replication is configured because records imported after replication is disabled by the SQL Server administrator are not replicated to other sites even when replication is reconfigured. You might also want to consider disabling the service if there is an interruption to the communications link. See Handling a Network Communications Failure for details.

Note: Disabling the Scheduler service, rather than stopping it, prevents the service from restarting if the server is rebooted.

© N. Harris Computer Corporation

Semantic types

The management of semantic types is simplified in a replicated database. You no longer need to distribute copies of the semantic type library from a central database using `.mtc` files. Semantic types and how they are assigned are distributed as part of the database schema.

For background information on semantic types in iBase, see the iBase Designer help.

Conflicts and how they occur

A data conflict occurs when a user changes a record that has been changed by a user at a different site, within the same replication cycle.

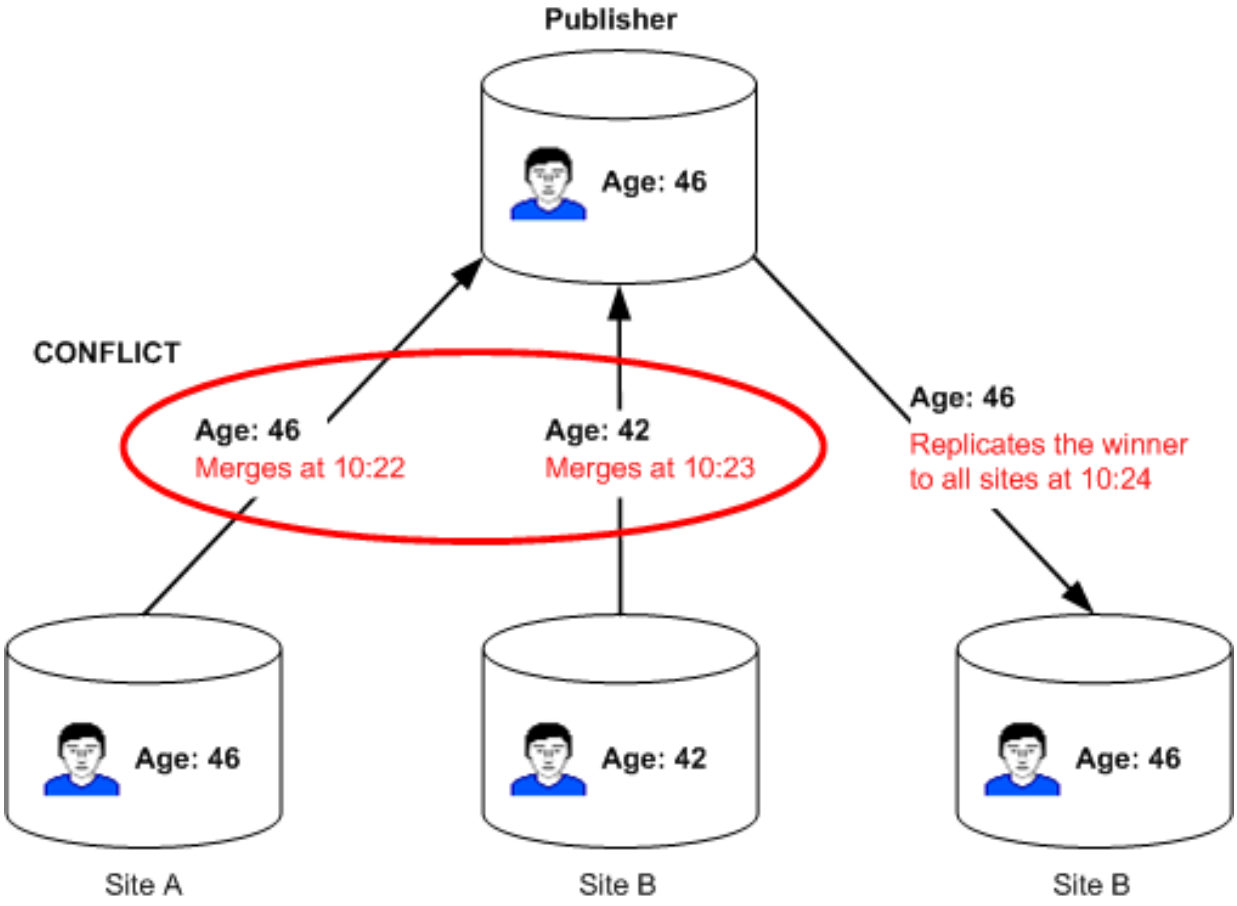
Examples of data conflicts are:

- Data in any field in the same entity is modified by two users, even if in different fields.
- One user modifies a link, for example gives it a direction, but a different user deletes it.
- One user adds a link but a different user deletes a link end entity for that link.

In normal circumstances, it is unlikely that any data conflicts occur in systems with fixed Subscribers only because changes from Subscribers are merged continuously with the Publisher and replicated to the Subscribers almost immediately.

For a conflict to arise, the first update to a record and the second conflicting update must merge with the Publisher before the Publisher can replicate the first update to the Subscribers. In this example, Sites A and B make conflicting changes to the same record:

© N. Harris Computer Corporation



In this example, a conflict occurs because Site B merges its change into the publication database before the Publisher can replicate the change made by Site A. The conflict is automatically resolved in favor of the first change to merge at the Publisher (Site A), and it is this record which is replicated to Site B. The outcome would be the same if Site A was actually the Publisher.

Conflicts are most likely to occur when disconnected Subscribers connect to the Publisher or when a failed communications link comes back online.

All conflicts are resolved automatically by SQL Server which replicates the modification or deletion to all sites. You need to review all the conflicts in the iBase Conflict Viewer. If required, you can change how the conflicts were resolved. Changing the outcome of a conflict updates the publication database and replicate the change to all the subscription databases.

Note: Conflicts must be reviewed before you can purge or restore soft deleted records, and before the SQL Server administrator disables replication. All conflict data is lost when replication is disabled by the SQL Server administrator.

Understanding the Conflict Viewer

Conflicts are reviewed using the iBase Conflict Viewer, in the publication database only.

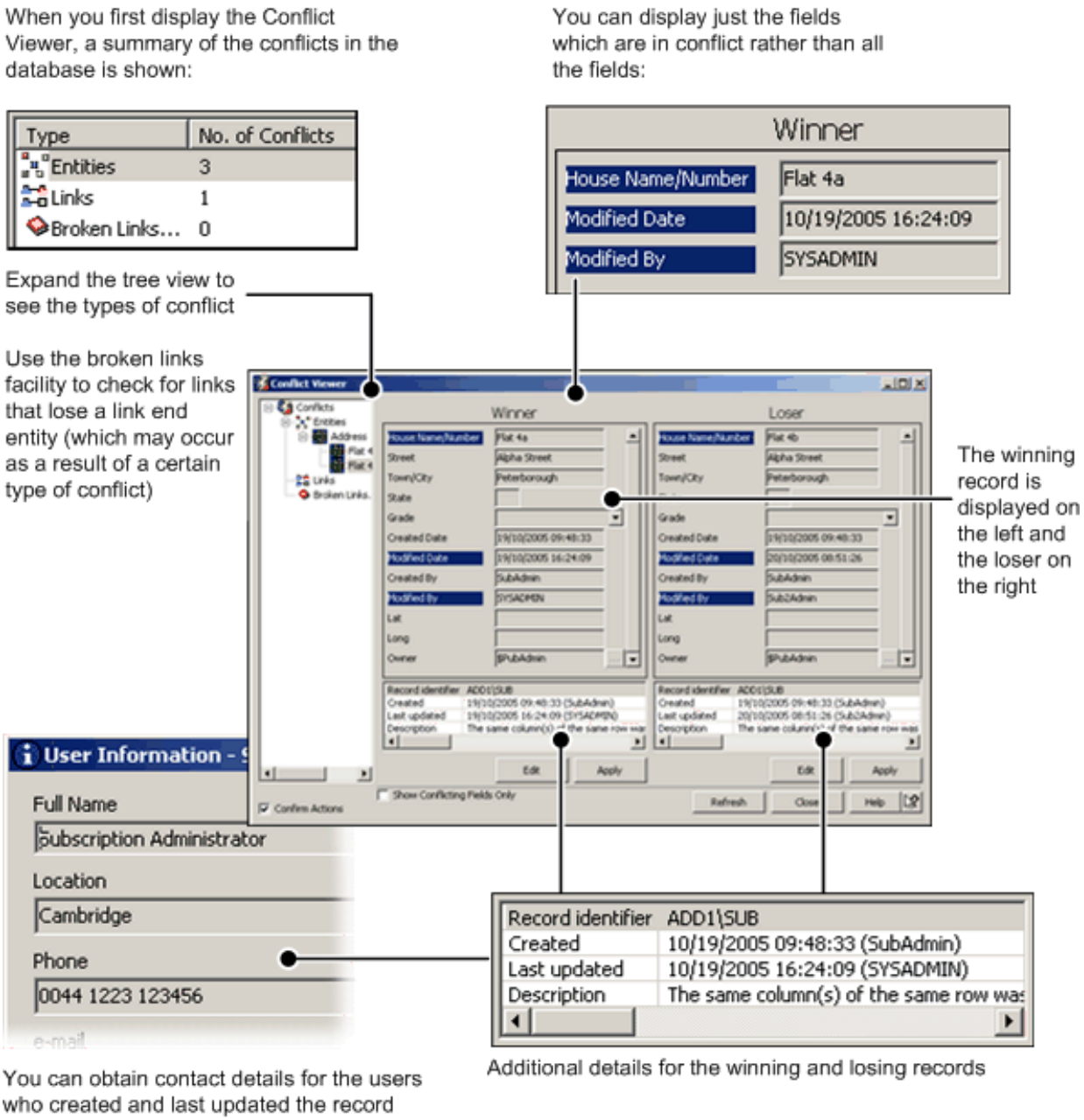
The user who reviews the conflicts must decide whether to keep the "winning" record in the conflict (the record currently seen at all sites) or replace it with the "loser". When they confirm their decision, the

© N. Harris Computer Corporation

record they choose to discard is purged from the database. Before they confirm their decision, they can edit the record they want to keep, for example, to include details from the record that is purged.

If the outcome of the conflict is changed, the records are replicated to all the subscriber sites to update the databases and produce a consistent view of the data across all the sites. If the decision is to keep the original outcome, then nothing is replicated and the databases remain unchanged because they already have the data.

The following figure summarizes the iBase Conflict Viewer. The Subscriber to merge first with the Publisher is presented as the winner, and the Subscriber to merge second is presented as the loser:



© N. Harris Computer Corporation

Note: The records that are displayed might contain several conflicting changes, for example to both the link direction and to a field in the link.

Note: It is important to remember the deletions and modifications that are shown in the Conflict Viewer have already taken place. The winning record is already seen at all fixed Subscribers, and at any disconnected Subscribers that are online and connected to the Publisher since the conflict was resolved. You use the Conflict Viewer to either confirm that the deletion or modification was correct, or to change the outcome.

The different types of conflict are described in detail in the following information:

Unless otherwise stated, the examples given in the following sections apply both to a conflict automatically resolved in SQL Server or to a conflict reviewed in the iBase Conflict Viewer.

Conflicting changes to an entity

In this example, the Subscriber at Site A deletes the entity but the Subscriber at Site B changes it in some way:

- If Site A wins the conflict, the entity is soft deleted at all Subscribers.
- If Site B wins the conflict, the entity is updated at all Subscribers, and users at Site A see a restored entity, in its updated form.

Conflicting changes to a link

In this example, the Site A Subscriber deletes the link while the Site B Subscriber adds a direction to the link:

- If Site A wins, the link is soft deleted at all Subscribers.
- If Site B wins, the link is updated at all Subscribers, and users at Site A see a restored link with the update made by Site B.

Note: There is a similar outcome if the change at the Site B Subscriber is a change to a link field.

Non-conflicting changes to a link

Conflicts on links occur only:

- Between fields in the link record
- Between the link direction, strength, or both
- Between the link end entities

A conflict cannot occur when, for example:

- One site changes a link field and another site changes a link end entity
- One site changes the link direction or strength and another site changes a field in the link record

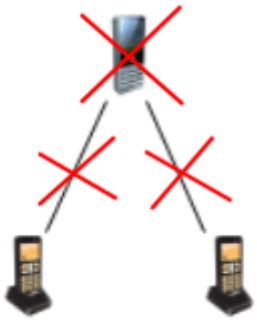
Even though these changes are not conflicts, they are detected by the Conflict Viewer, which highlight just the **Modified Date** and **Modified By** fields as the difference. However, if one site changes any of the fields in the link, then the conflict presents itself as a conflict between the link fields.

See Unsupported Conflicts for changes that involve both link end entities.

Conflicting changes to an entity with links

In this example, the Site A Subscriber deletes the entity, which automatically deletes the two links, and the Site B Subscriber changes fields in the entity but leaves the links untouched:

© N. Harris Computer Corporation

Initial record	Site A change	Site B change
		

How this type of conflict is handled by SQL Server and the Conflict Viewer is different from a conflict that involves just links or just entities.

If Site A wins, the conflict is resolved as you would expect by SQL Server, which replicates the deleted entity and links to the other databases. When you review the conflict in iBase (and click **Apply** below the Winner area to confirm that the change made at Site A is correct), nothing is replicated to the other databases because they already have the correct data. This is the usual behavior.

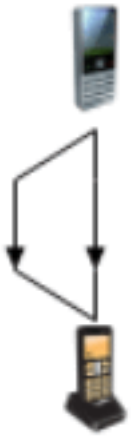
However, if Site B wins, the conflict is not resolved as you would expect in SQL Server because it cannot restore the soft deleted links which is a feature of iBase. Therefore, SQL Server replicates the updated entity as changed at Site B and the soft deleted links as deleted at Site A. As a result all sites see the data in this state until you review the conflict in the Conflict Viewer:

To correct this, use the Conflict Viewer. In this example, it reports that two links are affected by this update. When you click **Apply** below the Winner area to confirm that the change made at Site B is correct, the links are restored and replicated to the other databases.

© N. Harris Computer Corporation

Broken links

In this example, which is not a conflict, the Site A Subscriber deletes the link end entity (and therefore the connecting link) while the Site B Subscriber adds a link:

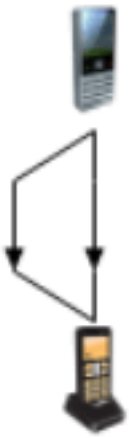
Initial record	Site A change	Site B change
		

If Site B wins, the new link is replicated to all Subscribers, and users at Site A see a restored entity with the new link added by Site B.

If Site A wins, the link end entity and its link are deleted at all Subscribers, however, at Site B, there is a broken link because the new link is left without one of its link end entities:

© N. Harris Computer Corporation

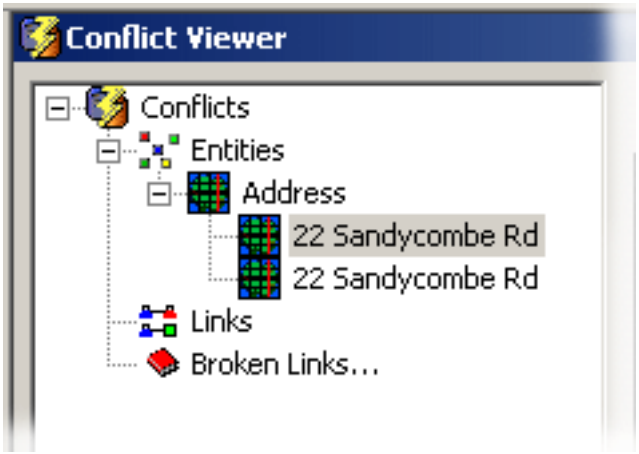
For this reason, after the conflict is resolved, you use an extra facility in the iBase Conflict Viewer to ensure the integrity of all records. You can either delete the broken link or restore the missing link end entity. Therefore, the final outcome will be one of the following:

Deleted link	Restored link end
	

Note: This type of conflict is specific to iBase, and is not visible in the SQL Server Conflict Resolver.

Conflicts between three or more sites

It is possible for three or more sites to make conflicting changes to the same data. The conflicts are listed in the Conflict Viewer in the usual way (by the record label of the winning record). The second conflict that is listed in this example occurs between a third site and the "winner" of the first conflict:



© N. Harris Computer Corporation

Conflicts when records are deleted

Conflicts occur when users at different sites delete the same entity, link, or entity with links. Although the outcome of these conflicts is the same regardless of which site wins the conflict, you might want to investigate why users at different sites are deleting the same records.

Unsupported conflicts

Merge conflicts and a specific type of link end conflict are not supported in the Conflict Viewer.

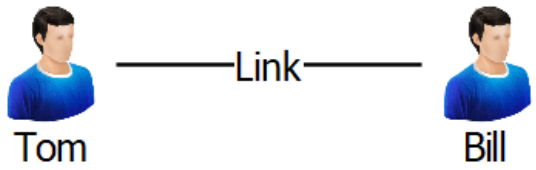
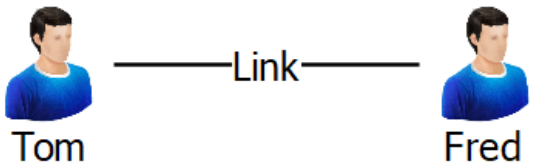
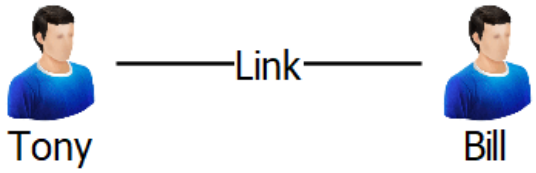
Conflicts resulting from merging

Conflicts arising from merging entities are not supported, and cannot be viewed in the Conflict Viewer. You need to set up a procedure to prevent these conflicts from arising. See Merging Entities for working practices.

Conflicts resulting from changing different or both link ends

A link where different, the same, or both link end entities are changed is detected by the Conflict Viewer as a change to the link, and not as a conflict. In the Conflict Viewer, there is no visible difference between the two records apart from the **Modified Date** and **Modified By** fields.

For example:

Changes made	**Site A - Site B**
This is the original link before the link end entities are changed:	
Site A changes one of the link end entities and is the first to merge with the Publisher.	
Site B changes the other link end entity and merges last. This overwrites the Site A change. You cannot change this outcome in the Conflict Viewer.	

© N. Harris Computer Corporation

Prepare iBase for database replication

Before the SQL Server administrator can set up replication for the first time, you (as the iBase administrator) need to prepare the publication and subscription databases.

The **publication database** is the database containing the data that you want to make available to other sites. This database is on a server that is known as the Publisher.

The **subscription databases** are the databases that receive this data. These databases are on servers that are known as Subscribers.

When you prepare an iBase system for replication, you must to:

- Upgrade both the iBase database and its security file if you are using iBase five files.
- Convert both the database and security file to SQL Server (you can only replicate databases that are in SQL Server format).
- Set up the security connection file at each site (all sites start with a copy of the same security data).
- Set the database properties required by replicated databases.
- Ensure that the database design and the folder objects for the database are up-to-date, and suitable for use at all sites.
- Create a database at each subscriber site. The database design must be the same at all sites.

Although you create the publication and all the subscription databases in iBase, the subscription databases are populated using SQL Server when replication is configured. This ensures that the Subscribers are synchronized with the Publisher.

How you prepare iBase for replication depends on the size of your database, the speed of the communication links between the publisher and subscriber sites, and the version of SQL Server.

All these steps are described in greater detail in separate topics. For an overview of the whole configuration process covering both iBase and SQL Server, see Project Management Overview.

Before you create any iBase databases

Before you create new iBase databases on a server, you might want to give the SQL Server administrator an opportunity to predefine the security for these databases. See Overview of Setting Up iBase database replication in SQL Server for details.

The names that you choose for the security file and database are used to generate the names of the SQL Server databases so you should discuss the naming convention to use with your SQL Server administrator.

SQL Server database names are derived from the names of the `.idb` and `.ids` files created in iBase.

Preparing new iBase databases and security files

For organizations that are starting with a new iBase database, see:

1. Creating New Security Files and a Publication Database
2. Creating Subscription Databases
3. Testing Replicated Security Files
4. Testing Replicated Databases

Note: You might also want to read Updating the Database Design. This explains which parts of the database design are replicated and suggests features that you might want to incorporate into your database.

© N. Harris Computer Corporation

Users can start work in the publication database after preparation is complete; you do not have to wait for replication to be configured. Users in the subscription databases must wait for replication to be configured before they enter data in the security file or database.

Preparing existing security files

If you have an existing security file, follow the steps in Preparing Existing Security Files.

Preparation involves:

1. Adding the administrative user accounts required by the sites involved in replication, to the existing security file.
2. Updating the System Commands Access Control groups with the new command groups for iBase.
3. Distributing copies of the security file to all the sites involved in replication. You need to distribute an SQL Server backup if security data is held in an SQL Server database.
4. At each site, converting the security file to SQL Server format to create a security database on the server machine at each subscriber site. If you are using an SQL Server database already, restore the database from the backup.

After replication is configured by the SQL Server administrator, you must to test the replicated security file; see Testing Replicated Security Files.

Users can start work in the publication database after preparation is complete; you do not have to wait for replication to be configured. Users at the subscriber sites must wait for replication to be configured in SQL Server before they change any of the security data.

Preparing existing iBase databases

For organizations that have an existing database that they want to replicate, where it is possible to populate the subscription databases over a communications link or from an SQL Server backup, see:

1. Preparing Existing Databases
2. Updating the Database Design
3. Creating Subscription Databases
4. Testing Replicated Databases

For a visual overview of how to set up your system, see Project Management Overview.

Users can start work in the publication database after preparation is complete; you do not have to wait for replication to be configured. Users in the subscription databases must wait for replication to be configured before they enter data in the security file or database.

Preparing the audit log

No preparation is required for existing or new audit log databases.

Creating security files and a publication database

The following section describes how to create a new security file and iBase database suitable for replication. The database that you create is used as the publication database. All subscriber sites must have a copy of the security file created at the publisher site, and a database created from a database template, which was also created at the publisher site.

© N. Harris Computer Corporation

Before you begin

The names that you choose for the security file and database are used to generate the names of the SQL Server databases so you might want to agree a convention for file naming with your SQL Server administrator. See Before creating any iBase databases for details.

Procedure

1. Create a new security file:
   a. At the publisher site, create a new database folder to hold the security file and database that you create. The folder must be shared and should be on the database or application server machine.
   b. Using iBase Designer, log on as a security or system administrator and create a new security file in Microsoft Access format. You convert this file to SQL Server (upsize it) later.
   c. In the Security Manager, add user groups and users. As a minimum, add the system, database, and security administrators for the publisher and subscriber sites. At this stage, you can also define Data Access Control groups but you cannot set permissions for them yet.
   d. Back up the security file.

      Attention: You will need the original Microsoft Access security file if you want to extend the iBase system in the future by adding additional subscriber sites.

      See Managing Security for further details.
2. Distribute the security file to the subscriber sites: The security files used at the publisher and at all subscriber sites must be converted from Microsoft Access format to SQL Server (upsized) from the same security file. For background information on why this is necessary, see Importance of the security file at the publisher site.
   a. Distribute the security file, which is in Microsoft Access format, to the subscriber sites using any appropriate method, such as copying to CD, sending by email, or copying over a local or wide area network.
   b. At each subscriber site, create a shared folder and copy the supplied security file to it.

   Attention: Do not make any changes to the security data at the subscriber site. You can rename the file if required but any other changes are lost when replication is configured. For information on file names, see Before creating any iBase databases.
3. Convert the security files to SQL Server, at the publisher site and at all subscriber sites:

   You must convert (upsize) the security file from Microsoft Access format to SQL Server before you can replicate it. This process leaves a security connection file (`.ids` file) in the iBase database folder and create an SQL Server database with the name `<file>_Sec` on the designated server (where `<file>` is the name of the `ids` file).

   After you upsize the security file, you need to assign a site identifier to the security database.

   a. Using iBase Designer, log on as a security or system administrator using the Microsoft Access security file distributed by the publisher site.
   b. Click **Cancel** at the prompt to create or open a database.
   c. Select **Tools** > **Upsize Security File to SQL Server**.
   d. Click **OK** when you are informed that a backup is made. This is a backup of the original Microsoft Access security file and has the file extension `.ids.bak` (appended with a number, such as `.ids.bak1`, if there is already a file with this extension in the folder).
   e. Enter the server name.

© N. Harris Computer Corporation

Note: Do not select the **Local** option from the **Server** list.

f. Select the security mode. This is Windows Authentication unless your SQL Server administrator directs otherwise.

g. In **Identifier**, enter a site identifier, up to 5 characters long. The identifiers used for the security connection file and the database at a site are generally the same but should otherwise be unique within the replicated system. For example, you might use the site identifier PUB for both the security file at the publisher site and the publication database.

h. Click **OK** to validate the settings and perform the upsize, then click **OK** when the upsize is complete. If you want to review the connection details and ID of the security connection file, select `File Security File Properties`. The path of the security connection file will also be displayed in the status area with (SSE) after the file name to indicate that it is SQL Server format.

i. Repeat these steps for each site involved in replication.

j. Back up the connection file at each site (`.ids` file). If you lose the connection file, you are not able to log on.

Attention: Do not make any changes to the upsized security file at the subscriber site. You can rename the file if required but any other changes are lost when replication is configured. For information on file names, see Before creating any iBase databases.

4. Protect the security connection file

In Windows, protect the SQL Server security connection file by making it read-only or by setting appropriate security permissions. This allows any user in iBase Designer to view the properties of the connection file but prevents anyone, including iBase security administrators, from changing the SQL Server connection details.

You should also ensure that the security connection file is included in any backup schedules for the database folder.

5. Create a new iBase database at the publisher site:

a. Create a database. This database is configured by the SQL Server administrator as the publication database. The database should have the following mandatory properties:

Field	**Description**
Database type	Select SQL Server. You cannot replicate MS Access databases.
Server	Enter the name of the server for the publisher site. (**Note:** Do not select the Local option from the Server list.)
Use Windows Authentication	Turn on the **Use Windows Authentication** checkbox, unless your SQL Server administrator asks you to use SQL Server authentication.
Database identifier	On the Advanced page, enter a unique site identifier, up to five characters in length. This is usually the same identifier as the security connection file used at the subscriber site. For

© N. Harris Computer Corporation

Field	Description
	example, the site identifier for a publication database and its security connection file might be PUB.
Soft Delete	On the Advanced page, turn on the **Soft Delete** checkbox. (**Attention:** You must use Soft Delete in a replicated database. Users are not able to open the database if this is turned off.)

For information on designing databases for replication, see Updating the Database Design.

Warning: It can be a time-consuming process to modify the database schema after replication is started so it is important to take the time now to review, test, and refine your design. You are not able to change the schema after replication is configured in SQL Server without first disabling replication.

b. Initialize the database for replication, in iBase Designer, select **Tools** > **Feature Availability** > **Initialize Database for Replication**and then click **Yes**.

Warning: Databases that are created from this template are not initialized for replication.

c. Create a database template from the initialized database, and send this template to the subscriber sites.

d. Back up the connection file (`.idb` file) to your publication database. If you lose the connection file, you are not able to open the publication database.

What to do next

Security connection file

Tell your SQL Server administrator whether you need to replicate the security audit log.

After the security connection file exists:

- **At the publisher site**, request the SQL Server administrator to configure replication for the security database. You can add user groups of any type, add users, and assign them to groups, and define permissions for SCAC groups either before or after replication is configured (but you should not change the security files at the subscriber sites).
- **At each subscriber site**, request the SQL Server administrator to configure replication for the security database. Do not change the security data at the subscriber sites until you are notified that replication is enabled for these.

After replication is fully configured, test the replicated security file at each site.

Publication database

Ask your SQL Server administrator to configure replication for this database and, optionally, for the audit log database.

Subscription databases

Create an iBase database at each subscriber site from the new database template. See Creating Subscription Databases for details.

Secure the publication database

© N. Harris Computer Corporation

Ask your Windows administrator to restrict access to the iBase database folder and to any other folders used by iBase, for example folders used when importing, exporting, and reporting. It is particularly important to restrict access to the database connection file.

Preparing existing security files

The publisher and all subscriber sites must start with the same security file or database.

Before you begin

Do not make any changes to the security data at the subscriber sites until replication is configured, for example do not add any groups or users - any changes that you make are lost when replication is configured in SQL Server.

You need to log on as a security administrator or as a system administrator to perform the following steps.

About this task

Using the Security Manager, verify that the user details are correct before you distribute the Microsoft Access security file to the other sites. Although, the security file does not need to contain the names of all the iBase users, it should as a minimum contain the system, database and security administrator accounts for the publisher and subscriber sites. See Managing Security for details.

Attention: Back up the security file if it is in Microsoft Access format. You will need the original Microsoft Access security file if you want to extend the iBase system in the future by adding additional subscriber sites.

Procedure

1. Distribute your security database to the subscriber sites.

   For SQL Server:

   - If your security file is already in SQL Server format, you need to send the following files as used at the publisher site to the subscriber sites, along with a user name and password:
     - A full SQL Server backup of the SQL Server security database
     - A copy of the iBase security connection file
     - The user name and password of a security administrator
     - The iBase database password (this is required if the security administrator at the subscriber site works on a machine that cannot connect to the publication database). This is obtained from the Options dialog in iBase Designer.
   - At each subscriber site:

     a. Ask your SQL Server administrator to restore the security database from the supplied backup. This should be done on the server that acts as the Subscriber. The name of the restored database is identical to that of the security database at the Publisher.

        Warning: If the SQL Server database component is to have a different name, create a new security file with the required name, upsize it and then restore the supplied SQL Server backup over the SQL Server component for the new security database.

     b. Redirect the connection file to the restored database on the Subscriber:

        1. Start the iBase Database Configuration utility from the Windows **Start** menu.

© N. Harris Computer Corporation

2. In the Database Configuration utility, browse to the security connection file.
3. Click **Next** and, if prompted, enter the database password.
4. Change the connection properties to direct the security connection file to the security database on the Subscriber.

c. Test and save the connection settings.

For Microsoft Access:

- The security files used at the subscriber sites must be upsized from a copy of the security file used at the publisher site. They are in Microsoft Access format. For background information, see Importance of the Security File at the Publisher Site.

  a. Distribute the upgraded security file, in Microsoft Access format, to the subscriber sites using any appropriate method, such as CD or email.

  b. At each subscriber site, create a shared folder and copy the supplied security file to it. The folder must be on an application or database server.

  Attention: Do not make any changes to the security files at the subscriber sites. You can rename the files if required but any other changes are lost when replication is configured. For information on file names, see Before creating any iBase databases.

- You need to convert (upsize) the security file from Microsoft Access to SQL Server format before you can replicate it. Upsizing creates an SQL Server database on the designated server, leaving the .ids file in the database folder as a connection file for the SQL Server database.

  Before upsizing, review the name of the security file because the logical name of the SQL Server database is derived from it. For information on file names, see Before creating any iBase databases.

  At the publisher and all subscriber sites:

  a. In iBase Designer, log on as security administrator or as a system administrator, using the Microsoft Access security file.

  b. Click **Cancel** when you are prompted to open or create a database.

  c. From the **Tools** menu, select **Upsize Security File to SQL Server**.

  d. You are informed that a backup is made of the Microsoft Access security file. This has the file extension `.ids.bak` (appended with a number, such as `.ids.bak1`, if there is already a file with this extension in the folder). Click **OK** to continue. The Upsize Security File dialog is displayed.

  e. Enter the name of the server for the site.

  Note: Do not select the **Local** option from the **Server** list.

  f. Select the security mode. This is Windows Authentication unless your SQL Server administrator directs otherwise.

  g. In **Identifier**, enter a site identifier, up to 5 characters long. The identifiers for the security connection file and the database at a site are generally the same but otherwise should be unique within the replicated system. For example, you might use the site identifier PUB for the security file at the publisher site and the publication database.

  h. Click **OK** to validate the settings and perform the upsize, and then click **OK** when it completes.

  The path of the security connection file will be displayed in the status area with (SSE) after the file name to indicate that it is now in SQL Server format.

  i. Repeat these steps for each site involved in replication.

© N. Harris Computer Corporation

j. Back up the connection files (`.ids` files). If you lose a connection file, you are not able to log on at that site.

Attention: Do not make any changes to the security data at the subscriber sites. You can rename the files if required but any other changes are lost when replication is configured. For information on file names, see Before creating any iBase databases.

2. Protect the security connection file.

In Windows, protect the SQL Server security connection file by making it read-only or by setting appropriate security permissions. This allows any user in iBase Designer to view the properties of the connection file but prevents anyone, including iBase security administrators, from changing the SQL Server connection details.

You should also ensure that the security connection file is included in any backup schedules for the database folder.

3. Assign an identifier to a security connection file:

If you upsized the security file without entering an identifier, you can enter one now.

a. Make sure that you have write access to the security connection file.

b. In iBase Designer, select **File** > **Security File Properties** and enter the site identifier in the **Identifier** field.

What to do next

Security connection file

Tell your SQL Server administrator whether you need to replicate the security audit log.

After the security connection file exists:

- **At the publisher site**, request the SQL Server administrator to configure a publication for the security file. You can add user groups of any type, add users, and assign them to groups, and define permissions for SCAC groups either before or after replication is configured (but you should not change the security data at the subscriber sites).
- **At each subscriber site**, request the SQL Server administrator to configure the subscriptions. Do not change the security data at the subscriber sites until you are notified that replication is configured for these.

After replication is fully configured, test the replicated security file at each site.

Database preparation

To continue, review the design of the database as discussed in Updating the Database Design. If no updates are required, see Preparing Existing Databases.

Preparing existing databases

You must convert (upsize) iBase databases in Microsoft Access format to SQL Server before you can replicate them.

About this task

Upsizing creates an SQL Server database (leaving the `.idb` file in the database folder as a connection file for the SQL Server database). The connection file stores the details that are needed to connect to the server that holds the database. Before upsizing, review the database name because the logical

© N. Harris Computer Corporation

name of the SQL Server database is derived from it. For information on file names, see Before creating any iBase databases.

You must set the database properties that are required for replication and then initialize the database. This database becomes the publication database. After you initialize a database, you made changes to the database tables that cannot be undone, and soft delete must always be used.

To perform these steps, you must log on as a database or system administrator.

Procedure

1. Upsize the database to SQL Server:
   a. In iBase Designer, log on using the correct security connection file. You might need to upgrade the security file first. Do not open the database.
   b. Select **Tools** > **Upsize Database to SQL Server**
   c. Follow the instructions displayed by the Database Upsize Wizard and, when prompted, enter the named server for the publisher site and the correct security mode. Your SQL Server administrator tells you which server and security mode to use.

      Note: Do not select the **Local** option from the **Server** list.
   d. Click **Finish** to perform the upsize, and then **Close** once it is complete.

      Note: Back up the connection file (`.idb` file) to the database. If you lose the connection file, you are not able to open the publication database.
2. Set the database properties required for replication, open the database in iBase Designer, then select **File** > **Database Properties** > **Advanced**.
   - **Database identifier**

     Each database in the replicated iBase system must have a unique site identifier, up to 5 characters in length, which is used by the Conflict Viewer and the audit log. The identifier is typically the same as the associated SQL Server security connection file for the site. For example, the site identifier for both the main publication database and its security file might be PUB.
   - **Soft delete**

     Soft delete must be turned on as this is required by the Conflict Viewer. You must not turn off soft delete after the database is initialized for replication as this prevents users from opening the database (even if replication is not currently configured in SQL Server).
   - **Audit level**

     Set the audit level to 4 or 5 if you want to log the results of conflict resolution. Although the audit level property is not replicated, it should be set the same at all sites involved in replication.
3. Initialize the database for replication:
   a. In iBase Designer, log on as a database or system administrator using the correct security connection file and open the database.
   b. Select **Tools** > **Feature Availability** > **Initialize Database for Replication**, and then click **Yes**.

      This operation can take a long time on large databases.

      You do not need to do this step at the subscriber sites. They are initialized for replication when the SQL Server administrator creates a subscription to the publication.
4. Create a database template.

© N. Harris Computer Corporation

It is important to ensure that the design of the iBase database that you want to replicate is up-to-date and suitable for use by all sites before you continue. For details, see Updating the Database Design.

If the database design is already up to date, then create a database template for use at the subscriber sites. The databases at all sites must use the same database design.

5.

What to do next

Step	Details
Publication database	Request that your SQL Server administrator configures a publication for this database and, optionally, for the audit log database. The users of the publication database can continue to work in the database while replication is configured.
Subscription databases	Send the new database template to the subscriber sites. The subscription databases can then be created from this template. See Creating Subscription Databases.
Secure the databases	Ask your Windows administrator to restrict access to the iBase database folders and to any other folders used by iBase, for example to the folders used when importing, exporting, and reporting. It is important to restrict access to the database connection files.

Updating the database design

It is important to ensure that the design of the iBase database that you want to replicate is up-to-date and suitable for use by all sites. When the database template is created, it is distributed to all sites and the SQL Server administrator configures the publisher site for replication.

About this task

The database design comprises more than its entity and link types, it also includes items that affect its usability, such as datasheets, and supporting files. The items to consider fall into these categories:

- Database schema, such as link types and code lists
- Folder objects, such as queries and report definitions; and non-folder objects, such as datasheets and labeling schemes
- Other files, which are database-specific but not part of the database, such as supporting files for report definitions

Print a database design report to obtain a list of items to review in your database.

Note: the report does not include items such as folder objects, or datasheets and charting schemes.

After replication is configured, iBase provides tools to update the database design at all sites. Nevertheless, it can take time to make these changes, mainly because of the cooperation that is

© N. Harris Computer Corporation

required between sites. You must also require the cooperation of the SQL Server administrators at each site as they must delete and then re-create the publication and subscriptions.

Items that must exist before the publisher site is configured for replication

The database schema must be set up correctly before replication is configured for the publisher site. For database replication, the items in the database schema are:

- Entity types
- Link types
- Fields
- Standard fields
- Pick lists, icon lists, and SCC lists (but not their contents, which can be changed by any authorized user at any time)
- Chart attributes
- Datasheets
- Common folder objects (folder objects that are marked as common folder objects in iBase but not dependent on any data)
- Semantic types and the way they are assigned

You might want to make certain changes to the database schema to support database replication. Specifically, adding a Merge Request entity to help manage the merging of records, and adding owner fields to reduce the risk that conflicts might occur.

The following objects are not replicated, and need to be added to the database template before you set up new databases at the subscriber sites:

- Labeling schemes

Public and private folder object control groups are not replicated but the objects they contain are copied for all users. (Folder object control groups are defined in the security file.)

Note: You might want to delete unwanted objects at this stage otherwise they are copied to the databases at the subscriber sites.

Items that can be added at any time

The items in pick lists, icon lists, and SCC lists can be updated at any time as the contents of these lists are replicated.

The following non-database files can also be replicated (by loading them in the File Manager) and therefore do not need to be considered when you prepare the database for replication. The exact list of files depend on how you use iBase:

- Command groups (`CommandGroups.mdb`)
- Word documents that are used for styles or as templates in reporting
- Lists of excluded words for search indexes (`WSexclude.txt`)
- Icon lists (`Iconlist.txt`) and, if customized, the associated graphic files (`.ico`)
- i2 Analyst's Notebook templates
- Mapping configuration files

If there are many files, for example Word documents, then consider archiving them in a `.zip` file before you load them in the File Manager.

© N. Harris Computer Corporation

Note: After replication, you will need to copy these files to the correct place in the database folder or to the correct folder in the iBase installation.

Items that are site-specific

The Word Search and Full-Text Search indexes must be built individually at each subscriber site.

Possible schema changes for replication

When you review the current database schema, you might want to consider making the following two changes to support iBase database replication.

Adding a Merge Request entity type

The Merge Request entity type is intended to support the procedure for merging entities. A procedure is required because the iBase Conflict Viewer does not handle conflicts that arise from merging entities or occurring in the records that are affected by a merge operation.

When you design a suitable entity type for Merge Requests, you might want to add fields for:

- The name and location of the analyst
- The reason for the merge
- Details of the records for the proposed merge, for example by listing the record IDs.
- A screen capture of the pre-merge data
- Room for an analyst from each site to enter comments
- Voting buttons, one for each site (you might use a checkbox or a list with **Yes**/**No**/**Discuss** options)

For more information, see Merging Entities.

Adding owner hyperlink fields to entity and link types

To reduce the likelihood that conflicts occur, you can define a hyperlink field that automatically inserts the name of the current user whenever a new record is saved. Users can see this information when you show or edit the record. For more information, see Checking the Ownership of Records.

To define this type of field, use iBase Designer to create new fields or a new standard field with:

- The hyperlink field type
- A name such as Owner
- A default value of $ (the $ symbol automatically inserts the name of the current user when the record is saved)

You must also update your datasheets to show the owner field.

Process for changing the design of a live database

To make changes to the design of a live database that is not yet configured for replication in SQL Server, work on an isolated copy of the iBase database.

After you finalize the design, you can apply the changes to the live database using the Update Database Schema dialog (you can use this dialog even though replication is not yet configured) or you can make them by hand.

The steps are:

1. Create a template from your live database.
2. Create a temporary database from the template, in either Microsoft Access or SQL Server format.

© N. Harris Computer Corporation

3. Make the required changes to the design of your temporary database, considering all the issues described in this topic.
4. Test the new design in iBase.
5. Create a template to capture the new design. This template is needed by the subscriber sites.

You can then repeat these changes in the live database or use the Update Database Schema dialog to apply some of the changes for you. Changes to folder objects and datasheets must always be made manually.

Procedure

Apply the updated design to a live database:

1. Create a template from the temporary database that contains the updated database design.
2. Create another copy of the live database for test purposes. The database must be:
   - SQL Server format, with the database properties listed in To create a subscription database.
   - Initialized for replication. For details, see Initializing the database for replication.
3. Apply the updated database schema to the test database:
   a. In iBase, log on as a database administrator and open the database.
   b. Load the database template into the database using the File Manager: select **Tools** > **Replication File Manager**. See Replicating non-database files for details.
   c. To prevent users from opening the database (and to inform users that the database is closing), select **Tools** > **Replication Take Offline**, and then click **OK**. See Shutting Down the Databases for details.
   d. Select **Tools** > **Replication Update Database Schema**.
   e. Click **Update** to apply the changes shown in the Update Database Schema dialog and then, after this is finished, click **OK** to reopen the database.
   f. Select **Tools** > **Replication Bring Online**.
4. Check that the template was applied successfully.
5. If the template was applied successfully, apply the changes to the live database.

What to do next

Creating a database template

For details of how to create and test a database template, see Updating the database design. The databases at all sites must have the same database design. After you finalize the design of the iBase database, and completed the preparation of the database:

Step	Details
Publication database	Request that your SQL Server administrator configures a publication for this database. The users of the publication database can continue to work in the database while replication is configured.
Subscription databases	Send the new database template to the subscriber sites. The subscription databases can then

© N. Harris Computer Corporation

Step	Details
	be created from this template. See Creating Subscription Databases. Users of subscription databases cannot work in the database until replication is configured.

Creating subscription databases

At each subscriber site, you need to create an iBase database from the correct database template as supplied by the publisher site.

About this task

The procedure described here is suitable for:

- Any database that is populated with data from the publication database by downloading the data over a communications link
- Large databases that are populated with data from a backup

Note: If you are using Windows Terminal Services, you need to log on to the application server machine for the site where iBase is installed and create the subscription database on the database or application server for the site.

To create a security connection file

All sites must start with a copy of the same security data as the publisher site. This is distributed either as a Microsoft Access security file that is upsized to SQL Server or as an SQL Server backup. If you do not have this yet, see:

- Creating New Security Files and a Publication Database
- Preparing Existing Security Files

Procedure

At the subscriber site:

1. Ensure that the database template supplied by the publisher site is copied to the `Templates` folder for your iBase installation.
2. In iBase Designer, log on as a user with the database creator role, using the correct security connection file for the site.
3. Create a database with the properties described in the following steps.
4. Enter an appropriate database name for the subscriber site. This name becomes the logical name of the SQL Server database so you might want to agree a convention for file naming with your SQL Server administrator. For more information, see Before creating any iBase databases.
5. Select the following for the new database:

Option	Description
Database type	Select SQL Server. You cannot replicate Microsoft Access databases.

© N. Harris Computer Corporation

Option	Description
Server	Enter the name of the server for the subscriber site. Do not select the **Local** option from the **Server** list.
Use Windows Authentication	Turn on the **Use Windows Authentication** checkbox, unless your SQL Server administrator asks to use SQL Server authentication.
Audit Level	On the Details page, select the same audit level as set for the publication database. For more information, see How the Audit Log Works in a Replicated Database.
Template	On the Template page, select the template that contains the database schema. This must be identical to the schema of the publication database. See Updating the Database Design.
Database identifier	On the Advanced page, enter a unique site identifier, up to five characters in length. This is usually the same identifier as the security connection file used at the subscriber site. For example, the site identifier for a subscription database and its security connection file might be SUB.
Soft Delete	On the Advanced page, turn on the **Soft Delete** checkbox. (**Attention:** You must use Soft Delete in a replicated database. Users are not able to open the database if this is turned off.)
Read Only	On the Advanced page, turn on the **Read Only** checkbox if the subscription database is to be read-only. The database still receive updates from the Publisher in the usual way.

6. Click **OK** to create the new database. The new database is opened if it is created successfully.
7. Verify that the database contains the correct entity and link types, semantic types, code lists.

   Note:

   - If the database schema seems incorrect, contact the iBase administrator responsible for the publication database. The corrections must be made in the publication database. The schema for the publication and subscription databases must be identical.
   - If changes are made to the schema, you must delete the connection file and the SQL Server database and start again.
8. Do not turn on Audit History at this stage. You do this later after the SQL Server administrator finishes configuring replication.

© N. Harris Computer Corporation

9. Close the database.
10. Repeat this procedure for each subscriber site.

What to do next

The preparation for the iBase database at the subscriber sites is now complete:

- Notify the SQL Server administrator that the subscription databases are ready for replication to be configured. You must tell the administrator if any of these databases are read-only. Notify the SQL Server administrator that the audit log database can also be configured at this stage.
- After you are notified that replication is configured, test that replication is working correctly. See Testing Replicated Databases.

Testing replicated security files

Test the replicated security files to ensure that the properties are correct, and replication is working correctly.

Before you begin

About this task

After the SQL Server administrator configures replication for the security data at all sites, you need to check that:

- The database properties are correct
- Replication is working correctly
- The audit log for the security data is working correctly (optional)

Procedure

1. Check that the security connection file has a site identifier:
   a. In iBase Designer, log on as a security or system administrator.
   b. Select **File** > **Security File Properties**, and enter a unique site identifier if required. To do this, you need write access to the security connection file.

   Note: You should protect the security connection file by making it a read-only file or by setting appropriate security permissions. See Protecting the security connection file for details.
2. Test that replication is working:

   This operation requires a security or system administrator at each site if you are unable to log on to the remote servers.

   When you use the Security Manager for this test, you can refresh the groups and users displayed in the Security Manager by closing and reopening it. You should also consider the frequency with which updates are replicated between sites. Depending on your organization, the frequency might, for example, be anything from every minute to once a day.

   a. At the publisher site, add a user and then close the Security Manager.
   b. After a suitable interval, log on at the subscriber site and check that the new user is replicated to this site.
   c. At the subscriber site, delete the user.

© N. Harris Computer Corporation

d. At the publisher site, after a suitable interval, redisplay the Security Manager and then check that the user is deleted.

e. Repeat these steps for any other subscriber sites in your iBase system. Contact your SQL Server administrator if there are any problems.

3. Add groups and users:

After you are satisfied that replication is working correctly, you can add groups and users. The access rights for the users who resolve conflicts, merge records, and perform batch edits and deletions are described in Managing Security.

You will not be able to set up or check the permissions for Data Access Control groups until the publication database exists.

Before you allow users to open any of the databases and start entering data:

- Check that there are no duplicate group or user names. See Handling Duplicate Group and User Names.
- If you are intending to record contact details for each user, predefine the location names. See Assigning a Location to Each User.

When you allow users to start entering data, you might want to ask the users to add their contact details to the database. Each user selects a predefined location from the **Location** list.

4. Test the security audit log.

The audit log for the security data is held in the security file and is configured in SQL Server as part of the security file. If you choose to replicate the audit log, check that the audit log is recording security related actions for other sites.

For example:

a. Open the security audit log for the publisher site. You do not need to open the publication database. You should see, for example, Database Opened actions for all the replicated security files.

b. Repeat this for each site.

Contact your SQL Server administrator if security actions are not being recorded.

Testing replicated databases

Test replicated database to ensure that replication is working correctly and the database properties are correct. You must also test that the Conflict Viewer dialog is accessible.

About this task

After the SQL Server administrator configures all the publication and subscription databases for replication, you need to check that:

- Replication is enabled
- The database properties are correct (at this stage you can turn on Audit History)
- Replication is working correctly
- The Conflict Viewer dialog is accessible in the publication database
- The audit log is working correctly (if used)

You must log on as a database or system administrator to perform these steps.

© N. Harris Computer Corporation

Procedure

1. To test that replication is enabled:
   a. At the publisher site, start iBase, log on as a database or system administrator, and open the publication database.
   b. Select **Tools** > **Replication Status Report**. Depending on the status of the publication database, you should see the following messages:

Message	Description
Publication is OK	The publication database is configured for replication.
<server>\<instance>:<SQL Server database name>	Lists the names of the subscription databases, one for each subscriber site.
No subscriptions	Indicates that the subscriber sites are not yet configured for replication.
Publication not found	Replication is not running at the Publisher. Contact your SQL Server administrator.
The publication is invalid because it permits anonymous subscriptions	Replication is incorrectly configured in SQL Server. Contact your SQL Server administrator. (**Note:** You are not able to review and resolve any conflicts until this problem is rectified.)

2. Verify that the user who uses the Conflict Viewer has the correct access rights in SQL Server:
   a. At the publisher site, start iBase, log on as the user who administers conflicts, and open the publication database.
   b. Select **Tools** > **Replication Conflict Viewer**. The Conflict Viewer dialog appears if the user has sufficient access rights to the SQL Server database. For more information about the iBase permissions required by this user account, see iBase system roles.

      Note: If there is a problem with how replication is configured in SQL Server, you see a message warning that the publication is invalid. This problem prevents you from using the replicated iBase database. Contact your SQL Server administrator.
3. Test that replication is working: A user is required at each site to test replication; the users do not need to be administrators. Alternatively, you might be able to perform the whole test yourself if you are able to log on to the remote servers.
   a. In the publication database, add a record.
   b. After a short interval, check that the new record is replicated to the subscription data

      Note: If you are using a read-only database, replication is working correctly if the new record appears in the subscription database. Remove the test record by deleting it in the publication database.
   c. If the subscription database is not read-only, delete the record.

© N. Harris Computer Corporation

d. After a short interval, check that the record is deleted from the publication database. For example, try searching for it.

e. Repeat these steps for any other subscription databases in your iBase system.

Note: Contact your SQL Server administrator if there are any problems.

4. Test the audit log: Check that the audit log is recording the appropriate actions for all the sites that are involved in replication, and that audit history is turned on (if required).

a. Open the audit log for the publication database. You should see the Database Opened and Database Closed actions for all the replicated databases.

b. Repeat this for each site. The audit level should be the same for all sites.

c. To test that the audit history is replicating, edit a record in the subscription database and confirm that you can see the audit history for that record. After a short interval, check the audit history for that record in the publication database.

Working with a replicated database in iBase

In a replicated database, you should only add data (whether manually or by scheduled import), or edit or delete data while replication is either configured or temporarily stopped. You must not add, edit, or delete data while replication is disabled.

There are a number of other differences between replicated and non-replicated databases that might affect day-to-day working. For an overview of these, see Differences between Working in a Replicated and Non-Replicated Database.

When replication is stopped

Replication might be temporarily stopped because it is suspended by the SQL Server administrator or interrupted by a technical problem. When this occurs, changes cannot be replicated between sites until replication is restarted.

Depending on the reason why replication is temporarily stopped, you can choose to:

- Ask users to log on using a different server and open a different replicated database. This requires Windows Terminal Services or similar.
- Allow your users to continue working but consider discouraging operations such as importing (because there is an increased risk that conflicts occur). After replication is running again, review any conflicts that might occur.
- Take the database offline to allow users who are currently working to save their work before closing their sessions, and to prevent others from starting new sessions.

When replication is disabled

Whenever replication needs to be disabled. You should take the database offline before the SQL Server administrator disables replication. iBase administrators can still open an offline database.

When replication is disabled, it is no longer configured in SQL Server and any changes that are made to the data cannot be replicated to other sites and are not replicated even when replication is reconfigured. In contrast, you can only change the database schema when replication is disabled.

Before replication is disabled, you need to review any existing conflicts. All Subscribers must be online and connected to the Publisher at this stage.

© N. Harris Computer Corporation

Note: If it becomes necessary to change the data, you need to work in the publication database, and inform the SQL Server administrator so that they can configure replication appropriately.

Managing security

Managing security in a replicated database is similar to an unreplicated database. However, the security file must be SQL Server format.

Replication ensures that your iBase security setup is identical at all sites. When you are using a replicated security file, any change that is made by a security administrator at one site is replicated to all the other sites, whether adding or modifying user groups, defining permissions for those groups, or adding or modifying users. Updates are replicated to all the sites at a frequency that is agreed with the SQL Server administrator and might, for example, be anything from every minute to once a day.

If one site requires slightly different security arrangements, you must set up user groups and users that are site-specific.

Each analyst should have their own user account, and any naming convention for these should include a location identifier in the user name. See Updating User Data for details.

Note: To refresh the user groups and users displayed in the Security Manager, close and reopen the Security Manager--- updates to user groups and users from other sites are not otherwise displayed.

iBase system roles

iBase administrators that work with security files, database properties, and the Conflict Viewer require different iBase system roles.

To work with the security file and set up groups and users:

- You need to be able to log on as a security administrator, and your user account must be a member of a group with the security administrator system role. This enables you to:
  - Upsize the security file to SQL Server
  - Change the properties of the security file (anyone with access to iBase Designer can view the properties of the security file but only a security administrator can change them)
  - Display the Security Manager (and check for duplicate groups or users)
  - Define the permissions for groups

To initialize a database for replication and use the options on the Replication menu:

- You need to be able to log on as a database administrator, and your user account must be a member of a group with the database administrator system role. This enables you to:
  - Initialize a database for replication (in iBase Designer)
  - In iBase, run all the commands on the **Replication** menu. You are able to display conflicts in the Conflict Viewer but not review them.

  Note: You also need permission to access the SQL Server conflict tables in the database. Contact your SQL Server administrator if you cannot display the Conflict Viewer but appear to have the correct iBase permissions.

To use the Conflict Viewer to review conflicts:

- Your user account must be a member of a group with the database administrator system role, and also have add, update, and delete permissions for entity and link records, including entity and link records created by other users.

© N. Harris Computer Corporation

Note: You also need permission to access the SQL Server conflict tables in the database. Contact your SQL Server administrator if you cannot display the Conflict Viewer but appear to have the correct iBase permissions.

To change database properties:

- You need to be able to log on as a system administrator. Your user account must be a member of a group with both the security administrator and database administrator roles.

Maintaining System Commands Access Control (SCAC) groups

In a replicated database, you can deny access to iBase commands using System Command Access Control groups. You create the groups for the whole organization using the Security Manager and assign permissions using the `System Commands Access Control` command at the following times:

- In the Microsoft Access security file before you distribute the security files to the subscriber sites.
- In the publication database, either before or after the security file is configured for replication.
- In the subscription database, after replication is configured for the security file at the subscriber sites.

Before you define permissions for System Command Access Control groups, you should review how the following access controls are assigned.

Soft Delete

Denies users access to the commands for purging and restoring soft deleted records. These commands are only available in the publication database, after all current conflicts are resolved.

Batch Modification

Denies users access to `Batch Edit` and `Batch Delete` which, when used especially on entities with large numbers of links, can introduce conflicts into your data. This group also denies access to **Merge Entities**. Conflicts arising from merging cannot be handled in the iBase Conflict Viewer and must therefore be avoided by good working practices. See Merging Entities for details.

Show User Information

Denies access to the User Information and Select User dialogs. This is available in the following places:

- On the **File** menu
- In the Conflict Viewer by clicking the user name
- In the Properties dialog by clicking the **User Information** button
- In the Show dialog and datasheets by clicking the user name (if owner hyperlink fields are used in the entity and link types)

Note: If this group is not displayed in the System Commands Access Control dialog, update the SCAC groups by selecting Update Command Groups from the Tools menu.

If a particular group of users at one site requires different permissions to similar users at another site, then you need to assign their user accounts to a separate SCAC group.

Maintaining Data Access Control groups

Extended Access Control (EAC) gives the ability to assign rights to Data Access Control groups (DAC groups). In a replicated database, you create the groups for the whole organization using the Security Manager and deny access to specific tables and fields on the database using the `Data Access Control` command at the following times:

© N. Harris Computer Corporation

- In the Microsoft Access security file before you distribute the security files to the subscriber sites, if the publication database exists. However, you are not able to deny access based on Security Classification Code lists as these are a feature of SQL Server databases.
- In the publication database, either before or after the security file is configured for replication, if the publication database exists.
- In the subscription database, after replication is configured for the security file at the subscriber sites, if the subscription database exists.

You need to open the database to define the permissions for Data Access Control groups. However, empty groups can be created and users assigned even if the database does not yet exist.

If a particular group of users at one site requires different permissions to similar users at another site, then you need to assign their user accounts to a separate Data Access Control group.

Updating user data

Additional user details can be recorded to provide contact information for other iBase users who need to establish the history of a particular record or modification.

The information is available in the Conflict Viewer and Property dialog, and also in the Show dialog and datasheets if owner hyperlink fields are added to the entity and link types.

Contact information consists of a user's:

- Full name
- Location (which is predefined by the security or system administrator)
- Email address
- Telephone number
- Notes®

It is simplest to ask your users to add their own details: in iBase, select `Change User Information` from the **File** menu. You can also enter these details when you create new accounts in the User dialog (but these details are not copied if you copy a user account). This feature is available in both replicated and unreplicated databases.

Note: Contact details are stored in the security file and are replicated.

Adding users

Add a user account for each analyst who uses iBase database replication. Each user must log in using a unique user name otherwise the Conflict Viewer might not be able to report conflicts between two users logged in with the same user name.

Devising a naming convention for users that includes a location identifier in the username allows you to continue to add new users if a communications link goes down without any risk of creating duplicate users. Ensure that the default password that is given to new accounts is specific to each site.

Duplicate names are checked whenever you display the Security Manager, and are automatically renamed.

Note: To refresh the user groups and users who are displayed in the Security Manager, close and reopen the Security Manager. This does not necessarily display the users and groups added by other sites as this depends on the frequency with which updates to the security file are replicated. Depending on your organization, the frequency might, for example, be anything from every minute to once a day.

© N. Harris Computer Corporation

When to set up the user data

You can add user groups and users, and assign users to groups at any of these stages:

- In the Microsoft™ Access security file, before it is distributed to the subscriber sites.
- At the publisher site, before or after the SQL Server administrator configures the security database for replication.
- At the subscriber sites, after the SQL Server administrator configures the security database for replication. Changes made to security data at any site is replicated to the other sites.

Note: For information on when to define the permissions for Data Access Control groups, see Managing Security.

Assigning a location to each user

In a replicated database where the audit log is also replicated, it is important to assign each user a location as part of their contact details. You can derive the location from the user name, however, the advantage of using the location field in the contact details is that the user can keep the same user name even if they move location. Typically the location is the physical location of the user rather than the location of the database.

You can structure the location name to facilitate wildcard searching in the Audit Viewer if you choose to use the location field in the contact details rather than rely on the user name containing a location identifier.

You must predefine the locations by adding contact information to a sample user from each location:

1. In the Security Manager, select an existing user and click **Edit**.
2. Click the **User Information** button

3. Enter the location name, up to a maximum of 50 characters.
4. Click **OK** twice. The location name is not saved until you click **OK** in the User dialog.

Note: As a security or system administrator, you can also add and modify location names in the User Information dialog in iBase.

Handling duplicate group and user names

You should periodically check for duplicate groups and users.

When you display the Security Manager at any publisher or subscriber site, iBase Designer checks for and corrects duplicate names (whether groups or users). For example, the duplicates are renamed _***1***, _***2***, . The group or user created first keeps the original name, and the second group or user is renamed.

If there are any duplicates, the Security Manager displays a warning message. You should make a note of the groupss and users involved because this message is displayed once only. However, name changes are recorded in the security audit log.

Note: Where there are duplicate user accounts with the same password, either user can log on using the usable account but the access rights might not be correct.

© N. Harris Computer Corporation

Checking the ownership of records

Everyone with a specific responsibility for managing the data in an iBase database can enter their own contact details. These details make it easier for other iBase users to contact them if there is a query that concerns a record that they own. Contact the owner of the record before you modify, delete, or merge records.

About this task

Note: It depends on your organization whether every record type has an owner, or a similarly named, field. Records that are entered in the database before the owner field was added to the entity or link type have a blank owner field.

Depending on your organization, you might need to record additional information about yourself. This information can assist other users with queries about the data for which you are responsible, or who might need to talk to you before they edit, merge, or delete records that you own.

The contact details are for the user name that you use when you log on.

Procedure

To add contact details in iBase:

1. Select **File** > **Change User Information**.
2. Enter your full name, telephone number, and email address. You may also need to select a location. Depending on your organization, this can be a geographical area, a division, or an area of responsibility.

   Note: If nothing suitable is displayed in the **Location** list, contact your system administrator.
3. Save your details by clicking **OK**.

Changing the owner of a record

The owner of a record can be contacted for more information. Keeping the ownership of records up-to-date ensures that the right people are contacted.

Procedure

When you add a record or updating an existing one, you can:

- Make yourself the owner by typing $. If it is not already displayed, your user name is inserted when you save the record.
- Select a different user as the owner:
  1. Click **Browse** next to the "owner" field to display the list of possible owners. If you know the first few characters of the name, enter these first - this then scrolls down to that position in the list.
  2. Double-click a name to select that person as the owner.

Finding the owner of a record

The record owner is the person that should be contacted for information about the record. The record owner can be a different person to the user who created or updated the record.

About this task

When you use either Show or a datasheet, there are two ways of displaying the contact details for the owner of the record:

© N. Harris Computer Corporation

- Click the username that is shown in the owner, or a similarly named, field. The user name is displayed as a hyperlink.
- Click **Edit** and then double-click the user name.

Procedure

To find out who these users were:

1. Right-click on the record in any record list and, from the menu, select **Properties**.
2. Click

to display the contact details for the person who created or updated the record.

Handling conflicts

Depending on the amount of data that is added and edited each day and the types of procedure that you have in place, the interval at which you should check for conflicts in the Conflict Viewer varies. To begin with, a suitable interval might be hourly, then after you know how frequently conflicts occur you can adjust the interval, for example, to once a day.

It is important to review the conflicts regularly as SQL Server will automatically delete the conflicts after a set time, for example after 14 days. You might want to discuss the length of this period with your SQL Server administrator.

Note: You must review all current conflicts before replication is disabled. The losing records in the conflicts are deleted when the SQL Server administrator disables replication. To make sure that you see all the conflicts, all fixed and disconnected Subscribers must be connected to the Publisher.

Minimizing the risk that conflicts occur

It is important to understand that the risk that conflicts occur is low where only fixed Subscribers are used--- the risk increases with disconnected Subscribers because users at these Subscribers are working with data that is only periodically synchronized with the publication database.

Your existing working practices and procedures (that control how data is entered and updated) might already go a long way to minimize the risk that data conflicts occur. Especially if your procedures require analysts to consider the ownership of records before any updates are made.

- Ask your SQL Server administrator to configure a message (an alert) to send an email, or a message to a pager, if replication is temporarily stopped.
- Ensure that each record has an owner and that the owner's contact details are available. For example, you can enforce the ownership of records by adding an owner hyperlink field, as a standard field, to every entity and link type (see Adding owner fields to entity and link types for details). Contact details can also be added to the database in iBase by individual users using the `Change User Information` command on the **File** menu.
- Review who has permission to perform batch edits and batch deletes as this type of operation is more likely to generate conflicts. In particular, batch deleting large numbers of entities.

© N. Harris Computer Corporation

Using the Conflict Viewer

You use the Conflict Viewer to review the conflicts that are automatically resolved in SQL Server. SQL Server resolves each conflict in favor of the Publisher and then in favor of the site that merges first. When you review a conflict, you either confirm that the conflict is resolved correctly or you change the outcome of the conflict, for example in favor of the other site or by editing the winning record.

When you finish, the losing record is purged from the database - this cannot be undone. If you change the outcome of the conflict, then the new outcome is replicated to all sites.

The record that is in conflict can be a modification or a deletion (held as a soft deleted record). When the outcome of a resolved conflict is a deletion, the record remains soft deleted.

To use the Conflict Viewer, you need to log on as a database administrator with full rights to the entity and link records. For more information. see iBase system roles. Your SQL Server administrator might also need to grant you specific rights.

To display the Conflict Viewer, in iBase select **Replication Conflict Viewer** from the **Tools** menu.

Working with disconnected Subscribers

You must review the conflicts that might occur when a disconnected Subscriber synchronizes with the Publisher.

Note: You might want to discourage operations such as importing, including imports scheduled by iBase Scheduler.

Handling a network communications failure

If network communications fail, changes cannot be merged with the Publisher. However, the database server can still be running so your users are able to continue working in the iBase database but with an increase in the risk that conflicts occur.

You can choose to:

- Ask users to log on using a different server and open a different subscription database. This requires Windows Terminal Services or similar.
- Allow your users to continue working and review any conflicts that might occur. However, you might want to discourage operations such as importing, including imports scheduled by iBase Scheduler.
- Take the database offline to prevent anyone else from logging on. This allows users who are currently working to save their work before they close their session.

Merging entities

The Conflict Viewer, handles conflicting changes to entities and their links that arise from direct editing or deleting of the records; it does not handle conflicts that arise from merging entities or occurring in the records affected by a merge operation. A typical example of this type of conflict is someone edits a link that is deleted as a result of a merge performed at a different site.

About this task

For this reason:

- Only authorized users can access the Merge Entities options.
- You need to establish a procedure for reviewing and merging entities so that only one user at a time merges entities, preferably using the publication database.

© N. Harris Computer Corporation

- You need to review any existing procedures that might add to the number of duplicates in the system.

It is important to restrict who has access to **Merge Entities** This ensures that only users who understand the impact of merging entities on a replicated database have authority to perform this operation.

Note: Merging is part of the same command group as batch editing and batch deleting.

The objective of this procedure is to ensure that only one user at a time merges entities to prevent merge conflicts from arising in the first place.

Each site needs to appoint a senior analyst with responsibility for carefully analyzing the records involved in a merge. Depending on the procedure, they might have permission to also perform the actual merge, or you might choose to restrict this to a single user account.

A senior analyst at each site must carefully analyze the records involved in the proposed merge. There are various ways of organizing this. For example, if you want to use iBase to manage this, a possibility is to define a Merge Request entity type in your database schema, which would contain the information that relates to the merge, and use this as a way of reviewing and getting agreement on the decision to merge.

The procedure would be, for example:

Procedure

1. When records are identified as candidates for a merge, a senior analyst charts the records involved to an Analyst's Notebook chart and saves the chart as a record of the pre-merge data.

   Note: This is also a useful way of restoring the data to its original state if a problem is discovered with the merge later.
2. Using iBase, the analyst adds a Merge Request entity that contains details of the records involved in the proposed merge. Depending on how you choose to design the Merge Request entity type, the analyst also includes a screen capture of the pre-merge data to help the review.
3. When the Merge Request is saved, it is automatically replicated to all databases for review. Analysts responsible for reviewing merge requests have set up a browse definition that auto runs when they log on, and that tells them when a new merge request is waiting to be reviewed.
4. Each analyst reviews the details of the proposed merge, and updates the Merge Request entity. There should be space for each analyst to add comments. After saving, their review comment is replicated to the other databases so that other reviewers can follow the progress of the review.

   Note: There is a risk that conflicts occur if two or more analysts edit the Merge Request record within the same replication cycle. In this situation, review the record in the Conflict Viewer in the usual way, taking care to copy the comments from the losing record to the winning record before you click **Apply**.
5. After all the analysts review the merge request, the nominated user merges the records. To do the merge, they:

   - Should work in the publication database
   - Check that all Subscribers are online (a conflict can occur when a communication link, which was down, comes back online)

   They can either:

   - Select one record to keep as the merge entity, and merge the other entity records into it
   - Create a new entity of a suitable type and merge the existing records into this.

© N. Harris Computer Corporation

6. After the merge is complete, you might want to update the owner of the merged record (if you use owner hyperlink fields).

What to do next

Designing a Merge Request entity type

For information on what to consider to design a suitable entity type for Merge Requests, see Updating the Database Design.

Note: Adding a Merge Request entity type is a change to the database schema and must be done before the publication database is configured for replication in SQL Server or when replication is disabled in SQL Server.

How are duplicate records created at the moment?

Consider whether there is any scope within your current working practices to reduce the number of duplicate records that are added to the system. For example:

- Are users importing data that contains duplicates and then using merge to remove the duplicates? In this case, consider how you might clean your data before you import it into a replicated database.
- Check that you have discriminators set on all your entity and link types to prevent users from accidentally entering duplicates.
- Avoid performing batch edits, especially on discriminator fields.

What to do if a conflict occurs

A typical conflict that arises from a merge is, for example, someone edits a link or entity before someone else performing a merge in which the entity or link is deleted. The iBase Conflict Viewer cannot display the context in which this conflict arose, and it is not possible to use the Conflict Viewer to restore it to its original state.

In this situation, the best course of action is to use the chart that contains the pre-merge data to recreate any missing data. It is also possible that your SQL Server administrator might be able to use an SQL Server Conflict Resolver to restore some of the data.

To avoid this situation from arising in the future:

- Always chart the data to Analyst's Notebook and then save a chart before a merge--- this gives you the opportunity to recreate the data if you discover a problem.
- Do not merge if any Subscribers are offline.

Restoring and purging soft deleted records

You can only restore and purge soft deleted records in the publication database after all outstanding conflicts are reviewed in the Conflict Viewer and when there are no broken links. You can never restore and purge soft deleted records when you work at the subscriber sites.

If the outcome of reviewing a conflict in the Conflict Viewer is to keep a soft deleted record, then that record remains soft deleted and might be restored or purged.

After a conflict is reviewed in the Conflict Viewer, the losing record is automatically purged from the database and you cannot restore it. See Conflicts and How They Occur for further details.

Note: Only purge records when all Subscribers are connected to the Publisher. Purging records when disconnected Subscribers are offline might result in hanging links that is links without any entities.

© N. Harris Computer Corporation

Note: Hanging links can occur when an entity and its links are purged at the Publisher but a disconnected Subscriber modifies one of those links. When the Subscriber connects with the Publisher, the modified link is merged into the publication database but the connecting entity no longer exists.

Replicating non-database files

Working in the publication database or in any of the subscription databases, you can use the File Manager dialog to load any type of file into the database for replication to the other sites. You can only load one database template. To load additional database templates, you should archive them or rename them with a different file extension.

You can add a description to the file that is also replicated. After the file is replicated, save it to disk but do not delete it from the File Manager dialog. Editing the description of a file or deleting a file also edits or deletes it at the other sites.

You should only delete files that you know to be out-of-date or that are no longer required.

Example file types

Examples of these file types are:

- A database template (`.idt` file)
- Analyst's Notebook templates (`.ant` files)
- Archive log files (`.idla` files)
- Microsoft Word documents that are used as templates for iBase report definitions, if there are many these, consider adding them to an archive file first (`.zip` file)

Note: After replication, copy these files to the correct place in the database folder or to the correct folder in the iBase installation.

Displaying the File Manager

To display the File Manager in iBase, log on as a database administrator and, select **Tools** > **Replication File Manager**.

Replicating the audit log

It is optional whether the audit logs for the database and security file are replicated. The audit logs for the database and security data are replicated separately.

The audit logs are optionally replicated:

- If the audit data for the database and security file is replicated, authorized users at any site can review all activities.
- If the audit data for the database and security file is not replicated, the audit logs records activity for the local database and security file only. The main effect of this is incomplete information on how conflicts between entity and link records, and in the security data were resolved. For example, log entries for the Conflict Viewer are only recorded in the audit log for the publication database (if the audit level is set to 4 or 5).

Note: The dates and times that are shown in the audit log are local to the server on which the user was working.

© N. Harris Computer Corporation

Filtering the audit log for the database

In a replicated audit log, the best way of filtering the log entries for a specific site depends on how user names and locations are defined in your organization:

Filter by	Description
User name	The user name might be a useful way of filtering the audit log if each user name contains an element to identify the site at which the user works.
Network Login	If users are restricted to logging on to the database on the local server, then the network login can indicate the location of the user.
Location	Locations can be a useful way of filtering the audit log if all user accounts have a location and that the location names are consistent.

Note: Filtering by location does not identify the records, which a user owns (if you are using owner hyperlink fields), only those records that they create or update.

Viewing log entries for conflict resolution

There are two actions for the Conflict Viewer: Conflict Detected and Conflict Resolved. These are recorded if the audit level of the publication database is set to 4 or 5.

Two examples showing the audit trail for these actions are included in the following information.

Note: All dates and times for the Conflict Detected/Resolved records are the SQL Server date/time on the Publisher as conflicts are always detected and resolved in the publication database.

Viewing the audit log for the security file

In addition to the standard entries for non-replicated databases, the audit log for the security file records changes to group and user names that results from resolving duplicate names in the Security Manager.

The site to which a user belongs is identified by the name of the user that created it.

The site to which a group belongs is identified by the group ID which contains the database identifier.

The first site to create the user or group keeps that user or group with the permissions they defined. The user or group at the second site is treated as a duplicate and loses any defined permissions.

For example:

Date/time	User	Action	Detail
09:35:09	Sub1Admin	Create User	User SmithT created
10:06:04	PubAdmin	Create User	User SmithT created
16:00:00	Sub2Admin	Change User	User SmithT renamed to ***1*** (**Note:** The

© N. Harris Computer Corporation

Date/time	User	Action	Detail
			user Sub2Admin displayed the Security Manager which detected this duplicate and automatically renamed it.)
17:05:06	PubAdmin	Change User	User ***1*** renamed to 03SmithT (**Note:** The SmithT created by Sub1Admin is kept with permissions assigned by Sub1Admin. The other SmithT is treated as a duplicate and its permissions are ignored.)

Archiving audit log files

You can archive audit log files in the usual way. It is not necessary to stop or disable replication to do this. However, notice that although the entries you delete are deleted from the audit log at all sites, the archive file is not replicated.

To make the archived entries available to all sites, you should load the archive file into the database by using the replication File Manager.

Example 1: Accepting the original outcome of the conflict

The audit log records two changes to the same record but made at different sites. To simplify the examples, only relevant details are shown:

Date/time	User	Action	Detail
10:50:19	01Smith	Record Modified	
10:50:25	03Malina	Record Soft-Deleted	

At this stage, there is no indication that a conflict has occurred. The date and time shows when the records were changed - it does not show which record merged first with the Publisher.

After the conflict is reviewed in the Conflict Viewer, the audit log shows which record won the initial conflict because it merged first with the Publisher (the change made at 10:50:25) and then which one won when the conflict was reviewed in the Conflict Viewer. In this example the initial outcome is unchanged:

Date/time	User	Action	Detail
10:50:19	01Smith	Record Modified	

© N. Harris Computer Corporation

Date/time	User	Action	Detail
10:50:25	ConfAdmin	Conflict Detected	Winner 10:50:25 (03Malina); Loser 10:50:19 (01Smith) (**Note:** The winner is the record kept by SQL Server and replicated to all sites. The date and time is logged only when the user runs the Conflict Viewer. It is always the later of the two update date/ times of the records in conflict, regardless of which record won or lost the conflict.)
10:50:25	03Malina	Record Soft-Deleted	
10:54:16	ConfAdmin	Conflict Resolved	Winner 10:50:25 (03Malina); Loser 10:50:19 (01Smith) (**Note:** The winner is the record chosen by the user as the one to keep. The user chose the same record as SQL Server so there is no change to the database and no additional log entry.)

Example 2: Change to the original outcome of the conflict

The audit log records two changes to the same record but made at different sites:

Date/time	User	Action	Detail
11:20:03	03Malina	Record Modified	
11:20:07	01Smith	Record Soft-Deleted	

At this stage, there is no indication that a conflict has occurred.

After the conflict is reviewed in the Conflict Viewer, the audit log shows which record won the initial conflict (the change made at 11:20:07) and then which one won when the conflict was reviewed in the Conflict Viewer. In this example the initial outcome is changed:

© N. Harris Computer Corporation

Date/time	User	Action	Detail
11:20:03	03Malina	Record Modified	
11:20:07	ConfAdmin	Conflict Detected	Winner 11:20:07 (01Smith); Loser 11:20:03 (03Malina) (**Note:** This entry is logged when the Conflict Viewer is run but takes the date/time of the later of the two records in conflict.)
11:20:07	01Smith	Record Soft-Deleted	
18:00:55	ConfAdmin	Record Modified	**Note:** Because this record was chosen by the conflict administrator as the one to keep, it is replicated to all sites and an extra entry appears in the audit log.
18:00:55	ConfAdmin	Conflict Resolved	Winner 11:20:03 (03Malina); Loser 11:20:07 (01Smith) (**Note:** The conflict administrator chose to keep the original losing record.)

Updates to code lists

The contents of code lists (that is pick lists, icon lists, and SCC lists) are not part of the database schema, and can be updated at any time by any authorized user. The modifications are replicated to the other sites in the usual way, and will be available to any user who logs on after the changed list is replicated.

Note: Adding or editing the description of a list, or assigning a pick list as a parent or child to create a filtered pick list, is a schema change and is not replicated.

Users who are logged on at the time the change is made continue to see the old pick list (see following information), icon list, or SCC list until they do one of the following:

- Close and then reopen the database.
- Display the pick list dialog and then click **OK** to close it (which updates the contents of all the lists).

This does not apply to filtered pick lists where changes to the contents are seen as soon as they are replicated.

© N. Harris Computer Corporation

Shutting down the databases

To make it easier to gain exclusive access to the databases, for example before you apply changes to the database schema, you can use the `Take Offline` command to broadcast a message to all active iBase users that asks them to end their sessions. Existing users are not ejected from the system by this message but new users (except for database administrators) cannot open the databases.

About this task

The message does not deny database access to services such as the Scheduler service, which should be disabled manually. For more information, see iBase Scheduler.

You can broadcast just the standard message (which is `*** WARNING *** This database is now offline`) or you can add a message to provide users with some instructions or information.

Note: Typically the Take Offline command is run from just one of the databases and the database status is replicated to all the other sites. However, it might need to be run locally if a site is unable to communicate with the publisher site because of a network communications failure.

Procedure

To take the databases offline:

1. Start iBase and log on as a database administrator.
2. Open one of the databases and from the **Tools** menu, select **Replication Take Offline**.
3. In the Take Offline dialog, enter a message (if required). Only users who are actively using iBase, or trying to log on, see the message.
4. Select the frequency with which the message is displayed and click **OK**.

   This prevents new sessions from being started (unless the user is a database administrator) but it does not end any current sessions.
5. To change the frequency with which the message displays or to change the message:
6. From the **Tools** menu, select **Replication Bring Online**.
7. Click **Yes** to bring the databases online.
8. From the **Tools** menu, select **Replication Take Offline** and re-enter the message, change the interval, or both.

What to do next

When the databases are offline, in iBase, select **Tools** > **Replication Bring Online** to bring the databases back online to make them available to users who want to start new sessions.

You can bring the publication database online as soon as your SQL Server administrator tells you that replication is configured for the publisher site. The subscription databases automatically come back online as the subscriber sites are configured for replication.

Note: Do not bring the publication database online while replication is disabled. You should only add data (whether manually or by scheduled import), or edit or delete data while replication is configured. This is because any changes that are made to the data while replication is disabled cannot be replicated to other sites even when replication is reconfigured.

© N. Harris Computer Corporation

Changing the schema of a replicated database

Schema changes cannot be replicated. To change the schema of a replicated database, the SQL Server administrator must disable replication for all sites while you update the database template at each site. After you apply the changes at each site, the SQL Server administrator must to reconfigure replication. For this procedure, all Subscribers must be connected to the Publisher.

About this task

The different stages of the operation are:

1. Preparing a new database template with the wanted schema (you need to work on a test database). When the new template is ready, you load the new template into your production database by using the File Manager - replication automatically distributes it to all sites.
2. Preparing the database so that replication can be disabled. This involves taking the database offline (to prevent further conflicts from occurring) and then reviewing all existing conflicts. After this is done the SQL Server administrator disables replication by deleting the publication and its subscriptions.
3. Applying the new schema at all sites. Only changes to entity types, link types, fields, standard fields, code lists, semantic types, datasheets, and common folder objects are applied.
4. After the new schema is applied, the SQL Server administrator reconfigures replication.

Note: Although the changes to the database schema are not discarded, they cannot be replicated. Changing the schema in this way, can also prevent you from using **File Manager** and **Update Database Schema** to update the schema as described in the following information.

What is a schema change?

Any of the following changes to the database design constitutes a schema change (and are not replicated):

- Adding, modifying, or deleting entity types, link types, fields or standard fields
- Adding, renaming, or deleting pick lists, icon lists, or Security Classification Code lists

  Note: Adding or editing the description of the list is a schema change. Changing the content of these lists is not a schema change, and any changes you make to the contents are replicated.
- Adding, modifying, or deleting common folder objects
- Adding, modifying, or deleting datasheets
- Adding semantic types (whether manually or by loading a custom semantic type library) and assigning them

Do not delete anything from a database schema once the database is live.

Procedure

1. To change the database schema, you can work at any one of the sites involved in replication:
   a. Create a database template to capture the current schema.
   b. Using the template, create a temporary database in which to develop and test the modified schema. The database can be either an Access or an SQL Server database, whichever is most convenient.
   c. Use the temporary database to develop the new database schema, and then test it to check that it works as intended.

© N. Harris Computer Corporation

d. Create a database template to capture the updated schema. You should test the database template by applying it to a parallel database.

Note: At this stage, you might want to plan what changes (if any) are required to other iBase object such as report definitions or labeling schemes.

2. Use the File Manager to replicate the new template to all the sites:

a. Log on as a database administrator, open one of the replicated databases, and load the database template into the database using the File Manager.

See Replicating Non-Database Files for details.

The database template is replicated to all sites.

b. Check that all Subscribers received the new database template.

Note: Do not delete the database template from the File Manager dialog.

c. **Optional:** At each site, save the new template to the `WorkgroupTemplates` folder of the local iBase installation.

3. After all sites receive the new database template, you need to prepare iBase so that replication can be disabled by the SQL Server administrator. This needs to be done in the publication database:

a. If you use iBase Scheduler, disable the Scheduler service to prevent any imports from running. Disabling rather than stopping the service prevents the service from restarting if the server is rebooted.

b. Make sure that all Subscribers are connected to the Publisher.

c. Log on as a database administrator and open the publication database.

d. Take the database offline: in iBase, select **Tools** > **Replication Take Offline**.

See Shutting Down the Databases for details.

This is replicated after a short interval, and the message is displayed to all users actively working in iBase or trying to open one of the databases.

e. Check that everyone closed the databases. For example, if you can log on to the remote servers, try to open each of the databases in iBase Designer. If necessary, ask the SQL Server administrator to log out the remaining users on all sites.

f. Review the current conflicts in the Conflict Viewer: in iBase, from the **Tools** menu, select **Replication Conflict Viewer**. If you change any of the records involved in the conflicts make a note of the last record that you change and the nature of the change. (This helps you to check later that the modified records reached the subscriber sites.)

Any changes that you make in the Conflict Viewer are replicated to all subscriber sites.

g. Check that any records modified as a result of using the Conflict Viewer reached all subscriber sites.

h. Ask the SQL Server administrator to disable replication.

Warning: After replication is disabled, you must not make any changes to the data in any of the databases. The changes are not replicated to the other sites after replication is reconfigured.

After the SQL Server administrator informs you that replication is disabled, you can apply the new schema to the databases. Before you start, ensure that you have a backup of the database and that replication is disabled.

© N. Harris Computer Corporation

- If you are working in the publication database, run the Status report to check that replication is disabled: in iBase, from the **Tools** menu, select **Replication Status Report**. It reports `Publication not found`.
- If you are working in a subscription database, open the database in iBase Designer. If replication is still running, you see the message:

  ```
  *** WARNING *** This is a replicated database. You must not change the
   database schema.
  ```

4. To apply the schema change at each site:

   a. In iBase, log on as a database administrator, and select **Tools** > **Replication Update Database Schema**.

      Note: You cannot display this dialog if you are a member of a Data Access Control group that denies access to any tables or fields in the database.

   b. Click **OK** to close the database. If you are warned that there is no database template, you can still load it manually. However, do not load any other files into the database as all the files listed in the File Manager can be overwritten after replication is reconfigured.

   c. **Optional:** Save a list of schema changes in a format that is useful for the SQL Server administrator: in the Update Database Schema dialog, click

      You can print this file later.

   d. Click **Update** to apply the changes and then, once this is finished, click **OK** to reopen the database.

   e. Check that the template is applied successfully.

   After all subscriber sites apply the schema change, notify the SQL Server administrator that replication can be reconfigured.

5. When the SQL Server administrator informs you that replication is reconfigured:

   a. Verify that replication is running: for example, open the publication database in iBase, and from the **Tools** menu, select **Replication Status Report**. The message `Publication is OK` is displayed.

      You might also want to test that replication is running correctly. See Testing Replicated Databases for details.

   b. In iBase, log on as a database administrator, open the publication database and make the database available again to other users: from the menu, select **Tools** > **Replication Bring Online**.

      After a short interval, the status is replicated to the subscriber sites.

   c. Notify users that they can log on again.

   d. If you use iBase Scheduler, restart the Scheduler service.

      Warning: After replication is configured, you should not open the database in iBase Designer unless necessary (unless you are a security administrator).

© N. Harris Computer Corporation

Check the integrity of iBase databases

There are some restrictions on when you can check the integrity of replicated iBase databases.

Schema integrity

You cannot run a schema integrity check while replication is configured in SQL Server.

Link integrity

You can run a link integrity check on a replicated database, preferably on the publication database, if there are no conflicts and you have exclusive access to the database. To obtain exclusive access to the database, use the `Take Offline` command.

Note: The Link Integrity Checker runs on physically deleted records and therefore, does not assist you in resolving any problems with merged entities (which apply to soft deleted records only).

For SQL Server administrators

When you set up iBase database replication in SQL Server Enterprise Manager or Management Studio, keep your replication setup as simple as possible and avoid using SQL Server replication features that increase the complexity of the setup.

iBase database replication uses SQL Server merge replication to distribute data from the Publisher to Subscribers, allowing the Publisher and Subscribers to make updates, and then to replicate the updates between sites. However, iBase database replication is unlike SQL Server merge replication in that Subscribers must always be connected to their Publisher - it is not suitable for disconnected users.

The setup is described in more detail in the following information.

iBase supports the following features of SQL Server replication:

- Merge replication.
- Servers running SQL Server 2005 or SQL Server 2008 (the replicated iBase system must not contain a combination of these).
- Native SQL Server format for the initial snapshots.
- Subscribers that synchronize with the Publisher--- the Subscribers can be either fixed or disconnected.
- Subscribers that receive subscription data from the Publisher.

For more information, see Supported Publication Options and Supported Subscription Options.

Prerequisites

All servers that are to be configured for merge replication must be set to the same time.

Unsupported features

iBase does not support:

- **Anonymous pull subscriptions** - publications that allow anonymous pull subscriptions are invalid.
- **On demand synchronization** - data from Subscribers must be synchronized at a regular interval, for example a polling interval for the Merge Agent of 1 second for the publication that contains the entity and link records. You can select a different interval for the publications that contain the security and audit data.

© N. Harris Computer Corporation

- Subscribers that receive subscription data from another Subscriber rather than from the central Publisher (**republishing**).
- Subscribers that synchronize with another Subscriber rather than with the Publisher (an **alternate synchronization partner**).
- **Column level conflict resolution** in iBase databases all updates are handled at row level. This is because a change to a single column changes the modification date and user, which results in a conflict occurring.
- **Data filtering** of either columns or rows when you add a table to an article, you must select all the columns and rows.
- **Dynamic data filtering** (you cannot filter the data that you provide for different Subscribers).
- **Publishing via the Internet**.
- **Custom resolvers** iBase has its own Conflict Viewer, which runs as part of iBase and requires that the first change to merge with the Publisher wins the conflict. The Conflict Viewer allows iBase users to review the conflicts and change the outcome if required.

Supported publisher options

You must select specific SQL Server options to configure Publishers for iBase databases.

When you configure Publishers for iBase databases, whether containing entity and link data, security data or audit data, you must select the following SQL Server options only, and accept all other defaults:

- Merge replication
- Either 'SQL Server 2005' or 'SQL Server 2008 or later' as the only subscriber type (the replicated iBase system must not contain a combination of these)
- Articles that contain tables only - there is no need to replicate the iBase stored procedures, and there are no views.

Although there are a wide range of Publisher properties, iBase support the default settings only. For this reason, you should not alter any of the default article properties (except where indicated in the following instructions).

Supported subscription options

iBase has various supported subscription options.

When you configure subscriptions to a publication that contains iBase data:

- Configure either push subscriptions or named pull subscriptions
- Accept all the default settings for the subscription options - the iBase Conflict Viewer is designed to work with the default conflict settings.
- Use continuous, rather than on demand or scheduled synchronization.

Never use the Pull Subscription wizard to create new subscription databases - databases are always be created by an iBase administrator using iBase Designer.

When you configure subscriptions for a system in which replication is already configured, you should not initialize the schema and data. The databases always have their schema and data, and are synchronized with each other (if the rules for schema changes are followed by the iBase administrators).

When you work with a large iBase database or a slow communications link, you can transfer iBase data to the Subscriber using a full backup rather than download a snapshot. The backup is a standard full

© N. Harris Computer Corporation

backup - there are no specific options that must be set. When you restore the backup, you must turn off the **Preserve the replication settings** checkbox on the **Options** page of the Restore Database dialog.

Setting Up iBase database replication in SQL Server

Set up iBase database replication in SQL Server. Ensure that the environment is correctly prepared.

Publications and subscriptions for entity and link data, security data, and audit data can be created in any order:

Data type	iBase file	SQL Server database
Security data	`<database name>.ids`	`<database_name>_Sec`
Entity and link data	`<database name>.idb`	`<database_name>`
Audit data	---	`<database_name>_Log`

Note: In database names that are derived from iBase file names, characters outside the range of A-- Z, a-- z, 0-- 9, and space are converted to underscores. An underscore is also appended to the database name if the iBase file name consists of a single word.

Typically security data is configured first and the audit data is configured last.

Before you can initialize a subscription, you must to transfer an initial snapshot of data to each Subscriber, and it is possible that the entity and link data is too large to download.

If you intend to apply the initial snapshot for the entity and link database over a communications link, then follow the steps in the Configuring Subscribers section. However, if the entity and link database is too large for this, follow the steps in Applying the snapshot manually using removable media.

Before any databases are created

Before the iBase administrators create any new iBase databases, you might want to predefine the security for these databases by restricting permissions to the Public role for the Model database (which is the template for new user databases). For details, see Managing access to data and functions.

You should also agree a database naming convention with the iBase administrator. The database name must not exceed 119 characters.

Note: All servers to be configured for merge replication must be set to the same time.

Configuring the Distributor and Publisher

1. Enable the server that you are using as the Distributor and Publisher for replication. If you want to create push subscriptions or named pull subscriptions, also enable the servers that you are using as Subscribers.
2. Configure the Distributor.
3. At the Publisher, create publications for the security data, entity and link data, and optionally the audit data.

   For more information, see

   - Publishing security data

© N. Harris Computer Corporation

- Publishing entity and link data (there is an additional step if you need to apply the snapshot manually)
- Publishing audit data

4. Back up the replicated databases, scripts, and appropriate system databases. See Backing Up and Restoring Replicated Databases for details.

   Note: The iBase administrator must set up the databases at the Publisher before you can configure replication. Details are given in each section.

Configuring Subscribers

Follow these steps if you intend to initialize the subscriptions by applying the initial snapshot over your communications link:

1. Create a subscription to the publication that contain the:
   - security data
   - entity and link data
   - audit data (optional)

   The subscription can be a push subscription or a named pull subscription. The iBase administrator has already created the subscription databases.
2. Set up the Merge Agents to run at an appropriate interval, for example 1 second for the entity and link data. If the Subscriber is to be read-only, configure the Merge Agents to prevent uploads to the Publisher.
3. Back up the replicated databases, scripts, and appropriate system databases. See Backing Up and Restoring Replicated Databases for details.

Applying the snapshot manually using removable media

Follow these steps if you cannot apply the initial snapshot of the entity and link data over a communications link:

1. At the Publisher, perform a full backup of the replicated, publication database. Distribute the backup file to the subscriber sites.
2. At each Subscriber, restore the publication database over the empty subscription database.
3. Create a subscription and then synchronize it with the Publisher.
4. Set up the Merge Agent for each subscription to run continuously at a 1 second interval. If a Subscriber is to be read-only, configure the Merge Agents to prevent uploads to the Publisher.
5. Back up the replicated databases, scripts, and appropriate system databases. See Backing Up and Restoring Replicated Databases for details.

Considerations for large databases

Transfer the initial snapshot of the entity and link data from a large publication database to your subscriber sites by backing up the publication database (in SQL Server) and then restoring it over the empty subscription databases.

This process:

- Ensures that all the required iBase system and user tables are created correctly - most of the iBase system tables are not replicated.

© N. Harris Computer Corporation

- Avoids the need to transfer large amounts of data over your communication links.
- Ensures that the Subscriber receives all the updates made to the publication database since the publication was created and the Subscriber configured (if the time lapse between the two does not exceed the expiration setting on the publication).
- Allows you to synchronize the newly configured Subscribers and Publisher at a convenient time.

Replication security

The following information describes the general security environment for iBase Database Replication. However, there are no differences between security for replicated iBase databases and security for other types of replicated database. For detailed information on general replication security, refer to the Microsoft SQL Server documentation.

The following section also explains the SQL Server logins required by iBase administrators. In particular, the permissions required for the iBase Conflict Viewer.

SQL Server Agent

Replication, in common with other SQL Server functions that use scheduled services, requires an SQLServerAgent service to be running on each SQL Server instance (whether Publisher, Distributor, or Subscribers). Configure the SQL Server Agent service to start automatically when the server starts up.

This service needs a Windows logon and password. For Windows logons, you can use one of the following:

- A domain account. If the account belongs to different domains, you might need to set up trust relationships between the domains. However, for SQL Server Agents, the account does not need to belong to the Windows Administrators group.
- Local accounts. Each local account must be identical on all servers and must be a member of the local Administrators group.

The Merge and Snapshot Agents used by merge replication run within the security context of the SQL Server Agent. For more information, see the Microsoft SQL Server documentation.

The SQL Server Agent account needs appropriate permission to the snapshot folder on the server that acts as the Distributor.

Snapshot files and folder

The folder in which the initial snapshot files of the iBase data are stored must be available to the SQL Server Agent account for the Publisher and Subscribers. Specifically:

- The Snapshot Agent that writes the initial snapshots from the Publisher requires Full Control.
- The Merge Agent for each Subscriber requires Read access (because the Merge Agent must read the snapshot at each Subscriber before replication can start).

To ensure secure access to the snapshot files, use an explicit share, rather than the administrative share because accounts must be a member of the Administrators group to access this share.

In addition to the default permissions on this folder, the SQL Server Agent account for the Publisher needs read and write access to the contents of the folder so that they can read and write snapshot files.

Security mode

Connections to a server (Publisher, Distributor, or Subscribers) can use Windows Authentication or SQL Server security. Windows Authentication provides greater security and general ease of use.

© N. Harris Computer Corporation

SQL Server login and roles for SQL Server administrators

You must be an SQL Server system administrator to enable the servers, on which the iBase databases are held, for replication.

After replication is enabled, you do not need to be an SQL Server system administrator to configure publications and subscriptions, or to invoke or schedule the replication agents. You must be in the sysadmin or db_owner roles to create publications, create subscriptions, and attach subscription databases.

SQL Server login and roles for iBase administrators

If you follow the configuration method for iBase database replication that uses attachable subscription databases, then you must grant the logins for the iBase administrators access to the subscription databases as they lose their access rights as a result of this configuration method.

The user responsible for conflict resolution in iBase must have the necessary permissions to access the SQL Server conflict tables. You grant this access by adding their login name to the Publication Access List.

Setting Up Database Replication in SQL Server

Create publications and subscriptions for iBase.

You can create publications and subscriptions for iBase based on:

- Entity and link data from the main iBase database
- Security data (this can be shared by several iBase databases)
- Audit data

And, optionally:

- Suitable procedures for creating publications and subscriptions are described in the following section. Unless stated otherwise, the procedures are the same for all supported versions of SQL Server.

  For the main iBase database that contains the entity and link data, you publish all the user tables and a specified set of iBase system tables. You do not publish any user-defined stored procedures or views. The procedure is slightly more complicated when the initial snapshot for the main database is too large to download over a communication link. For details of the deployment process used in this situation, see:
- Overview for Large Databases

For background information on the SQL Server options supported by iBase, see Overview of Supported SQL Server Replication Features.

Note: Only qualified SQL Server administrators should implement replication for iBase databases.

© N. Harris Computer Corporation

Setting the time on the servers

All servers that are to be configured for merge replication must be set to the same time.

Configure the distributor

As with any type of SQL Server database replication, deployment starts when you configure a Distributor. Configuring the Distributor for iBase is no different than configuring one for other types of SQL Server replication, and there are no special considerations for iBase.

Publishing security data

The following section describes how to publish the security data associated with the main iBase database. Before you can configure the Publisher for a security database, the iBase administrator should have already set up the iBase security database on the server that is the Publisher.

Before you begin

If you intend to replicate the security audit log, include the `_ AuditLog` table when you create the publication.

Note: In iBase, this database is typically referred to as the security file.

Procedure

To create a publication for security data:

1. From the Object Explorer list, select **Replication** > **Local Publications**, right-click and select **New Publication**.
   - **Server:** select the server that is also the Publisher of the entity and link data for the iBase system. (Use the same server for all publications.)
2. On the **Publication Database** page, select the existing database that contains the security data you want to publish. The database name is `<database>_Sec` where `<database>` is derived from the name of the `ids` file created at the publisher site.
3. On the **Publication Type** page, select **Merge publication**.
4. On the **Subscriber Types** page, ensure that only **SQL Server 2008 or later** is selected.
5. On the **Articles** page, expand the **Tables** node. Select these iBase system tables:
   - `_ AccessDenied`
   - `_ AuditLog` (omit this table if you do not want to replicate the security audit log)
   - `_AutoLogon` (optional but only relevant if Windows Single Sign-On is used and users are able to log on to the different systems using the same Windows credentials)
   - `_CommandGroup`
   - `_Group`
   - `_PasswordHistory`
   - `_SecurityPolicy` (optional)
   - `_User`
   - `_ UserGroup`
   - `_UserSettings`

   Note: You must not include these tables:

© N. Harris Computer Corporation

- `_Configuration_Binary`
- `_Configuration_Def`
- `_Configuration_Text`
- `_ NextId`

6. Accept whatever article issues are reported - typically SQL Server warns that "`Uniqueidentifier columns will be added to tables`".
7. Enter the publication name. For example, `iBase_security_data`.
8. Generate the publication in the usual way. There are no additional properties to set for this type of publication.

What to do next

After you create the publication:

1. Restrict access to the tables in the SQL Server database. For more information, see Managing access to data and functions.
2. Back up the databases. For more information, see Configuring and maintaining databases.
3. Tell the iBase administrator that the security data at the Publisher is configured for replication.

Subscribing to security publications

You can create push or named pull subscriptions for iBase security data. Unless specified below, always accept the default options for the subscription (notice that there are no conflicts for SQL Server to resolve in this type of iBase database).

Before you begin

Before you can start, the iBase administrator should have already created iBase security databases on the Subscribers. These can contain some data but the data is overwritten by the initial snapshot.

Procedure

1. Create a subscription and select the following options for the subscription:
   - The Subscriber does not have the schema and data and should therefore be initialized.
   - Depending on the subscription type, set the Merge Agent to initialize the subscription immediately (optional).
   - The Merge Agent should continuously check for updates.
   - The subscription is a client subscription type
2. Verify that the initial snapshot is applied. For example, check that a rowguid column is added to one of the tables included in the publication, such as the `_User` table.
3. Test that replication is working correctly for the new subscription. Use the `_User` table for this purpose:
   a. At the Subscriber, in the `_User` table, add a user.
   b. After the change is merged, check the corresponding row in the `_User` table at the Publisher.
   c. At the Publisher, delete the user you added.
   d. After the change is replicated, check the corresponding row in the `_User` table at the Subscriber.

© N. Harris Computer Corporation

4. Set the polling interval for the Merge Agent to the required frequency. See Setting Up the Merge Agents.
5. Restrict access to the tables in the SQL Server database. For more information, see Managing access to data and functions.

   Note: The iBase administrator who created this database might already have a suitable SQL Server login as they are the database owner but you might need to add the logins for any other administrators to the database.
6. Back up the databases. For more information, see:
   - Configuring and maintaining databases
   - Back up and restore replicated databases
   - Microsoft SQL Server documentation for information on backing up the replicated SQL Server databases
7. Tell the iBase administrator that the security data at the Subscriber is configured for replication.

Publishing entity and link data

The following section describes how to publish the entity and link data in the main iBase database. The procedure for publishing the security and audit data is slightly different. There are some optional steps at the end of the procedure that are required if you need to transfer the initial snapshot of data to the subscriber sites by using removable media rather than over a communications link.

Before you begin

Before you can configure the Publisher, the iBase administrator should have prepared the database, and advised whether Audit History is being used.

Use the same server for all publications of iBase data.

About this task

To create a publication for entity and link data, whether for a new database or a database that is already replicated, follow these steps. You should accept all the default options unless indicated otherwise in the instructions.

Procedure

1. From the Object Explorer list, select **Replication** > **Local Publications**, right-click and select **New Publication**.
   - **Server:** select the server that is also the Publisher of the entity and link data for the iBase system. (Use the same server for all publications.)
2. On the **Publication Database** page, select the existing database that contains the iBase entity and link data that you want to publish.
3. On the **Publication Type** page, select **Merge publication**.
4. On the **Subscriber Types** page, ensure that only **SQL Server 2008 or later** is selected.
5. On the **Articles** page, expand the **Tables** node. Select these iBase system tables:

© N. Harris Computer Corporation

Option	Description
_ AccessDenied	iBase system table that stores data for Data Access Control groups.
_Codes	iBase system table that stores the contents of pick lists, icon lists, and SCC lists.
_DataTable_Audit	Select this only if you are using Audit History.
_Field_Audit	Select this only if you are using Audit History.
_ GlobalConfiguration	iBase system table that stores the online/offline status of the database.
_ LinkEnd	iBase system table that stores the link connections made by users between entity records, the link direction, link strength, and the Security Classification codes (SCC codes).
_ ReplicationFiles	iBase system table that stores any files uploaded to the database. If this table is missing, cancel the publication and ask the iBase administrator to initialize the database for replication.
iBase user tables	You must replicate all the user tables in the iBase database. There is one table for each user-defined object in the database schema. User table names do not start with an underscore.

Note: Do not include the tables beginning _AL_, _FTS_, the SQL Server system tables or the remainder of the iBase system tables. These iBase system tables are not designed for replication---the names of these system tables begin with an underscore.

6. If you are using Audit History, in Article Properties, for all the selected tables, set **Copy User Triggers** to False.
7. Accept any article issues that are reported. Typically SQL Server warns that `" Uniqueidentifier columns will be added to tables"`.
8. Enter the publication name. For example, `iBase_entity_and_link_data`.
9. Generate the publication in the usual way.
10. Set the publication property **Anonymous pull subscriptions** to False as this option is not supported.

    Note: Do not change any other publication properties. Specifically, allowing anonymous pull subscriptions invalidates the publication.
11. If you are replicating this database for the first time and you need to transfer the initial snapshot to each Subscriber using removable media, then perform a full backup of the database.

© N. Harris Computer Corporation

What to do next

After you create the publication:

1. Restrict access to the tables in the SQL Server database. For more information, see Managing access to data and functions.
2. Back up the databases. For more information, see Configuring and maintaining databases.
3. Tell the iBase administrator that the security data at the Publisher is configured for replication.

Subscribing to entity and link publications

Create push or named pull subscriptions for an entity and link database.

About this task

Note: If you are creating a subscription for the first time to a database transferred to the subscriber site as a backup or attachable subscription database, follow the instructions in:

In this situation,

- Subscriptions for Restored Databases

  If you are recreating a subscription that has been dropped, follow the instructions in Recreating Publications and Subscriptions.

  Before you can start, the iBase administrator should have:
- Created a database for the entity and link data on the Subscriber
- Told you if the database is to be read-only because this affects how you set up the Merge Agent

Unless specified below, you should always accept the default options when you create the subscription.

Procedure

1. Start a new subscription of the required type. The subscription should be as follows:

Option	Description
Publication	The database that contains the iBase entity and link data at the Publisher.
Subscription database	The existing entity and link database at the Subscriber.

   Select the following options for the subscription:

   - The Subscriber does not have the schema and data and should therefore be initialized.
   - Depending on the subscription type, set the Merge Agent to initialize the subscription immediately (optional).
   - The Merge Agent should continuously check for updates.
   - Use the Publisher as a proxy for the Subscriber when resolving conflicts (described as First to Publisher Wins).
   - The subscription is a client subscription type

© N. Harris Computer Corporation

2. Set the Merge Agent for the Subscriber to run at the required frequency, for example a polling interval of 1 second. For more information, see Merge agent for the entity and link database.

   Note: If the iBase database at this Subscriber is read-only, configure the Merge Agent to prevent any uploads to the Publisher. For more information, see Merge agent for a read-only database.
3. Verify that the snapshot has been applied by checking that a rowguid column exists in one of the user tables selected for the article. If this column is missing, check that the snapshot has been applied.
4. Test that replication is working correctly for the new subscription. Use one of the user-defined tables:
   a. At the Publisher, in one of the user tables (table names do not start with an underscore), change the data in one of the columns for a row. For a new database, carry out this test on the `_GlobalConfiguration` table and change the version number but take care to change it back to its original value.
   b. After the change is replicated, check the corresponding table at the Subscriber.
   c. At the Subscriber, change the column back to its original value.
   d. After the change is replicated, check the table at the Publisher. If the Subscriber is supposed to be read-only, and the change is merged at the Publisher, check the setting of the Merge Agent.
   e. If the Subscriber is read-only, change the data at the Publisher back to its initial value.
5. Restrict access to the tables in the SQL Server database. For more information, see Managing access to data and functions.
6. Back up the databases. For more information, see:
   - Configuring and maintaining databases
   - Back up and restore replicated databases
   - Microsoft SQL Server documentation for information on backing up the replicated SQL Server databases
7. Tell the iBase administrator that the security data at the Subscriber is configured for replication.

Subscriptions for restored databases

This procedure only applies to systems where the initial snapshot is transferred manually using a full backup.

Before you begin

You need to restore the iBase entity and link data if you chose to transfer the initial snapshot of data to the subscriber site by using a full backup. Your iBase administrator should have already created an empty iBase database to receive the data.

Perform the restore in the usual way. When you restore:

- Turn on **Overwrite the existing database**
- Leave **Preserve the replication settings** turned off

After you restore the database, you apply the initial snapshot of data and you are ready to configure a subscription. After you configure the subscription, you must to synchronize the Subscriber with the Publisher.

Procedure

1. Start a new subscription, either a push subscription or a named pull subscription:

© N. Harris Computer Corporation

Option	Description
Publication	The database that contains the iBase entity and link data at the Publisher.
Subscription database	The existing entity and link database at the Subscriber.

Select the following options for the subscription:

- The Subscriber does not have the schema and data and should therefore be initialized.
- The Merge Agent should continuously check for updates.
- Use the Publisher as a proxy for the Subscriber when resolving conflicts (described as First to Publisher Wins).
- The subscription is a client subscription type

2. Synchronize each Subscriber with the Publisher:
   a. If the Merge Agent is set to use On Demand synchronization, start the Merge Agent manually to upload any changes from the Publisher.

      Note: If there has been many changes at the Publisher, you might want to delay this step until there is a quiet period on your communications link.

   b. Check whether you are synchronized with the Publisher by verifying that a recent change in one of the user tables at the Publisher is replicated to the Subscriber.
3. Set the Merge Agent for the Subscriber to run at the required frequency, for example a polling interval of 1 second. For more information, see Merge agent for the entity and link database.

   Note: If the iBase database at this Subscriber is read-only, configure the Merge Agent to prevent any uploads to the Publisher. For more information, see Merge agent for a read-only database.
4. Verify that the snapshot has been applied by checking that a rowguid column exists in one of the user tables selected for the article. If this column is missing, check that the snapshot has been applied.
5. Test that replication is working correctly for the new subscription. Use one of the user-defined tables:
   a. At the Publisher, in one of the user tables (table names do not start with an underscore), change the data in one of the columns for a row. For a new database, carry out this test on the `_GlobalConfiguration` table and change the version number but take care to change it back to its original value.
   b. After the change is replicated, check the corresponding table at the Subscriber.
   c. At the Subscriber, change the column back to its original value.
   d. After the change is replicated, check the table at the Publisher. If the Subscriber is supposed to be read-only, and the change is merged at the Publisher, check the setting of the Merge Agent.
   e. If the Subscriber is read-only, change the data at the Publisher back to its initial value.
6. Restrict access to the tables in the SQL Server database. For more information, see Managing access to data and functions.
7. Back up the databases. For more information, see:
   - Configuring and maintaining databases

© N. Harris Computer Corporation

- Back up and restore replicated databases
- Microsoft SQL Server documentation for information on backing up the replicated SQL Server databases

8. Tell the iBase administrator that the security data at the Subscriber is configured for replication.

Publishing audit data

The audit log database records events that affect the entity and link data. iBase automatically creates this database alongside the database that holds the entity and link data, giving it the same database name but with the suffix `_log`. For instance, the database `Crime_` has an audit log database `Crime__log` (note the double underscore).

About this task

Audit records for the security data are held in the `_ AuditLog` table in the security database. This information is already replicated if you included the `_ AuditLog` table when creating the security publication.

To create a publication for audit data, follow these steps. You should accept all the default options unless indicated otherwise in the following instructions.

Procedure

1. From the Object Explorer list, select **Replication** > **Local Publications**, right-click and select **New Publication**.
   - **Server:** select the server that is also the Publisher of the entity and link data for the iBase system. (Use the same server for all publications.)
2. On the **Publication Database** page, select the existing database that contains the audit data (it ends with the suffix `_log`). For example, the publication database `Crime_` has an audit log database called `Crime__log` (note the double underscore).
3. On the **Publication Type** page, select **Merge publication**.
4. On the **Subscriber Types** page, ensure that only **SQL Server 2008 or later** is selected.
5. On the **Articles** page, expand the **Tables** node. Select these iBase system tables:
   - Select the `_AuditLog` table if Audit History is not used.
   - If Audit History is used, select:
     - `_Audit`
     - `_AuditCodes`
     - `_AuditData`
     - `_AuditDataBinary`
     - `_AuditLog`
     - `_Field`
6. If you are using Audit History, in Article Properties, for all the selected tables, set **Copy User Triggers** to False and **Copy Permissions** to True.
7. Accept any article issues that are reported. Typically SQL Server warns that `" Uniqueidentifier columns will be added to tables"`.
8. Enter a publication name. For example, `iBase_audit_data`.

© N. Harris Computer Corporation

9. Generate the publication in the usual way.

What to do next

After you create the publication:

1. Restrict access to the tables in the SQL Server database. For more information, see Managing access to data and functions.
2. Back up the databases. For more information, see Configuring and maintaining databases.
3. Tell the iBase administrator that the security data at the Publisher is configured for replication.

Subscribing to audit publications

You can create either push or named pull subscriptions for an audit log database. When you create the subscription, you should always accept the default options unless specified otherwise in the following instructions (notice that there are no conflicts to resolve in this type of iBase database).

About this task

Note: The audit log database is created automatically when the database that contains the entity and link data is first opened in iBase.

Procedure

1. Start a new subscription, either a push subscription or a named pull subscription:

Option	Description
Publication	The database that contains the iBase entity and link data at the Publisher.
Subscription database	The existing entity and link database at the Subscriber.

   Select the following options for the subscription:

   - The Subscriber does not have the schema and data and should therefore be initialized.
   - The Merge Agent should continuously check for updates.
   - Use the Publisher as a proxy for the Subscriber when resolving conflicts (described as First to Publisher Wins).
   - The subscription is a client subscription type
   - Depending on the subscription type, set the Merge Agent to initialize the subscription immediately.
2. Verify that the snapshot is applied by checking that a rowguid column exists in the `_ AuditLog` table. If this column is missing, check that the snapshot is applied.
3. Set the Merge Agent for the Subscriber to run at the required frequency, for example a polling interval of 1 second. For more information, see Merge agent for the entity and link database.

   Note: If the iBase database at this Subscriber is read-only, configure the Merge Agent to prevent any uploads to the Publisher. For more information, see Merge agent for a read-only database.
4. Test that replication is working correctly for the new subscription. Use one of the user-defined tables:

© N. Harris Computer Corporation

a. At the Publisher, in the `_ AuditLog` table, change one of the user names.
b. After the change is replicated, check the corresponding table at the Subscriber.
c. At the Subscriber, change the column back to its original value.
d. After the change is replicated, check the table at the Publisher. If the Subscriber is supposed to be read-only, and the change is merged at the Publisher, check the setting of the Merge Agent.
e. If the Subscriber is read-only, change the data at the Publisher back to its initial value.

5. Restrict access to the tables in the SQL Server database. For more information, see Managing access to data and functions.
6. Back up the databases. For more information, see:
   - Configuring and maintaining databases
   - Back up and restore replicated databases
   - Microsoft SQL Server documentation for information on backing up the replicated SQL Server databases
7. Tell the iBase administrator that the security data at the Subscriber is configured for replication.

Setting up the merge agents

The synchronization process in merge replication ensures that data at the Publisher and Subscribers eventually converge and that all sites end up with the same data values. However, the sites are never completely synchronized unless there is a period of complete inactivity in iBase.

The best way to ensure that the sites are as closely synchronized as possible is to set a low polling interval, particularly for the entity and link data, for example a 1 second polling interval. Different considerations will apply for the iBase security data and audit log.

Note: After you change agent properties, stop and start the agent, or start the agent manually by using Start Synchronizing.

The SQLServerAgent service should be set to auto-start whenever the operating system starts.

Configuring the Merge Agent to run continuously

The pull subscriptions that you use to deploy replication to the subscriber sites for large entity and link databases are configured for On Demand synchronization. However, in day-to-day use, the Merge Agent must run continuously for iBase database replication to function correctly. To reconfigure the Merge Agent, change the properties of the Merge Agent by adding the following to the Run Agent command line:

```
-Continuous
```

This parameter specifies that the agent queries the Publisher or Subscriber for data changes at the specified interval, even if there are no updates pending.

You also need to set the polling interval as described in the following information.

Start or restart the agent to use the new agent properties.

Note: Until the Merge Agent is set to run continuously, a pull subscription at the Subscriber has the status Never Started and at the Publisher, the status Deactivated.

© N. Harris Computer Corporation

Merge Agent for the entity and link database

By default, the Merge Agent runs every 60 seconds. Reconfigure the Merge Agent to run at a 1 second polling interval to reduce the risk that data conflicts occur. Add the following to the Run Agent command line:

```
-PollingInterval 1
```

Start or restart the agent to use the new agent properties.

Merge Agent for the security database

For the security data, you can set the polling interval for the Merge Agent for each Subscriber to any required frequency, for example 1 second. To decide the polling interval for this database, consider:

- How often does the user data in the security database change?
- How quickly must changes be replicated to other sites?

Note: Users can add records at the Subscriber before their user login is known at the Publisher.

Merge Agent for the audit log

If you replicate the audit log, you can set the polling interval for the Merge Agent for each Subscriber to any suitable frequency, for example hourly. However, the best frequency for your organization depends on:

- How often analysts consult the audit log
- How long it takes for a problem to be identified and for analysts to decide to review the audit log
- The audit level of the database

Where the audit level is set to 5, all database activity is logged including tasks that do not change the data, such as browsing. In this case, you might want to configure an interval that coincides with quieter periods on your communication links.

Merge Agent for a read-only database

Configure the Merge Agent for a read-only iBase database to prevent any changes made at the Subscriber from being uploaded to the Publisher. For example, changes made through SQL Server Query Analyzer.

To reconfigure the Merge Agent, add this parameter to the Run Agent command line:

```
-ExchangeType 2
```

The argument specifies that the agent only downloads data changes from the Publisher to the Subscriber.

Replicated databases in SQL Server

How to recreate publications and subscriptions after replication is already deployed but is dropped, for example to allow changes to be made in iBase to the schema of an iBase database in which data is already replicated. It is important to establish a backup policy for the iBase connection files and the replicated SQL Server databases.

Disabling replication

You must inform the iBase administrator before you drop any publications and subscriptions for an iBase database, especially the databases that contain the entity and link data. In the case of this

© N. Harris Computer Corporation

database, the iBase administrator must to review any current conflicts, check any changes that arise from this review are replicated to the Subscribers, and attempt to get all users out of the iBase system.

Before you begin

When you are ready to drop the publications and subscriptions:

Procedure

1. Check that all Subscribers are online and connected to the Publisher.
2. Log out any users who are still accessing the databases.
3. Check that no errors are reported for the Merge Agents which might prevent the iBase administrator from completing the preparation of the iBase databases.
4. You might want to check that there are no outstanding conflict records. Any records in the conflict tables are deleted when you drop the publication.

Recreating publications and subscriptions

How to recreate publications and subscriptions in a system where the iBase database was previously enabled for replication and the previous publications and subscriptions are dropped.

About this task

Creating publications for the database that contains the entity and link data is the same regardless of whether the database was previously enabled for replication or not. For detailed information, see Publishing Entity and Link Data.

When recreating subscriptions, you should not need to generate a snapshot that contains the data and schema because the databases should already have all the data. The iBase administrator should have denied database access to all users until replication is reconfigured. The databases have the complete database schema--- if there were any schema changes, the iBase administrators applied the changes by using the appropriate iBase utility.

Procedure

1. Start a new subscription of the required type. The subscription should be as follows:

Option	Description
Publication	The database that contains the iBase entity and link data at the Publisher.
Subscription database	The existing entity and link database at the Subscriber.

Select the following options for the subscription:

- The Subscriber has the schema and data.
- The Merge Agent should continuously check for updates.
- Use the Publisher as a proxy for the Subscriber when resolving conflicts (described as First to Publisher Wins).
- The subscription is a client subscription type

© N. Harris Computer Corporation

2. Test that replication is working correctly for the new subscription. Use one of the user-defined tables:
   a. At the Publisher, in one of the user tables (table names do not start with an underscore), change the data in one of the columns for a row. For a new database, carry out this test on the `_GlobalConfiguration` table and change the version number but take care to change it back to its original value.
   b. After the change is replicated, check the corresponding table at the Subscriber.
   c. At the Subscriber, change the column back to its original value.
   d. After the change is replicated, check the table at the Publisher. If the Subscriber is supposed to be read-only, and the change is merged at the Publisher, check the setting of the Merge Agent.
   e. If the Subscriber is read-only, change the data at the Publisher back to its initial value.
3. Set the Merge Agent for the Subscriber to run at the required frequency, for example a polling interval of 1 second. For more information, see Merge agent for the entity and link database.
4. Validate the subscriptions to check that there are no discrepancies between the data at the Publisher and Subscribers. For example, use the `Validate All Subscriptions` dialog, selecting the option to use both the checksum and rowcount validation methods. For more information, see the Microsoft SQL Server documentation.
5. Back up the databases. For more information, see:
   - Configuring and maintaining databases
   - Back up and restore replicated databases
   - Microsoft SQL Server documentation for information on backing up the replicated SQL Server databases
6. Tell the iBase administrator that the security data at the Subscriber is configured for replication.

Back up and restore replicated databases

It is important to establish a backup policy that covers all the elements of a replicated iBase database.

The following artifacts must be completed at all sites:

- The iBase security connection file (`ids` file)
- The iBase database connection file (`idb` file)
- The iBase database and application folders - the locations of database templates, Analyst's Notebook templates, icons

Then, depending on the type of site, the backup can also include the:

- Distribution database
- Publication databases
- Subscription databases
- SQL Server system databases (the primary, msdb, and model databases)
- SQL Server replication scripts

For information about backing up SQL Server databases, see the Microsoft™ SQL Server documentation.

© N. Harris Computer Corporation

Setting the retention period for conflicts

iBase has its own Conflict Viewer in which iBase users, in the publication database holding the entity and link data, review the conflicts automatically resolved by the default SQL Server Conflict Resolver. iBase users check how each conflict was resolved and can choose to keep the losing record.

You might want to check how frequently the iBase administrator reviews the resolved conflicts to determine the length of the conflict retention period set for the entity and link publication.

This does not apply to the database holding the audit or security data.

Setting up replication alerts

You can enable predefined replication alerts to notify users, such as the iBase administrators at the different sites, if a Merge Agent fails because of a problem with a communications link.

The iBase administrator might want to know whether a database is not receiving and sending updates to restrict the type of work done in that database. For example, the might want to postpone specific tasks that increase the risk that conflicts occur.

Note: When you configure alerts, consider how communication links are used in your organization. If replication uses a dedicated communications link, then you might be able to send messages through email otherwise it might be preferable to send messages to a pager.

Creating a database index

The Index Service Configuration tool indexes one or more i2® iBase databases. You use this tool to enable the databases for Search 360.

For each database, you can set up an index service and a job schedule. Each time the job runs, it starts the index service, and obtains the location of the iBase database to index from the index database (IBaseIndexDB). No sensitive information is stored in this database as the index service uses Windows™ authentication to connect to the SQL Server database.

To use Index Service Configuration tool, you require:

- An SQL Server installation on the machine on which you want to create the iBase index database. The tool can only connect to a local instance. The iBase databases to be indexed can either be on the local machine or elsewhere on the network.
- An SQL Server login with suitable permissions to allow databases and jobs to be created.
- For each index service, a suitable Windows account or a proxy account

Installing the Index Service configuration tools

When you are installing iBase with a custom setup, you can select the Server feature to install the following in the iBase installation folder: a command-line application, `i2.iBase.SearchIndexerExe.exe`, and a configuration tool, `i2.iBase.SearchIndexerConfig.exe`.

Running the Search Index Configuration tool for the first time creates a database, called IBaseIndexDB, and a file `Searching Config.xml` in the program data folder, specifically: `C:\ProgramData\i2\i2 iBase <n>\<language>\Searching`

If necessary, the database name is appended with a number to make this name unique.

Note: The database and configuration file should be included in your backup schedule. If either file is lost, you need to reconfigure your iBase databases for indexing.

© N. Harris Computer Corporation

SQL Server logins and Windows accounts

The SQL Server login, or Windows™ account, you use to run the Index Service Configuration tool requires the following permissions.

When the index service configuration tool is started for the first time to create the IBaseIndexDB database:

- `dbcreator` server role

To create the IBaseIndexDB database table:

- `db_owner` database role on the `IBaseIndexDB` database

To detect the IBaseIndexDB database table:

- `db_datareader` on the `IBaseIndexDB` database

To add new records into the table:

- `db_datawriter` on the `IBaseIndexDB` database

To schedule an index service job:

- `db_datareader` on the actual iBase database
- `SELECT, INSERT, UPDATE` on `_Configuration_Text` on the actual iBase database
- `SELECT, INSERT, UPDATE` on `_Configuration_Def` on the actual iBase database

To create a job:

- `db_datawriter` on the `IBaseIndexDB` database
- `SELECT` on `sysjobs` within the `msdb` database
- `EXECUTE` on `sp_add_job` within the `msdb` database
- `EXECUTE` on `sp_add_jobstep` within the `msdb` database
- `EXECUTE` on `sp_add_jobschedule` within the `msdb` database
- `EXECUTE` on `sp_add_jobserver` within the `msdb` database

To delete a schedule:

- `EXECUTE` on `sp_help_job` within the `msdb` database
- `EXECUTE` on `sp_help_jobstep` within the `msdb` database
- `EXECUTE` on `sp_help_jobschedule` within the `msdb` database
- `EXECUTE` on `sp_help_jobserver` within the msdb database
- `EXECUTE` on `sp_delete_job` within the msdb database

Logging on

You can only connect to a local SQL Server instance. The iBase index database is created in the data folder for this instance.

Procedure

1. If you are not connecting to the default instance, select the SQL Server instance .
2. Log in to the SQL Server instance in the usual way.

© N. Harris Computer Corporation

Configuring an iBase database

The Search index service must have the connection details of the database that need indexing. In addition, the frequency to run the indexing job needs to be set.

Procedure

To set up the Search index service for an iBase database:

1. Click **Add** to display the Configure Database dialog.
2. Enter the name of the server that hosts the iBase database.
3. Enter the SQL Server name of the iBase database.
4. In **run as**, enter the Windows™ account to be used to run the index service.
5. Enter the Database Access Token for the iBase database.

   Note: A copy of the Database Access Token for the current database is available from the iBase Designer Database Properties dialog. System administrators can choose to **Show** or **Copy** the token from the **Database Access Token** tab.
6. Select the frequency at which the index jobs run and then set up the schedule.
7. To start the index service on the date and time that is specified for the first job, turn on the **Enabled** checkbox.

   Note: Unlike most other SQL Server job properties, the enabled flag enables both the job and the schedule.
8. Click **OK** to create the index service. If the **Enabled** checkbox is turned on, a job (`i2_Search_Indexing_<server>_<InstanceName>_<DatabaseName>`) is run at the next scheduled time and starts the index service.
9. Advise the iBase database administrator of the date and time of the first scheduled job as this is not visible in iBase Designer.

   Important: If Search 360 is turned off in iBase Designer, jobs run, but exit without indexing.

Checking that the service is configured successfully

When you complete the steps for creating or modifying an index, ensure that the index is configured successfully.

Information that the index service is configured successfully for a specific database is available in several places:

- In SQL Server Management Studio, SQL Server Agent lists an `i2_Search_Indexing` job for the database (the Job Activity Monitor reports whether the last run was successful).
- In the Windows™ Event Viewer, a started event is listed for the database.
- In iBase Designer, the last date and time that the index was updated is displayed in Search 360 Administration.
- In iBase, the last date and time that the index was updated is displayed in Search 360.

If the service is not running:

- Check the Windows Event Viewer - the service logs when the service starts and stops.
- Try running the Index Service from the command line (type just the command name `i2.iBase.SearchIndexerExe.exe` in the root of the iBase installation directory to display the arguments) for example:

© N. Harris Computer Corporation

```
C:\Program Files\i2 iBase> i2.iBase.SearchIndexerExe.exe
 SearchIndexerExe /iBaseDBName User_guide
```

Managing the index service

Index services that are set up for iBase databases are managed using the Index Service Configuration tool. Using the Index Service Configuration tool rather than modifying the index service directly through SQL Server prevents synchronization issues.

Procedure

- To modify the schedule for the index service, click **Configure**.
- To stop or suspend the index service, click **Configure** and turn off the **Enabled** checkbox.

  Note: This allows iBase users to continue to search but without any updates for new information.
- To remove the index service, for example before moving the database or uninstalling the Index Service Configuration tool, click **Delete**. This deletes the job and prevents the index tool from running.

  Note: The database (IBaseIndexDB) and configuration file (Searching Config.xml) should be included in your backup schedule. Losing either of these files will require you to reconfigure your iBase databases for indexing.

Troubleshooting

If you accidentally delete the iBase index database, you will not delete the jobs but they will not be visible in the Search Index Configuration dialog. To resolve this, you need to either restore the database from backup or reschedule the jobs (which will delete and then recreate the jobs).

Configuring and maintaining databases

You can create and maintain iBase databases, whether in Microsoft™ Access or SQL Server format.

Create databases

You can create different types of iBase database for operational use.

The different types of database include:

- Empty databases without any schema. You must define the schema, or copy and paste it from another database.
- Databases that use the schema of another database (the new database is based on a database template). You can add to the schema, or modify and delete the objects in it.
- Databases that partition their data by case. A database of this type must contain a minimum of one case and the only access to the data is through the case or cases to which the user is assigned. See What is case control?.

The database is either MS Access or SQL Server:

- Microsoft™ Access databases offer most of the database features of iBase and creating each database is simpler. However, they are only suitable for small amounts of data (up to 2 Gb) and small numbers of simultaneous users (up to 5 or 6). For more information, see Before Creating a Database.
- SQL Server databases are more suitable for large databases with large numbers of simultaneous users. They also provide additional features.

© N. Harris Computer Corporation

Before creating a database

Consider the following points, at least before you create your first operational database, and then review your decisions as you create other databases.

Should the database be SQL Server or Microsoft™ Access?

Microsoft Access databases offer most of the features of iBase and creating each database is simpler. However, over time and with growing size or numbers of databases, you might find that administration becomes difficult.

For personal use, and especially for use with a portable computer, an iBase Access database might be the best choice. You can always upsize any iBase Access database to SQL Server, allowing a straightforward transfer of all data and folder objects to the new format.

In general, the advantages of SQL Server databases make it the preferred choice:

- The advantages include the ability to work with larger databases, more users, better performance with large databases, and a higher level of data security with more flexible access control.
- You need to use one or more of the features specific to SQL Server databases. For a summary of these features, see Comparison of Access and SQL Server Databases.

The different combinations of Microsoft Access and SQL Server databases and security files are summarized in Configuration Options for an iBase System.

System requirements

All iBase installations can use Access databases. Multi-user sites need only a shared disk folder on a suitable server.

If you decide to use SQL Server, you need the following before you create an iBase database:

- SQL Server instance on a server or locally
- Suitable logins on that server

For more information about SQL Server logins, see Access control.

Identifying other database requirements

There are also some standard decisions to make for each database.

- Should the database have Federal Information Processing Standards (FIPS) compliance enabled?

  If you are working in environments that enforce FIPS compliance, you must enable FIPS when you create your database. FIPS enablement cannot be modified later because this changes the encryption algorithms, and existing users are prevented from logging in.

Before you create database records, you should consider the following questions:

- Do you want to identify records in this database uniquely when combined with records from other iBase databases?

  If so, you need to choose a text string, up to 5 characters long, that is unique to this database and that can be guaranteed to remain unique as new databases are created. (This is mandatory for replicated iBase databases.)
- Should the data be read-only to users?

  For example, this state might be appropriate if the database is used only for analysis of historical data collected from other databases. Database administrators can change this setting at any time,

© N. Harris Computer Corporation

but you might prefer to make such a database read-only from the time of creation and change it to an editable state only when necessary for a specific task. Only the data is read-only, users (depending on their permissions) can still add, modify, and delete folder objects, such as queries.

- Should the database be partitioned by case?

  Do you want to restrict access to the records in the database on a case-by-case basis? If so, you need to create a case-controlled database. However, this setting cannot be changed at a later date. For details of how case control works, see What is case control?

With the exception of case control and FIPS, most other decisions can be made now or easily modified after the database has been in use for some time. For example:

- What level of auditing is appropriate?

  A low or intermediate level of detail is often a good starting point, because it is easy to modify settings for operational databases.
- Should audit logs contain a cross-reference for records from external data sources?

  If you do not have this need, there is nothing to do now. If you want this functionality, the process is complex and extends across database design, configuration choices, and auditing.

With the answers and information that is prompted by these questions, you are ready to create the database.

Logging on to the correct security file

You must be logged on to the correct security file when you create the database. The new database can only be accessed through this security file. In iBase Designer, the name of the security file is displayed in the second area from the left of the Status Bar at the bottom of the application window.

Note: Each database shares a unique identifier with the security file used when you create the database. You can only use the database with this security file (or with a copy of the security file).

Database templates

You can create a new empty database from a template that is created from an existing iBase database. Creating databases in this way reduces the time that is taken to define core components.

Depending on the type of database, the template contains:

- Entity types, link types, fields, and standard fields
- Pick lists, icon lists, and SCC lists
- Datasheets
- Charting and labeling schemes
- Folder objects such as report definitions and queries
- Mapping configurations
- Common folder objects

The template does not contain anything that relies on the existence of specific records. For example, it does not contain:

- Sets
- Alert definitions
- Database subset definitions

© N. Harris Computer Corporation

- Data Access Control group permissions
- Cases (even if the source database is case-controlled)
- Support files, such as Analyst's Notebook templates
- Access permissions for folder objects (permissions are always set to Public unless you are using iBase database replication)

A template is saved with the file extension `idt`.

It is important to make sure that the template you select for use is up-to-date. It can sometimes be difficult to change the schema of a database that is in constant use or is off-site. A Schema Update utility is available to reduce the time that is taken to apply schema changes. For more information, see Updating Database Schemas.

Templates and database formats

You can create templates from both Microsoft™ Access and SQL Server format databases, and create a database of any format from that template.

However, a Microsoft Access database that is created from a template based on an SQL Server database does not contain any objects that rely on SQL Server. For example:

- Queries containing semantic conditions or distinct counts
- Import specifications and Import Batch specifications

Note: A template that is created from a case-controlled database is also case controlled. You can never create a Microsoft Access database from this type of template.

Where database templates are stored

Templates are stored in either the `Templates` or `WorkgroupTemplates` folder. By default the workgroup folder contains the database templates that are supplied by i2 and the `Templates` folder contains the ones that are created by the user locally.

Database templates are always created in the `Templates` folder. To distribute a database template for general use, you need to copy it to the `WorkgroupTemplates` folder. For more information, see Installation and Application Data Folders for details of paths.

Any user can change the path of their `Templates` folder.

Note: To prevent users from moving the templates folder, change the permissions for the `Settings.xml` file. See Location of Templates, Icons, and other Files for details.

Backing up database templates

Make sure that the folder in which you keep your database templates is included in any backups that are made of the iBase system.

Creating a database template

Database templates hold no data records but that do contain definitions of database objects to allow databases to be created quickly that match frequently used configurations. You can use any database that you can access to create a database template.

Procedure

To create a template from the database:

© N. Harris Computer Corporation

1. In iBase Designer, log on to the security file associated with the database but do not open the database.
2. Select **File** > **New Database Template**.
3. Use the file browser to locate and select your database. You can only create a template from a database associated with the current security file. The name of the security file is displayed in the second area from the left of the status bar at the bottom of the application window.
4. Review the entity and link types listed in the dialog to check that you have selected the correct database.
5. In the **Template Name** box, enter a name. You might also want to include the version number of the template in the file name. For example `Crime_v1_0`.
6. Click **OK** to create the template. Templates are always saved in your folder for user templates.

What to do next

You can now create a new database from this template or use it to update the schema of a copy of the database (from which you created the template).

Creating a Database

Databases can be created in both iBase User and iBase Designer.

Before you begin

Before you start to create a database, check that:

- You are logged on to a suitable security file (see Logging on to the correct security file for details).
- The security file is stored in the correct folder because iBase Designer creates the new database file (`.idb` file) in the same folder as the security file (`.ids` file).

When you are ready to create the database:

Procedure

1. In iBase Designer or iBase User, select **File > New Database**.
2. In the **Name** box, enter a unique name for the database. When you choose the name, consider:
   - Whether the name uniquely identifies the database, not only within your iBase system but also when the database is used with other iBase databases from other organizations, which is possible if maps and Analyst's Notebook® charts are created using data from multiple iBase databases.
   - For SQL Server databases, the name you choose is used to generate the name of the SQL Server database so you might want to discuss the naming convention to use with your SQL Server administrator. For more information, see SQL Server Database Names.
3. From the **Database Type** list, select the file type of the database you want to create:
   - **Microsoft Access**

     Creates a Microsoft™ Access database. Click the **Details** tab to continue.
   - **SQL Server**

     If you have a suitable server available, you can create an SQL Server database. To do this:

     a. For the **Database Type**, select **SQL Server**.

© N. Harris Computer Corporation

b. Enter a **Server** name in the box to use a known server. Only select the **local** option, if available, if the database is for personal use.

c. Choose how your computer connects to that server, using one of these options:

- If your SQL Server database administrator has given you a login name and password for SQL Server, type these items in the **Login Name** and **Password** boxes. Each iBase user connects to the server using this login.
- Turn on the **Use Windows Authentication** checkbox if you want to use integrated security, where SQL Server accepts the fact that a user has logged on to a Windows™ domain as sufficient permission to connect to the server. If you choose this option, the SQL server login entered above is never used, and each user that attempts to connect to use the iBase database is validated by the server using their network credentials.

The different methods of authenticating a connection are described in detail in Authenticating Connections to SQL Server.

d. Click the **Details** tab to display the Details page.

4. On the Details page, add a **Title** for your database. The title will appear in the title bar of the application window when the database is open in iBase.
5. **Optional:** Enter a **Description** of the database. You might want to enter a brief description that is seen by users opening the database. You might also want to record the name of its database template and the version number of the schema or template.
6. Set the **Audit Level** which you want to log changes.

   Level 1 means that the audit log collects the lowest level of detail, and level 5 the highest. (If you are creating an Microsoft Access database, the highest setting is 4.) Level 4 and higher collect large amounts of data about user activities so you should use these levels with care, and monitor the size of the log file as the database is used.

   You have now set enough properties to create a blank database.
7. Depending on your requirements:

   - Click **OK** to close the dialog and create the database now.
   - Specify a template for the database. For more information, see Creating a database from a template.
   - Set advanced properties for your database. For more information, see Setting advanced properties.

What to do next

The final step is to control the type of access allowed to the database folder and its files. By default, Windows users do not have sufficient permissions to log on and open the new database.

Creating a database from a template

Database templates contain standard components. Creating a database from a template reduces the time that is taken, and ensures that databases for a specific task are created consistently.

Procedure

To create a new database from a template:

1. Ensure that you are logged into iBase, but have no databases open.

© N. Harris Computer Corporation

2. Select **File > New Database**.
3. Click the **Template** tab.
4. Select a type from *Blank*, *User* (user created) or *Workgroup* (system created) templates. You have the option to create and delete templates from this tab. You can also click **View** if you want to see the entity types, link types, and fields in a selected template.
5. Click **Use Database** to create a database from an existing template. You can browse to any database accessible from the currently logged in security file. A temporary template is created and given a type of *Database*. This template is automatically deleted if you change to a different template, create a database using the template, or close the New Database dialog.

   Note: You can also create a template from a different database, and use that template instead. For more information, see Creating a template from an existing database.
6. Click the **Configuration** tab, and select the database type.
7. Click the **Details** tab, and enter the name of the database and some information about the purpose of the database or its contents.
8. Click the **Advanced** tab, and enter the details:

   - **Database Identifier**

     Optionally, enter a short string of text in the **Database Identifier** box. Do this if you want to identify entity and link records as belonging to this database. This database identifier is only necessary if you plan to perform operations outside iBase on records taken from different databases.

     Attention: The use of a database identifier has an impact on performance since the database identifier is appended to the record identifier on every record.
   - **Extra Detail Field for Audit Log**

     Type the name of a field (in this database) in the **Extra Detail Field for Audit Log** box if you want the audit log to record the value of this field when recording actions that affect records.
   - **Soft Delete**

     Turn on the **Soft Delete** checkbox if you want to use a two-stage process for deleting records. With **Soft Delete** turned off, all delete operations take place immediately. If the **Soft Delete** checkbox is turned on, all **Delete** commands mark records for deletion and make those records unavailable for most analysis, but do not delete the records.
   - **Read Only**

     Turn on the **Read Only** checkbox if you want to make the entire database read-only, and prevent any changes to records. Users can still create sets, queries, and other folder objects.
   - **Security Classification Codes / Case Control**

     Determines whether the database uses Standard Security Classifications or restricts information based on specific cases. If you select **Standard (SCC)**, you can additionally opt to **Restrict SCC lists to accessible items only**. Turn on this option to restrict any lists of Security Classification Codes to accessible ones only. This setting applies when you add or edit a record that includes an SCC list.
   - **First Day of Week**

     Displays the first day of the week as set for this database. This defaults to <System>, which is Sunday for Microsoft Access databases. For SQL databases, this is derived from the current locale as set on your machine or via the locale ID of the SQL Server machine.

© N. Harris Computer Corporation

You should only need to change this if the locale on the SQL Server machine is different to your local machine or you are working with statistics and you want your week to start on a different day.

Note: The start day of the week may affect calculations on dates and date parts.

9. Click **OK** to create the database with the settings you have chosen.

Setting advanced properties

When you create a database, you can set certain advanced properties. The following information describes the properties that need to be set before data is entered.

About this task

Database Identifier

If you want to identify entity and link records as belonging to this database, enter a short string of text (up to five characters). This is appended to the identifier of each new record; for example, `PER475\GEN` where `GEN` is the database identifier. This identifier is only necessary if you plan to perform operations outside iBase on records taken from different databases or if you use iBase database replication.

Note: If you intend to use this feature, you must enter this string before you create any records in the database. The field remains editable after database creation and the addition of records, but any change you make will mean that records created before and after the change will have different database identifiers.

Attention: The use of a database identifier has an impact on performance since the database identifier is appended to the record identifier of every record.

Extra Detail Field for Audit Log

If you want the audit log to record the value of a particular field when recording actions that affect individual records, enter the name of a field (from this iBase database).

The audit log always records the iBase Record ID so this extra recorded field is a free choice from data entered in iBase or imported from another database. Typically, the database designer sets up the schema so that the named field or standard field contains an external reference number or some other way of assessing the history or validity of a record.

For example, this feature can be used to maintain an audit log with details of data and record identifiers imported from external databases.

Standard (SCC) Control

Leave the **Standard (SCC) Control** option selected. This gives each user access to all the records in the database, depending on their user permissions. For details of creating a database that is partitioned by case, see Creating a Case-Controlled Database.

Other properties can be set, or changed with caution, at any time during the life of the database. For more information about all of these properties, see Summary of the Database Properties.

Procedure

To set advanced database properties:

1. Click the **Advanced** tab to display the Advanced page.
2. Select the properties that you would like to use.

© N. Harris Computer Corporation

3. Click **OK** to create the database with the settings you have made.

What to do next

For any open database, you can view the properties by displaying the Database Properties dialog. When viewed in iBase Designer, you can change the Title, Description and, with caution, the settings displayed on the Advanced page.

Summary of the database properties

The properties of the database provide detailed information about the database.

At any time you can view the properties of the database in iBase Designer, by selecting **File** > **Database Properties**.

Title

The title for the database, as displayed in the application title bar.

Description

The description of the database, as displayed when any user first opens the database.

File

The location of the database (`.idb`) file.

Version

The database version number.

Audit Level

The detail level at which the audit log collects data on changes to the database and security file. You can change the audit level: level 1 means that the audit log collects the lowest amount of detail and level 5 collects the highest amount of detail (SQL Server databases only). Level 4 and higher collect large amounts of data about user activities so you should use these levels with care, and monitor the size of the log file.

Audit History

In SQL Server databases only, all updates to data, including code lists, are logged and can be viewed either in Audit Viewer or in the iBase History. In a database that is set to audit level 5, records that are viewed but not updated are also logged.

Note: This property is automatically turned on if the database is initialized for alerting and cannot be turned off while alerting is in use.

Database Logo

You can replace the default database logo with an alternative image. Click **Select...** to browse the available supported file types in your Windows Pictures folder. Select an image no larger than 0.5MB and click **OK**. To remove the image from the database, click **Clear** and **Yes** to confirm. Your image is replaced with the default iBase icon.

Configuration

The configuration page shows details of the database file and format, and the security mode. You can change the authentication mode when connecting to the SQL Server instance on this page or by using the Database Configuration tool (see Managing SQL Server Connection Settings).

© N. Harris Computer Corporation

Database Type

The file format of the database, either Microsoft Access or SQL Server.

Database Name

The name of the SQL Server database on the server. See SQL Server Database Names.

Note: You cannot rename an SQL Server database in iBase Designer. See SQL Server Database Names for further details.

Server

The name of the database server. You can change to a different server provided that the database exists on that server. Enter a name in the field to use a known server. Only select the **(local)** option if the database is for personal use.

Note: This and the following changes do not take effect until you reopen the database.

Login Name, Password

An SQL Server login name and password is displayed if SQL Server authentication is used to secure access to the SQL Server instance. See Authenticating Connections to SQL Server for details.

For security reasons, the login that is used to create the database might be different from the one used after creation. After creation, you might prefer to change the login to one with a lower level of SQL Server permissions. After creation, you might also want to change the authentication mode by turning on the **Use Windows Authentication** checkbox.

Use Windows Authentication

The **Use Windows Authentication** checkbox is turned on if Windows authentication (integrated security) is used to secure access to the SQL Server. Each user that attempts to connect to use the iBase database is validated by the server using their network credentials. See Authenticating Connections to SQL Server for details.

Advanced

The Advanced page displays the current setup of the database, which you can change with caution. See Advanced settings for details.

Database Access Token

Use the Database Access Token page to obtain a copy of the token for the current database. The token is used by some applications, such as the Search 360 Indexer, to connect to SQL Server databases. You can save time by copying the token from this page and pasting it into the application, rather than having to regenerate the token in the Database Configuration tool.

Passwords for Microsoft Access databases

A 20-character password is generated for you when the Microsoft Access database is created. You should keep a record of this password. The password is the same for all the Microsoft Access databases created from the same security file.

If you are using iBase Designer and want to see the password, select **Tools** > **Feature Availability** > **Options** > **Advanced**. This option is not available in iBase User.

© N. Harris Computer Corporation

SQL Server database names

The names that you choose for the security (`ids`) file and database (`idb`) file in iBase are used to generate the names of the SQL Server databases. For this reason, you might want to discuss the naming convention to use with your SQL Server administrator.

Main iBase database

A complete logical iBase database (for entity and link data) contains two Microsoft™ SQL Server databases:

- An iBase database:

  Typically the database name is similar to the name of the connection file, but is subject to modification to comply with SQL Server naming rules.

  The database name always contains an underscore (_). For example, if the requested database name is `Intelligence`, SQL Server uses the name `Intelligence_` and the connection file remains `Intelligence.idb`. Additionally, any spaces in database names are replaced by underscores (_). T
- An Audit Log database:

  The Audit Log database is the database name with _log added at the end, for example `Intelligence__log`. (Notice the double underscore in this single-word database name.)

These two databases are always present.

iBase security database

Optionally, iBase security data can be held in an SQL Server database. The SQL Server name follows the rules for the main iBase database but is appended with _sec. For example, if the name of the Access security file is `Intelligence.ids` then the SQL Server name is `Intelligence__sec`.

Renaming SQL Server databases

To rename an SQL Server database that contains entity and link data (not security data), create a new database in iBase Designer with the wanted name. The name must uniquely identify the database within your iBase system and also when used with third-party iBase databases. You must be logged on to the correct security file when you do rename a database. The connection file that is required by iBase to connect to the database on the server is also created. To move the data to the new database, your SQL Server administrator must make a backup of the SQL Server database that you want to rename and then restore the backup over the new database.

Attention: You cannot rename an SQL Server security database in this way. You lose the connection between the security file and the databases that it secures and prevent your users from opening the databases.

Database subsets

A database subset is a portion of records in the database that are copied into a separate database. This collection of records are selected by creating a database subset definition that consists of the results of queries and sets.

You might want to create a database subset for a number of reasons:

Creating an environment that matches your current production environment for testing or training.

© N. Harris Computer Corporation

Adding a smaller amount of real data from a production environment lets you test changes to the database, or train users in as close to the production environment as possible.

Working with a set of data that relates to a specific department or organization.

By creating an environment that only contains specified data allows sanctioned data to be shared.

A database subset can be created from a query at any time, unlike the information in a case, that is assigned as the data is added.

To create a database subset:

1. Define the records to include using a subset definition.
2. Create the database subset in either Microsoft™ Access or SQL Server.

The database subset can then be used independently, and if required, you can synchronize any changes with the original database.

Creating a database subset definition

The records in a database subset are selected by creating a database subset definition. When you have created the definition, you can use it to export the data you selected as XML, or you can create a database containing the selected records.

Procedure

To define the records in a database subset:

1. Log on as a user with permission to add folder objects, and open the database.
2. Select **File** > **Data** > **Database Subsets** > **Database Subset Definitions**.
3. Click **New**.
4. Select the records by adding queries and sets to the definition. The queries and sets form a part of the definition and deleting any of these sets or queries, as opposed to just removing them from the definition, invalidates the definition and any database subsets created from it.

   Note: If the subset definition is being used to create database subsets in Microsoft Access, you can use parameterized queries and the values required to run these queries are entered when the database subset is created (or synchronized). If you include parameterized queries, then you must enter values for them when creating database subsets (and when synchronizing). Advanced subsets cannot be created using subset definitions that include parameterized queries.
5. Click **Save** to save the definition.

To create a database subset from your definition:

1. Select the type of database storage to use for your subset:
   - To create a subset in a Microsoft Access database, select **Create Subset**, and follow the instructions in Creating database subsets (Microsoft Access).
   - To create a subset in a Microsoft SQL Server database, select **Create Advanced Subset**, and follow the instructions in Create advanced database subsets (SQL Server).

Results

The database subset definition is created.

At any stage, you can:

© N. Harris Computer Corporation

- Change the definition by adding new sets and queries or by removing them (during synchronization the database subset will be re-created).
- Rename and move the sets and queries that are listed in the definition (this updates the definition).
- Rename the definition.
- Move the definition to a different folder.

You can also delete the definition if it is:

- No longer required to create new database subsets.
- No longer required to update database subsets at the end of synchronization.

Creating database subsets (Microsoft Access)

You can create a database subset from the records that are included in the results of running queries or sets that are specified in a database subset definition. If you use the **Create Database subset** option, the subset database will be in Microsoft Access format.

Before you begin

Before you can create a database subset, you need to specify the records that you want to copy to the new database by creating a database subset definition.

Note: Only database administrators can initialize the database for database subsets.

Procedure

To create a database subset:

1. Log on as a user that has the Database Creator role.
2. Open the database from which you want to create the database subset.
3. Select **File** > **Data** > **Database Subsets** > **Create Database Subset**.
4. In the **Identifier** box, enter a unique ID for the database subset. The ID is up to five alphanumeric characters long. Previously-used identifiers are listed in the **Utilized Identifiers** list.
5. In the **Name** box, enter a name that will be used for both the subset security file and subset database.
6. A new user account with system administrator permissions will be created in the subset security file. Enter the username and password for this account. This account will be used to synchronize the database subset with the main database and to log on to the database subset if no other user accounts are added to the security file.

   Note: Any records added to the database subset will have this user as their "Create User". You may therefore want to select a username that will be meaningful once these records are uploaded to the main database.
7. In **Destination folder**, browse to the folder where you want to create the subset security file and database. You can create a new folder if you have sufficient Windows permissions. The folder you use can contain only one iBase database and security file.
8. In **Subset Definition**, browse for the definition that defines the data to be copied to the new database. At this stage, it is not possible to know whether the definition is still valid or whether the total number of records exceeds 50,000 (the maximum allowed records).
9. Click **Create** to continue. You will be warned if the definition is invalid because it contains deleted queries or sets, or if the total number of records exceeds the 50,000 record limit.

© N. Harris Computer Corporation

10. Click **OK** to create the database subset. If the definition contains any parametrized queries then you will be prompted for the values. You can click **Cancel** but doing so will also cancel the creation of the database subset.

Synchronizing database subsets

Database subsets are used remotely, and the records they contain must be synchronized with the main database regularly. Although in most cases, records are modified either in the subset or in the main database, you might need to resolve conflicts that arise.

Before you begin

You must connect to that database as a system administrator of the database subset to ensure that you have access to all records in the database subset and the necessary permissions.

During synchronization, you can choose whether the database subset expires after synchronization is complete.

About this task

Synchronization begins by identifying the records that are needed to repopulate the database subset by examining the queries and sets in the database subset definition. If the definition comprises any parameterized queries, then you are prompted for the parameter values. If you cancel entry of these values, then synchronization is also canceled. This step is not necessary for database subsets that are set to expire.

The main database is then updated in three phases:

1. All newly created records in the database subset are added to the main database, with the same record identifiers, create date and time, and the same create user.
2. Any updated records in the database subset are copied to the main database:
   - An update to a database subset record is applied to the main database if the main database record is unchanged since the last synchronization.
   - If a record is changed in both the main and the subset databases since the last synchronization, then the conflict is resolved by applying the rule that is selected by the user.
   - All soft deleted records in the database subset are ignored. They do not delete the corresponding record in the main database.
3. Finally, either:
   - The database subset is updated with the changes and additions that are made in the main database.
   - Or, if the database subset is set to expire, then all the entity and link records are deleted, and the database subset is set to read only.

Procedure

To synchronize a database subset:

1. Back up the main database if it is an Microsoft™ Access database. This step is unnecessary for SQL Server databases because updates are committed to the main database after each phase of the synchronization process.
2. Log on to iBase as a user with the Database Administrator role, and then open the main database.
3. Select **File > Data > Database Subsets > Synchronize Database Subset**.

© N. Harris Computer Corporation

What to do next

When a conflicting change is made in the same record (in any field in that record) in both the main database and database subset, you can choose to:

- Keep the changes made to all the main records that are in conflict, and discard all the changes that are made to the corresponding subset records
- Keep all the changes made to the subset records that are in conflict, and discard all the changes that are made to the corresponding main records

The user decides without knowing which records are affected or what the conflicts are, and the rule that is selected applies to all records with conflicting changes.

Notes:

- If the main record is deleted, and is changed in the subset, then it is either restored and updated (to match the subset record) or re-created (depending on whether it was soft deleted or purged).
- Restoring or re-creating a link always results in its link ends being restored or re-created if necessary.
- Restoring or re-creating an entity results in its links being restored or re-created; but only for those links where the other end of the link is still active.

Create advanced database subsets (SQL Server)

You can create a database subset from the records that are included in the results of running queries or sets that are specified in a database subset definition. If you use the **Create Advanced Subset** option, the subset database will be in Microsoft SQL Server format.

Before you begin

Before you can create a database subset, you need to specify the records that you want to copy to the new database by creating a database subset definition.

Note: Only database administrators can initialize the database for database subsets.

Procedure

To create an advanced database subset:

1. Log on as a user that has the Database Creator role.
2. Open the database from which you want to create the database subset.
3. Select **File** > **Data** > **Database Subsets** > **Create Advanced Subset**.
4. In the **Name** box, enter a name that will be used for both the subset security file and database subset. This also generates the **Database Name** displayed in the **SQL Server (subset)** section.
5. The subset security file will be generated with the same users as the master database.
6. Enter the Server connection URL in the **Server** box, and enter your database credentials, these can be:
   - An exact duplicate of the credentials used to access the master database.
   - A specified user name and password
   - Windows Authentication

   Tip: Test your connection each time you change the server or the credentials used to access it.

© N. Harris Computer Corporation

7. In **Destination folder**, browse to the folder where you want to create the subset security file and database identifier. You can create a new folder if you have sufficient Windows permissions. The folder you use can contain only one iBase database identifier and security file.
8. In **Subset Definition**, browse for the definition that defines the data to be copied to the new database. At this stage, it is not possible to know whether the definition is valid.
9. Confirm the iBase password that was entered when you logged into the database.
10. Click **Create** to continue. You will be warned if the definition is invalid if it contains parameterized queries, deleted queries or sets, or if the total number of records exceeds the record limit (5 million).
11. Click **OK** to create the database subset.

Advanced synchronize

Synchronizing databases, uploads the data from the database subset to the main database and downloads new and updated records in the subset definition to the database subset. You can update the database subset using the original subset definition or you can select a different subset definition.

About this task

When you synchronize an advanced database subset, the records are compared, any records that have been updated either the main database or the database subset is updated in the other location.

A conflict occurs when an entity or link is changed in both the main database and the database subset. To resolve the conflict, you need to decide which record you want to keep. You can either:

- Discard the subset record changes, keeping the changes to the record in the main database and lose the information in the record from the database subset.
- Keep the subset record changes, keeping the information in the record in the database subset and overwriting the changes in the main database.

If the main record is deleted, then it is:

- Restored and updated to match the subset record if Soft Delete is in use.
- Re-created if the record is deleted or purged.

Restoring or re-creating a link always results in the link ends being restored or re-created if necessary. Restoring or re-creating an entity also restores or re-creates any associated links if the other end of the link is still active.

During synchronization, the following error messages might be displayed:

- The database subset has expired. - You cannot reuse an expired database subset. Re-create it from its database subset definition.
- The database subset has an incompatible schema. - The database subset is invalid because the schema of the main database was changed after the database subset was created. To fix this problem, use the **Database Schema Update** option in iBase Designer.
- The database subset is read-only. - Use iBase Designer to change the database properties of the database subset so that it is no longer read-only. Although you can change the Read-only property in an expired database, you cannot reuse it.
- This is not a valid database subset. - The selected database subset is either not a database subset or it might be a subset of a different database. You can set the database subset to expire if you do not need it any longer. This deletes the contents of the database subset and mark it as read-only. The database subset can never be reused.

© N. Harris Computer Corporation

When you synchronize an advanced database subset with the main database:

- Newly created entities and links in the database subset are added to the main database, with the same record identifier, create date or time, and create user.
- All (soft) deleted records in the database subset are ignored - they have no effect on the main database.
- Records in the main database are updated to match the changes in the database subset if there are no conflicts.
- If a record has changed in both the main database and database subset, since the last synchronization, then conflict resolution is applied. See below for details.

At the end of synchronization, you are informed of the changes made to the main database:

- The number of new records added to the main database.
- The number of records updated in the main database with changes made in the database subset
- If Soft Delete is used: the number of records restored as a result of conflict resolution
- If Soft Delete is not used: the number of records that are re-created as a result of conflict resolution
- The total number of conflicts resolved (at record level)

When synchronization is complete, an updated database subset, re-created using the latest version of the subset definition, is available for reuse in the field. Alternatively, the database subset is set to read-only if the database subset was set to expire.

Procedure

To synchronize an advanced database subset:

1. Log on using a user account that has the Database Administrator role and permission to add records, update records, delete records, and update or delete records created by other users.
2. Open the database from which the database subset was created.
3. Select **File** > **Data** > **Database Subsets** > **Advanced Synchronize (SQL Server)**.
4. Browse for the database subset containing the records that you want to load.
5. Enter the iBase username and password used to access the database subset.

   Note: This user account should also have the Database Administrator role and permission to add records, update records, delete records, and update or delete records created by other users.
6. **Optional:** Use the **Options** to determine whether field attachments and records that have been deleted are included in the synchronization.
7. Click **Next**.
8. Decide how you want to handle any conflicts between changes made in the main database and those made in the database subset. By default, synchronization will never overwrite changes in the main database.

   Note: At this stage it is not possible to know whether there are actually any conflicts.
9. Click **Next**.
10. View the list of changes and use **Ignore Selected** to stop selected records from being updated.
11. **Optional:** Decide whether to update field attachments where they differ using **Include field attachments when repopulating**.
12. **Optional:** To discard the subset after uploading the records to the main database, turn on **The database subset should expire after synchronization**.

© N. Harris Computer Corporation

13. Click **Syncronize**.

Creating case-controlled databases

If you need to restrict access to data on a case by case basis, you must to create a case-controlled database. Because you cannot convert a case-controlled database to a standard (SCC) controlled database, consider carefully whether you require this facility.

Note: You cannot use cases and Security Classification (SC) codes in the same database. You cannot use cases in a replicated database (or if iBase database replication is installed).

What is case control?

All data in a case-controlled database is partitioned by case. Every record belongs to a single case, and each user is assigned to one or more cases. Data cannot be shared between cases - data can only be entered, whether by manual entry or by importing, when a single case is selected. However, records from two or more cases can be analyzed together, for example by running queries and reports.

A record in a case-controlled database:

- Belongs to a single case.
- Might be duplicated across cases, such as a repeated telephone number, address, vehicle, but with distinct case ownership - updating one record does not update the other.
- Can only be edited or deleted when working in a single case.
- Cannot be linked to records in other cases.
- Is always read-only if it belongs to a closed case (but still appears in searches and queries).

A user in a case-controlled database:

- Sees only the records in the cases to which they are assigned.
- Sees all the records to which they have access when they work in multi-case analysis mode.
- Can only update records (either manually or by importing) in a single, open case. When working in a case, all reports, queries, browses, and so on, apply just to the records in that case.

Note: Sets are not specific to any case. A set can contain records from all the cases assigned to the user and, if the access to the set is Public, might also contain records added by other users, from other cases. However, a user only ever sees the records that belong to the current case (or all their assigned cases if working in multi-case analysis mode).

The history of each case can be recorded in the audit log. Actions include: Case Added, Case Closed, Case Deleted, Case Name Changed, Case Reopened.

Defining access to cases

All users, including database administrators, must be assigned to one or more cases before they can open a case-controlled database in iBase.

You do not need to assign system administrators to cases. Any system administrator can, in iBase Designer, add cases and assign users to the cases. They can also assign all the users who are members of a Data Access Control group.

See Creating and Managing Cases in iBase for details of creating cases and assigning users to them.

© N. Harris Computer Corporation

Displaying case names

By default, the case name is not displayed as part of the record - to display the case name in iBase you need to add a case field to each entity and link type. You might prefer to use a standard field for this.

A case field is useful to users working in multi-case mode who might want to know which case a record belongs to. Case fields are less useful to users who work only in one case at a time because the name of the case is displayed in the title bar of the iBase window at all times.

Note: It is not possible to change the value of a case field.

Creating a case-controlled database

You can create a new case-controlled database. This is similar to creating a standard database.

About this task

In the Create New Database dialog:

1.
2.

Procedure

1. In iBase Designer, select **File** > **New Database**.
2. On the **Configuration** page, select SQL Server as the database type.
3. On the **Advanced** page, select **Case Control**. This option is not available in the following situations:
   - If you also select a database template that uses SCC control - you must to create the database first and then convert it to case control later (before any records are entered).
   - If iBase database replication is installed on the machine.

   Note: After you create the database, you are not able to select the **Standard (SCC) Control** option, on the **Advanced** page of the Database Properties, which would allow the use of security classification (SC) codes. It is not possible to create a case-controlled database from a template that uses standard (SCC) control and vice versa. A template always inherits this setting from the database used to create it.

Converting a database to case control

SQL server databases without database replication can be converted to case-controlled databases. This allows copies of the data to be used remotely without connection and synchronized when connection is available.

About this task

Turning on case control is an irreversible process. Ensure that you back up your database before you experiment with this facility on a database that contains data. Where there is no data yet, you might create a template from your database instead of backing up.

After you convert to case control:

- You cannot convert the database back to standard (SCC) control.
- Menu commands and options that apply to security classification (SC) codes and SCC lists are no longer available.

© N. Harris Computer Corporation

- Any records that are classified by using security classification codes are stripped of those codes and assigned to a default case.

Note: You can preserve the classifications represented by security classification codes, for example to place Confidential records in a separate case to other records. Before you turn on case control, you would need to export the Confidential records that you want to allocate to a different case. If required, you can delete those records from the database - this means that the Confidential records will not become part of the default case. After the database is case controlled, import the Confidential records into the required case(s).

Procedure

To convert an existing database to case control:

1. If you have an Access database, convert the database to SQL Server format. For more information, see Upsizing a Database to SQL Server.
2. Select **File** > **Database Properties**.
3. On the Advanced page, select **Case Control** and click **OK**.

   Note: This option is not available if iBase database replication is installed on the machine.
4. Confirm that you want to proceed with the conversion to case control and the removal of existing SC codes and SCC lists.
5. Enter the name of the default case. All existing records, folder objects, and alert definitions will be moved to this case. You can use export and import to move records between cases.
6. If alerting is in use, assign all users who have created alert definitions to the default case. You must do this before you close the database.

   Warning: Alert definitions without owners are deleted when you close the database.

iBase cases

Cases can be created in either iBase or iBase Designer. Cases can be accessed by people who are assigned to the case.

Before a user can open a case-controlled database, you as a system administrator must assign them to one or more cases. When a user opens a case-controlled database, they then select:

- Either, a single case to obtain read/write access to the case, if the case is not closed and that they belong to a user group that grants add, modify, and delete permissions. (Some analysts only ever require read-only access to the data.)
- Or, all their cases (by turning on **Multi-Case Analysis**) to obtain read-only access to all the cases assigned to them.

Regardless of the type of access, a user who selects all their cases, when they open a database, only ever has read-only access to their cases.

When no cases are defined, only administrators with the Database Administrator and Security Administrator roles can open the database in iBase; for example, to create a new case. To add data to the new case, they must select the case by selecting **File** > **Change Case**.

Note: To obtain information on the cases in a database, run a Security Design report.

© N. Harris Computer Corporation

Creating new cases

Cases can be created and updated by any administrator with both Security Administrator and Database Administrator roles.

1. To create a new case:
   - In iBase Designer: Select **New** > **Case** from or in the Database Explorer, right-click on Cases, and select **New**.
   - In iBase: If required, select a single case on opening the database and select **New** > **Case**. Creating a case does not select it. To change to the new case: select **File** > **Change Case**. The name of the current case is shown in the title bar of the application window.
2. Enter the details of the case.
3. Assign users to the case.

Giving and revoking access to cases

You authorize users to work on a case when you create or edit it. You can either assign users one at a time or you can assign all the users who are members of a Data Access Control group.

Note: The Users page lists the users who are assigned individually to the case. If a user has access to a case because they belong to a group that is assigned to that case, then they are not listed on the Users page of the Case.

You do not need to assign system administrators to cases. Administrators (with both the Database Administration and Security Administration roles) always have access to all cases.

New or amended access to a case only takes effect the next time a user opens the database.

When you revoke access to a case, note:

- If a user has access to a case because they belong to a group that is assigned to a case, then you can only revoke access by removing the user from the group.
- If alerting is in use, then the user is removed from any alert definitions that they own. These alert definitions remain active for other users. A system administrator can delete the alert definitions if required.

Listing existing cases

You can view a list of all the cases in the database in two places:

- In iBase Designer: Click **Cases** in the left pane of the Database Explorer to list the cases in the right pane.
- In iBase: Log on with an account with both the Security Administrator and Database Administrator roles and select a database to open.

To obtain information on who is assigned to which case, run a Security Design report, select **Security** > **Security Design Report**.

Modifying cases

In both iBase Designer and iBase, you can modify the description of a case, change who has access to it, and open or close it (see the following sections). You can only change the name of the case in iBase Designer. When you change the name, all the records belonging to the case are updated as well.

In iBase, for users with current sessions, the changes take effect next time they open the database.

© N. Harris Computer Corporation

To modify a case:

- In iBase Designer, right-click on the case in the Database Explorer, and select **Edit**.
- In iBase, select from the **Edit** menu, select `Case`. You can only modify the case that you selected when you opened the database.

Viewing case statistics

You can view statistics by case or by database. To view statistics:

- In iBase Designer, select **File** > **Database Statistics**.
- In iBase, select **File** > **Properties Database Statistics**.

In iBase Designer, record statistics are displayed by case (both open and closed) for the whole database. For example, click the **Entity Types by Case** tab to see the number of entity records by type for each case. The number of entity records for the database is displayed on the Entity Types page.

In iBase, **Database Statistics** display only the statistics for the cases that are accessible to the user. For example:

- A user who selects a single case sees statistics for that single case.
- A user who works with all their cases sees statistics for entities in all their cases on the **Entity Types** page, and a breakdown by case on the **Entity Types by Case** page.
- A system administrator always sees statistics for all the cases in the database, even if they select a single case when opening the database.

You can also print the database statistics with a breakdown by case if required.

Closing and reopening cases

In both iBase Designer and iBase, you can close cases. When you close a case, the closure date is recorded. The closure date is taken from the date set on the user's machine. If required, you can reopen cases, and the closure date is retained until you close it again.

A closed case is read-only and no one can edit the records that belong to the case. However, any user who is assigned to the case can select it when they open the database. When more than one case is selected, operations such as finding and querying includes results from both closed and open cases.

To review the complete history of a case, use the audit log.

To close or reopen one or more cases:

- In iBase Designer: in the right pane of the Database Explorer, right-click on a case, and from the menu, select **Edit**.
- In iBase: log on as an administrator with the Security Administrator and Database Administrator roles, select the case that you want to close or reopen, and select **Edit** > **Case** .

You can then change the status of the case on the **General** page.

Deleting cases

When you delete a case in iBase Designer, you delete all the records that belong to that case, all the entries in the audit log for those records, and all alert definitions.

Before you delete a case:

1. Archive the records in the case. Either by backing up the database, or alternatively, exporting all records to text files.

© N. Harris Computer Corporation

2. Archive the audit log for the case, making sure that you have archives that cover the period of the case.

To delete a case in iBase Designer:

- In the Database Explorer, right-click on the case, and select **Delete**.

Routine maintenance

There are several areas in iBase that require routine maintenance to ensure that your database continues to run correctly. Where possible, you can use tools that are provided in iBase Designer to run maintenance tasks.

The commands for routine database maintenance are available from the **Tools** menu in iBase Designer:

Maintaining database tables and indexes

All databases and security files operate more slowly as deletions and changes increase the fragmentation of the data.

For Microsoft™ Access databases and security files, use the relevant **Tools** > **Database Administration** > **Repair/Compact** option.

For more information about maintaining tables and indexes in SQL Server databases, see Performance Tuning in iBase Designer.

Maintaining search indexes

Depending on the type of search, the method of maintaining the search index varies:

- For Search 360, ensure that:
  - The Index Service is scheduled to run regularly.
  - The `IBaseIndexDB` database and Searching `Config.xml` configuration file are included in your backup schedule.
  - The transaction log is monitored, and cleared when it becomes too large.
- For Word Search indexes, run **Tools** > **Search** > **Word Search Indexing** each time that you want to update the index.
- For Full-Text Search indexes, you can use **Tools** > **Search** > **Full-Text Search Indexing** to set up ongoing updates, either with a regular schedule or in response to changes in the database content. On a less regular basis, you might want to respond to user comments or new types of recorded data by updating the lists of excluded words or synonyms.

For more information, see Setting Up Search.

Check for responsiveness and integrity of the database.

If users report slow performance or recurring errors in normal operation, it might indicate fragmented disk files or some kind of corruption.

In iBase Designer, you can use the commands: **Repair/Compact Database File**, **Schema Integrity Check**, and **Link Integrity Check**. There might also be causes external to the database system, such as other processes that run on the server or client computers or poor network connections.

For more information, see Checking a database.

Managing databases that use Soft Delete

In databases using soft deletion of records, purge or restore records as required. For more information, see Batch delete.

© N. Harris Computer Corporation

Managing databases that use cases

In databases that use cases, add new cases, give and revoke access to cases, and close old cases as required. For more information, see iBase cases.

Monitoring iBase usage

Monitor the following regularly:

Audit logs

Use the external iBase Audit Viewer to monitor usage and identify any repeated events such as failed logons or repeated editing or analysis activities that might indicate user difficulties.

Pick lists

You might find that users are frequently typing to supply alternatives to entries suggested in a pick list. You can add these entries to the pick list if required.

Datasheets

Datasheets provide alternative ways of creating or viewing records. Their effectiveness might need monitoring. For example, you might find that analysts want to view an entity in association with selected fields from a particular type of link and linked entity, or that data entry might be made faster by reordering and regrouping the fields of information.

Select **Tools** > **Datasheet Manager** to start editing or creating datasheets.

Reviewing database design, statistics, and security

At any time, you can use iBase Designer to view, or change, the database properties that are chosen when you created the database, and view data statistics and a database design report. With default access control, all users of iBase can view, but not change, database properties, database statistics, and a database design and statistics report. Select the relevant command from **File** > **Properties**.

The security design report can have several forms, but always lists security groups, users, and their consequent permissions or restrictions. You can choose to include user information if required.

The security design report presents all the information held in the security file to which you are logged on. The report does this first by group, listing the group's properties (if any) and user membership; then by user, listing the accumulated permissions of the user, possibly gained by membership of several groups, and the groups of which the user is a member.

If you have databases open, the report includes the use made of Data Access Control groups in the active database.

Note: The security design report does not include details for the use made of Folder Object Control groups.

Routine maintenance on the database servers

You need to maintain adequate free space on disk for databases, search indexes, audit logs, and any linked documents. This is largely a matter of using tools supplied with Windows™ to monitor both the free space and the size of the files that are growing most quickly to reduce that free space.

The strategy that you adopt for databases might vary from moving old data to archives with iBase batch export, batch delete, or creating new databases to hold current data for each year or other time period. For audit logs, the external iBase Audit Viewer provides a way to view, archive, and delete old audit records.

© N. Harris Computer Corporation

You need to maintain adequate backups of the database, security file, and audit logs. You should schedule backups for a time when no users are using the database. See Backing Up iBase Databases.

Database backup procedures depend upon the type of database, Microsoft Access or SQL Server. Back up each security file frequently, as a complete file. Back up audit logs using the external iBase Audit Viewer to identify data for archiving. For further details, see the Audit Viewer help.

Routine maintenance in SQL Server

For large SQL Server databases, disk operations have a significant effect on the performance of the database. To reduce the amount of data that is read from disk during queries, iBase applies indexes to the data. Over time as data is added to and deleted from the database the indexes become fragmented and larger than they need to be. This reduces performance because more data blocks are read into memory to process a query. Eventually, without corrective action, the result is queries that run many times slower than in a newly indexed database.

You need to maintain the indexes of an SQL Server database. The larger the database, the more benefit comes from regular maintenance of the indexes:

- For databases that use legacy indexing or that are tuned in SQL Server, this is done by your SQL Server administrator.
- For databases that use indexes that are optimized in iBase (and that are not tuned in SQL Server), this can be done in the Performance Tuning dialog in iBase Designer. See Performance Tuning in iBase Designer for details.

The indexes should be rebuilt regularly, at intervals that are determined by your SQL Server administrator who is able to measure the fragmentation of the indexes using tools in Enterprise Manager or Management Studio. For example, a database that is updated with imports that use the Bulk Import method might require reindexing after each bulk import.

Each rebuild takes some time and should be scheduled to take place when the database is not in use.

Batch delete

Users can delete records from the database either individually or in batches. When an entity record is deleted, all links to that entity are also deleted - the link end entities are not deleted.

Deleting records is a permanent and irreversible operation unless soft delete is enabled for your database. When soft delete is enabled, deleted records, whether deleted individually or in batches, are removed from the user's view of the database but you have an opportunity to undo the deletion.

You can use Batch Delete with or without soft delete.

Note: You can deny users access to the **Batch Delete** menu command by using a System Commands Access Control group.

Soft delete

Soft deleted records do not appear in search results or in record lists (when listing and browsing records), but are not removed from the database.

For system administrators, soft deleted records:

- Can be restored using **Restore Deleted Records** in iBase.
- Can be permanently removed using **Purge Deleted Records** in iBase.

If soft delete is not enabled, then deleting records is a permanent and irreversible operation.

© N. Harris Computer Corporation

To check the setting of soft delete:

1. In iBase Designer, select **File** > **Database Properties**.
2. On the Advanced page, view the **Soft Delete** setting.

Batch Delete

To delete batches of records:

- In iBase, select **Edit** > **Batch Delete**.

The deletion can take a while to complete if you are deleting many records. The speed of the deletion depends on several factors:

- Whether Data Access Control is used and is restricting access to the records selected for deletion.
- The format of the database (Microsoft™ Access or SQL Server)
- The audit level if the database is in SQL Server format

The following information describes these factors.

Batch Delete and Data Access Control

The actual records that are deleted depend on whether Data Access Control restricts access to the records selected for deletion.

Consider this example: there are a 1000 telephone entities in the database with many telephone call links between them. A user has permission to view all 1000 telephone records but has restricted access to the telephone call links. In fact, of the 1000 telephone entities, only 200 of them have unrestricted telephone call links. This means that although the user has full access to all the telephone entities, they can only delete the 200 telephone entities with unrestricted telephone call links.

Entity 1 access restriction	**Link access restriction**	**Entity 2 access restriction**	**Delete entity 1 and/or link?**
None	None	None	Yes
None	None	Read-only table	Yes
None	None	Hidden table	No
None	None	Record is restricted (using an SC code)	Yes
None	Any restriction	Any restriction	No
Any restriction	Any restriction	Any restriction	No

Note: Any restriction includes making a table or field read-only, hiding a table or field, or applying a Security Classification (SC) code to deny access to a record.

If Batch Delete encounters a record with restricted access, iBase skips that record. It does not report that it encountered a record that it could not delete. At the end of the operation, it reports on the number of records that it successfully deleted.

© N. Harris Computer Corporation

Batch Delete in Access databases

After batch deletion starts in an Access database, you can press the **Esc** key to stop the deletion but you cannot cancel the deletion of records already deleted. A message is then displayed that tells you how many records have been deleted.

Batch Delete in SQL Server databases

How Batch Delete works in a SQL Server database depends on the audit level of the database. Batch Delete runs fastest with the audit level set to 1, 2 or 3:

Audit level 1, 2 or 3

After batch deletion starts, you can press the **Esc** key to cancel the deletion and, provided that **Esc** is pressed before the deletion finishes, no records are deleted.

Audit level 4 or 5

After batch deletion starts, you can press the **Esc** key to stop the deletion, however you cannot cancel the deletion of records that have already been removed. A message is then displayed that tells you how many records have been deleted. An entry is made in the audit log for each deleted record.

Backing up iBase

Like any critical system, it's important to make and maintain regular backups of your iBase databases. This part of the documentation describes how and when to make backups, and provides guidance on establishing a backup policy.

As well as the database itself, which can be in Microsoft Access or SQL Server format, iBase uses a number of files to store configuration and other data. Backing up iBase requires you to consider all parts of the system.

Database folder

iBase's database folder contains the main database files (for Access) or connection files (for SQL Server). This folder also contains the security file, the log file, and the word search index for Microsoft Access databases.

Shared application data folder

iBase stores files that are shared by all users in Windows' shared folder for application data. Specifically, the folder contains database templates (although the installation can be customized so that workgroup templates are held in a different folder) and icon lists.

Note: Users do not write to this folder, but to their own areas for application data. See Installation and Application Data Folders for details.

SQL Server databases

For an iBase installation that uses SQL Server, you must back up the databases that store the iBase data. These databases include the main database, the security database, and the audit log database.

For more information, see Backing up SQL Server databases.

© N. Harris Computer Corporation

Other information

Every installation of iBase is different, and you might have additional files or databases that you need to back up. For example, for Microsoft Access databases, there might be separate folders for archive log files. For SQL Server databases, there might be separate databases that contain archived data.

Moving and Copying Databases

The procedures used to move and copy databases are slightly different for Microsoft Access and SQL Server databases. The principles of moving a database applies equally to copying a populated database.

An iBase system might contain several related databases, for example copies on laptops of a main database on a central server. In this situation, you might want to define the folder objects that are shared by all the databases before you create or copying the main database. You might also want to consider Database Subsets as an alternative to copying the database.

Note: For greater control over these folder objects, use the separately-licensed Schema Update utility; for details, see Handling updates to the schema and folder objects.

There are several reasons why you might need to move, copy, or rename a database and its security file. Among these reasons are:

- Migration to a different computer or server
- Providing a copy for use at another site or on a laptop

  Warning: Consider using database subsets where only a portion of the records is required.
- Routine backup

When you copy or renaming a database, you should select a name that uniquely identifies it within your iBase system and also when used with third-party iBase databases.

The procedures for moving or copying a database and its security file are different for Microsoft Access and SQL Server databases and are described separately in:

- Moving a Microsoft Access Database or Security File
- Moving an SQL Server Database or Security File

You always need both the Windows permissions to move the files, and the ability to log on in iBase Designer as an iBase Security Administrator. If you are moving an SQL server database, you also require an SQL Server login name and password for connecting to each of the relevant SQL Server instances.

Note: Each database records the location of the security file that protects it. Each database is secured by only one security file but there might be several databases secured by the same security file. There must be only one security file in any one folder. The folder should be shared and referenced by a UNC path.

Handling external files

Databases can make references to, or otherwise use, external files. Many of these files must to be moved or copied with the database:

Hyperlink fields

As file names given in Hyperlink fields within records.

© N. Harris Computer Corporation

For a database with multi-user access across a network, good practice would mean that all such files are held in a shared folder and named in the field using a UNC path, such as: `\\server\sharedfolder\Report99.doc`.

- No action is needed if this is true and the shared folder is accessible to users of the database in its new location.
- If UNC names are not used, copy the files to a corresponding drive letter on the destination computer, as discussed next for a single user database.

For a single user database, it is possible that the files are held on a local disk and named using a drive letter and local folder names, as in this example: `C:\Artwork\House.bmp`.

You must copy these files to a similar location on the destination computer. This might not always be possible if there is a conflicting use of drive letters.

Support files

As support files, such as chart and report templates, held in the same disk folder as the database `.idb` file.

You must copy or move these files if you copy or move the `.idb` file so that the files stay together in one folder.

Audit log files

As a log file with extension `.idl`, only present for an iBase Microsoft Access database.

You can move this file if you want to maintain a single log file for the database. If you do not move this file, iBase creates a new log file in the new location.

Word Search index

As a Word Search index with extension `.idx`, only present for an iBase Microsoft Access database that uses Word Search.

You do not need to move this file. You can use iBase Designer to create a new index file in the new location.

You must make and restore true binary copies of all files mentioned in this section, using any convenient method supported by Microsoft Windows. If all you do is make copies for backup and occasionally restore from these copies to the original location, there is no special iBase procedure to follow. The procedures for handling external files are the same for both Access and SQL Server databases.

Handling updates to the schema and folder objects

If you have a Schema Update license, you can keep the copies of a main database, for example held on laptops, in step with changes made to the main database. Changes could include the addition of new fields, new pick lists, or changes to folder objects such as import specifications.

To facilitate the maintenance of copy databases on laptops, you can mark the folder objects that you want to be able to update in the future as common folder objects. These objects can then be added to, updated and deleted from the copy databases--- standard folder objects cannot be maintained in this way.

Common folder objects can also be used to facilitate the updating of data in a copy database. For example, you could:

© N. Harris Computer Corporation

1. Add import specifications and an import batch specification to the main database, and export the data from the main database to create import files for use with the import batch specification.
2. Mark the import specifications and import batch specification as common folder objects.
3. Save a template from the main database to give to your laptop users.
4. Each user applies the template to their copy of the database. This adds the specifications to their database.
5. Each user runs the import batch specification to load the new and amended records.

For further information, see Using Common Folder Objects.

After the move

After a database is moved, users must find the new location of any moved files. After users open a moved database, iBase records any change of connection file location in the most recently used (MRU) section of their **File** menu.

What happens in subsequent use depends on the relative positions of the connection and security files:

- If the security and connection files are in the same folder, users see no change from behavior before the move.
- If the security and connection files are in different folders, users see a Security File browser each time that they need to log on and must navigate to the security file. (Where possible, you should always keep the security file and database in the same folder.)

Moving Access databases or security files

Move or rename an Access database or security file.

The necessary iBase database files to move are:

- The security file, with extension `.ids`
- The database file, with extension `.idb`

Moving or renaming the security file

If you move the security file to another location or rename it, you must open each related database in iBase Designer to update the stored location.

What happens when you open the database depends on the location of the security file. The possibilities are:

- If there is a security file in the same folder as the database file, iBase Designer opens that file immediately, even if it is the wrong security file.
- If it is the wrong security file, an error message notifies the user that the database is not associated with the current security file.
- If there is no security file, iBase Designer displays a Security File browser for you to locate the moved security file and click **Open**.

An example message:

```
Incorrect Security File
You normally connect to this database via a different security file
(\\ SERVER\databases\my_security.ids).
Are you sure you want to connect via this security file
(\\SERVER2\databases2\my_security.ids)?
```

© N. Harris Computer Corporation

If you confirm that you want to use the new file, iBase Designer stores the location. You can close the database immediately, or continue working.

If you have other databases that are linked to this security file you can open and close them for update now, without needing to log on each time.

Moving or renaming the database file

After a database file is moved or renamed, there is no need to open an Access database in iBase Designer if the security file is in its original location.

As mentioned for moved security files, you (and users in iBase) need to confirm the location of a security file that is not in the same folder as the database file. However, keep the security file in the same folder as the databases it secures.

Note: Renaming a database file prevents any existing Analyst's Notebook charts from accessing that database.

Note: If your organization creates maps or Analyst's Notebook charts that use data from multiple iBase databases, the name of the database must be unique.

What users see

At their next use of each database, users must find the new location of the moved or renamed file.

After users open a moved database, iBase records any change of database location in the recently used (MRU) section of their **File** menu.

What happens in subsequent use depends on the relative positions of the database and security files:

- If the security and database files are in the same folder, users see no change from behavior before the move.
- If the security and database files are in different folders, users see a Security File browser each time that they need to log on and must locate the security file.

Moving SQL Server databases or security files

For a SQL Server iBase database, moving or renaming the database also involves making changes to its connection and security files. In some cases, iBase Designer can make the changes automatically. In others, you might need to intervene.

An iBase SQL Server database consists of:

- The security file, with extension `.ids`
- The database connection file, with extension `.idb`
- The Microsoft™ SQL Server database that holds the main iBase database
- Optionally, extra Microsoft SQL Server databases that hold the iBase audit log, archived audit logs, and the Search 360 indexes

If the security file has been upsized to SQL Server, the database also consists of:

- The security *connection* file, with extension `.ids`
- The SQL Server database that holds the security data

It is simplest for administrators and users if you keep the security file (or security connection file) in the same folder as the database connection files that it secures. Share the folder and reference it by a UNC path.

© N. Harris Computer Corporation

Note: Do not copy connection files to client workstations. This might compromise the security of your system and adds to the administrative workload. Keep the `.ids` file in the same folder as the `.idb` file, in a central location.

Moving or renaming the security connection file

If you move or rename a SQL Server security connection file, you must open each related database in iBase Designer to update the stored location.

Note: If you move the security and database connection files to the same folder, you can update both locations in one operation by opening the database.

What happens when you open the database depends on the location of the `.ids` file relative to the `.idb` file. The possibilities are:

- If there is a `.ids` file in the same folder as the database connection file, iBase Designer opens that file immediately, even if it is the wrong security file.
- If there is no `.ids` file, a Security File browser is displayed for you to locate the moved security file and click **Open**.

When you open the database, log on as a user with the Security Administrator role. iBase Designer then recognizes that the security file is in a new location, and asks if you want to store that new location in the database.

An example message:

```
Incorrect Security File
You normally connect to this database via a different security file
(\\SERVER\databases\my_security.ids).
Are you sure you want to connect via this security file
(\\SERVER2\databases2\my_security.ids)?
```

If you confirm that you want to use the new file, iBase Designer stores the location. You can close the database immediately, or make any changes that you want.

If you have other database connection files and databases linked to this security file, you can open and close them for update now, without needing to log on each time.

Moving or renaming the database connection file

You can move or rename database connection files, but you should not copy them to client workstations. After you move or rename a database connection file:

- In iBase Designer, open the database connection file in its new location. The new location is stored in the database. There is no accompanying message.

Note: Renaming a database connection file prevents any existing Analyst's Notebook charts from accessing that database.

Note: If your organization builds up maps or Analyst's Notebook charts that use data from multiple iBase databases, the name of the database connection file must be unique.

Moving the SQL Server database

To move a Microsoft SQL Server database to another server:

1. Disable Search 360 and Alerting.
2. Backup the iBase `.idb` and `.ids` files.
3. Backup the iBase database in Microsoft SQL Server Management Studio.

© N. Harris Computer Corporation

4. You should install the versions of iBase and SQL Server on your new server; or upgrade your Production environment before moving servers.
5. Copy the iBase `.idb` and `.ids` files to the new server.
6. Use Microsoft SQL Server Backup and Restore to copy the database from server to server. You must use the same name for the database on the new server as it had on the old server.

   Note: It is possible to rename the database if it is the main database containing the entity and link data. For details, see SQL Server Database Names. You cannot rename the database containing the security data.
7. Use the Database Configuration utility to open the database connection file that connects to the database, and update the connection details for the new server. See Managing SQL Server Connection Settings.

   Note: If the security (`.ids`) file has been upsized, you might need to provide the Database Configuration utility with the *database* password in order to remake the connection with the security database. See the next section for information.
8. Open the database in iBase Designer to confirm that the connection is working correctly.
9. If you are using Bulk Import, Alerting, or Search 360 for this database, you must set up the new server. See:
   - Bulk import details
   - Configuring alerting
   - Setting up Search 360

Note: If you move a full-text indexed database to another server, you must also install the Microsoft Search Service on the new server if you want to continue to use Full-Text Search.

Updating an upsized security file

An upsized security (`.ids`) file stores the location of the SQL Server database that contains the security data. If, after you copy the database to a new server, the `.ids` file no longer has access to the old environment, the Database Configuration Utility displays this message:

```
iBase was unable to connect to the security database. Please enter the
 database password for this security file to confirm that you have security
 administrator permissions.
```

To find the password and open the database on the new server:

1. In the *old* environment, open the `.idb` connection file in iBase Designer.
2. Go to **Tools > Feature availability > Options**.
3. In the **Options** dialog, click the **Advanced** tab and make a note of the value in the **Database Password** field.
4. Return to the new environment and launch the Database Configuration Utility.
5. Open the `.ids` file and enter your administrator credentials. This results in the message requesting the password.
6. Enter the password that you noted in Step 3.
7. Click **OK** to open the security database in your new environment.

© N. Harris Computer Corporation

Database schema updates

Schema changes to an operational database on a server are typically made and tested in a temporary copy of the database before application to the operational database itself. You can use the **Update Database Schema** command in iBase Designer to manage this process, making the changes and then applying them to the other databases by applying a new database template.

This process is only suitable for compatible databases. A compatible database is any database that is created from the same database template or any copy of a database. These databases are compatible because their entity types, link types, fields, and standard fields share underlying table names, column names, and identifiers. For example, you cannot make a database 'compatible' by adding an apparently identical entity type because the entity type might not have the same table ID as the other databases.

A source database becomes incompatible with the other databases if you turn on case control - any action that you take must be repeated in all the related databases. Adding, modifying, or deleting entity types, link types, fields or standard fields does not make it incompatible because these changes can be updated to the target databases by saving a template.

A target database becomes incompatible if there is a conflict between the identifiers in the source and target databases. For example, if you manually add an entity type to the target database that has the same identifier as a different entity type in the source database. It also becomes incompatible with the source database if you turn on case control when the source database is not case-controlled.

Updating the original schema

Elements of a database schema that can be updated:

- Entity types, link types, fields, and standard fields
- Datasheets
- Pick lists, icon lists, and SCC lists
- Common folder objects, such as import specifications, report definitions, queries, charting schemes and so on (but not labeling schemes).

You can add to and edit these items as required.

Attention: Removing entity types, link types, fields or standard fields from the schema of an operational database deletes the data held for those database objects.

Creating a template for a schema update

To create a template that captures the updates to a database schema, including any changes to the common folder objects, create a template from the database that contains the required updates.

You should always test the new template before you apply it to the operational database or any copy databases. To do this, create a copy of the operational database and apply the update template to it (using the steps in the following section). Only when you verify that the database was updated correctly, should you apply these steps to your operational database.

Note: You can also create new databases from this template if required. Any database created from the template contains both the ordinary folder objects and the common folder objects.

Updating the schema of a database from a template

After you create a suitable template, you can apply the new schema to the operational database and to any copies of it. Before you start, make sure that you have:

- A backup of the databases

© N. Harris Computer Corporation

- Permissions to create and delete files in the same folder as the main database `.idb` file

To apply the schema change:

1. In iBase Designer, log on as a database administrator and open the database.
2. From the **Tools** menu, select `Database Design Update Database Schema`. An empty Update Database Schema dialog is displayed.

   Note: You cannot display this dialog if you are a member of a Data Access Control group that denies access to any tables or fields in the database.
3. Select the template that contains the schema changes.

   After you select a template, you can review the entity types, link types, and fields in the template by clicking

   .
4. On the Additions and Modifications page, and the Deletions page, review the changes that are listed. For example, the Additions and Modifications page summarizes the changes made to:
   - Entity types and their fields
   - Link types and their fields
   - Standard fields
   - Datasheets
   - Pick lists, icon lists, and SCC lists
   - Common folder objects (listed separately for each type of folder object)
   - Semantic Type Library (but specific changes are not listed)
5. If required, click

   to save a list of the schema changes in a file that you can print later.
6. Click **Update** when you are ready to apply the changes. When this is finished, you are warned if any folder objects were renamed because they have the same name as a common folder object in the template.

Common folder objects

You can simplify the administration of several common databases, by defining a core set of folder objects (common folder objects).

Common folder objects across all the databases:

- Have identical names
- Are in the same categories
- Have an identical definition
- Are set to Public access (unless you are using iBase database replication in which case the original access setting on the folder object is preserved)

Any authorized user can define folder objects as common items.

© N. Harris Computer Corporation

There is otherwise no visible difference between an ordinary folder object and a common folder object. For this reason, you might want to use a naming convention for common folder objects or keep them in a specific category.

How common folder objects are updated

In order for the Common Folder Objects option to be available in the iBase **Tools** menu, you need to have the Schema Update Option installed. You can modify your iBase installation in the usual way from the Windows Control Panel. From the Custom Setup page in the installation wizard, select the Schema Update Option, under Extended Features.

Common folder objects are updated by running the `Schema Update` command in iBase Designer. This command applies changes held in a database template to the schema of the database in which it is run.

When a folder object, such as a report definition or a charting scheme, is defined as a common folder object, it can be:

- Added to databases that do not already contain it
- Updated with the changes held in a database template
- Removed from a database if it exists in the database but not in the template

Ordinary folder objects remain unchanged (but are renamed if they have the same name as a common folder object).

To update a compatible database with the current folder objects, create a template from the database containing the folder objects, and then apply that template to the other database. For more information, see Updating Database Schemas.

Defining a common folder object

To define an existing folder object as a common folder object:

- From the **Tools** menu in iBase, select `Common Folder Objects`. The Common Folder Objects dialog is displayed. Click **Help** in the dialog for information on how to use the dialog.
- Dependent on a set. Being data-dependent, sets cannot be saved in a template.
- Dependent on a folder object that is not selected as a common folder object (or that is deleted).

A folder object cannot be defined as a common folder object if it is:

The settings that are made in the Common Folder Objects dialog are saved in the database. Redisplaying the dialog displays the common folder objects defined in the database.

Any template that is saved from the database, distinguishes between ordinary and common folder objects.

Effect of adding, modifying, and removing common folder objects

What happens when you define a new folder object as a common folder object in the source database on folder objects in the target database is summarized below:

- In the source database, add a folder object and define it as a common folder object

  An identical common folder object is added to the target database. If any ordinary folder object with the same name exists, then the object is not overwritten but it is renamed by adding an underscore to the beginning of the name.

  Note: The access permission is not copied, unless you are using iBase database replication.

© N. Harris Computer Corporation

- In the source database, modify a common folder object

  The common folder object in the target database is updated to match the definition in the source database, including any updates to the name or category. If the common folder object was renamed in the source database, then any ordinary folder object in the target database with the same name is not overwritten. It is renamed by adding an underscore to the beginning of the name.

  Note: The access permission is not copied, unless you are using iBase database replication.
- In the source database, make a common folder object into an ordinary folder object

  The common folder object is deleted from the target database.
- In the source database, delete a common folder object

  The common folder object is deleted from the target database.

Note: You are informed if any name changes are made during the update process. The renamed folder objects are identifiable as they appear at the top of any lists (because of the underscore prefix).

Checking a database

You can check a database after upsizing, or after large changes or prolonged editing, or at any time that you suspect problems.

There are several ways to check a database and, if necessary, repair any inefficiencies or errors found. In approximate order of use, you should use these commands:

- **Database Statistics** (or **Database Design Report**)
- **Repair/Compact Database File**
- **Schema Integrity Check**
- **Link Integrity Check**

Except for **Database Statistics** (or **Database Design Report**), all these methods work with a database that is not open in iBase Designer.

Reviewing the database statistics

Use the **Database Statistics** command to review the database statistics. These statistics provide a quick way of seeing how many entity and link records a database contains.

About this task

One statistics report is only a count of the records, but seeing reports with identical counts, before and after any conversion process, provides a quick confidence check that all data is transferred to the new database.

Procedure

1. Select **File** > **Database Statistics**.
2. **Optional:** If you want to make a paper record for later reference, click **Print**.

Repairing and compacting the database file

Use the **Repair/Compact** command to reduce the fragmentation of tables in a Microsoft™ Access database where many records have been changed or removed. This command removes only the space

© N. Harris Computer Corporation

that is marked as being unused in the database. For SQL Server databases, it only compacts the connection file.

About this task

You must be logged on to the relevant security file but have the database closed.

Procedure

1. In iBase Designer, select **Tools** > **Database Administration** > **Repair/Compact** > **Database File**.
2. Check that the result is success.

What to do next

Whether or not you see a successful report, complete the other checks described next:

- Checking the integrity of the schema
- Checking the integrity of the links

Checking the integrity of the schema

Use the Schema Integrity Check wizard to check the integrity of the database schema, that is, whether the database structure follows the rules set up in the database design.

About this task

The wizard reports any problems found and offers to fix those that it can repair. Some repairs can involve additions to the schema to make them consistent. For example, the wizard might add indexes or fields (that can be blank or use the default field value), or change the size of fields. You can choose to abandon repairs at any point up to final approval, allowing you to assess what the repairs would mean. Eventually, you must repair problems to avoid the possibility of prolonged and greater errors.

Note: Schema Integrity Check is unable to check or repair the indexes if you have run the Performance Tuning wizard but not yet completed the reindexing process. For further information, see Performance Tuning in iBase Designer.

Procedure

To use this wizard:

1. Log on to the relevant security file as a database administrator but do not open the database.
2. From the **Tools** menu in iBase Designer, select **Database Administration Schema Integrity Check**. A list of database files is displayed.
3. Select a database from the list. If necessary, select **More Files** to display a file browser where you can locate the database.
4. Click **Next**. The next page displays a list of all entities and links, together with details of system and user data for each.

   Note: You can expand the list into a tree by clicking the + signs. At first viewing, the **All** option is selected; showing you all entities, whether or not they have errors. (Any entries without a tick or check mark to the right of the checkbox have an error.)
5. Select the **Errors** option to see only problems. In a properly functioning database, the list for Errors should be empty. If there are errors, you see a message below the list as you move the mouse

© N. Harris Computer Corporation

pointer over the errors with an X in a red circle. Typical messages include: `Incorrect schema size and Index missing`.

6. If there are errors, turn on the checkbox for each of the errors that you want the wizard to repair.
7. Click **Next**. The next page displays a list of all errors that you have selected for repair, and the corresponding repairs that the wizard will perform. You can click **Back** to alter your selection or **Cancel** to abandon all changes.
8. Click **Finish** to perform the listed repairs, if any.
9. The wizard performs the repairs and then displays **Close**. Click **Close**.

What to do next

The database is opened, whether or not you asked for any repairs. You might want to check the integrity of links.

Checking the integrity of the links

Use the Link Integrity Check wizard to check the integrity of the link records for a database, that is, whether the data held for links is consistent with that held for the entity records at the ends of the links. You should check the integrity of the database schema before you check link integrity.

About this task

In a properly functioning database, the pages of the Link Integrity Check wizard should be empty. However, issues can occur and the wizard reports any problems found with links, or the entities they reference, and offers some repair options. Most repairs are safe and non-destructive, but some repairs might involve removing invalid data. You can use the list of proposed repairs to further inspect suspect data and perhaps recover it by other means. After repair, you should look at places where the wizard has added entities and links, possibly with blank mandatory fields, and decide how to make these records usable. Eventually, you must repair problems to avoid the possibility of misleading analysis based on faulty data.

Procedure

To use the Link Integrity Check wizard:

1. Log on to the relevant security file as a database administrator but do not open the database.
2. In iBase Designer, select **Tools > Database Administration > Link Integrity Check**. The Link Integrity Check wizard opens.

Wizard Page	Action
Step 1	Select a database from the list and click **Next**.
Step 2	Any link IDs listed on this page are missing one of the two link records that hold direction information. You can select list items to view more information that is displayed in the other panes on this page. All listed items are checked for repair by default, you can deselect any record that you do not want the wizard to repair.

© N. Harris Computer Corporation

Wizard Page	Action
Step 3	Any link IDs listed on this page are missing the description records that hold their link attributes. All checked items are repaired with blank data, which is the only possible repair. Deselect any record that you do not want the wizard to repair.
Step 4	Any link IDs listed on this page are missing both of the required records, and are selected by default to be deleted. This is the only repair available. You can choose to uncheck any link IDs that you do not want the wizard to delete, make a note of the details and fix the problem by other means.
Step 5	The entity IDs listed on this page cannot be found in the database, although they should be a link end record. The wizard can repair any missing entity records with blank data, which is the only possible repair.
Step 6	Links should only represent the relationship between two end records. Any link IDs listed here appear to connnect more than two ends, and cannot be repaired. Make a note of these issues for further investigation and fix the problem by other means.
Step 7	Any link IDs listed on this page have inconsistent record status values across the link ends table, and the link's own table. For consistency, the wizard addresses the problem by replacing the values in the link ends table with the value in the link table. All link IDs are checked for repair by default, they can be unchecked if required.
Step 8	You can review the list of fixes that you selected in the previous steps, and take the appropriate actions.

3. Click:
   - **Cancel** To abandon all repairs.
   - **Back** If you want to select a different set of repairs in earlier pages of the wizard.
   - **Finish** To perform the listed repairs.
4. Click **Close**. The database is opened, whether or not you asked for any repairs. What you do next depends on whether you repaired errors:
   - If the wizard did not report errors, the database is ready for use.

© N. Harris Computer Corporation

- If the wizard reported errors and you chose to repair them, close the database and run the wizard again. (Some errors are only revealed after the wizard has made its first repairs.)

5. Run the wizard again until you see no errors. After two uses with repairs performed, the third use of the wizard should always be error free.

What to do next

After repairs, you might need to add data for any records that are created with blank data fields or to replace removed entities or links, perhaps by importing data from a suitable source.

Managing SQL Server databases

iBase provides the capabilities to store data in Microsoft SQL Server databases and Microsoft Access databases. Microsoft Access should be used as the supporting database only if the number of simultaneous users is five or less. When a database of more than 200 Mb is accessed by a number of users simultaneously then consideration should be given to using SQL Server.

Upgrading an iBase database to SQL Server

You can use iBase Designer to convert a Microsoft Access database to SQL Server format. The upsize process creates an SQL Server database and an `.idb` file that contains the connection details. For details of this process, see Upsizing a Database to SQL Server.

Managing the security of the data in an SQL Server database

iBase supports SQL Server's **Transparent Data Encryption** (TDE) feature. TDE ensures data is encrypted at rest; that is the physical files on the disk where SQL Server stores your data are encrypted. Data in the server's memory or on the network is not encrypted. You can add a layer to protect data in transit over the network with **Transport Layer Security** (TLS). iBase does *not* support the **Always Encrypted** feature, introduced in SQL Server 2016.

For detailed information about configuring the security of the overall system, see Managing access to data and functions.

Populating the SQL Server database

If you need to import very large quantities of data, then you might want to consider using bulk import or XML import. Bulk import makes use of the SQL Server BULK INSERT statement and requires the database and server to be configured before it can be used. For further information, see Overview of Bulk Import.

Attention: You cannot use the general SQL Server tools to populate iBase SQL Server databases. The iBase application must have complete control of the data in the database to ensure the integrity of the entities and the links between them. Any data that is not entered or imported by iBase tools can render the whole database corrupted.

Keeping data safe and available (backup)

This is probably the most complex area of managing a database installation, and iBase with SQL Server is no different. SQL Server provides tools for completing the backups and automating them, although your SQL Server administrator might use other backup tools if the right files are backed up at suitable intervals.

With your SQL Server administrator, you must to decide on your backup regime. This can depend on how the iBase SQL Server databases are populated: for example, whether the database is populated

© N. Harris Computer Corporation

by users entering data continuously or by users importing large sets of data. For further information, see Backing Up iBase Databases.

Note: Perform database backups at a time when no users are using the database. This is because some iBase operations can take place over a relatively long time and affect multiple database records. Examples of such operations are data imports, batch edit, batch delete, merge, or deletion of entities with many links. If the backup was performed during such an operation and the database is subsequently restored from the backup the restore operation restores data on which work was in progress at the time the backup was taken and is therefore potentially in an incomplete state. It is safest if backups are completed when no users are performing operations on the database.

Modifying the database schema

Your SQL Server administrator cannot modify the schema of an iBase SQL Server database using SQL. The schema is part of the structure of iBase, and must remain unchanged to ensure data integrity and the success of future upgrades. The only way that you can modify the schema is to use iBase Designer.

Note: It is possible for an SQL Server administrator to modify the indexes of an iBase database to improve performance in areas such as querying although there is a tool for doing this in iBase Designer, see performance tuning section for details. Completing this step manually needs careful planning, and your SQL Server administrator should keep detailed notes and take SQL scripts of the changes to default indexing. Completing this step manually prevents the use of the iBase Designer Performance Tuning wizard.

Note: Before you modify the indexes, your SQL Server administrator must to stop the Microsoft Search service if it is used to continuously update the Full-Text Search indexes in iBase. Other services, such as alerting, are stopped automatically when you open the database in iBase Designer.

Performance tuning in SQL Server

The performance of an iBase SQL Server database can be maintained by regular reindexing in SQL Server. A decline in performance might become apparent after the database grows larger than possibly 10 to 15 GB, and is most noticeable when you run iBase queries. If you are using a database upgraded from iBase 4, then you might be able to improve the performance of queries by optimizing the database indexes. A tool for doing this is available in iBase Designer, see Performance Tuning for details.

If you are already using query-optimized indexes (which is the case for databases created or upsized in iBase 5) and query performance is still poor, you need to discuss the problem with your SQL Server administrator. Setting aside issues with hardware and network infrastructure, the decline in performance might occur for various reasons in Microsoft SQL Server:

- Frequent data imports caused the data and indexes to become fragmented
- Databases that are set to grow/shrink automatically on the same disk became fragmented
- Inserting, updating, or deleting large amounts of data caused the SQL Server database statistics to become out-of-date

There are a number of steps that an SQL Server administrator can take to address these problems:

- Data and index fragmentation can be addressed by rebuilding or defragmenting the indexes on the database tables. An SQL Server administrator can do this while the database is online but, for the best results, it is preferable to first take the database offline.
- Operating system fragmentation can be resolved by defragmenting the disk files. This can be done by a server administrator rather than by an SQL Server administrator. It also requires the database to be taken offline so that the files can be moved around the physical disk.
- If automatic statistics updating is disabled, an SQL Server administrator can update them manually.

© N. Harris Computer Corporation

Effect of auditing on performance

Standard auditing of updates and deletions has a low impact on performance. However, the read auditing that can be configured as an option for iBase SQL Server databases does have an impact. The design of this auditing is such that only records, which have been displayed, charted, or reported are audited. This means that activities such as finding and querying do not run noticeably slower. Activities that result in a revealing a record, such as charting, can take more time to complete. If you intend to use read auditing extensively, it is possible to configure the Audit log database to write to files on disks with fast write performance (see Server machines for details).

Read audit places a higher load on the network and so network performance is more important when using this option. The read audit logs grow relatively quickly and should be archived regularly.

SQL Server Replication and iBase

See Setting Up iBase Database Replication in SQL Server for details of how to set up SQL Server replication for iBase databases.

For more information on hardware requirements, see SQL Server Clients, Servers and Networks.

You can use iBase installation to work with data in both SQL Server and Microsoft Access database formats. This allows you to work with the scale of data appropriate to your analysis. iBase automatically recognizes the type of database and you can switch between them within an iBase session.

Upsizing a Database to SQL Server

You can upsize (convert) an iBase Microsoft™ Access database to SQL Server format. You must have a backup of the original database if you want continued access to the Microsoft Access version of the database.

Before you begin

Before you can upsize a Microsoft Access database, you need:

- An iBase logon for the original database with at least the Database Administrator role.
- Exclusive access to the database that you are upsizing.
- A backup of the iBase database that you are upsizing, or sufficient space to make a disk copy if you want the upsize process to make a copy for you.
- A printout of the database statistics for the Access database--- you might want to compare these with the statistics of the upsized database.
- The identity (network name) of the server on which Microsoft SQL Server is running.
- The login name and password of an SQL Server user that belongs to the dbcreator server role. For more information, see Managing access to data and functions.
- Sufficient disk space and time to complete the operation.

Note: The upgraded databases use twice the disk space of the original iBase database.

About this task

The upsize process creates an SQL Server database and an `.idb` file, which contains the connection details.

Make sure that you have a backup of the database that you intend to upsize. If this is an operational database, it is a good idea to restore the backup and make sure that you can read the restored version before you complete the upsize.

© N. Harris Computer Corporation

You can complete the upsize from any iBase client machine. For large databases, however if possible, run this iBase Designer session on the server machine to reduce network traffic.

Note: If you are upsizing any database that is likely to exist already on the server, such as the supplied example database `User Guide.idb`, you may need to rename the original database (`.idb`) file to a name expected to be unique on the server. For example, you might rename the database file `User Guide.idb` to `User GuideAB.idb`. After the upsize is completed, rename the database connection file created by the upsize back to its original name to make sure that any report templates work. For example, you would rename the connection file `User GuideAB.idb` back to User `Guide.idb`.

Procedure

1. Start iBase Designer.

   Note: Do not open any database.
2. Select **File** > **Logon**.
3. In the Security File browser dialog, navigate to the folder and select the security file used to secure the database you are upsizing.
4. Click **Open**.
5. When you successfully log on, click **Cancel** in the i2® iBase dialog. You cannot have the database open when upsizing.

   Note: You might want to open the database briefly, to confirm that you have used the correct security file and, perhaps, from the **File** menu to select `Database Statistics` and view or print the information so that you can compare it with statistics for the database after upsizing. Close the database before you continue. You are now ready to upsize the database:
6. From the **Tools** menu in iBase Designer, select **Database Setup** > **Upsize** > **Database to SQL Server**.
7. Select a database from the list. If necessary, select the entry **More Files** and click **Next** to display a file browser where you can locate the database.
8. Name the backup file or, if you do not want a disk backup file, delete the suggested name to leave an empty field. Click **Next** to continue. A backup is created if required.
9. Enter the name of the server or select it from the **Server** list.

   Note: Do not use the aliases (**local**) or '.' because they refer to the client machine when the connection file is opened remotely.
10. Enter the logon details for the SQL Server instance on the server. Use SQL Server authentication for the upsizing, not Windows™ authentication. See Authenticating Connections to SQL Server for details.
11. Click **Next** to continue. Your choices are checked and any problems are reported. For example, if the database exists on the server, you must choose another server, or exit and change the name of the original database, before you restart the process. Provided there are no problems, the settings for the new database are displayed.
12. Check that these settings are what you want and click **Finish** The upsizing process starts and progress is displayed by listing each stage with a time and success or failure.
13. Click **Close**. When the upsize process is complete, the iBase database file is overwritten with a file of the same name and extension. For example, `User Guide.idb` is now a connection file to an SQL Server database, and it is likely to be significantly smaller than before. The new SQL Server database is opened automatically.

© N. Harris Computer Corporation

14. **Optional:** Close the database and change the name of the connection file back to the original database name.
15. Before you use the database, check the database properties to see that the settings in the Configuration and Advanced pages of the Properties dialog are what you expected.
16. If success is reported for all stages of the upsize process, there is no reason to expect problems. It is still wise to check the upsized database as described fully in Checking a Database and summarized here:
    a. Select **Tools** > **Database Administration** > **Schema Integrity Check**. Select the new database and complete each page of the wizard. When you finish the wizard, the database is reopened. Close the database.
    b. Select **Tools** > **Database Administration** > **Link Integrity Check**. Select the new database and complete each page of the wizard. When you finish the wizard, the database is reopened.

What to do next

If you want to use Word Search with the upsized database, you must to rebuild the index. The original index (`.idx`) file is no longer be used by the upsized database. However, it might be required if you plan to allow continued access to the Microsoft Access version of the database.

Note: When users create or upsize to a new SQL database, a Database Access Token is silently generated. Before the Search 360 indexer can be manually run, or a schedule set up, you must change and record the Database Access Token using the database configuration tool. See Managing SQL Server Connection Settings.

Managing SQL Server Connection Settings

You use the Database Configuration utility (iBaseConfig) to manage SQL Server settings held in an iBase connection file (whether a security connection file or a database connection file).

About this task

You can change:

- The name of the server that holds the database
- The server login name and password for all users if SQL Server authentication is used
- The security mechanism that is used: SQL Server authentication or Windows™ authentication (integrated security)
- Database Access Tokens

Typically, you use the Database Configuration utility when you use SQL Server tools to change the server instance or login details for existing databases. For example:

- After you create a database, you can change the SQL Server login that is used by the iBase application to one with fewer permissions.
- After you use backup and restore tools to move a database from one server to another, you can reestablish a connection between iBase and SQL Server.

You can inspect many of these details in the Database Properties dialog within iBase or iBase Designer. The advantage of using the Database Configuration utility is that it displays these settings without opening the database on the server, so that you can specify a different server and test the connection.

© N. Harris Computer Corporation

Note: You must update any copies of the connection files held on other machines. Users are unable to connect to the server if the path or file name is different and see the message: The security file has failed an integrity check. Access is denied.

Procedure

1. In iBase Database Configuration, enter the following details and then click **Next**:

Option	Description
Security File Name	Enter the name of the security (`.ids`) file or the security connection file that secures the database connection file. If you want to change the connection details for a security connection file, leave **Database File Name** blank.
Database File Name	Enter the name of the database connection (`.idb`) file. By entering a database file name, you change the connection details for the database that contains the entity and link data rather than the security data.
User Name, Password	Enter the user name and password of an iBase System Administrator (that is, a member of an iBase database management group with all permissions granted).

   When you click **Next**, the connection file is opened, the connection settings are read, and the database and SQL Server information is displayed.

2. You can change many of the settings, for example if you move the database to another server or want to change the method of login to an existing server. However, you cannot change the database type or database name.

   - **Server**

     Specify the name of the server. You must enter a name that can be seen from network client computers. If you are working on the server computer, this means that you cannot choose (local) or its equivalent presentation as a single period (.).

   - **Login Name, Password**

     After selecting a server, you must choose the authentication method to be used for connection to the SQL Server instance. You can use either SQL Server or Windows authentication.

     To use SQL Server authentication, enter the SQL Server login name and password. You can enter the details of any user who has the appropriate access rights on the server.

   - **Use Windows Authentication**

     To use Windows authentication, turn on the **Use Windows Authentication** checkbox. Each iBase session will log on to the database using the Windows login name with which the user started their Windows session.

   Note: The Database Name box displays the name of the Microsoft™ SQL Server database that the connection file (.idb file) connects to. It is not possible to change this name. This prevents a user

© N. Harris Computer Corporation

from connecting to a database where they do not have access by using a connection file for which they do have access permissions.

Click **Test** to check that the details are valid. When you click **Next** the Database Access Tokens are displayed.

3. To create Database Access Tokens, SQL Server users must have `db_owner` database role and `Alter Any Application Role` permissions on the database.

   If you change a token on a database that has Search 360 enabled, you will receive a notification when you click "Generate". You either need to update the Database Access Token in the Configure Database dialog of the iBase Service Configuration tool, or add the new token to the Search 360 Indexer command line arguments.

4. Click **Save** to update the connection file. A summary of its actions is then displayed. A typical summary looks like this:

```
Test connection succeeded.
Server Name
Server Login Name
Server Login Password
Integrated Security setting
Unicode setting
Security access token
Database access token

Completed.
```

Performance Tuning in iBase Designer

iBase automatically indexes certain system tables when an iBase SQL Server database is created or upsized. It will also index those columns within user-defined tables where the

Before you begin

Running Performance Tuning on an SQL Server database requires `VIEW DEFINITION` permission on the SQL Server database. You need to grant this permission to the user mapped to the SQL Server log in. You can use an SQL script similar to this:

```
GRANT VIEW DEFINITION TO username
```

For example, if users connect to iBase using Windows™ authentication, and the user who is running Performance Tuning is called iBaseAdmin and is a member of the YourDomain domain:

```
GRANT VIEW DEFINITION TO [YourDomain\iBaseAdmin]
```

You should revoke this permission after you run Performance Tuning:

```
REVOKE VIEW DEFINITION TO username
```

In addition, if present remove the Full-Text Search index. It is not possible to run the Performance Turning wizard while a Full-Text Search index exists.

About this task

Query optimized indexing is of significant benefit even if your database has no user-defined indexes. The index rules are used whenever you:

- Create an SQL Server database.
- Upsize an existing database to SQL Server.

© N. Harris Computer Corporation

Note: Upsizing removes any indexes that were created manually in Microsoft™ Access.

Procedure

1. In iBase Designer, select **Tools** > **Database Setup** > **Performance Tuning**.

   Note: If necessary, you can stop the process and resume it later. However, until you complete this process, the database is only partially indexed and some parts of iBase might perform slowly. Also, certain commands such as `Schema Integrity Check` do not display, check, or repair the indexes.

2. You can use the Schema Integrity Check to restore missing indexes on user-defined tables in an SQL Server database. In iBase Designer, log on as a database administrator but do not open the database, and from the **Tools** menu, select **Database Administration** > **Schema Integrity Check**. For information on using this dialog, see Checking the integrity of the schema. If the database uses the original iBase index rules, the command restores the indexes to conform to those rules. If the database uses the query optimized index rules, the command restores the indexes to conform to the query optimized rules.

Scheduling imports and exports

In i2 iBase, you define the data that you want to import or export as one or more import or export specifications. You can group them together in the sequence you want them to run, in import batch or export batch specifications. These batch specifications are run by iBase Scheduler by using a Windows™ service that you configure with the Scheduler Configuration utility.

Using Scheduler, you can run batch imports and exports:

- Immediately
- Once, at a specified time
- At preset intervals (hourly, daily, weekly, monthly)
- Whenever a specific file is updated
- Until an optional cut-off date or time is reached

Before or after you run the batch process, you can run a command prompt program, a custom plug-in, or an operating system command. For example, to move data files to a different folder or to send a notification that the batch ran.

Details of scheduled imports or exports are stored in the Scheduler database. This is monitored by the Scheduler Service, which automatically runs the scheduled batch imports or exports. The Scheduler service runs as a Windows Service, and can be started, stopped, paused, resumed, or disabled through the Services application.

The Scheduler Configuration utility can be installed on any machine on which iBase is already installed; you do not have to run the Scheduler Service on this machine. The Scheduler Service must also be installed on a machine that iBase is installed on.

Why is it useful?

Scheduler can be used to schedule data imports and exports to occur at times when system resources are lightly loaded or when iBase would otherwise be unattended. For example, at night or during lunch breaks. It is useful for controlling automatic imports of data that is generated by systems external to iBase and that might be required to occur regularly.

© N. Harris Computer Corporation

How Scheduler works

You can use iBase Scheduler to control the automatic import and export of data for a database. You can set up tasks to import or export when file changes are detected, or at set intervals.

Triggers

The Scheduler Service uses triggers to activate a task within an iBase database. A trigger can be configured either to monitor a file for changes, or to wait for a specific date and time before the task is activated.

For each trigger, an activation date and optionally a deactivation date can be specified.

File change triggers

File change triggers are a useful means of completing a task on an ad hoc basis, or for near real-time database updates.

For example, if an iBase database is to be kept up to date with data from another system, but the precise time when the export files from the other system are created is not known, it might be possible for the other system to create or touch a triggering file after the export is completed.

Scheduler can be configured by using file change triggers to monitor the triggering file for a change in the modification date. After a change is detected in the modification date, Scheduler will start the task to import the data contained in the import files, or export data from another database.

Interval triggers

Interval triggers are useful for completing a task regularly - for example, once a day, on particular days of the week, or on a specific day in the month. The trigger can instruct the task to run at a specific time in the day, or at regular intervals within a time window, with a minimum interval of a minute.

Note: The time window cannot span midnight. To do this, two triggers must be configured, one running from the start time up to midnight, the other running from midnight to the end time.

Service interval

The Scheduler Service is configured to run at regular intervals using the **Run schedule service every** option in the configuration utility. This service checks to see if tasks are available that need to be run, and can be configured to run as frequently as once per second. The maximum value that can be specified for the service interval is 9999 minutes.

When the schedule service is running, additional checks are not made, and extra tasks will not be triggered. Checks will only resume when the tasks complete or are stopped.

For example, if the service interval is set to 5 minutes, the trigger defines an execution frequency of 5 minutes. The task takes a minute to run, then the interval between executions of the task would be a minimum of 6 minutes.

Note: The service interval should be set to monitor triggers more frequently than the smallest interval trigger that is defined. Setting the service interval to a value that is greater than the smallest interval causes the service to miss one or more expected schedules. For example, a trigger that is configured to run a task every fifteen minutes with a service interval of twenty-five minutes will run only three in every five schedules. However, having a service interval that is less than fifteen minutes will not necessarily solve this problem if the time taken to run the tasks extends the service interval to a value greater than the task frequency.

© N. Harris Computer Corporation

Task actions

Before a task is run and after the successful completion of a task, it is possible to complete one or more actions. These actions can take the form of an operating system command, a batch file, or a custom plug-in written in a high-level language that supports COM interfaces.

- **Program actions**

  Program actions can be a single operating system command, a batch file, or an executable program. These can be used to complete a simple data clean on a source file to remove unwanted records, data formatting to ensure that dates are in the correct format ready for import, or sending a notification message to a user by using the NET SEND command.

- **Plug-in actions**

  Plug-in actions are written for customers in C# or another .NET language. Plug-ins can be used to manipulate the contents of the database or complete a custom action that is not easily completed through a program action, such as sending an email message.

  A plug-in action must adhere to the following rules:

  - It must provide a means of accepting a reference to an iBase database.
  - It must provide a means of accepting a reference to the Scheduler log.
  - The processing must be completed inside a single entry point.

  To ensure that the developer of a plug-in provides the three preceding functions on its object interface, the Scheduler Framework includes an interface definition for use by plug-ins.

Installing and configuring Scheduler

The Scheduler Configuration utility can be installed as a stand-alone application, or with the Scheduler Service component.

It is suggested that you install the Scheduler Service component on an application server that continually runs the Windows operating system.

If the machine on which iBase Scheduler is installed is in a remote location, or a location that is not easily accessible, the folder that the Scheduler database is installed into must be shared so that it can be accessed from other machines on the network by the Scheduler Configuration Utility.

If the Scheduler service is to complete imports or exports solely with Microsoft Access iBase databases that are on the same machine as the service is running, a standard installation of Scheduler suffices. However, if iBase SQL Server databases are to be used with integrated security or the iBase database, or one or more files are to be accessed across the network, any combination of the following might have to be altered:

- The Scheduler Service account startup context
- SQL Server permissions
- File and folder security permissions
- Folder share permissions

For example, when import files, export files, or trigger files must be accessed over the network, one or more of the following changes must be made:

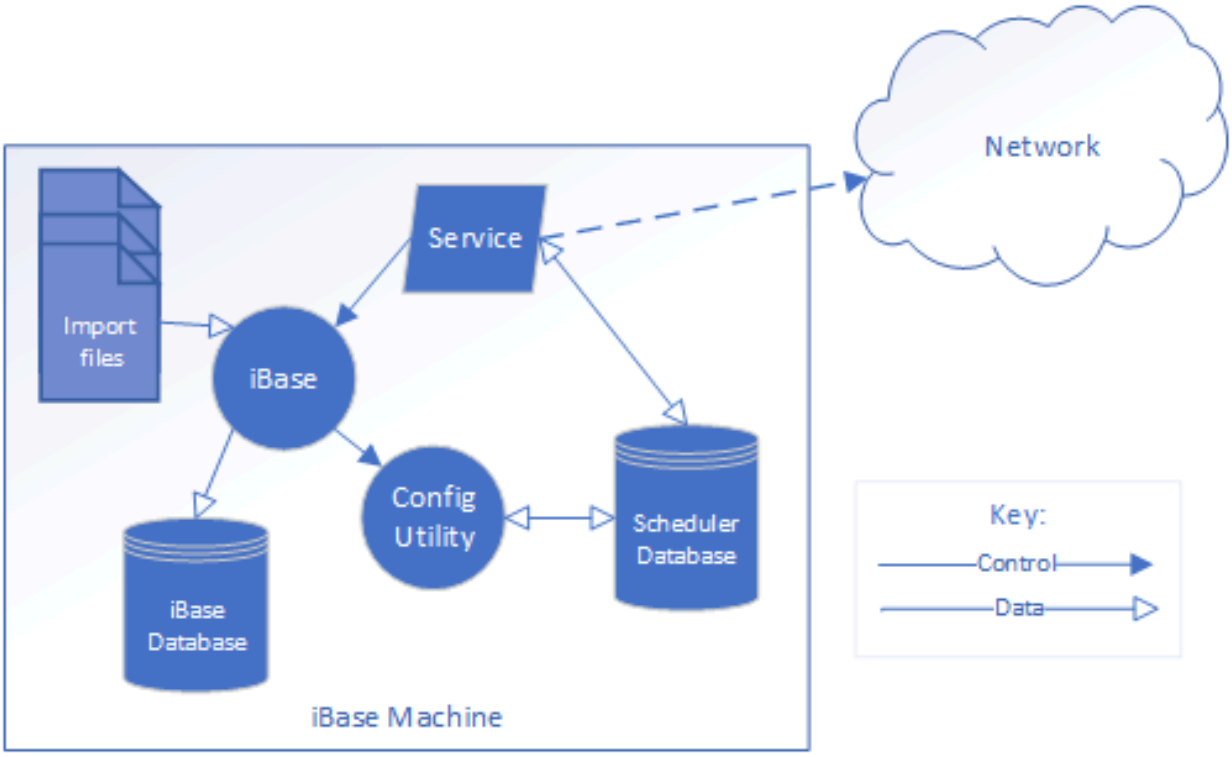
- Change the configuration of the Scheduler service to start by using the security context of a domain account. Grant this account the relevant access rights to the files.

© N. Harris Computer Corporation

- If the account context under which the service is configured to start must remain as 'Local System', the permissions to all files and folders that are to be accessed must be changed. The local system account has limited network access and is only able to access files that are authorization that is granted to the 'Everyone' group.
- If the database is an SQL Server database and the SQL server is on another machine, the service must be configured to run under the context of a domain user with sufficient rights to access the SQL Server database on the SQL server.

Set up a standard configuration

The default Scheduler installation installs all the components of the Scheduler on a single box. This installation is useful for application, file, and database servers, or individual users who want to schedule imports into their personal databases.



Use the Local System account

Installing Scheduler without changing any of the default installation options configures the Scheduler service to use the Local System account. The Local System account has administrative rights on the local machine but limited access to the network. On a secure network, where the Everyone group has no rights, the service is unable to access resources that are not on the local machine.

An installation that uses the Local System account is able to access the import files or export folder on the local machine if the local Administrators group has access to them.

If the iBase database is a Microsoft™ Access database, the service must be able to open the database through iBase and import data into it or export data from it. However, if the iBase database is an SQL server database, the service uses the sysadmin fixed server role. The sysadmin role is able to open

© N. Harris Computer Corporation

the database and import or export data, provided it has the `BUILTIN\Administrators` group or `NT AUTHORITY\System` login assigned.

Access to the Scheduler database through the configuration utility depends on the rights of the user that is running the application.

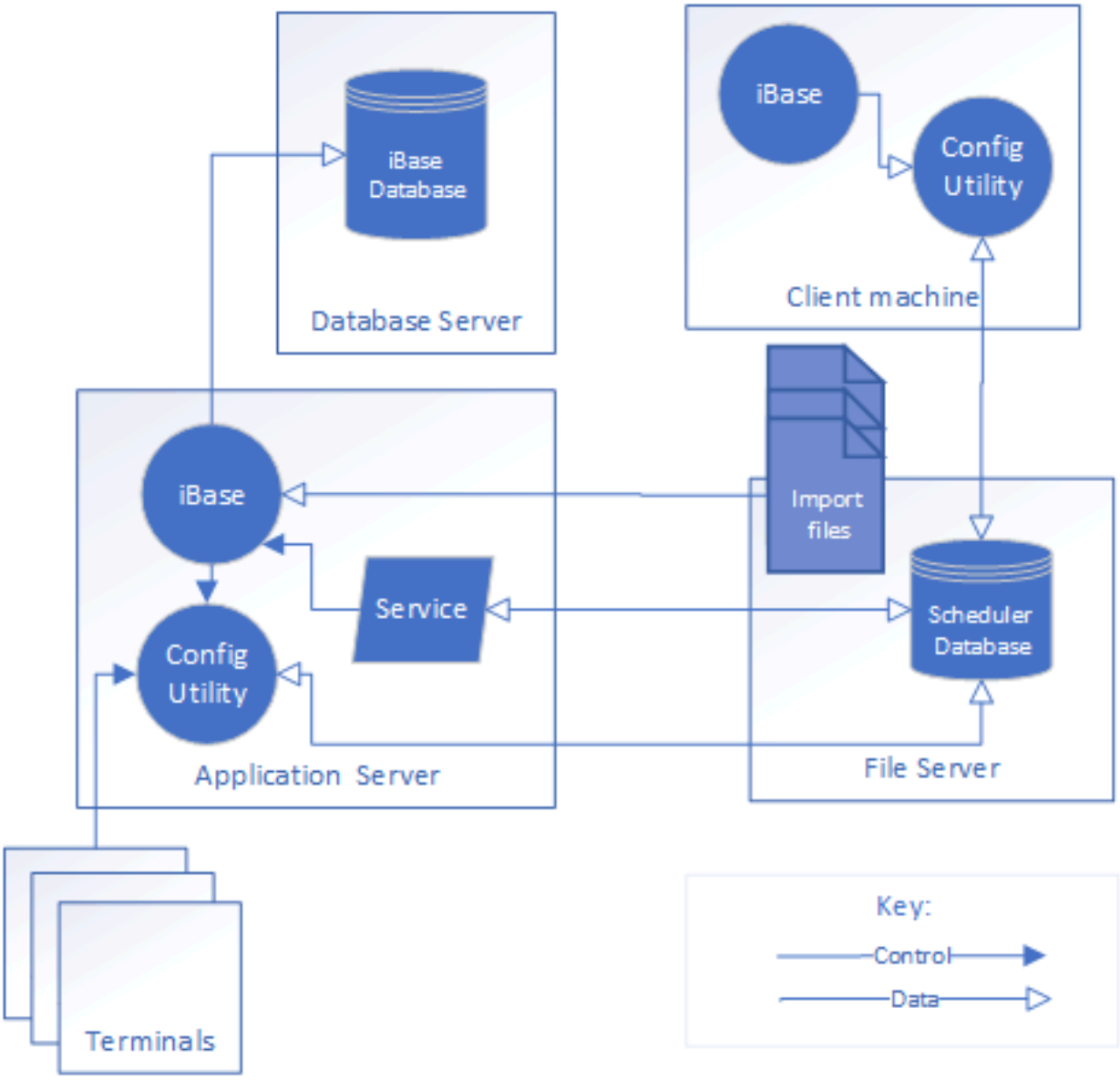
Set up an advanced configuration

Within an enterprise environment, it is unlikely that all the components required for iBase and Scheduler are on the same machine.

It is possible that:

- A dedicated SQL Server machine stores the iBase database.
- A file server has the iBase database connection file, Scheduler database, import files, and trigger files and is the destination for export files.
- An application server runs the Scheduler Service and is accessed through terminals to run iBase and the Scheduler Configuration utility.
- The configuration utility can also be installed on users' PCs.

© N. Harris Computer Corporation



Before you install Scheduler

Before you install Scheduler, the following preparatory actions must be carried out:

- Create a domain account for the Scheduler service. Set the 'Password never expires' option for this account.
- Create a folder on the file server for the Scheduler database. Share this folder and ensure that the Everyone group has Full Control of the share.
- Assign the required security permissions through NTFS security on the Scheduler database folder to the following domain accounts:
  - Scheduler Service account.

© N. Harris Computer Corporation

  - Any users that configure Scheduler. For example, create a Windows™ group for this purpose and assign the permissions to the group rather than to individual users. When a new user is to configure Scheduler, they can be added to this group.
  - Installation user. This is not required if the user that installs the Scheduler service is already a member of the Windows group for Scheduler users.
- Map a drive on the application server to the Scheduler database folder on the file server.

Installing Scheduler on the server

Install the Scheduler on the application server, selecting both the Service and Configuration options in the installer. You can choose where to install the Scheduler database. Typically, the default location for the Scheduler database is:

```
C:\ProgramData\i2\i2 iBase\en-US\Scheduler\Scheduler.mdb
```

You can move the database to a different shared folder if required. You are prompted for the location of the Scheduler database when you first start the Scheduler Configuration utility.

To display the Scheduler Configuration, select **Start** > **Programs** > **i2 iBase 10** > **Tools** > **iBase Scheduler Configuration**.

Setting the permissions

After the installation, the Scheduler service account will require permission to access the following information:

- Connection and security files for any of the iBase databases in use by the Scheduler Configuration utility.
- Import and trigger files.
- The destination folder for export files.

For SQL Server databases, the Scheduler service account requires access to the databases on the SQL server that are referenced by the connection files.

When you configure Scheduler in a distributed environment, it is important to use UNC paths throughout the Scheduler Configuration utility and within the import and export specifications themselves.

Note: The use of more than one Scheduler service monitoring the same Scheduler database is not supported in this release because of the risk that each instance of the Scheduler service might run the same import specification. This might result in duplicate entities and links created in the iBase database, depending on how the individual import specifications are set up to handle duplicated data read from import files.

Setting up a domain account for the Scheduler service

The Scheduler service is installed to run under the local Windows™ system account, which is only applicable if the service is running on the local machine. To run Scheduler that uses a Scheduler database that is installed on a different machine, then the service needs to log on using a domain user name and password. The user and password are set in the Windows Services application.

Directories and files	**Min. permission**
The `Scheduler` directory that contains the file `Scheduler.mdb`	Modify

© N. Harris Computer Corporation

Directories and files	**Min. permission**
`Scheduler.mdb`	Read, Write
Directory containing the DLLs and EXEs for iBase Scheduler	List Folder Contents
DLLs and EXEs for iBase Scheduler	Execute
Directory containing the file on which a file trigger is created--- no permissions are required on the file itself.	List Folder Contents
Directory containing the batch scripts or plug-ins that are required to execute as task actions. Alternatively, you might assign `List Folder Contents` permission to the directory and `Read` and `Execute` permission to the files.	Read & Execute
Data files for import or the directories that contain the data files. You require `Write` permission if the file is to be altered by a batch script or plug-in, and `Modify` permission if the file is to be deleted by a task action.	Read
Directory containing the iBase security file.	Modify
iBase security file	Read, Write
iBase database directory (JET databases only)	Modify
iBase database file (JET databases only)	Read, Write
iBase database file (SQL Server databases only)	Read

The Windows Services application automatically grants the domain account the permissions that are required to run services.

If you are using Windows integrated security (NTLM authentication), then the domain account needs to have `permit` access for the relevant SQL Server databases.

Service configuration for remote access

When iBase Scheduler is installed on a user's machine solely to keep their personal database up-to-date, no configuration changes are required to the service or SQL Server databases. It is anticipated that, in this case, Scheduler can be configured by the administrator of the iBase database.

If Scheduler is installed on an application server, possibly accessing files on a file server or databases on SQL Server, the service configuration must be changed:

- Users must have write access to the Scheduler database, `Scheduler.mdb`.
- Users must log on with an iBase user name and password (they cannot use single sign-on).

© N. Harris Computer Corporation

The configuration of the Scheduler connections, tasks and triggers, and the iBase import or export specifications becomes more involved because of the need to use UNC paths throughout. Therefore, it is beneficial for the user who configures Scheduler to understand Windows™ security and folder sharing in an enterprise environment.

After the Scheduler service is installed on the machine that is to run the tasks, the Scheduler Configuration Utility can be installed on a remote machine for administering the Scheduler database.

Note: During the installation, the Scheduler database is installed in its default location. You can move the database if required. See Installing Scheduler on the server for details.

Configuring Scheduler

Note: To open iBase Scheduler Configuration, you require write access to the Scheduler database, `Scheduler.mdb`. You must log on with an iBase user name and password (not single sign-on).

This involves the following steps:

1. Setting up database connections

   A database connection contains the information that is required by Scheduler to open the database, that is, the location of the database and its security file, and the user name and password. Notice that Scheduler does not require exclusive access to the database when scheduled tasks are run.
2. Add one or more tasks to specify the data to import or export.

   A task specifies the batch specification that defines the import or export specifications to run. (Specifications and batch specifications are created in iBase itself.) You can also set up programs or custom plug-ins to perform specific actions (task actions) either before or after you import or export data.
3. Set up one or more triggers for each task to determine the scheduling frequency.

   A trigger is the event that causes the task to run, for example either when a specified date and time is reached or when the modification date on a file changes. A trigger can recur or be a once-only event.

Configuring iBase Scheduler

iBase Scheduler provides automatic scheduling for data imports and exports for multiple iBase databases on either local or remote machines. Data is imported or exported by a Microsoft™ Windows™ service that you configure in iBase Scheduler.

To import or export data on a set schedule:

1. Define the data that you want to import or export as one or more import or export specifications.
2. Group the specifications in the sequence in which you want them to run, in batch specifications.
3. Schedule the import or export by setting up tasks in iBase Scheduler.

Using iBase Scheduler, you can run scheduled tasks:

- Immediately
- Once, at a specified time
- At preset intervals (hourly, daily, weekly, monthly)
- Whenever a specified file is updated.
- Until a specified cutoff date or time is reached.

© N. Harris Computer Corporation

In addition, before or after importing or exporting data, you can run a command-line program, a custom plug-in, or an operating system command. For example, to move data files for import to a different directory or send a notification that the batch export is complete.

Details of scheduled tasks are stored in the Scheduler database, which is monitored by a service, which automatically runs the scheduled tasks. This service runs as part of the Windows Services application. You use this application to start, stop, pause, resume, or restart the service.

iBase Scheduler can be installed on any machine on which iBase is already installed; you do not have to run the service on this machine.

Scheduling imports and exports

Before you use iBase Scheduler to schedule the import or export of data, you need to configure it in the iBase Scheduler Configuration dialog.

This involves the following steps:

- Set up a connection to the required iBase database.

  A database connection contains the information required by iBase Scheduler to open the database, that is the location of the database and its security file, and the user name and password. Notice that iBase Scheduler does not require exclusive access to the database when running scheduled tasks.
- Add one or more tasks to specify the data to import or export.

  A task specifies the batch specification that defines the import or export specifications to run. (Specifications and batch specifications are created in iBase itself.) You can also set up programs or custom plug-ins to perform specific actions (task actions) either before and/or after importing or exporting data.
- Set up one or more triggers for each task to determine the scheduling frequency.

  A trigger is the event that causes the task to run, for example either when a specified date and time is reached or when the modification date on a file changes. A trigger can recur or be a once-only event.

Scheduler and iBase security features

The user account for running iBase Scheduler does not need any security restrictions applied. However, iBase Scheduler does support these iBase security features:

- **Database Management groups**. These determine whether users will be denied access to the database when they try to test a connection or create a new task. The minimum user permissions are described in Minimum user permissions below.
- **Data Access Control groups**. These restrict access to specific entity and link type records or to specific fields, and will prevent iBase Scheduler from running tasks that access those records and fields. In these circumstances, the completion status of the task will indicate that the task failed.
- **Folder Object Control**--- specifically this restricts access to the folder containing the batch import or export specifications.

The following iBase security features are not supported by iBase Scheduler, and any restrictions imposed by these features will be ignored in iBase Scheduler:

- System Command Access Control
- Reason for Action
- Auditing--- nothing is recorded in the audit log

© N. Harris Computer Corporation

Minimum user permissions

Scheduling can only be run by authorized users with the following minimum user permissions in iBase:

- Add Entity/Link Records
- Update Entity/Link Records
- Update/Delete Entity/Link Records created by other users

These are the same as those required to run import specifications in iBase itself.

Note: If you need to change the permissions granted to a user account which is used to run iBase Scheduler then make sure you re-test the connection to the database. If you are using Data Access Control groups then you need to run a trial import/export to verify that these are set correctly.

Setting up database connections

Before you can schedule any batch imports or exports, you need to set up a connection to the iBase database. The database can be local or on the network.

Procedure

1. If the database is on the network, check that it can be opened:
   a. Close iBase Scheduler.
   b. Open iBase Designer.
   c. Select **File** > **Open Database**
   d. In the **File name** box, type in the UNC path of the database (.idb) file, and then click **Open**.

      Note: Paths that are entered by using a mapped network drive are always converted to UNC (Uniform Naming Convention) paths when you click **OK**.
   e. Close iBase Designer.
   f. Reopen iBase Scheduler.

      Note: A system administrator must first open the database using the UNC path before other users can open it using this path.
2. In the Connections area, enter the name that you want to use for the connection to the database.
3. Enter the path to the database file (.idb).
4. Enter the path to the security file (.ids) for this database.
5. Enter the iBase username and password that you want the Scheduler service to use. These credentials will be saved in the Scheduler database in encrypted form. For details of the permissions required by this user account, see Scheduling imports and exports. To verify that the connection works, click **Test Connection**.
6. Click **OK** to save the database connection.

Results

The database connection is configured. You can change any of the details of an existing connection, such as the connection name, database path, username, and password without affecting any scheduled tasks. To edit a connection, click **Edit** in the Connections area.

Although the changes to the connection are made as soon as you click **OK**, the changes only affect the Scheduler service when the next task is scheduled. To update the schedules, cancel the current scheduled tasks by making the trigger inactive.

© N. Harris Computer Corporation

Note: It might not be appropriate to change the database path to another database because the tasks may no longer be relevant to the batch specifications in the new database.

To remove a database connection and all the tasks, triggers, and scheduled tasks that are associated with it, click **Delete** in the Connections area.

If you use the logging option, when tasks are scheduled to run, any scheduled tasks for this database are marked as **Trigger deleted** in the log description.

The connection to a database might be unsuccessful for various reasons. For example:

- The username or password are no longer valid.
- The user account has insufficient permissions to open the database. See Scheduling imports and exports for the required permissions.
- It references a security file that is not associated with the database.
- Someone is logged on to the security file in iBase Designer, and made a change but has not yet logged off.
- The database has moved so the path is no longer correct.
- The database has been deleted.

Immediately after you open iBase Scheduler, you may be unable to select any of the database connections and this message appears:

```
Could not connect to the iBase database for this connection.
The database may have been moved or deleted, or your connection details are
 incorrect.
```

This occurs if the database for the first connection in the list is open in iBase Designer. To continue, close the database in iBase Designer. A similar problem also occurs later in the session if one of the other databases for which there are connections is open. To continue, find out which of the databases is open in iBase Designer, and close it.

Setting up the logging options

Details of batch imports and exports for each database connection can be logged in the Scheduler database. You control the level of detail by setting the logging options on the Schedules page of the Scheduler Configuration dialog; these can be different for each database connection. Nothing is recorded if no logging options are set.

Procedure

To set the logging options for a database connection:

1. Click the **Schedules** tab to display the Schedules page.
2. Select the required database connection. By default, all the logging options are selected when the database connection is created.
3. In the Logging Options area, turn on or off any of the following logging options. Your changes take effect immediately.
   - **When tasks are scheduled to run**

     The date and time that the next batch import or export will run. This provides a useful confirmation that the task and trigger are set up correctly as the Scheduler service will add a log entry for the scheduled import or export the next time that the service runs.
   - **When tasks start and finish**

     When the batch specification started and finished running.

© N. Harris Computer Corporation

- **When task actions start and finish**

  The startup and shutdown of programs and plug-ins that run before or after the import or export.
- **When components start and finish**

  The start and finish of individual specifications in the batch specification. (For example, an import specification is a component of a batch import specification.)
- **Statistics when a component finishes**

  The number of records that are added, updated, unchanged, or result in errors for the main entity, or the link end entities, in the specification (component).

  For a link import, will also include the number of records added, updated, unchanged or result in errors for the link itself.

  For an export, will include the number of records exported.
- **The completion status when a task finishes**

  Whether the batch specification ran successfully or whether there was a problem and, if so, what the status of the scheduled task is.
- **Warnings, Errors**

  Any warnings and errors generated by the batch import or export.

Setting up batch import and export tasks

Batch imports and exports are defined as tasks. A task consists of a single batch specification that lists one or more import or export specifications in the order in which they are to run and, optionally, actions to perform before and after the batch import or export.

About this task

The batch specifications and import/export specifications are created in iBase.

The only restriction is that the import specifications cannot contain any prompts for user input. You will not be able to select an import batch specification that contains such an import specification. You should not modify an import specification that is used by iBase Scheduler so that it requires input; if you do so, the Scheduler service will not run the associated task.

You must set up a connection to the database before you can add any tasks. Once you have added a task, you can set up the triggers to schedule when the task runs.

Note: You cannot add tasks to an iBase database that cannot currently be opened by iBase Scheduler. For example, if another user has opened the database in iBase Designer.

Procedure

1. Open the iBase Scheduler Configuration and click the **Schedules** tab to display the Schedules page.
2. Select the required database connection from the **Connections** list.
3. In the tasks area, click **New**.
4. Enter a descriptive task name. For example, you might include the name of the batch specification in your task name. The maximum length of the task name is 255 characters.
5. In the **Type** list, select either:
   - **Batch Import**
   - **Batch Export**

© N. Harris Computer Corporation

6. Select the batch specification that you want to use.

   Note: The list displays only suitable batch specifications - it does not display any that contain import or export specifications that require input from the user. However, when you browse for a specification, all batch specifications are listed regardless of suitability.
7. If required, enter the programs or plug-ins that you want to run before or after the batch import or export. For details, see Setting up extra task actions.
8. Click **OK** to save the task.

Results

The task is now set up.

How soon the task runs will depend on how frequently the service runs.

Scheduling a task to run immediately

You can schedule a task to run when the Scheduler service next runs, even if there is no trigger for it:

1. Select the task that you want to schedule.
2. Click **Run** below the list of tasks.

Editing a task

You can change any of the details of the task without affecting any scheduled tasks. The specification name shown may be different if the batch specification has been renamed in iBase.

To edit a task, select the connection, and click **Edit** below the list of tasks.

Note: Changing the name of the task will change the task name against all entries in the log, including log entries for completed scheduled tasks. If required, you can save the existing log file.

Although the changes to the task are made as soon as you click OK, the changes only affect the Scheduler service when the next task is scheduled. To update the schedules, cancel the current scheduled tasks by making the trigger inactive.

If the batch specification used by a task is deleted, then the list of tasks will display #Not Found next to the name of the task.

Canceling scheduled tasks

To prevent iBase Scheduler from running a scheduled task, and prevent further scheduling, you need to make the trigger inactive. For details, see Setting up triggers.

Deleting a task and its triggers

You can remove a task, and any triggers defined for it, from a database connection by selecting the connection to the database, then selecting the task and clicking Delete below the list of tasks.

If you use the logging option, When tasks are scheduled to run, any scheduled tasks are marked as Trigger deleted in the log description.

Deleting a database connection will also delete all the tasks and triggers set up for it.

Setting up extra task actions

In addition to importing and exporting data, you can add actions to your batch task. For example, to ensure that the data is handled correctly, the right people are notified, and the files are in the right

© N. Harris Computer Corporation

format. You can set up external systems for managing regular actions, and associate them with your batch imports and exports.

About this task

Task actions can be completed by running a program, a custom plug-in, or by using operating system commands. The program or custom plug-in must have a command line so that it can be run from the operating system, and neither can prompt for user input.

Example actions that are needed before an import

You can import data from a number of different sources. To ensure that the data matches your data model, and is imported as quickly as possible, you can use an Extract, Transform, Load (ETL) process.

Extract

Before data can be imported into iBase, you might need to extract data from another data source. For SQL Server databases, you can call an SQL Server Integration Services (SSIS) package to move the data from the source.

Transform

To ensure that the data is in the best format to be imported, you can create and call a stored procedure to prepare it. To speed up the import, you might want to convert the data from a view to a table, or remove indexes from the tables that are imported.

Example actions that are needed after import

Clean up

After data is imported successfully, you can trigger various tasks to leave your system ready for future use. For example:

- Deleting import files.
- Exporting the import log to a timestamped text file.
- Clearing the scheduler database.
- Setting the date and time of the next import.
- Reinstate indexes that were removed before the import.

iBase modifications

As the data that is present in your database has changed, you might want to update your database design. For example:

- Regenerate code lists present in the database based on the data that is now present.
- Adjust the icons used to highlight different aspects of the data. For example, age, gender, profession, and for links aspects of the connection such as the age the person connected to the event was when the event occurred.

Depending how the Scheduler service is configured, any program steps that follow a task is canceled if the program takes too long to run, and the task marked as canceled in the log. This cancellation does not undo any completed import or export.

Note: The cancellation does not apply to plug-ins.

Environment variables

When running a program before the task, the following environment variables will be set: `IBASE_CONNECTION`, `IBASE_TASK`, `IBASE_TRIGGER`, `IBASE_SCHEDULE_ID`. For running export

© N. Harris Computer Corporation

tasks, additionally the variables `IBASE_FAILED_EXPORT_COUNT`, `IBASE_FAILED_EXPORTS`, `IBASE_SUCCEEDED_EXPORT_COUNT`, and `IBASE_SUCCEEDED_EXPORTS` are available. For import tasks the environment variables are the same, with IMPORT rather than EXPORT in the variable names.

Procedure

To set up an additional action to a bulk import or export task:

1. Open the iBase Scheduler Configuration and click the **Schedules** tab to display the Schedules page.
2. Select the required database connection from the **Connections** list.
3. In the tasks area, click **New**.
4. Enter a descriptive task name.
5. In the **Type** list, select either:
6. Select the timing of the action:
   - **Perform actions before the task runs** - the program or plug-in will run before the batch specification.
   - **Perform actions after the task runs** - if the import or export completes successfully, the program or plug-in will run after the last specification in the batch specification is finished.
7. From the **Action Type** list, select:
   - **Program** - For operating system commands or programs that can be run from the command prompt.
   - **Plug-in** - For custom plug-in actions.
8. Enter the details of the task actions:
   - For a program, the command-prompt parameters of an operating system file that you want to run. Browse for the location of the program file.
   - For a plug-in, its project and class names in the format `project.class`.
9. Click **OK** to save the task.

Setting up triggers

A trigger defines the event that causes the task to run, for example either reaching a specified date and time or when the modification date on a file changes.

Before you begin

You must create a task before you can set up any triggers.

About this task

You can trigger tasks in three ways:

- **Trigger when file changes** runs a task whenever a specified file is updated (or created if the file is copied to a directory before the import or export). The file does not have to be associated with any import specification.

  The Scheduler service checks the modification or creation date on the file at the interval that is set for the service. It runs the scheduled task when the modification date for the file is later than the start date set in the trigger.
- **Trigger by date/time** runs a task to import or export data after an interval of time.

© N. Harris Computer Corporation

The service will calculate the interval from the details in the trigger, and run the scheduled task when the calculated date and time is reached.

- **Run** runs the selected task the next time the service runs. For details, see Setting up batch import and export tasks.

With the exception of **Run** triggers, triggers last indefinitely unless you set an end date. Triggers that have passed their end date remain on the system and can be reactivated when needed.

The following date and time options are available:

Option	**When the task is scheduled to run**
Daily	After the specified number of days. For example, enter 10 to run a batch import every 10 days.
Weekly	After the specified number of weeks, on the required days of the week.
Monthly	After the specified number of months, on the required day of the month. (**Note:** A task that is scheduled to run on day 29, 30, or 31 will always run on the last day of the month if there are fewer than 29, 30 or 31 days in the month.)
Once	Only on the run date, at the selected time.
Every	Every *n* minutes or hours, on the run date, between the specified times.

Each time the Scheduler service runs, it calculates the next scheduled task for any active triggers it finds that do not have a schedule. The run date is shown in the log if the logging option **When tasks are scheduled to run** is selected. The time that a task runs is always approximate.

If for some reason there is a delay, such as pausing the service, then the service will attempt to run it later when the service resumes. The next scheduled task will not be added to the queue, or shown in the log file, until the current scheduled task has completed.

The first run date depends on the start date and time in the active trigger, but all dates are inclusive:

Start date	**Effect on scheduling**
Today's date	The task will run today. If it is set to run at a time in the past, then the task runs immediately unless the **Ending at** time is also passed.
Past date, future date	This date is used to calculate the next run date only, at the time(s) specified in the trigger.

When you edit a trigger, the service recalculates the run date of the next task to be scheduled for the active trigger, and marks the current scheduled task as updated (in the log description).

© N. Harris Computer Corporation

Procedure

1. Click New below the list of triggers to display the Trigger dialog.
2. Enter a descriptive name for the trigger, such as Every Monday Night, Alternate Weeks (until Jun 2005). The maximum length of the name is 255 characters.
3. If you do not want to schedule the task at this stage turn off the Active checkbox.
4. In Start date, enter the date that iBase Scheduler will use as the starting point when calculating the intervals between scheduled tasks. You can choose a past or future date if required. Selecting a past date and/or time will affect how the next task is scheduled; for details, see How tasks are triggered above.
5. Triggers are active until further notice. To turn off this feature, click End date and enter the end date.

For example, if you enter the 5th May then the last possible date on which the task can run is the 5th. Triggers that reach their end date can be re-used by editing them.

1. Choose the event that makes the task run. To trigger the task to run:
2. When the modification date on a file changes, select Trigger when file changes, and then enter the path to the file or browse for it. The file does not have to exist at this stage.
3. When a date and time is reached, select Trigger by date/time and then enter the number of days, weeks or months, and an optional time.

See Types of trigger above for details.

1. Click OK to save the trigger.

The first task scheduled for this trigger will appear in the log the next time the Scheduler service runs (provided that the logging option When tasks are scheduled to run is selected).

Reading the scheduler log

The Scheduler log can record the details of scheduled and completed batch imports and exports - the actual events that are logged will depend on the logging options set on the Schedules page of the iBase Scheduler Configuration dialog. Activity for each database connection is logged to the same file.

To display the log viewer

1. Click the **Schedules** tab to display Schedules page.
2. Click **View Log** to open the log files folder, and then click the required log.csv file to view it. Files can be opened and viewed in a number of applications and text editors, such as Microsoft Excel, or Notepad.

Completion status

The different statuses for scheduled tasks are shown in the following table:

Task	**Status**
Task execution successful	The task ran successfully.
Trigger updated	The trigger has been edited.
Trigger inactive	The trigger has been made inactive and will not run again until you make it active.

© N. Harris Computer Corporation

Task	Status
Trigger expired	The batch import or export did not run because the end date on the File Change type trigger had passed.
Trigger deleted	The trigger has been deleted. Also, the task was scheduled but was canceled by the service because the trigger had expired. This might happen, for example, if a previous task took much longer to run than expected.
Error	The task was not completed successfully.

Scheduled batch imports and exports that fail

If the logging option **When tasks are scheduled to run** is selected, entries are added to the log for tasks that are scheduled to run in the future.

The file associated with a file trigger does not have to exist; the task is still scheduled and an entry appears in the log. If the file does not exist, or if the permissions on the folder containing the file prevent viewing of the date and time of the file, then these messages are logged once every 24 hours:

- The file X does not exist or there are insufficient permissions on the directory it is located in to read the file.
- The File Change trigger file X does not exist or there are insufficient permissions on the directory it is located in to read it.

A batch import or export fails if the batch specification contains a specification that requires input from or is output to a text file, and this text file is missing. In this instance the message "Failed to create import set X. File not found" is displayed.

Note: Tasks fail to run if the service cannot get exclusive access to the database, for example because it is open in iBase Designer.

Completed batch imports and exports

To prevent a type of message from appearing in the log, display the Schedules page, and click to turn off the required Logging option, as described in the Log Description section below.

Message	Description
Date & Time	Timestamp recorded when this phase of the batch import or export was started or completed. Most recent events are listed first.
Connection Name	Name of the connection to the iBase database.
Task Name, Task Type	Name of the trigger that scheduled the task, and its type: Interval or File Change.

© N. Harris Computer Corporation

Message	Description
Log Type	The log type corresponds to the logging options set on the Schedules page.
Log Type Schedule	When a task is scheduled to run.
Log Type Execute Task	When a task starts and finishes.
Log Type Execute Program	When the program for a task action starts and finishes.
Log Type Plugin	When a plug-in for a task action starts and finishes.
Log Type Execute Component	When components start and finish.
Log Type Statistics	Statistics when a task completes.
Log Type Completion Status	The completion status when a task finishes. See Completion Status above for details.
Log Type Warnings	Warnings produced by a batch specification running.
Log Type Errors	Errors produced by a batch specification running.

Log Description

The messages detail, in reverse order, the progress of the task. For example:

Message	Logging option
Completed execution of batch X	When batch X finished. This marks the end of this task. Exclude this message by turning off "When tasks start and finish".
Completed execution of X	When the named import/export specification finished running. The messages given below relate to this specification. Exclude this message by turning off "When components start and finish".
Entity, Links, Entity End 1/2	When importing entities and links, this reports statistics concerning the entity or link records. If any errors are reported, then they are logged in the text file specified in the specification. When exporting, reports the number of records exported. Exclude this message by turning off "Statistics when a component finishes".

© N. Harris Computer Corporation

Message	Logging option
Execution <status>	Shows the status of the completed batch import/ export. See Completion status above. Exclude this message by turning off "The completion status when a task finishes".
Starting execution of X	When the service began to run the named specification.

Managing the scheduler service

The Scheduler service is installed in the Windows services database, and is managed by the Windows Services application. You configure it using the Service page of the iBase Scheduler Configuration dialog.

About this task

Scheduled imports and exports (tasks) are stored in the Scheduler database, which is a Microsoft Access database. This is monitored by the service, at a frequency that you specify when you configure it. The service automatically runs the scheduled tasks as they fall due.

You can start, stop, pause, resume, or disable the service by using the Services application. Service activity may be recorded in a log file if required.

You can only run the service on a machine where iBase is installed.

The Windows Services application provides these commands for managing the service.

Command	Description
Start	Starts the service after it has been stopped. Any scheduled tasks are run if they have passed or reached their start date and time. By default, the service starts automatically after restarting the computer.
Stop	Stops the service, after completing any tasks that are currently in progress.
Pause	Suspends the service, after completing any tasks that are currently in progress.
Resume	Resumes the service at the beginning of the next iteration of the service loop. Any scheduled tasks are run if they have passed or reached their start date and time.
Restart	Stops and then immediately restarts the service.

These actions may be recorded in the Service log depending how the service is configured.

© N. Harris Computer Corporation

Procedure

You can find out whether the service is running by using the Service log - you can only do this if the activities of the service are recorded in the Service log:

1. Open the Windows Services application.
2. Click the **Service** tab to display the Service page.
3. Click **View Log**. A folder opens in File Explorer, showing the log files.

Configuring the scheduler service

Before you can use the iBase Scheduler, you must configure the iBase Scheduler service.

About this task

To configure the iBase Scheduler service, you need to:

- Set the frequency with which the service scans the Scheduler database for scheduled tasks.
- Choose which events are recorded in the Service log file - by default, all logging options are selected.
- Set the interval after which a scheduled task is canceled if the program that runs before or after the batch import/export is taking too long to run.

Note: You can also configure the service using the Windows Services application. For example, you might want to change the startup type from Automatic. There are no startup parameters for the service. For further information, see the Help for the Services application.

Procedure

To configure the service:

1. Click the **Service** tab to display the Service page.
2. Specify the frequency with which the service scans the Scheduler database, by entering the number of minutes and seconds. The default interval is 10 seconds.

   How frequently you want the service to run depends on the number of tasks that you are running, and available computing resources.

   Specify which details are recorded in the service log by turning on or off these checkboxes:

Option	**Records**
Starts up	When the service is started by using commands in the Windows Services application, or when the machine starts up (if the service is set to use Automatic startup).
Shuts down	When the service is shut down by using the Services application, or when the machine is shut down.
Pauses	When the service is suspended by using the Services application.

© N. Harris Computer Corporation

Option	Records
Continues	When the service is resumed by using the Services application.
Fails	When a problem occurs that prevents the service from running. An exception is when the database cannot be opened.

3. If the programs associated with a task take too long to complete, enter the interval after which the service should cancel this and any other programs associated with the task. This is 10 minutes by default. This feature does not apply to plug-ins.
4. Click **Save**.

What to do next

The service is installed to run under the local Windows system account, which is only applicable if the service is running on the local machine. If you intend to run iBase Scheduler using a Scheduler database that is installed on a different machine then the service needs to log on using a domain user name and password. This is set in the Windows Services application.

The account under which the service runs must have the following minimum file permissions:

Directories and files	Min. permission
Scheduler directory containing the file Scheduler.mdb	Modify
Scheduler.mdb	Read, Write
Directory containing the DLLs and EXEs for iBase Scheduler	List Folder Contents
DLLs and EXEs for iBase Scheduler	Execute
Directory containing the file on which a file trigger is created--- no permissions are required on the file itself.	List Folder Contents
Directory containing the batch scripts or plug-ins required to execute as task actions. Alternatively, you could assign List Folder Contents permission to the directory and Read & Execute permission to the files.	Read & Execute
Data files for import and/or the directories containing the data files. You will require Write permission if the file is to be altered by a batch script or plug-in, and Modify permission if the file is to be deleted by a task action.	Read

© N. Harris Computer Corporation

Directories and files	Min. permission
Directory containing the iBase security file.	Modify
iBase security file	Read, Write
iBase database directory (JET databases only)	Modify
iBase database file (JET databases only)	Read, Write
iBase database file (SQL databases only)	Read

Managing the scheduler database

If you have set up the scheduler database, there are a number of administrative tasks that you will need to carry out to manage your installation. These tasks should be carried out within the iBase Scheduler dialog.

The Scheduler database is a Microsoft Access database, which stores the:

- Scheduler service configuration
- Details of connections, tasks, and triggers
- Details of batch imports/exports that are scheduled, completed, or inactive (canceled)

Do not open or edit the Scheduler database in Microsoft Access. You should configure the database using the iBase Scheduler Configuration dialog, and view the log files using the View Log button. The Scheduler database is password protected to prevent accidental changes to the data that might be incompatible with iBase Scheduler.

You should back up the Scheduler database in the same way that you back up your other iBase databases.

Setting a password to control access to iBase Scheduler configuration

It is possible to apply a password to the Scheduler database to prevent users from using Scheduler to attempt an import into, or export from, an iBase database that is prevented through their usual iBase account. Assigning the password is accomplished through the **Service** page within the configuration utility. After a password is assigned, any user that starts the Scheduler Configuration utility must enter this password.

Procedure

To set the password:

1. Click the **Service** tab to display the Service page.
2. Enter the password in the **Scheduler Configuration Password** field, and then re-enter the password when prompted.
3. To apply the password immediately, click **Close**. When you close the dialog, iBase Scheduler checks the password and prompts you to re-enter it if there is a mismatch between the passwords you typed.

Saving the log files

© N. Harris Computer Corporation

You can save the Scheduler log file or the Service log file as a text file.

Procedure

1. Click the appropriate tab to display either the Schedules or Service page depending which log file you want to save.
2. Click **View Log** to display the Log Viewer.
3. Click **Save**, and then enter the file name and the directory that you want to use. The current date and time is automatically added to the file name.

Purging the log files and the scheduler database

You can delete all the entries from the Scheduler or Service log files, with the option of saving them to a text file first. Purging the Scheduler log file deletes all the log entries for scheduled tasks, however this has no effect on scheduled tasks which will run at the correct time. It also has the effect of purging the Scheduler database of any deleted connections, tasks, and triggers.

Procedure

To purge the log file (and also purge the Scheduler database):

1. Click the appropriate tab to display either the Schedules or Service page depending which log file you want to purge.
2. Click **View Log** to display the Log Viewer.
3. Click **Purge**. You are asked whether you want to save the log entries to a file before they are deleted.
4. Click **Yes** to save the entries in a text file, and then enter the file name and the directory that you want to use, or No to delete the entries immediately.

Moving the scheduler database

At any point you can move the scheduler database. When iBase Scheduler detects that the database file has moved, you will be prompted to identify the new location.

About this task

The Scheduler database is located in a directory that you specify when you install iBase Scheduler, and its location is shown on the Service page of the iBase Scheduler Configuration dialog. You cannot edit the database path on this page.

Procedure

1. Close the iBase Scheduler Configuration dialog and then stop the iBase Scheduler service.
2. Move the file `Scheduler.mdb` to the new directory.
3. Start iBase Scheduler and then, when prompted, select the new location of the database file.
4. Restart the iBase Scheduler service.

© N. Harris Computer Corporation

The Scheduler Log

Scheduler cannot send progress or error information for an interactive user to act upon because importing or exporting data by using Scheduler is a non-interactive automatic process. Any actions the Scheduler Service makes are recorded in the Scheduler log.

The Scheduler log is an important source of information about what happens. In particular, its record of the sequence of events provides valuable diagnostic information about the import or export. The level of detail that is recorded in the Scheduler log can be configured on a per connection basis and includes:

- When a task is scheduled
- When a task runs and its completion status
- When task actions run and their completion status
- When an individual component within a task runs, and its completion status
- Statistics that are returned from a task component
- Errors and warnings detected by the service

NLog is the logging framework that is used by Scheduler, and is fully configured using the `NLog.config` file. The default configuration is to write a log.csv in `C:\ProgramData\i2\i2 iBase \Scheduler`. Scheduler log files display in Microsoft File Explorer, and can be selected using the View Logs button in the Scheduler. For more information about customizing the log, contact i2 Support.

Interpreting the Scheduler Log

The Scheduler Log contains information about each task that is run.

The most recent event is at the top of the log.csv file.

Each line of the log displays a summary in the **Log Type** column of what occurred.

Reading from the bottom of the log upwards and examining the dates and times, that the task was scheduled for execution a number of days before it was run. The log activity for any successful execution of a task is as follows:

- Task execution commences
- Pre-task actions are run
- Task components are run
- Post-task actions are run
- Task execution completes
- Completion status is logged
- Task is rescheduled

In the previous log, which contains the results of the 'Example of a Successful Import Batch' task, the first import specification creates only a single statistics line. From this, it can be concluded that the import specification is importing entity records. The details for the statistics show the number of entities that were added, updated, unchanged, or were in error. The second import specification creates three separate statistics lines: it can be concluded from this that the import specification is importing link records. The statistics show the number of entities or links that were added, updated, unchanged, or were in error for the links, end 1 entities, and end 2 entities.

Note: A task does not run if a user changes an import or export specification within the batch that requires a response from the user. Changes of this type would be an import specification that prompts

© N. Harris Computer Corporation

for confirmation of the action to take when a matching entry is found in the database or an export specification that relies on a parameterized query.

If a pre-task plug-in, post task plug-in, or component fails to complete successfully the task execution stops at that point and the completion status indicates that a problem was encountered. The log does not report on any pre-task or post-task programs that fail to complete.

Pre-task action errors

The following example log shows that a problem was encountered when a pre-task action is run:

Date & Time	Level	Description	Connection Name	Task Name	Task Type	Trigger Name	Trigger Type	Log Type	Module
2024-09-14 11:31:05	Info	Task scheduled to run at 14/09/2024 11:31:35	User Guide DB	Example Pre-Task Error	Batch Import	Daily import 11:30	Interval	Schedule	i2.Base.Apps.Scheduler
2024-09-14 11:31:04	Info	Starting execution of 'Batch Import'	User Guide DB	Example Pre-Task Error	Batch Import	Daily import 11:30	Interval	ExecuteTask	i2.Base.Apps.Scheduler
2024-09-14 11:31:04	Info	Starting execution of pre-task program 'example.exe'	User Guide DB	Example Pre-Task Error	Batch Import	Daily import 11:30	Interval	ExecuteTaskAction	i2.Base.Apps.Scheduler
2024-09-14 11:31:04	Error	System.ComponentModel.Win32Exception: The system cannot find the file specified	User Guide DB	Example Pre-Task Error	Batch Import	Daily import 11:30	Interval	Error	i2.Base.Apps.Scheduler
2024-09-14 11:31:04	Info	Failed to execute pre-task program 'example.exe'	User Guide DB	Example Pre-Task Error	Batch Import	Daily import 11:30	Interval	ExecuteTaskAction	i2.Base.Apps.Scheduler
2024-09-14 11:31:04	Info	Completed execution of 'Export Batch	User Guide DB	Example Pre-Task Error	Batch Import	Daily import 11:30	Interval	ExecuteTask	i2.Base.Apps.Scheduler

© N. Harris Computer Corporation

Date & Time	Level	Description	Connection Name	Task Name	Task Type	Trigger Name	Trigger Type	Log Type	Module
		Specification Example'							
2024-09-14 11:31:04	Error	Errors detected during task execution	User Guide DB	Example Pre-Task Error	Batch Import	Daily import 11:30	Interval	Completion	i2Base.Apps.Scheduler
2024-09-13 14:03:26	Info	Task scheduled to run at 13/09/2024 14:04:00	User Guide DB	Example Pre-Task Error	Batch Import	Daily import 11:30	Interval	Schedule	i2.Base.Apps.Scheduler

Log management

The default NLog configuration for the Scheduler is to show today's entries in the log.csv file. Older log entries are archived into log.*date*.csv. files. After 30 days the archived log files are automatically purged. You can change the default settings in the NLog.config file.

Scheduler error messages

The following errors are reported in the Scheduler log if problems with the configuration of connections or errors in tasks or triggers occur. Each error is explained together with possible corrective action.

Many of the errors are caused by inadequate permissions on files, folders or within SQL Server. For details of the minimum permissions that are required for access to all the files and folders that are required by the Scheduler Service, see Setting up a Domain Account for the Scheduler Service.

Service errors

Could not start the iBase Scheduler Service on Local Computer.
Error 1053: The service did not respond to the start control request in a timely fashion.

This error message is displayed after a short interval when an attempt is made to manually start the Scheduler service by using the Windows Services application.

Refer to the next error message if this message is displayed immediately after an attempt is made to start the service. However, when there is a delay of thirty seconds or more between attempting to start the service and the error message appearing then the possible causes for the error are as follows:

Configuration	Possible Cause	Solution
N/A	The database file has the 'Read Only' attribute enabled.	Disable the 'Read Only' attribute on the database file.
The Scheduler database is on either the local machine or a remote machine and is accessed	The folder share permissions are set to 'Read' for the Scheduler Service account.	Alter the folder share permissions to grant the Scheduler service account at

© N. Harris Computer Corporation

Configuration	Possible Cause	Solution
through a folder share by using a UNC path. The file system in use is either NTFS or FAT.		least 'Change' permissions on the folders and files available through the share.
The Scheduler database resides on the local machine or a remote machine. The file system in use is NTFS.	The folder or file security permissions prevent write access to the database.	Alter the folder and file security permissions to grant the Scheduler service account at least the minimum permissions required for the Scheduler database.
The Scheduler service is configured to execute under the context of the local system account. The database is on a remote machine.	The local system account uses the 'Anonymous' login when is accesses files on remote machines.	Change the Scheduler service account context to a domain account with the required permissions on the database folder and file **OR** Grant the 'Everyone' group the required permissions on the database folder and file.
The Scheduler service is configured to run under the context of a local machine account. The database is on the local machine and is accessed with a local path. The file system in use is NTFS.	The local machine account must have permission to access the database file.	Grant the local machine account the required permissions to the database folder and file.
The Scheduler service is configured to run under the context of a local machine account and the database is on a remote machine. The file system in use is NTFS or FAT.	A local machine account is unable to access files across the network.	Configure the Scheduler service account to run under the context of either the Local System account, or a domain account.
The Scheduler service is configured to run under the context of the local system account and the database is on the local machine.	The local system account is a member of the Administrators local group. The Administrators local group does not have permission to access the database file.	Grant the Administrators local group the required permissions on the database folder and file.

Could not start the iBase Scheduler Service on Local Computer.
Error 1053: The service did not respond to the start control request in a timely fashion.

This error message is displayed immediately after an attempt is made to manually start the Scheduler service by using the Windows Services application.

© N. Harris Computer Corporation

Refer to the previous error message if this error message is displayed thirty seconds or more after an attempt is made to start the service. However, when the error message is displayed immediately after an attempt to start the service the possible causes for this error are as follows:

Configuration	Possible Cause	Solution
N/A	The Scheduler database is moved.	Open the Scheduler Configuration dialog and choose the new location of the Scheduler database when prompted. After the location of the database is verified and the Scheduler Configuration utility starts, try starting the service again from the **Services control** pane.
The Scheduler service is configured to run under the context of a local machine account. The database is on the local machine and is accessed by a local path. The file system in use is NTFS.	The local machine account must have permission to access the database file.	Grant the local machine account the required permissions to the database directory and file.
The Scheduler service is configured to run under the context of a local machine account and the database is on a remote machine. The file system in use is NTFS or FAT.	A local machine account is unable to access files across the network.	Configure the Scheduler service account to execute under the context of either the Local System account, or a domain account.
The Scheduler service is configured to run under the context of the local system account and the database is on the local machine.	The local system account is a member of the Administrators local group. The Administrators local group does not have permission to access the database file.	Grant the Administrators local group the required permissions on the database directory and file.

Scheduler Configuration utility errors

The Microsoft Jet database engine cannot open the file '<scheduler_database_file>'. It is already opened exclusively by another user, or you need permission to view its data.

This error is displayed when the configuration application is started.

Configuration	Possible Cause	Solution
The Scheduler database is on either the local machine or a	The user that is attempting to access the database does not have sufficient permissions to	Grant the user the required permissions to the Scheduler database folder and file.

© N. Harris Computer Corporation

Configuration	Possible Cause	Solution
remote machine. The file system in use is NTFS.	the Scheduler database folder or file.	

Unable to write to the database - you may not have permission.

This error is displayed when an attempt is made to save or edit a connection, task, or trigger.

Configuration	Possible Cause	Solution
The Scheduler database is on the local machine or a remote machine and is accessed through a folder share by using a UNC path. The file system in use is either NTFS or FAT.	The user that is attempting to access the Scheduler database has sufficient rights to read from the database, but not to modify its contents. The folder share permissions for the user are set to 'Read' for the user.	Alter the folder share permissions to grant the user 'Change' permissions on the folders and files available through the share.
The Scheduler database is on the local machine or a remote machine. The type of file system in use is NTFS.	The user that is attempting to access the Scheduler database has sufficient rights to read from the database, but not to modify its contents. The folder or file security permissions prevent write access to the database for the user.	Alter the folder and file security permissions to grant the user at least the minimum permissions required for the Scheduler database.

Unable to delete the connection/task/trigger -- you may not have permission.

This error is displayed when an attempt is made to delete a connection, task, or trigger. Refer to the previous error for possible solutions.

Could not connect to the iBase database for the '<connection_name>' connection.

The database is moved or deleted, or your connection details are incorrect.

This error message is displayed when either the Scheduler Configuration dialog is started, or an entry is selected from the list of connections.

Configuration	Possible Cause	Solution
N/A	The iBase security file has the 'Read Only' attribute enabled.	Disable the 'Read Only' attribute on the iBase security file.
The iBase connection and security files is on the local machine or a remote machine and are accessed through a folder share by using a UNC path. The file system in use is either NTFS or FAT.	The user that is attempting to access the iBase database does not have permission to access this share, or they do not have permission to change the files and folders available through the share.	Ensure the user that is accessing the iBase database has at least 'Change' permission on the folder share.

© N. Harris Computer Corporation

Configuration	Possible Cause	Solution
The iBase connection and security files reside on a local machine or remote machine. The file system in use is NTFS.	The folder or file security permissions are preventing `write` access to the iBase security file for this user.	Alter the folder and file security permissions to grant the user at least the minimum permissions required for the iBase database and security file.
The iBase security file is on a remote machine and is accessed through a folder share by using a UNC path. The file system in use is NTFS or FAT.	The folder share permissions prevent write access to the security file for the Scheduler service account.	Alter the folder share permissions to grant the Scheduler service account 'Change' permissions on the folders and files available through the share.
The iBase security file is on the local machine or a remote machine. The file system in use is NTFS.	The folder or file security permissions prevent write access to the security file for the Scheduler service account.	Alter the folder and file security permissions to grant the Scheduler service account at least the minimum permissions required for the iBase security file.

Log errors

The file <file_name> does not exist or there are insufficient permissions on directory to read the file.

The user successfully created a file trigger through the Scheduler Configuration dialog, but the service immediately logs the error in the Scheduler log.

Configuration	Possible Cause	Solution
The trigger file is on the same machine as the Scheduler Configuration utility and is accessed through a local path. The file system in use is either NTFS or FAT, and the Scheduler service is on a remote machine.	The Scheduler service is looking for the file on the machine that the service is running on. This file does not exist on this machine.	Create a folder share on the Scheduler Configuration utility machine and reference the trigger file through this share.
The trigger file is on either the local machine or a remote machine and is accessed through a folder share by using a UNC path. The file system in use is either NTFS or FAT, and the Scheduler service is on either the local machine or the remote machine.	The Scheduler service account does not have permission to view the trigger file **OR** The Scheduler service account does not have access to the folder share.	Ensure that the Scheduler service account has at least the minimum folder and file security permissions required for the trigger file **OR** Add the Scheduler service account to the list of authorized users on the folder share. The account must have a minimum of 'Read' permissions on the files and

© N. Harris Computer Corporation

Configuration	Possible Cause	Solution
		folders available through the share.

Login failed for user '<windows_user_name>'

The Scheduler service creates an entry in the log when a batch import or export specification is run.

Configuration	Possible Cause	Solution
SQL Server is hosting an iBase SQL Server database **AND** The Scheduler service is running under the context of a domain account **AND** The iBase database connection file is configured to access the SQL Server machine by using Windows security.	The Scheduler service account does not have a corresponding login, which uses the Windows authentication security model, on the SQL Server machine.	Create an SQL Server login that uses the Windows authentication model and uses the domain account in use by the Scheduler service **AND** Grant the SQL Server login rights to access the iBase database on the SQL Server machine.

Login failed for user 'NT AUTHORITY\SYSTEM'

The Scheduler service creates an entry in the log when a batch import specification is run.

Configuration	Possible Cause	Solution
SQL Server is hosting an iBase SQL Server database. The `BUILTIN\Administrators` group has been removed from the SQL Server security configuration **AND** The Scheduler service is on the same machine as the SQL Server instance that is hosting the iBase database and is running under the context of the local system account.	The local system account does not have rights to access the iBase database **OR** The local system account is a special account with administrative rights on the local machine and is a member of the Administrators group at the operating system level **OR** Removing the `BUILTIN\Administrators` account from SQL Server has revoked rights to the database for the local system account.	Create a security login within SQL Server with the name `NT AUTHORITY\SYSTEM` that uses the Windows authentication model and add this new login to the System Administrators server role **OR** Configure the Scheduler service to run under the context of a domain account that has rights within SQL Server to access the iBase SQL Server database.

Login failed for user '<domain_name>\<computer_name>'.

The Scheduler service creates an entry in the log when a batch import specification is run.

Configuration	Possible Cause	Solution
SQL Server is hosting an iBase database that is on a different machine to the machine on which the Scheduler Service	The local system account can only access iBase databases that are hosted on the same	Change the Scheduler service configuration to run under the context of a domain account **OR** Create a SQL Server login that

© N. Harris Computer Corporation

Configuration	Possible Cause	Solution
is running **AND** The Scheduler service is running under the context of the local system account **AND** The iBase database connection file is configured to access the SQL Server using Windows security.	machine as that running the service.	uses the Windows authentication model and uses the domain account in use by the Scheduler service **OR** Grant the SQL Server login rights to access the iBase database on the SQL Server machine.

Logon failed for user '(null)'. Reason: Not associated with a trusted SQL server connection.

The Scheduler service creates an entry in the log when a batch import specification is run.

Configuration	Possible Cause	Solution
The Scheduler service is running under the context of a local machine account **AND** SQL Server is hosting an iBase SQL Server database on a different machine to that running the Scheduler service **AND** The iBase database connection file is on the same machine as the Scheduler service, and is configured to access the SQL Server by using Windows security.	It is not possible to grant a local Windows account on one machine access to an SQL server instance on another machine.	Change the Scheduler service configuration to run under the context of a domain account **OR** Create a SQL Server login that uses the Windows authentication model and uses the domain account in use by the Scheduler service **OR** Grant the SQL Server login rights to access the iBase database on the SQL Server machine.

Cannot open database requested in login '<database_name>'. Login fails.

The Scheduler service creates an entry in the log when a batch import specification is run.

Configuration	Possible Cause	Solution
The Scheduler service is running under the context of a domain account **AND** SQL Server is hosting an iBase SQL Server database **AND** The iBase connection is configured to access the SQL Server by using Windows security.	The Scheduler service account does not have sufficient rights within SQL Server to access the iBase database.	Grant the SQL Server login for the Scheduler service domain account rights to access the iBase database on the SQL Server machine.

Invalid procedure call or argument.

The Scheduler service creates this entry in the log when a task action is run.

© N. Harris Computer Corporation

Configuration	Possible Cause	Solution
The task that is run has a Program Action that is to be run.	The Program Action is defined to run an operating system batch file **OR** The batch file has no content.	Remove the batch file from the list of actions for the task **OR** Ensure that the batch file is not zero bytes by entering a carriage return into the file.

Permission Denied file path: <path>.

The Scheduler service creates this entry in the log when a batch import specification is run and an attempt is made to read an import file.

Configuration	Possible Cause	Solution
The Scheduler service is running on the same machine as the import file location.	The Scheduler service has insufficient permissions to open the import file.	Grant the Scheduler service account the required permissions to the import files.
The Scheduler service is operating under the context of the Local System account. The import files is on a remote machine.	The Scheduler service has insufficient permissions to open the import file.	Change the Scheduler service account to a domain account with the required permissions on the import files **OR** Grant the 'Everyone' group the required permissions on the import files.

Path/File access error.

The Scheduler service creates this entry in the log when a batch export specification is run and an attempt is made to create or overwrite an export file.

Configuration	Possible Cause	Solution
The Scheduler service is running on the same machine as the export file location.	The Scheduler service has insufficient permissions to create an export file in the folder specified.	Grant the Scheduler service account the required permissions to the folder in which the export file is to be created.
The Scheduler service is operating under the context of the Local System account. The export file folder is on a remote machine.	The Scheduler service has insufficient permissions to create an export file in the folder specified.	Change the Scheduler service account to a domain account with the required permissions on the folder intended for export files.

Miscellaneous problems

A task that is set up to execute on the change of a trigger file executes more than once, or does not execute when expected.

© N. Harris Computer Corporation

Configuration	Possible Cause	Solution
The Scheduler service and the trigger files are on two separate machines.	The machine on which the Scheduler service is executing has a different date, time, or both to the machine on which the trigger file is.	Ensure that the Windows Time service is running on each machine. This automatically synchronizes the times on the two machines with the domain controller **OR** Manually synchronize the times on the two machines by using the NET TIME command.

The Scheduler database lock file (Scheduler.ldb) persists in the directory after both the service and configuration applications have been shut down.

The `.ldb` file is a Microsoft™ Jet locking file that is removed if the applications are shut down cleanly.

Configuration	Possible Cause	Solution
N/A	The last application to use the Scheduler database stopped abnormally.	The file will be removed the next time an application that uses the Scheduler database is stopped cleanly and it is the last application that is accessing the database.
The database is accessed through a folder share. The file system in use is NTFS or FAT.	The folder share permissions are preventing users from deleting files available through the share.	Alter the folder share permissions to allow users that access the Scheduler database with the Scheduler Configuration utility permissions to delete the Jet locking file.
The database is accessed through a folder share. The file system in use is NTFS.	The folder and file security permissions are preventing users from deleting the Jet locking file.	Alter the folder and file security permissions to grant the users that use the Scheduler Configuration utility at least the minimum permissions required for the Scheduler database.

Managing access to data and functions

You can install and operate i2® iBase and ensure that access is securely controlled. Access to data and functions is based on the user name and password that are specified by the user when they log on

© N. Harris Computer Corporation

to iBase, but to fully ensure control of the data it is also necessary to manage the physical and logical security of other aspects of the whole system.

Setting up iBase clients

After you create the iBase database, you might need to configure each iBase installation to suit the work to be done in that database.

Setting options on local machines

You can specify some general options for how users use iBase in the Options (on the **Tools** menu in either iBase or iBase Designer). You do not need to log on to a security file or open a database, the options apply to the application not to any databases.

Page	**Detail**
General	Basic options for using iBase, for example, how users use categories. Any user can change these options without affecting any other user of the machine. These options take effect immediately.
Charting	Defaults that are used when charting in Analyst's Notebook® (unless specified otherwise in a charting scheme or the Charting Settings dialog). Any user can change these options without affecting any other user of the machine. These options take effect the next time records are charted.
Advanced	User options that can be changed by any user of the machine; changes to the Local Machine Settings affect all users of the computer. You can prevent users from changing the local machine settings by denying `write` access to the `Settings.xml` file.

Hiding plug-ins

You can deny access to certain iBase features for any user that is working on a specific machine. This is a local machine setting and can be copied between machines.

For more refined control set at the user group level, you might prefer to define System Commands Access Control groups.

Copying the local machine settings to other machines

The local machine settings in the **Options** dialog and **Plug-In Manager** are saved in the file `Settings.xml`. You can copy this file to the other machines on which iBase is installed. For details of where to copy the file, see Installation and Application Data Folders.

© N. Harris Computer Corporation

SQL Server connection files on local machines

Users should connect to iBase databases that are in SQL Server format by using connection files stored in a shared folder on the network.

Copying connection files to client machines might compromise the security of your system and adds to the administrative workload. If it becomes necessary to copy a connection file, then the file name and path must be identical on each machine to which you copy it.

Setting and maintaining user access rights

This topic describes how you can manage access to iBase by using features within iBase and also other measures outside of the control of iBase.

iBase application security

iBase provides the tools that are needed to set and maintain the access rights for groups of users, controlling what they can do within the application. One security file can control access to any number of databases. After a user has access to a database, they are subject to the following types of security:

- Database-wide permissions, to read or alter data
- Command access or denial
- Usage monitoring, with audit logs
- Folder object control, to provide private storage of analysis methods

An iBase SQL Server installation also supports Extended Access Control (EAC) that provides an extended range of Data Access Control features so that you can manage the degree to which users can manipulate data.

Security measures outside of iBase

There are measures must take outside of iBase to ensure that only authorized users access iBase and its data. The degree to which you want to exercise these external controls depends on your own data and environment. Managing the overall security of your system is likely to result in more administrative work. For this reason, it is important to apply only the degree of control that you need.

It is good practice to implement the controls that are discussed in this document incrementally to isolate the effect of each change. It is a necessary implication of managing access that some people are denied rights. For this reason, it is important that changes are tested and documented so that if anyone loses the ability to work with the database the cause can be easily established.

To fully control access to the data managed by iBase, you must to understand and implement:

- Access to the network on which iBase runs
- physical access to the iBase design and security features
- logical access to the iBase design and security Features

Some of these features require that particular users or groups of users have user identities that provide direct or indirect access to iBase data. The number of such users must be restricted, and the identities and passwords are held securely.

© N. Harris Computer Corporation

Physical access to iBase Designer

For maximum security, it is advisable to limit physical access to i2® iBase Designer and its capabilities. If a user cannot gain access to the tools that implement access control for the application, they cannot modify the settings and compromise security.

There are several precautions that can be taken to minimize the possibility of unauthorized access to iBase Designer.

When you install iBase, limit the number of machines on which iBase Designer is installed. For example, do not install the following unless necessary:

- **Designer**
- **Tools** (including the Database Configuration utility and the Audit Viewer)

The smaller the group that has access to iBase and the machines on which it runs, the easier it is to manage access control. Strategies that place the machines and database on a separate network without contact with other parts of the organization are inherently easier to control than general installations.

iBase is not designed to operate over a WAN. Implementation over a WAN impairs performance and also increase the exposure of the database to a wider audience through the network infrastructure.

Logical access to iBase (roles)

iBase application security provides three system roles to control a user's access to the database design and administration functions in iBase Designer. To manage access, you need to ensure that users have the appropriate roles.

Role	**Description**
Database Creator	This role should only be granted to users who need to create databases.
Database Administrator	This role allows a user to create and modify the definition of the database by adding, amending, and deleting the information that defines the entity and link types and the associated information. It also allows users to complete database administration tasks. A member of a group with this role can see the complete definition of the database and can view records. This is essential to allow a user to define and maintain the database.
Security Administrator	This role allows a user to create and modify group definitions and their access settings. It also allows a user to use Audit Viewer to manage audit logs. Security administrators cannot list records in iBase Designer; but they can list records in iBase and show individual records.

A user who through group membership has all three roles along with the full set of database management permissions is known as a System Administrator and has rights to complete any task

© N. Harris Computer Corporation

in iBase. For detailed information on setting up administrative users, see the Administration Center document Security Files, Users and Groups.

To design and administer the database, a user must be able to see the structure of the data. Any user to whom you grant access to iBase Designer using an iBase user name with a Database Administrator or System Administrator role is able to examine the entity and link types. An iBase System Administrator always has access to all of the data records to be able to check that changes they make are working as they expect. In contrast, an iBase Database Administrator can be denied access to all or specified records in iBase by a Security Administrator, but this might complicate database administration.

You should minimize the number of people who are given user accounts that include Database Administrator or Security Administrator roles.

iBase files on the network (permissions)

The iBase security file and database are managed through the security (`.ids`) and database (`.idb`) files.

For Microsoft™ Access format security files and databases, the `.ids` and `.idb` files contain all the data in the iBase system (apart from the audit log data held in the `.idl` file). However, when you use the SQL Server format security file and databases, these files contain only the information that is required to connect to the SQL Server instance and databases. All other data is stored within the SQL Server databases.

For successful operation of iBase, all users need the ability to create and delete files in the folder that contains the security file and database. This is because iBase creates and deletes multi-user lock files in this folder.

It is important for both access control and normal system management that users are prevented from deleting the security (`.ids`) and database (`.idb`) files. You should apply security settings to these folders and files to prevent this.

The access permissions that are required on the iBase security file and database depends on whether Microsoft Access or SQL Server is in use for storing the data:

- It is necessary for iBase to write to the Access format security file whenever a user logs in to the security file because a Microsoft Access format security file contains the Security Audit log.
- When a SQL Server format security file is used, iBase only needs to write to the Microsoft Access connection file when the database properties or connection details are changed. This operation can be performed through iBase Designer or the Database Configuration utility by an iBase user with both the Security Administrator and Database Administrator roles only.

The following tables detail the permissions necessary on the folder, security file, and database files for the various configurations and roles in the iBase system.

Config 1: Microsoft Access security file and database	**Role**	**Minimum Windows permissions**
Folder containing the iBase security file and database	All roles	Modify
Security file	All roles	Read, Write
Database file	All roles	Read, Write

© N. Harris Computer Corporation

Config 1: Microsoft Access security file and database	**Role**	**Minimum Windows permissions**
Audit log file	All roles	Read, Write

Config 2: Microsoft Access security file and SQL Server databases	**Role**	**Minimum Windows permissions**
Folder containing the iBase security file and database	All roles	Modify
Security file	All roles	Read, Write
Database connection file	Users with Security Administrator role	Read, Write
	All other roles	Read
Audit log file	-	-

Config 3: Microsoft SQL Server security file and databases	**Role**	**Minimum Windows permissions**
Folder containing the iBase security file and database	All roles	Modify
Security connection file	Users with Security Administrator roles	Read, Write
	All other roles	Read
Database connection file	Users with both Security and Database Administrator roles	Read, Write
	All other roles	Read
Audit log file	-	-

SQL Server login and user accounts

The SQL Server installation must be set up and configured for iBase users.

The main activities in managing security for iBase in an SQL Server installation are to:

- Set up login accounts and user accounts for iBase users.
- Deny access to the iBase databases from applications other than iBase.

© N. Harris Computer Corporation

- Allow access to other applications that must access the database such as the backup tool or the Microsoft Search service that creates the index for Full-Text Search.

Note: Access from iBase to an SQL Server database is managed through an application role. Application roles are an SQL Server mechanism that allows external applications to assume control of a database in a secure manner. Database access in Analyst's Notebook through the iBase API.

In an SQL Server installation of iBase, each user must gain access to an instance of Microsoft SQL Server through a login that establishes the user's ability to connect (authentication). This login is then mapped to an SQL Server user account, which is used to control activities performed in the database (permissions validation). Therefore, a single login is mapped to one user account that is created in each database the login is accessing. If no user account exists in a database, the user cannot access the database even though the user may be able to connect to an instance of SQL Server.

Almost all users that work with iBase need only have permissions to access the SQL Server database. They do not need any other permissions. To configure this, the SQL Server administrator should use the public database role. Every user of a database will be given the public database role.

In an SQL Server installation, the default permissions for this role allow a user to access system tables and run predefined stored procedures. It does not allow any access to any of the user tables that hold the iBase data. All access to user tables is handled by iBase. If a user only has the public database role they cannot use other applications to access the data in the database. This will prevent them from using tools such as Microsoft Query, Enterprise Manager or Management Studio to see data in the database.

In addition, the SQL Server login that is used to connect to the SQL Server database must have execute permission on the `sp_help_jobschedule` stored procedure in the `msdb` system database. SQL Server administrators can grant this permission by editing the properties of the Public database role defined in the `msdb` database.

SQL Server administrators can further reduce access to the database by removing access to all of the system tables within the database from the public role. This can be done by accessing the roles for the database and changing the permissions to not allow access. This measure prevents an iBase user with valid access from using the public role to open the database from another tool and reading the contents of the system tables which would, for instance, allow them to determine the names of tables and fields.

iBase security

All users of an iBase database must have access to the database (security file) which secures the main iBase database.

The name of the database is provided by the iBase security administrator at the time of creation. The security database on the SQL Server machine will have the SQL Server equivalent name of the Microsoft Access component with the suffix `_sec`. For example, a security file that has been given the name Vehicle Crimes will have a connection file named `Vehicle Crimes.ids` and an SQL Server database name of `Vehicle_Crimes_sec`.

iBase database

iBase manages the way that users can access the data within the database. Users can access only the data for which they have access permissions, which are set using the optional iBase Extended Access Control (EAC) module.

The name of the database file is provided by the iBase administrator at the time of creation. The database on the SQL Server machine will have the SQL Server equivalent name of the Microsoft Access connection file. The SQL Server database does not have a suffix. For example, a database file that has been given the name Vehicle Crimes will have a connection file named `Vehicle Crimes.idb` and an SQL Server database name of `Vehicle_Crimes`.

© N. Harris Computer Corporation

Audit log database

In an SQL Server installation, iBase creates an audit log database alongside the main database. The name of the database is the same as the main database name with the suffix `_log`. For instance, the database Vehicle_Crimes has an audit log database `Vehicle_Crimes_log`.

You must also ensure that iBase users can access this audit log database. If you do not provide access to the audit log database iBase attempts to create a new audit log database and fail with a message that says it could not do so successfully.

Managing data access

The model database is used by SQL Server as a template for all new databases that are created. Setting the permissions of the public database role on the model database ensures that newly created security databases, iBase databases, or audit databases copy the permissions that are set on the model database. This applies to all databases created on the server and not only those that are created through iBase.

In a Microsoft Windows environment, a simple way to manage data access is to use a Windows group for iBase users. For instance, create a windows group with the name `iBase_users` and add to it all users that need access to the database. In SQL Server, create a login for this group and grant it access to the iBase database (or even to the model database). Do not give this login any server roles or database roles other than public.

SQL Server login for creating databases in iBase Designer

When you create SQL Server databases using iBase Designer, users must have the iBase role Database Creator. These users must also have a corresponding SQL Server login that is a member of the `dbcreator` fixed server role.

A simple way to manage the creation of databases is to have a dedicated SQL Server login that uses either SQL Server or Windows authentication and is a member of the `dbcreator` fixed server role. iBase administrators use this login whenever they create new SQL Server databases or convert Microsoft Access databases to SQL Server. The databases are created with this login mapped to the `dbo` user.

The login used to create the database will continue to have `dbo` rights to the database and audit log database. It is important that either the login and password are kept secure, or the ownership of the databases is changed by an SQL Server administrator.

Note: After creating an iBase database, an iBase administrator can change the connection file so that users will connect to the database via a less powerful SQL Server login. The SQL Server administrator must grant this login access to the database on the SQL Server instance. For details of the login, see section above on managing data access. Alternatively, they can change the connection to use Windows authentication. For further information, see Managing SQL Server Connection Settings.

Working with `BUILTIN\Administrators`

Any Windows account that is an Administrator of the machine on which SQL Server is running is automatically a member of the SQL Server `BUILTIN\Administrators` group. In some SQL Server installations, it is a policy that this powerful group is removed so that SQL Server administration is separate from administration of the server machine on which it runs. If this group is removed or modified, it will be necessary to take additional measures to ensure that SQL Server and iBase continue to function.

© N. Harris Computer Corporation

Attention: If the `BUILTIN\Administrators` login is removed, ensure that there is at least one other login that is a member of the `System Administrators` fixed server role. If this login is validated using SQL Server authentication, also ensure that the SQL Server is configured to allow both SQL Server and Windows authentication.

If, during the installation of the SQL Server instance, the option was chosen to run the SQL Server service and SQL Server Agent service under the context of the `localsystem` account, it will be necessary to set up specific

Service	**Windows user account**
SQL Server	This user account should only be a member of the Windows group Domain Users.
SQL Server Agent	This user account usually only needs membership of the Windows group Domain Users, however to perform certain advanced tasks, for example, executing xp_cmdshell or using the AutoRestart feature of the SQL Server Agent service, it will also need to be a member of the Local Administrators group.

You may need to provide a login for backup operations. This login will need to have the `db_backupoperator` database role.

Access for Full-Text Search

The Microsoft Search Service must have permission to read the database. By default this service runs as `localsystem` and has access to the databases on the server via the `BUILTIN\Administrators` login.

If you remove access for `BUILTIN\Administrators` login you must add a new login with the name:

`NT AUTHORITY\System` (notice the space between NT and AUTHORITY)

This login must be a member of the SQL Server `System Administrators` fixed server role.

Use iBase with other products

iBase provides a sophisticated security model that controls all aspects of access to the data and functions. iBase users must provide a username and password to enter the iBase system, and determine the level of permissions.

If iBase Extended Access Control is used, some users might be able to read any data, but not add or modify anything; other users might not be able to see specific entity or link types, specific data fields, or individual records.

Consider carefully the folders in which data extracted from iBase in the form of charts, reports, or export files are stored. Ensure that these folders are given appropriate access permissions. After data is outside of iBase, it is potentially exposed to unauthorized access and aspects of the iBase database schema and data can be revealed.

Attention: Data that is taken from an iBase database, in the form of a chart, a report, a map or as an export file for use in a third-party system, is no longer protected by iBase access control.

© N. Harris Computer Corporation

The security implications of exporting data from iBase for use in other third-party products must always be carefully considered.

Analyst's Notebook and Chart Reader

Many customers use Analyst's Notebook with iBase to extend their visualization and analytical capabilities. When using Analyst's Notebook® with iBase, the user can only chart data and include those field values to which they are permitted access in the iBase database. In particular, consideration must be given to data that is exposed in the Analyst's Notebook chart by using cards, attributes, and data records, which might reveal the presence of a particular database field to which the user that creates the chart has access, but which is hidden from other users.

Chart Reader enables chart data, extracted from an iBase database and charted using Analyst's Notebook, to be shared as a snapshot of the database content with other non-iBase users. Some organizations are happy to send charts as email attachments for browsing using Chart Reader within, or even outside, their organization. However, after is is e-mailed, information tends to disseminate further in unexpected ways. Ensure that iBase data that is exposed in this way complies with your organization's data distribution and disclosure policies.

iBase GIS Interfaces

When using the optional i2® iBase Geographic Information System Interfaces to plot iBase data on a map, the user can only plot data and transfer those field values into the mapping application to which they are permitted access in the iBase database. Caution should be exercised if it is intended to save the data in a map for sharing with others who may not have the same level of data access permission as the iBase user who originally plotted the data on the map.

Physical security

After extracted from the iBase database and printed as charts, reports or maps, your data is more accessible.

You should review your physical security if the printouts of extracted data are going to be used in parts of your premises that are not usually used by iBase users. Is it permissible for iBase data and details of your iBase database schema to leave your secure working areas? If so, you must plan and restrict how much you do want to reveal.

Deploying Analysis Studio

Analysis Studio adds significant new functionality to an existing deployment of i2 iBase. In particular, it enables analysts to enrich their investigations with records from i2 Connect data sources.

After you deploy all the components of Analysis Studio, analysts can:

- Use **External Searches** in the Analyst's Notebook desktop client to search for information in data sources that are connected through i2 Connect.
- Visualize records from both iBase and i2 Connect data sources on charts in the Analyst's Notebook desktop client.
- Use iBase records as the starting point for external searches. For example, an analyst might look for a record that shares field values with an iBase record.
- If they have the Export-to-iBase plug-in, export records from connected sources from the Analyst's Notebook desktop client directly into iBase.

© N. Harris Computer Corporation

To provide these features to the analysts in your organization, you need to understand the iBase database that you already have, and the i2 Connect connectors that you want to use. Everything else is about joining those two things together.

Understanding Analysis Studio and iBase

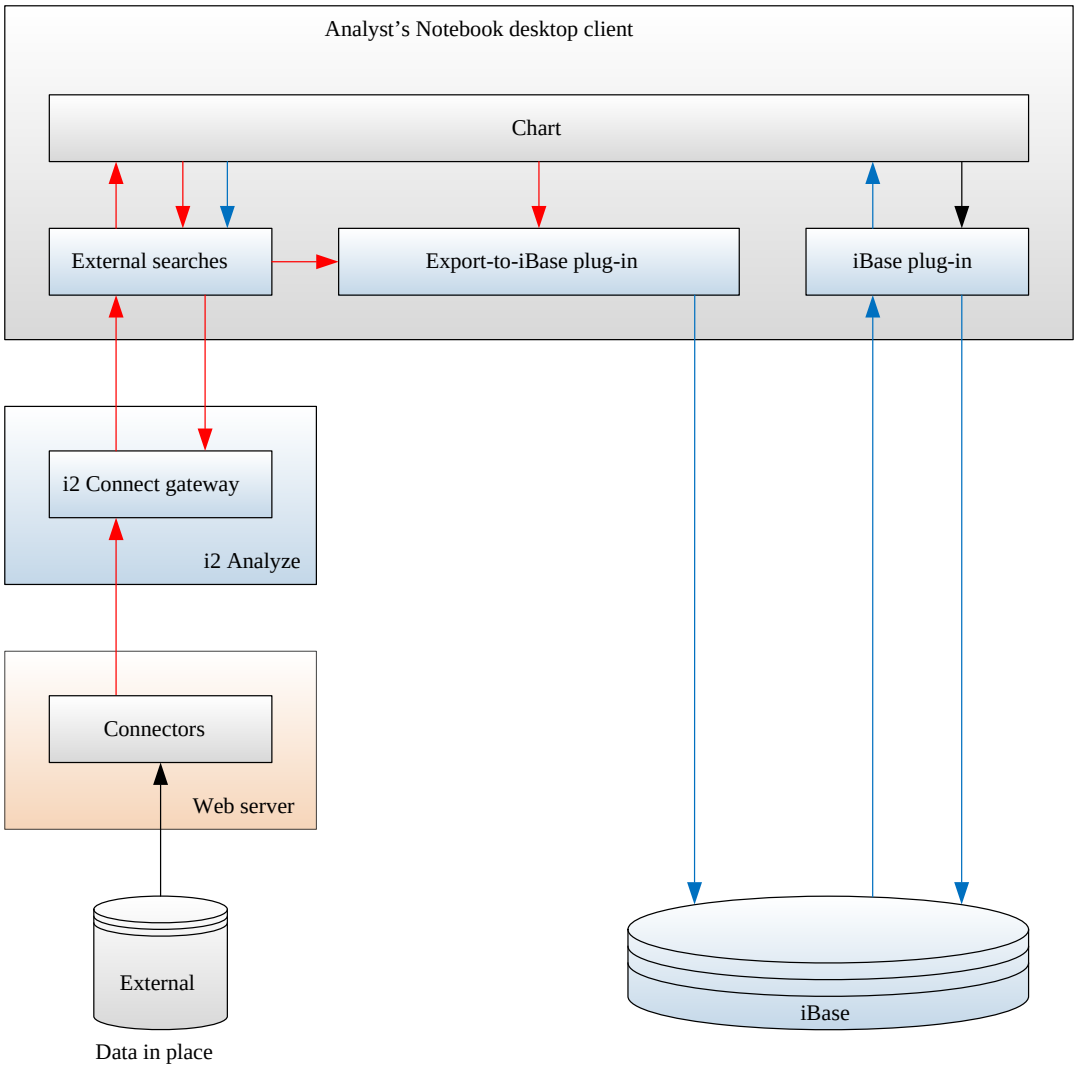
iBase users have always been able to use the iBase plug-in to create and edit iBase records on Analyst's Notebook charts. With Analysis Studio, they gain the ability to retrieve records from additional data sources, to interact with those records on charts, and the potential to add the information they contain to iBase.

In the Analyst's Notebook desktop client, the external searches that users can run are provided by *connectors* to data sources. Connectors define what kind of information they can retrieve, and what information users must provide in order to use them successfully.

Connectors are enabled by i2 Connect, which in turn is underpinned by i2 Analyze, which is i2's enterprise-grade, server-based analytics platform. You can buy connectors from third-party suppliers, or create them for yourself with the i2 Connect Node SDK.

You can think of a complete deployment of Analysis Studio and iBase like this:

© N. Harris Computer Corporation



The elements on the right side of the diagram should be familiar: they show the interaction between Analyst's Notebook, the iBase plug-in, and the iBase database.

The blue arrows represent the iBase plug-in's ability to create records in the iBase database, and to retrieve records from it and visualize them on a chart.

The elements on the left side of the diagram show how i2 Connect enables external searches in the Analyst's Notebook desktop client. i2 Analyze hosts the i2 Connect gateway, which provides the interface between the desktop application and the connected data sources.

The red arrows represent *i2 Analyze* records being created by connectors and eventually being visualized in charts. They also show the Analyst's Notebook Premium desktop client using i2 Analyze and iBase records as the starting point for further queries of external sources.

© N. Harris Computer Corporation

In the center of the diagram, the Export-to-iBase plug-in acts as a translator between iBase and i2 Analyze. With the plug-in installed, users can export i2 Analyze records from the Analyst's Notebook Premium desktop client, to create or augment records in iBase.

The following documentation describes how to deploy and then maintain Analysis Studio so that your iBase users can derive maximum value from it.

Deploying Analysis Studio with iBase

Deploying Analysis Studio to work with iBase means deploying i2 Analyze on a server and the Analyst's Notebook desktop client on workstations. To make the combination work, you must configure both iBase and i2 Analyze so that they can exchange data effectively.

The following topics describe the actions that you need to take, for a successful deployment of Analysis Studio that works with an existing iBase deployment.

Preparing iBase and generating an i2 Analyze schema

In successful deployments of Analysis Studio that work with iBase, the data structures of i2 Analyze and iBase records are as closely aligned as possible. To enable the Export-to-iBase plug-in, you must also prepare the existing iBase database so that the plug-in can communicate with it.

First, make the following changes to the iBase database:

1. If you haven't done so already, upsize the iBase database to SQL Server. The Export-to-iBase plug-in is not compatible with Microsoft Access databases.
2. Use the charting scheme editor to modify at least the first charting scheme in the list so that **Data Records: "All"** is set as the option for all entity and link types.

   The Export-to-iBase plug-in requires iBase records on the chart surface to contain all possible fields, no matter if they contain values.

   Important: All Analyst's Notebook desktop client users must use the same charting scheme to control how iBase records appear on charts.

Second, you must export the iBase schema in a form that i2 Analyze can use to define the structure of the records that it retrieves. Configure and run the command that creates an i2 Analyze schema from the iBase schema:

1. Install i2 Analyze on the server that will host your i2 Analyze deployment. The installation includes the toolkit whose commands you will use in the rest of this procedure.
2. The toolkit command that creates the i2 Analyze schema must be able to connect to the iBase database. Create and populate a file named `iBaseConnection.properties` in the `toolkit/scripts` directory of the i2 Analyze installation. The file provides information about the iBase database in the following form:

   ```
   DBUsername=[the name of a SQL Server user with permission to read the
    iBase database]
   DBPassword=[the password of the above user]
   DBName=[the name of the iBase database]
   DBServer=[the fully qualified name of the server]
   DBPort=[the port number on which SQL Server runs - for example, 1433]
   ```
3. Before you can create the i2 Analyze schema, you need to start the i2 Analyze deployment process. You'll learn more later, but for now:

   a. Navigate to the `toolkit\examples\configurations\daod-opal` directory of the i2 Analyze installation.

© N. Harris Computer Corporation

b. Copy the `configuration` directory that `daod-opal` contains to the root of the toolkit, so that it becomes `toolkit\configuration`.

4. The toolkit needs to communicate with the iBase database:

   a. Download the Microsoft JDBC Driver 7.4 for SQL Server archive from https://www.microsoft.com/en-us/download/details.aspx?id=58505.

   b. Extract the contents of the download, and locate the `sqljdbc_7.4\enu\mssql-jdbc-7.4.1.jre11.jar` file.

   c. Copy the `mssql-jdbc-7.4.1.jre11.jar` to the `toolkit\configuration\environment\common\jdbc-drivers` directory.

5. Run the `generateAnalyzeSchemaFromIBase` toolkit command to generate an i2 Analyze schema and charting scheme from the iBase schema. It supports the following parameters:

   - `locale` - The locale of the iBase deployment, which defaults to `en_US`.
   - `outputPath` - The absolute path to the output directory, which defaults to `C:\i2\i2analyze\toolkit\scripts\output` on Windows, or `/opt/i2/i2analyze/toolkit/scripts/output` on Linux.
   - `connectionProperties` - The absolute path to the connection properties file that you created earlier.
   - `fileNamePrefix` - The prefix that's given to all the generated schema and charting scheme files, which defaults to `ibase`.

   For example:

```
setup -t generateAnalyzeSchemaFromIBase -p locale="en-US"
 -p outputPath="C:\i2\i2analyze\toolkit\scripts\output"
 -p connectionProperties="C:\i2\i2analyze\toolkit\scripts
\iBaseConnection.properties" filenamePrefix="ibase"
```

For troubleshooting assistance, and for more details about the generated schema and its limitations, see the reference information.

Deploying i2 Analyze with the exported iBase schema

After you've *installed* i2 Analyze and created the schema, the next step is to *deploy* i2 Analyze with the i2 Connect gateway that enables the use of connectors. You also need to configure i2 Analyze so that your iBase users can log in.

You started the process of deploying i2 Analyze for use with iBase when you created the `toolkit/configuration` directory in Preparing iBase and generating an i2 Analyze schema. To complete the deployment:

1. Follow the instructions in the documentation to create a schema development environment for i2 Analyze.

   A "schema development environment" of i2 Analyze provides only the i2 Connect gateway, which is what you need here.

   Important: In the procedure, you can ignore the second part of Step 1 (installing the Analyst's Notebook desktop client) for now, and you have already completed the copy in Step 2. In Step 5, instead of creating a schema, use the schema and the charting schema that you created from iBase.

2. Configure Liberty so that iBase users can log in to the i2 Analyze server and run external searches.

© N. Harris Computer Corporation

iBase and i2 Analyze manage their users independently. However, it's easier to use the Analyst's Notebook desktop client with iBase and i2 Analyze when both environments use the same user registry type.

For example, if iBase uses a local user registry, then configuring the same user registry in i2 Analyze means that there is only one set of users and groups to manage. See the documentation about configuring the Liberty user registry in i2 Analyze.

Alternatively, if iBase is configured to use Active Directory, then configuring i2 Analyze to use the same directory will give a similar result.

See the Open Liberty documentation for information about configuring LDAP as a user registry.

3. Optionally, use i2 Analyze's command access control feature to permit or deny access to specific connectors on a per-user or per-group basis. For more information, see the documentation about controlling access to features.

For troubleshooting assistance, see the reference information.

Adding a connector to a deployment of i2 Analyze

When you've deployed i2 Analyze and configured it to use the schema that you generated from iBase, the next step is to add a connector for one of the data sources that you plan to use.

Setting up the connector correctly has two parts. First, you must provide i2 Analyze with details about the connector. Second, you need to ensure that the data from the connector is compatible with your schema.

1. Follow the instructions in Connecting to external data sources, and especially in Adding a connector to the topology, to add a connector to the deployment.

   The provider of the connector will be able to supply all the information that you need to complete the procedure.

2. Follow the instructions in Item type conversion, and especially in Creating type conversion mappings, to map types from the connector schema to types in the gateway schema.

After you successfully deploy and test one connector, you can repeat the steps above to add further connectors to your i2 Analyze deployment.

Preparing workstations for Analysis Studio and iBase

With the i2 Analyze server and at least one connector in place, you can switch your attention to preparing the client workstations. Each workstation needs iBase and the Analyst's Notebook desktop client. If you have it, you can also install the Export-to-iBase plug-in.

On all workstations from which users will access connectors and use their data with iBase, you must complete the following tasks:

1. Install iBase, if it is not already installed.
2. Install the Analyst's Notebook desktop client, or upgrade to it from Analyst's Notebook.
3. Start the Analyst's Notebook desktop client, and ensure that the connection to the iBase database is working correctly.
4. Ensure that you can log in to the i2 Analyze server that you deployed from the Analyst's Notebook desktop client, and that you can successfully run **External Searches**.
5. Finally, install the Export-to-iBase plug-in.

© N. Harris Computer Corporation

At this point, you can move on the final part of the deployment process, which is to configure the Export-to-iBase plug-in for the best user experience.

Configuring the Export-to-iBase plug-in

If you have deployed the Export-to-iBase plugin to your users, then the last part of the Analysis Studio deployment procedure is to configure the plug-in itself. By doing so, you can improve the fidelity of data exchanged between iBase and the external sources, and make local modifications to the plug-in's behavior.

To configure the behavior of the plug-in, you use the settings in the **Export to iBase** dialog in the Analyst's Notebook desktop client. The settings are available to you as you deploy Analysis Studio to work with iBase, and to any users who also have the security administrator role in iBase.

For full information on configuring the Export-to-iBase plug-in, see Configuring export settings and specifications.

Maintaining a deployment of Analysis Studio and iBase

After you've deployed Analysis Studio to work with iBase, several parts of the system are subject to change. On the i2 Analyze side, you might add or remove connectors. On the iBase side, you might change the database schema or its charting schemes.

Adding and removing connectors

Over time, the set of connectors that your deployment of i2 Analyze uses can change. You might add new connectors, or remove old ones, or change the behavior of an existing connector.

When you change a connector, you must check some other parts of the deployment to ensure that they continue to behave correctly:

- Do you need to add, remove, or edit the type conversion mappings from a connector schema to the gateway schema?
- If you have the Export-to-iBase plug-in, do you need to add, remove, or edit export specifications for connectors that have been added, removed, or edited?

Removing a connector

When you remove a connector, complete the following steps to ensure that the rest of the deployment continues to perform correctly:

1. Communicate to your users that the connector will be permanently removed.
2. If an export specification in the Export-to-iBase plug-in contains a group of custom settings for the connector, delete it.
3. Remove the connector from the i2 Analyze topology by reversing the procedure for adding a connector.

Modifying an existing connector

When you modify a connector, complete the following steps to ensure that the rest of the deployment continues to perform correctly during and after the modification:

1. Communicate to your users that the connector will be temporarily unavailable while the update takes place.
2. If you have the Export-to-iBase plug-in, create or edit a group of custom settings for the connector in which the checkboxes for both **Update iBase records...** and **Create iBase records...** are cleared.

© N. Harris Computer Corporation

It is then impossible for users to export records from the connector to iBase.

3. If you need to, use iBase Designer to modify the iBase schema to reflect the changes in the connector. Regenerate the i2 Analyze schema, and then redeploy it as described below.
4. If you need to, create or edit type conversion mappings for the modified connector.
5. Back in the plug-in, edit the custom settings for the connector by modifying the mappings (if you need to) and then re-selecting the **Update iBase records...** and **Create iBase records...** checkboxes.
6. Communicate to your users that the connector has been updated and is available once again.

Adding a connector

When you add a connector, complete the following steps to ensure that the rest of the deployment continues to perform correctly:

1. If you need to, create or edit type conversion mappings for the new connector.
2. If you need to, use iBase Designer to modify the iBase schema to reflect the data that the new connector returns. Regenerate the i2 Analyze schema, and then redeploy it as described below.
3. If you have the Export-to-iBase plug-in, create a group of custom settings for the new connector that contains any mappings you need, and in which the checkboxes for both **Update iBase records...** and **Create iBase records...** are selected.
4. Communicate to your users that the new connector is available.

Changing the schema or a charting scheme

If the schema of the iBase database that you're using with Analysis Studio changes after the initial deployment, then you must update the generated i2 Analyze schema to match it.

To replace the gateway schema with the updated one, and assuming that the name of the schema and its charting schemes have not changed, you can put the new files in this location:

```
toolkit/configuration/fragments/common/WEB-INF/classes
```

And then redeploy i2 Analyze in the usual way:

```
setup -t deployLiberty
setup -t startLiberty
```

When the server restarts, you must check that any type conversion mappings from connectors to the gateway schema are still consistent with that schema. You might also create mappings that take advantage of new features in the updated schema.

Remember that records from connectors that are not correctly aligned with the schema cannot be exported to iBase.

Reference information

This topic provides detailed information about some aspects of the interaction between iBase and i2 Analyze that can affect the behavior of a joint deployment.

Limitations of schema generation

In general, there is a high degree of compatibility between the iBase schema and the i2 Analyze schema that you create from it. However, there are some differences that the creation process cannot resolve.

© N. Harris Computer Corporation

Incompatible logical types

The following iBase logical types have no corresponding types in i2 Analyze. The schema generator does not generate property types from iBase fields that have the following logical types:

Type	Supported
Hyperlink	No
Picture	No
Document	No
Time_Zone	No
Calculated_Date_Part	No
Calculated_Number	No
Calculated_Date	No
CREATE_DATE_TIME	No
UPDATE_DATE_TIME	No
CREATE_USER	No
UPDATE_USER	No
RECORD_ID	No
ICON	No
STRENGTH	No

Custom icons

If the iBase database uses custom icons, then the i2 Analyze deployment must include corresponding icon files. In iBase, mappings for both standard and custom icons appear in the `Iconlist.txt` file, which by default you can find in the `C:\Program Files\i2 iBase 10\Resources\en-US\Configuration` directory.

The start of the file looks like this:

```
Account account
Action action
Adult adult
Aeroplane airplane
Afghanistan Flag AF
Air Rifle Air Rifle
Airplane airplane
Airport airterm
```

Where the syntax of each line is:

© N. Harris Computer Corporation

```
<iBase icon name> <Tab character> <File name of PNG file>
```

If there are custom additions to the `Iconlist.txt` file, you need to make the icons available to i2 Analyze as well. To do this, create a folder named `icons` in the `toolkit/configuration/fragments/common` directory and copy the PNG files to it - *but rename them according to the* left *column of `Iconlist.txt`*. For example, the path within the toolkit to the icon for Account entities (if that were a custom icon) would be:

```
toolkit/configuration/fragments/common/icons/Account.png
```

You also need to copy custom icon files to every client workstation, in the following location:

```
C:\Users\<username>\Documents\i2\i2 Shared\Custom Images\Screen\Icons
```

Finally, after making all of the above changes, redeploy and restart the Liberty server:

```
setup -t deployLiberty
setup -t startLiberty
```

Note: Even after you follow the above steps, the custom icons will not appear in i2 Analyze Schema Designer. However, the names of the custom icons will be present.

Glossary

This glossary provides terms and definitions for the iBase software and products.

The following cross-references are used in this glossary:

- *See* refers you from a nonpreferred term to the preferred term or from an abbreviation to the spelled-out form.
- *See also* refers you to a related or contrasting term.

A C E G I L M N P S V

A

abstract semantic type

A semantic type that only serves as the parent of other semantic types. Abstract semantic types categorize their child semantic types, but are never associated with real data.

administrator

A person responsible for administrative tasks such as access authorization and content management. Administrators can also grant levels of authority to users.

alert

A message or other indication that signals an event or an impending event that meets a set of specified criteria.

alert definition

The statement of criteria that trigger an alert.

ALPR

See automatic license plate recognition.

ancestor

A member that exists at a higher level than another member in a hierarchy and is connected by a series of parent-child relationships.

© N. Harris Computer Corporation

ANPR

See automatic number plate recognition.

audit

To record information about database or instance activity by applications or individuals.

audit log

A log file containing a record of system events and responses.

audit trail

A chronological record of events or transactions. An audit trail is used for examining or reconstructing a sequence of events or transactions, managing security, and recovering lost transactions.

automatic license plate recognition (ALPR)

A technology composed of separate devices that is used to detect, capture, and store visual information pertaining to a license plate.

automatic number plate recognition (ANPR)

See automatic license plate recognition.

C

calculated measure

A measure whose values are calculated from other measures, calculated measures, functions, and numeric constants in an arithmetic equation.

case

The information that is contained within a database that pertains to a particular investigation.

chart

A visual representation of real-world objects, such as organizations, people, events, or locations, and the relationships between them.

charting scheme

A definition that describes how item data behaves when it is visualized on a chart. For example, how data is copied into chart item properties, the chart template and labeling scheme to use, and whether to display attributes and pictures.

child

In a generalization relationship, the specialization of another element, the parent.

condition

A specified property, a value, and an operator that defines a comparison relationship between them. One or more conditions can be used to create a query or a conditional formatting specification. See also parameterized query.

E

end

An entity that is attached to a link. See also end constraint.

end constraint

© N. Harris Computer Corporation

A constraint on the types of entities that can be the end of a particular link. See also end, valid end type.

entity

A set of details that are held about a real-world object such as a person, location, or bank account. An entity is a kind of item.

entity semantic type

A semantic type that can be assigned only to an entity or an entity type. See also semantic type.

entity type

A descriptor of the characteristics of an entity, including the properties it can contain and its appearance in visualizations.

excluded word list

A list of words that are ignored when they are entered as search terms.

expansion

A process that searches for entities within a data source that are directly related to some selected entities.

G

grade

A rating that indicates the accuracy of a piece of information or the reliability of an intelligence source.

grading system

A rating scale that is used to classify information in a data store or on a chart. A grading system is a measure of reliability and accuracy.

I

import design

A specification of how data from an external source will be transformed into chart or repository items during an import procedure.

item

An entity or a link. Items are characterized by the values of their properties. See also merged item.

L

labeling scheme

A specification for combining property values to be displayed on screen, or as chart item labels.

layout

The arrangement of items on a chart.

line strength

An indication of confidence in the information underlying a particular link. Line strength is represented as a solid, dashed, or dotted line on a chart.

link

© N. Harris Computer Corporation

An association between two entities, such as an ownership relationship between a person and a vehicle.

link direction

An indication that the meaning of a link is different for each of its ends. For example, the direction of a telephone call makes one end the caller and the other the recipient. Link direction can influence the centrality measures used in social network analysis.

link semantic type

A semantic type that can be assigned only to a link or a link type. See also semantic type.

link type

A descriptor of the characteristics of a link, including the properties it can contain and its appearance in visualizations.

M

mapping scheme

A set of rules that determines how information is displayed on a map.

match

The part of a result that met a condition during a search operation. A search can yield a perfect match or a partial match.

merged item

An item that is created by merging the information held in two or more items. See also item.

N

noise word file

See excluded word list.

P

parameterized query

A query with conditions in which one or more parameters are defined. The parameter values are set by the user. See also condition.

parent

In a hierarchy or auto-level hierarchy, a member that has one or more child members at the level immediately below.

path

A route on a chart between two entities. A path may include intermediate entities.

pick list

A data category that has a limited number of permissible values, which are often presented in a drop-down list in the user interface.

pivot

A method of rearranging data in a data set to reveal patterns in the data.

© N. Harris Computer Corporation

property

A container for a single piece of information about an item.

property semantic type

A semantic type that can be assigned to a property type, a property in a data record, or an attribute class. See also semantic type.

property type

A descriptor of the characteristics of a property, including the type of information it can contain.

S

schema

A complete description of all the entity types, link types, and their associated property types that are available for items within a system.

semantic type

A category that defines the real-world meaning of data, and therefore how applications should interpret that data. For example, Person is a semantic type that could be assigned to entity types such as Male, Victim, and Witness. See also entity semantic type, link semantic type, property semantic type.

source reference

An identifier that indicates the source of information, for example, a document reference number.

V

valid end type

An entity type that conforms to the end constraints of a particular link. See also end constraint.

Using i2 iBase

Information about using i2 iBase to further your analysis and investigations.

Licensing information

i2 iBase is a licensed product and requires a valid software license. Depending on your organization's arrangements, you might need to provide a license directly or connect to a license server.

There are a number of ways you might provide your license information in order to activate iBase.

Standalone licenses

Standalone licenses are stored on the same workstation as your installation, and are provided in the form of license keys or license files.

Network licenses

You can use a floating license when your workstation is connected to a license server on your network. These licenses can also be borrowed for offline access, and returned when no longer required.

Internet licenses

When you have internet access, you can use a license from i2's internet license server.

Support

© N. Harris Computer Corporation

For detailed information about licensing, refer to the i2 Software licensing guide.

If you need further help with iBase licensing, contact your system administrator or i2 Support.

Overview of i2 iBase

i2® iBase provides powerful solutions for configuring, capturing, controlling, analyzing and displaying complex information and relationships in link and entity data. iBase is both a database application and a modeling and analysis tool.

Entering data

As a database application you can do all the things you would expect, such as adding records, deleting records, and importing, and exporting records.

Modeling and analysis

The modeling and analysis facilities are based upon the concepts of entities and links. Entities are the things that are being represented, such as people and addresses. Links represent relationships between entities, such as 'shareholder' and 'account holder'. For each entity and link, there is a database record.

Analysis is made easy in a number of ways; entities and links can be grouped together in a variety of ways using sets. Sets can be analyzed to find common members. There are sophisticated visual query facilities that allow you to construct queries as simple or as complicated as you need to reveal the information in your database. Data can be reported upon using flexible reporting tools.

Additional features

The features available in iBase depend on the type of database you have. If you have an SQL Server database, you may be able to manage data on a case-by-case basis, and add alerts to monitor when records of interest change or are viewed. Search 360 provides powerful tools for both simple and complex searches.

Integration with other products

iBase is closely integrated with i2® Analyst's Notebook® and some third-party mapping applications. If these products are installed on your machine, you can use:

- Analyst's Notebook as a 'front end' or interface to iBase, so you can use it to create records for example, and you can send records from iBase directly onto an Analyst's Notebook chart for further analysis and presentation.
- i2® iBase Geographic Information System (GIS) Interfaces to send iBase data to third-party mapping applications.

Creating a database

If your user account has the permission Database Creator, you can create a new database from a template supplied by your database designer. The template defines the type of information that can be stored in the database: the entity types, link types, and other items including any queries and browse definitions.

Certain details, such as the file type, are not defined in the template and you should discuss these details with your system administrator before creating a new database. Specifically, you should discuss:

- Which database type to use, either Microsoft Access or SQL Server.

© N. Harris Computer Corporation

- How auditing should be used, specifically which audit level to use and whether the value of a specific field should also be audited.
- Whether a database identifier is required.
- Whether soft delete is used.

For background information on creating new databases, see the help for iBase Designer.

Note:

A 20-character password is generated for you when the Microsoft Access database is created. You will need to keep a record of this password as it will be required in the event, for example, that you need to repair the database using Microsoft Access utilities.

To see the password, start iBase Designer, select **Tools** > **Options**, and select the **Advanced** tab.

Creating a database from a template

Database templates contain standard components. Creating a database from a template reduces the time that is taken, and ensures that databases for a specific task are created consistently.

Procedure

To create a new database from a template:

1. Ensure that you are logged into iBase, but have no databases open.
2. Select **File > New Database**.
3. Click the **Template** tab.
4. Select a type from *Blank*, *User* (user created) or *Workgroup* (system created) templates. You have the option to create and delete templates from this tab. You can also click **View** if you want to see the entity types, link types, and fields in a selected template.
5. Click **Use Database** to create a database from an existing template. You can browse to any database accessible from the currently logged in security file. A temporary template is created and given a type of *Database*. This template is automatically deleted if you change to a different template, create a database using the template, or close the New Database dialog.

   Note: You can also create a template from a different database, and use that template instead. For more information, see Creating a template from an existing database.
6. Click the **Configuration** tab, and select the database type.
7. Click the **Details** tab, and enter the name of the database and some information about the purpose of the database or its contents.
8. Click the **Advanced** tab, and enter the details:
   - **Database Identifier**

     Optionally, enter a short string of text in the **Database Identifier** box. Do this if you want to identify entity and link records as belonging to this database. This database identifier is only necessary if you plan to perform operations outside iBase on records taken from different databases.

     Attention: The use of a database identifier has an impact on performance since the database identifier is appended to the record identifier on every record.
   - **Extra Detail Field for Audit Log**

© N. Harris Computer Corporation

Type the name of a field (in this database) in the **Extra Detail Field for Audit Log** box if you want the audit log to record the value of this field when recording actions that affect records.

- **Soft Delete**

  Turn on the **Soft Delete** checkbox if you want to use a two-stage process for deleting records. With **Soft Delete** turned off, all delete operations take place immediately. If the **Soft Delete** checkbox is turned on, all **Delete** commands mark records for deletion and make those records unavailable for most analysis, but do not delete the records.

- **Read Only**

  Turn on the **Read Only** checkbox if you want to make the entire database read-only, and prevent any changes to records. Users can still create sets, queries, and other folder objects.

- **Security Classification Codes / Case Control**

  Determines whether the database uses Standard Security Classifications or restricts information based on specific cases. If you select **Standard (SCC)**, you can additionally opt to **Restrict SCC lists to accessible items only**. Turn on this option to restrict any lists of Security Classification Codes to accessible ones only. This setting applies when you add or edit a record that includes an SCC list.

- **First Day of Week**

  Displays the first day of the week as set for this database. This defaults to <System>, which is Sunday for Microsoft Access databases. For SQL databases, this is derived from the current locale as set on your machine or via the locale ID of the SQL Server machine.

  You should only need to change this if the locale on the SQL Server machine is different to your local machine or you are working with statistics and you want your week to start on a different day.

  Note: The start day of the week may affect calculations on dates and date parts.

9. Click **OK** to create the database with the settings you have chosen.

Types of use

iBase can be used by both Designers and Users. Designers are responsible for designing databases and setting up the properties of entities and links and for configuring security, these tasks are referred to as administrative tasks. Users can use databases, add or import records, and manipulate or analyze the data.

For example if you wanted to add a new type of entity to your database, such as an airport, then the Designer would do that and set up all the required properties of an airport. But if you wanted to record the details of an actual airport, and then run queries about airports, these would be user tasks.

Typically a database would have maybe one or two people who carry out the administrative tasks but may have many people who can carry out user tasks. iBase has separate modules for Designers and Users so you log on to whichever is appropriate for you.

The roles of designers and users can be summarized as follows:

Designers

Designers use iBase Designer to:

- Design databases - this process usually involves consulting with the intended users to decide exactly what information is to be stored and in what form.

© N. Harris Computer Corporation

- Create new databases - includes creating all the required entity and link types.
- Update databases - adding new entity and link types as requirements arise.
- Configure databases - setting up code lists, labeling schemes and all the other things that affect how the database is used.
- Administer the database - for example: adding users, setting security, deleting old records, and backing-up the data.

Users

Users use iBase to:

- Add, modify and delete records.
- Create sets and queries to find data of particular interest.
- Analyze the data using a variety of tools.
- Create reports or charts based upon the data.

Opening a database

To use iBase you need to log on and open a database. Logging on happens automatically if iBase is set up to use your existing Windows username and password. You will be prompted to enter an iBase username and password if iBase is not set up in this way.

If you have previously opened a database, iBase might offer to open that database the next time you start. You can only have one database open at a time in any iBase session. When you have opened a database, you can perform all of the actions that your access control settings allow.

You can only have one database open at a time in any iBase session. When you have opened a database, you can perform all of the actions that your access control settings allow.

Note: An administrator, using iBase Designer, must have created a database before you can use it. Alternatively, you might be able to create a new database from a template supplied by the database designer.

Note: You cannot open a non-Unicode database created in a different language environment. For details of supported language groups, see the iBase Release Notes.

Selecting a database

To use iBase, you need to log on and open a database. Logging on happens automatically if iBase is set up to use Windows™ authentication, alternatively, you are prompted to enter an iBase username and password.

About this task

You can only have one database open at a time in any iBase session. When a database is open, you have access to all of the actions that your access control settings allow.

Procedure

1. Select **File** > **Open Database**, and browse for the database (*.idb) to open.
2. Click **Open**.
3. If prompted, enter your iBase username and password (the password is displayed as asterisks (*) for added security).

© N. Harris Computer Corporation

4. Click **OK** to view the database summary.
5. Click **OK**.

Opening the example user guide database

The iBase example user guide database provides you with an environment with sample data. This example database can be used to explore the features of iBase without changing your live databases.

Procedure

To open iBase and the example database at the same time:

1. From the start menu, select:**i2 iBase** > **iBase User Guide Database**
2. Enter your username and password and click **OK**. The username and password for the User Guide database are 'general' (both are lowercase).
3. Click **OK**.

Changing a database

You can change to another iBase database if it has the same `.ids` security file, without the need to log in again.

About this task

You can only have one database open at a time in any iBase session. If you want to quickly change to another database that shares the same security file, you can use the **Change Database** option.

Procedure

1. Select **File** > **Change Database**. The confirmation message is displayed. Click **Yes** to close the current database.
2. In the **Open Database** window, select a database from the available `.idb` files. Click **Open**.
3. The iBase database opens.
4. Click **OK**.

Note: If your chosen database is associated with a different `.ids` security file, you are automatically logged out and prompted to login to the `.ids` for the database you want to connect to.

Adding your contact details

Depending on your organization, you may need to record additional information about yourself to assist other iBase users. For example, other users may have queries about the data that you add to the database, or may need to talk to you before editing, merging, or deleting records that you own.

About this task

The contact details are for the username that you used when you logged on.

Procedure

1. From the **File** menu, select **Change User Information**. The User Information dialog is displayed.
2. Enter your full name, telephone number, e-mail address, and any notes. You should be able to select your location from the **Location** list (if not, contact your system administrator).

© N. Harris Computer Corporation

3. Save your details by clicking **OK**.

Exploring the Database

The Database Explorer window is a complete view of your database. The top window shows all the objects in the database, and the lower Explorer Detail window displays information on the selected folder or folder object. Depending on the database design some of these object types might be empty, for example if chart attributes are not used.

Object type	**Description**
Name of your database	Right-click to see statistics for the database, for example how many entity types are defined in the database.
Entities	Click to list the entity types defined in this database. The entity types are shown in the Explorer Detail window. To view the fields and field types, expand Entities, and then select the required entity type.
Links	Click to list the link types in this database. The link types are shown in the Explorer Details window. To view the fields and field types, expand Links, and then select the required link type.
Labeling Schemes	Click to list the labeling schemes defined in this database for use in iBase itself and on Analyst's Notebook charts. To see which fields have labels, expand Labeling Schemes, and then double-click the required scheme.
Alert Definitions	Click to list the alert definitions.
Sets	Click to list the sets in this database. The sets may be organized into categories (folders).
Queries	Click to list all the queries saved in this database. The queries may be organized into categories.
Report Definitions	Click to list all the report definitions saved in this database. The report definitions may be organized into categories.
Browse Definitions	Click to list all the browse definitions saved in this database. The definitions may be organized into categories. For background information, see Listing and Browsing Records.

© N. Harris Computer Corporation

Object type	Description
Scored Matching	Click to list the queries that have been run with scored matching to identify how well the results match a query.
Import/Export Specifications	Click to list the specifications for importing (or exporting) data for single entity types or single link types. These may be organized into categories.
Import/Export Batch Specifications	Click to list the batch specifications for importing (or exporting) data for a series of entity types or link types.
Charting Schemes	Click to list the charting schemes defined in the database.
Mapping Configurations	Click to list the mapping configurations defined in the database. These will only be available if GIS Interfaces is installed.
Functions	Click to list some of the tools available in iBase. Some of these are also available on the Tools menu or the Analysis menu.

Summary of the database properties

The properties of the database provide detailed information about the database.

At any time you can view the properties of the database in iBase Designer, by selecting **File** > **Database Properties**.

Title

The title for the database, as displayed in the application title bar.

Description

The description of the database, as displayed when any user first opens the database.

File

The location of the database (`.idb`) file.

Version

The database version number.

Audit Level

The detail level at which the audit log collects data on changes to the database and security file. You can change the audit level: level 1 means that the audit log collects the lowest amount of detail and level 5 collects the highest amount of detail (SQL Server databases only). Level 4 and higher collect large amounts of data about user activities so you should use these levels with care, and monitor the size of the log file.

© N. Harris Computer Corporation

Audit History

In SQL Server databases only, all updates to data, including code lists, are logged and can be viewed either in Audit Viewer or in the iBase History. In a database that is set to audit level 5, records that are viewed but not updated are also logged.

Note: This property is automatically turned on if the database is initialized for alerting and cannot be turned off while alerting is in use.

Database Logo

You can replace the default database logo with an alternative image. Click **Select...** to browse the available supported file types in your Windows Pictures folder. Select an image no larger than 0.5MB and click **OK**. To remove the image from the database, click **Clear** and **Yes** to confirm. Your image is replaced with the default iBase icon.

Configuration

The configuration page shows details of the database file and format, and the security mode. You can change the authentication mode when connecting to the SQL Server instance on this page or by using the Database Configuration tool (see Managing SQL Server Connection Settings).

Database Type

The file format of the database, either Microsoft Access or SQL Server.

Database Name

The name of the SQL Server database on the server. See SQL Server Database Names.

Note: You cannot rename an SQL Server database in iBase Designer. See SQL Server Database Names for further details.

Server

The name of the database server. You can change to a different server provided that the database exists on that server. Enter a name in the field to use a known server. Only select the **(local)** option if the database is for personal use.

Note: This and the following changes do not take effect until you reopen the database.

Login Name, Password

An SQL Server login name and password is displayed if SQL Server authentication is used to secure access to the SQL Server instance. See Authenticating Connections to SQL Server for details.

For security reasons, the login that is used to create the database might be different from the one used after creation. After creation, you might prefer to change the login to one with a lower level of SQL Server permissions. After creation, you might also want to change the authentication mode by turning on the **Use Windows Authentication** checkbox.

Use Windows Authentication

The **Use Windows Authentication** checkbox is turned on if Windows authentication (integrated security) is used to secure access to the SQL Server. Each user that attempts to connect to use the iBase database is validated by the server using their network credentials. See Authenticating Connections to SQL Server for details.

© N. Harris Computer Corporation

Advanced

The Advanced page displays the current setup of the database, which you can change with caution. See Advanced settings for details.

Database Access Token

Use the Database Access Token page to obtain a copy of the token for the current database. The token is used by some applications, such as the Search 360 Indexer, to connect to SQL Server databases. You can save time by copying the token from this page and pasting it into the application, rather than having to regenerate the token in the Database Configuration tool.

Passwords for Microsoft Access databases

A 20-character password is generated for you when the Microsoft Access database is created. You should keep a record of this password. The password is the same for all the Microsoft Access databases created from the same security file.

If you are using iBase Designer and want to see the password, select **Tools** > **Feature Availability** > **Options** > **Advanced**. This option is not available in iBase User.

Viewing the database design report

The database design report summarizes the design of a database that you have open. For example, database statistics, entity and link type fields, code lists, and semantic types. To gain a full view of the database design, you must ensure that you run the report with the correct access permissions.

Procedure

To generate a report:

1. Click **File** > **Properties** > **Database Design report**.
2. Once you have generate the report you can:
   - Browse through the pages
   - Refresh the information it contains
   - Print the report
   - Export the report as a spreadsheet, a PDF, or a Microsoft Word document.

Reporting on database security

The security design report provides details about the security groups, users, and their consequent permissions or restrictions that have been applied to the database. You can select the items you want to include in the report.

Before you begin

To view the details of the database security, you must be assigned the `SecurityAdministrator` role.

Procedure

1. In iBase User, Select **File** > **Properties** > **Security Design Report**.
2. Select the types of details to include in the report:

© N. Harris Computer Corporation

- Groups - For each security group that has been defined for the database, you can list:
  - Permissions and restrictions - What members of the group can access, and what they are explicitly prevented from doing (for example, editing read-only items).
  - Users - The users that are currently members of the group.
  - Denied SCC items - The types of entities or link that have restrictions in place, to prevent members of this group from accessing particular records.
- Users - For each user that can access a database, you can list:
  - User Information - The information entered centrally about the specified user.

    Note: This report only uses information added in iBase Designer.
  - Groups - The groups the user is a current member of.
  - Permissions - The specific permissions for the user.

3. Once you have generate the report you can:
   - Browse through the pages
   - Refresh the information it contains
   - Print the report
   - Export the report as a spreadsheet, a PDF, or a Microsoft Word document.

Database statistics

Database statistics are provided to give you a summary of the number of records in the database for each of the entity types and link types. When you are working in multi-case analysis mode, you can view the number of entity types and link types in each case.

To view the database statistics, select **File** > **Properties** > **Database Statistics**.

Entities

The modeling and analysis facilities in iBase are based on the concepts of entities and links. Entities are the things that are being represented, such as people, vehicles, and addresses. Links represent relationships between entities, such as owner, daughter, associate, or account holder.

The different entity types and the details that are held for each (in the entity fields) are already defined by the database designer in iBase Designer.

In **Database Explorer**, select an entity type to display the fields for that entity type in the **Explorer Detail**.

There is one record in the database for each entity and link in the database, and each record has a unique record number. Generally, you create entities and then the links between the entities.

You can add entities to the database or case:

- On an individual basis
- By importing them
- By using a custom data sheet, in which case you can enter an entity and its links in one operation.

To speed up the entry of data, you can set up:

- Session defaults to automatically insert data into those fields that are common to all the entities and link types in the database, for example for a source reference field.

© N. Harris Computer Corporation

- Lists, such as code lists and pick lists, to include those values that you use frequently.

Links

The modeling and analysis facilities in iBase are based on the concepts of entities and links. Links represent relationships between entities, and the characteristics of those relationships.

For example:

- People can be linked to organizations as shareholders, directors, or employees.
- A pair of telephones can be linked by a specific call that is made between them.

You can add links to the database:

- On an individual basis
- By importing them
- By using a custom data sheet, in which case you can enter an entity and its links in one operation.

Generally, you add entities and then the links between them. Each link is a separate record in the database, and has a unique record number.

Note: A link can also have a direction, as shown by an arrow, and a line style that indicates the strength, such as whether it is confirmed or unconfirmed. In the **Database Explorer**, you might be able to get more information about a link type by moving the pointer over it to display a tooltip. For information on which semantic types are assigned to the link types in your database, run the **Database Design** report.

To list the link types available to you, click the plus sign to the left of the word Links. Select a link type to display the fields for that link type in the Explorer Detail window.

To find out which combinations of entities and links can be used together, right-click on a link type, and from the menu, selecting **End Types**.

Setting Up the Example Database

An example database (User Guide) is installed with the product. You can use it to experiment with the new features in iBase.

Opening the example user guide database

The iBase example user guide database provides you with an environment with sample data. This example database can be used to explore the features of iBase without changing your live databases.

Procedure

To open iBase and the example database at the same time:

1. From the start menu, select:**i2 iBase** > **iBase User Guide Database**
2. Enter your username and password and click **OK**. The username and password for the User Guide database are 'general' (both are lowercase).
3. Click **OK**.

© N. Harris Computer Corporation

Standard user accounts

The user guide database is supplied with a set of example users with different roles. These can be used to demonstrate different workflows.

User account	Password	Role
general	general	A user with all the permissions required to work through basic examples.
SYSADMIN	SYSADMIN	A full system administrator with all the permissions required to work through iBase Designer examples.
DataEntry	DataEntry	A Data Entry User with restricted menu functionality and access to fewer links
Analyst	Analyst	An analytical user with read-only access

Reverting to a clean copy of the User Guide database

When you use the User Guide Database for the first time, the database is automatically copied to your application data area. This means that you can change the database and the records as you want. At any point, you can restore the database to an unmodified state (provided you are using a Microsoft Access database):

About this task

Procedure

1. Select the following from the Programs group on the Windows Start menu: **i2 iBase** > **Tools** > **Reset iBase User Guide Database**.
2. Click **Yes** when prompted to reset the database.

   Attention: Reverting to an unmodified User Guide database will mean that you will lose any changes that you made to the User Guide database. For example, you will delete all entities, links, sets, queries, and other folder objects that you created or modified. Any files created independently, such as export files will remain.

Moving the User Guide database to a new location

If you move the User Guide database from the standard location, you will need to open it in the iBase Designer tool to re-register the location of the security file which controls it. You may need to ask your system administrator to do this for you.

Note: It is important that the association of the database and security file is maintained. iBase ensures that this is done and consequently if you move the database and security file you need to confirm the new location and association by opening the database using iBase Designer.

© N. Harris Computer Corporation

Using the User Guide database on SQL Server

The User Guide database is supplied as a Microsoft Access database. However, if you want to try out the features that are only available in SQL Server then you will need to ask your System Administrator to convert the database to SQL Server for you.

About this task

The following instructions are intended for a System Administrator, who will need permission to create a database on your SQL Server machine.

Procedure

1. Check that the server does not have an existing database called User Guide. If it does, rename the User Guide.idb file, for example to User Guide 2.idb. You will also need to rename the other files associated with the database (with the suffixes .dot, .doc, .ant and .idx).
2. Copy the User Guide Database folder to a suitable place. It is located in: C:\Program Files\i2 iBase <n>\Resources\language>\Examples\User Guide Database
3. Start iBase Designer, and then log on to the security file User Guide.ids as user SYSADMIN and cancel the option to open a database or create a new one.
4. From the Tools menu, select **Database Setup** > **Upsize** > **Database to SQL Server**.
5. Accept the option to make a backup.
6. Enter the name of the SQL Server machine and a login and password that has the dbcreator role on the server. Do not use the server name (local) since other clients will not be able to use the database. This server name is intended only for local use on the server computer. If the database name does not appear when you refresh the list, type in the machine name of the server.
7. Click **Finish**.
8. In iBase Designer, use the appropriate search option on the **Tools** menu to build a search index. For further information, click **Help** in the appropriate search dialog.

Entering records

A record is a discrete collection of information about a real world object (an entity) or the relationships between real world objects (a link). You can add records directly, or use a datasheet to target the information that you are adding towards a specific purpose.

Entities

The modeling and analysis facilities in iBase are based on the concepts of entities and links. Entities are the things that are being represented, such as people, vehicles, and addresses. Links represent relationships between entities, such as owner, daughter, associate, or account holder.

The different entity types and the details that are held for each (in the entity fields) are already defined by the database designer in iBase Designer.

In **Database Explorer**, select an entity type to display the fields for that entity type in the **Explorer Detail**.

There is one record in the database for each entity and link in the database, and each record has a unique record number. Generally, you create entities and then the links between the entities.

You can add entities to the database or case:

- On an individual basis

© N. Harris Computer Corporation

- By importing them
- By using a custom data sheet, in which case you can enter an entity and its links in one operation.

To speed up the entry of data, you can set up:

- Session defaults to automatically insert data into those fields that are common to all the entities and link types in the database, for example for a source reference field.
- Lists, such as code lists and pick lists, to include those values that you use frequently.

Creating an entity

You can add new data to create entities. The data is organized into different entity types and each type represents a "thing" such as a person, a crime, or a bank account.

About this task

When you add an entity, you are warned of matching records if you enter an entity that is a potential duplicate of an entity already in the database. iBase identifies matches by comparing the values in the discriminator fields. You might see a warning as soon as you enter data that matches an entity that is already in the database or when you try to save an entity that matches a record already in the database.

Procedure

1. In the Database Explorer, right-click on the entity type, and from the shortcut menu, select **New**.

   Note: Depending on how the database designer has configured the entity types in your database, you will display either the New (entity) dialog or the Datasheet dialog. The Datasheet dialog is displayed for entity types where the datasheet is set to override the standard form; this can only be changed using iBase Designer. To choose which to use, select **New With > New Record** for a standard dialog or **New With > `datasheet name`** for a datasheet.
2. Enter the details for the new entity in the New dialog.

   Notes:

   - Mandatory fields - those that you must fill in - are shown with a blue label.
   - You may be able to get help on what you should enter in each field by hovering the mouse pointer over the field labels. This displays a tooltip telling you something about the type of data to enter in each field.
3. Depending on your organization, you may need to select a user as the owner of the record, and there will be a **Owner** box or something similar. This person is the point of contact if there are any queries about the record. You can either type $ to make yourself the owner (your username will be inserted when you save the entity), or type $, click the **Browse** button, and then select a different person from the list.
4. When you have finished entering the data, click **Save** to save the record.

Results

The new record is displayed in the Show dialog. The title bar of this dialog also shows the unique number given to the record when it was saved, such as `[ADD48\GEN]`. If required, in the Show dialog, click **New** to add another record of the same entity type.

© N. Harris Computer Corporation

Copying an entity

You can create a new entity by copying a similar entity and editing it. Copying an entity can be used when related items are added that share information.

Procedure

1. Find the entity that you want to copy.
2. Right-click the entity and from the menu, select **Show With** > **Show Record**.
3. Click **Copy**.
4. Edit the entity details.
5. Click **Save** to save the copy as a new entity.

Data sheets

Your database designer can set up data sheets to use instead of the standard process for adding, editing, and deleting data. Data sheets are customized forms that are designed especially for your work.

Data sheets show only the fields that you need, arranged in groupings that reflect your way of working. Depending on the data sheet, you might be able to use it to enter data for related entities and links. For data sheets that contain linked entities, the upper part is used to enter details of the main entity while the lower part is used to enter data about any linked entities. The fields in the lower part can be a mixture of the link fields and the link end entity fields.

To use a data sheet to create a new entity and create links to new or existing entities, you can:

- Select **New** > **Datasheet** > **`datasheet name`**.
- Right-click on the entity type, and select **New With** > **`datasheet name`**. Data sheets are listed in the lower half of the menu.

To use a data sheet to review or edit an entity and its linked entities, including adding new links: select the entity, right-click and from the menu, select **Show With** > **`datasheet name`**.

Note: Your database designer might decide to make the data sheet the default method for entering record data for a particular entity. When you want to show records that you can choose whether to show the details in a data sheet. To make this choice, select one of the **Show With** options.

Duplicates found when entering data

When a match is found while you are entering data in a data sheet, a message is displayed. The record that you are trying to create is a duplicate of existing records.

They are identified as duplicates by comparing the values in their discriminator fields.

You can:

- Review the matching records and then continue with the 'new' record.

  Click **Yes** to review the matching records. Examine the matching records, and click **Ignore** to return to your new record. You can either edit the data in the new record so that it is no longer a duplicate or you can continue creating a potential duplicate.
- Discard the 'new' record and work on one of the matching records.

  Click **Yes** to review the matching records. Select one of the matching records (to use instead of the 'new' record that you were working on) and click **Select**. Edit the details of the matching record as required and then click **Save**.

© N. Harris Computer Corporation

- Ignore the message.

  Click **No** to close the message. You can either edit the data in the new record so that it is no longer a duplicate, or you can continue creating a potential duplicate.

Duplicates found when an entity is saved

If you enable **Check for matching records whenever a discriminator field value changes**, you are notified if a match is found when you save an entity. You can then decide if the information you added is new, or already exists in the database.

About this task

Records can be identified as duplicates by comparing the values in specified unique fields. For example, details about a person, a passport number or tax code are unique. By flagging records that might contain duplicate information, you can ensure that your database contains all the information about an item in a single record. When you attempt to save a record that contains duplicate fields, a message notifies you.

You can:

- Click **Yes** to save the record and create a potential duplicate without reviewing any of the records that are found. Depending on how iBase is configured, there might be a second warning message. Click **Yes** to save the record.
- Click **No** to review the list of matching records. You can review the summary of each record or use the commands on the menu to determine whether the record really is a duplicate. For example, values in the other fields of the entity might indicate that it is not one.

Note: If you want to stop being notified of duplicate records, you can turn off **Check for matching records whenever a discriminator field value changes**, in **Tools** > **Options**.

Procedure

To review potential duplicate records:

1. Review the list of matching records to determine whether the record really is a duplicate, and if any new information about a duplicate is available. You can do this by:
   - Clicking each record in turn to display a summary of the details.
   - Selecting a record and clicking **Show Record** to display the full details, and edit if necessary.
2. When you are finished, click **Close** to return to the record you were editing.
3. Do one of the following:
   - Edit the record and change the details that make it a potential duplicate.
   - Click **Save**, and select **Yes** to save the record as a potential duplicate.

Show an entity

You often need to see the contents of an individual record, either to read the details or to edit the information. To display or work with individual entity records that you find, use **Show**.

Directly opening a record (**Show**) always shows all of the fields that are defined for an entity whereas data sheets might omit some fields.

After you have opened an entity, you can edit its details, copy (and then edit) it to a new entity, or delete it. Depending on the database, you might also be able to the history of the record.

© N. Harris Computer Corporation

Showing a record might raise an alert.

Different ways of showing an entity

There are different menu commands for showing the complete record or showing in a data sheet:

- To use the default method for showing records of this type, right-click on the record and from the menu, select **Show**.
- To show the complete record, right-click on the record and from the menu, select **Show With** > **Show Record** .
- To use other data sheets (if there are any), right-click on the record and from the menu, select **Show With** > **`datasheet name`**.

Note: If you selected more than one record, the records are listed to compare the records.

When you have reviewed the record, you can start other operations involving the shown entity.

Contact details for the owner of the record

Depending on your organization, a record might have an "owner" that you need to contact before you edit, delete, or merge the record. The user's name and their contact details are listed in **User Information**.

There are two ways of displaying their contact details:

- Click the username.
- Click **Edit** and then double-click the username.

The record owner can be a different person to the user who created or updated the record. To find out who these users were, right-click the entity in any record list and from the menu, select **Properties**.

Note: Contact details might not be available in this database, or might be available on certain entity and link types.

Changing the owner of a record

If you have the correct permissions, you can assign a record to a different owner. For any record that has an owner, the record must be in edit mode before the owner can be changed.

To select a user as the owner of the current record:

- Select the user from the list and click **OK**.
- Double-click the user.

Note: If there is a long list of names, enter the first letter of the name to scroll to that position in the list.

Merging entities

You might have two or more entities that you decide relate to the same person or thing. You can merge these entities into one, however, after entities are merged, you cannot reverse the operation.

About this task

Merging means that one entity (the 'merge' entity) inherits the links from other entities that are transferred to it before the other entities are deleted. Only entities of the same type can be merged. Optionally, you can:

- Use the data from the entities that are deleted to replace blank field values in the 'merge' entity.

© N. Harris Computer Corporation

- Add the data in any Multi-line Text (Append Only) fields to the end of the corresponding fields in the 'merge' entity.

The original creation dates and usernames on the links are preserved, and the update details are modified to show who merged the entities and when it was done. Where a change is made to a Multi-line Text (Append Only) field, the text MERGE is also inserted after the username.

Note: The record that is displayed in the upper part is the record that is kept, and the records displayed in the lower part are deleted once they are merged.

Procedure

1. Select **Edit** > **Merge Entities**.
2. In the Merge Entities dialog, select an **Entity Type**.
3. In the Merge the records below into this record area, either click **Select** to find and select the 'merge' entity, or drag a single record from a different dialog into the blank icon area.

   Tip: The pointer changes shape when you are over the correct part of the dialog.
4. You can check that you have selected the correct entity by reviewing its details. For example, you can right-click on the icon and from the shortcut menu, select **Show**.
5. In the Merge these records into the above record area, click Add to find and select entities to add to the list of entities that you want to be merged. Alternatively, drag a record from a different dialog into this area.

   Tip: You can check the details of the entities by using commands, such as Show, on the menu. To display the menu, right-click on a record in the record list.
6. Repeat as required to add more entities.
7. Optionally, turn on **Use the values from the records above to substitute blank values in the 'merge' record (and merge append-only fields)**. Turn this on if you want to assign values from the entities that are to be deleted to blank fields in the 'merge' entity. If there is more than one record in the Merge these records into the above record area, then the value is taken from the first record in the list. Any values in Multi-line (Append Only) fields will also update with any values from the entities that are to be deleted.
8. Check that only the required entities are listed. If you do not want to merge any of these, select them and then click **Remove**.
9. Click **Merge** and then click **Yes** to confirm that you want to merge and delete the entities.
10. If the link end entity at the other end of the link is not valid for the link type, you see a warning. Click **Yes** to perform the merge (you can modify the link later) or **No** to cancel the merge in order to modify the link.

Editing entities

Entities can be edited to add new information or to modify existing information if you have permission to do so. If necessary, check with the owner of the record before making any changes.

About this task

Saving your changes stores the text along with your username, and the date and time that you made the change. Saving your changes can also raise an alert.

© N. Harris Computer Corporation

Note: On some fields, your administrator can set permissions such that you may not be able to modify the current contents; you might only be able to append new text. These fields are normally displayed with a color background (uses the tooltip color, typically yellow).

Procedure

To edit the details of an individual record:

1. Find the entity that you want to change and show the record.
   a. From the **Edit** menu, select **Find Entity** and then select the required entity type.
   b. Enter a piece of information about the entity into the appropriate field and click **Find**.
   c. Right-click the required record, and from the menu, select **Show**.
2. Depending on your organization, you might need to check with the owner of the record before you edit the entity. If you need to do this, the owner's username is displayed as part of the record details.
3. Click **Edit**. All fields that you can change are displayed with a white background.

   Tip: If you double-click a multi-line field, a larger window is displayed.
4. Make your changes and then click **Save** to store the changes in the database.

   Note: You can make the same change to a number of entities at the same time: from the **Edit** menu, select **Batch Edit**.

Changing the icon used for an entity

Icons are used in records and whenever you chart items. You can change the icons that are used to represent entities of a particular type.

About this task

You can change the icon and icon shading color for an individual entity, all the entities in a set, or all the entities that are found by a query.

Note: When you apply an icon with a semantic type to an entity, you can affect the results of queries that use semantic types.

Procedure

To assign a different icon or shading color:

1. Select **Edit** > **Assign Icons** and select the entity type from the **Entity Type** list. The standard icon for the type is selected automatically in the list.
2. In the **Icons** list, select the icon that you want to assign to the entity or entities.

   The icons that are included in this list depend on the selected entity type. If this entity type has an Icon type field, then the list is filtered so that you can choose icons from the list that is assigned to this field. If the entity type does not have an Icon type field, then you can choose from the whole set of icons.
3. You can apply a shading color to the icon.

Option	Description
Ignore	When you are changing the icon for more than one record, select this option to keep the

© N. Harris Computer Corporation

Option	Description
	existing icon shading setting for each record. The shading that is set for each record is applied to the new icon.
Apply	Use a color that you select. Click **Browse** and select a color from the Color Picker, or click **Custom** to define your own color. If you select a different icon to assign to one or more records, it also uses the selected color. By default, the color picker shows the 40 standard colors that are used in Analyst's Notebook. (To see the name of a color in the picker, move the pointer over it.) Selecting one of these colors enables searching for the textual value of the color (as an analysis attribute) in Analyst's Notebook. Choosing a custom color results in only the RGB values being used for the analysis attribute.
Restore to Default	Any icon assigned to the specified records uses the standard icon color.

4. Specify which entity or entities use the selected icon by selecting one of the following options:

Option	Description
Assign to this record	The icon is assigned to the record that you select. (You can also drag a record onto the **Assign to this Record**.)
Assign to records in this query	The icon is assigned to records of the entity type that are in the results of a query, which you select. You can only select from queries that output records of the specified type.
Assign to records in this set	The icon is assigned to records of the entity type that are contained in a set, which you select. You can only select from sets that contain records of the specified type.

5. Click **Assign** to assign the icon.

© N. Harris Computer Corporation

Batch editing

If you have permission, you can edit the same field in multiple records (of a specified entity or link type) simultaneously. You can edit all the records of the same type in the database, in a case or a set, or found by a query.

Before you begin

An iBase database that contains very large numbers of sets and queries can cause performance issues. Before you use Batch Edit, delete any sets and queries that are no longer in use.

Procedure

To edit a batch of records:

1. Select **Edit** > **Batch Edit**, and specify the entity or link type of the records you want to edit from the **Entity/Link Type** list.
2. Select the origin of the records from the **Source** list:
   - **All records** - to work on all records of the selected entity or link type.
   - **Query** - to work on the records that are output by a query, which you select.
   - **Set** - to work on the relevant records contained in a set, which you select.
3. In the **Update Field** list, select the field whose values you want to edit.
4. Specify how you want to change the field's value by selecting from the different editing options. If you selected a Text type field for example, you might opt to change the case of the text, or add a prefix.

Option	**Description**
Set To	Enter the value that replaces the existing field value. If you want to change the text in a Multi-line Text (Append Only) field, you need to log on as a system administrator.
Append	For Multi-line (Append Only) fields, you can append the change to the field. To amend the existing value, you must log on as a system administrator.
Convert To	Changes the case of the text.
Prefix, Suffix	Enter the value that is inserted at the beginning of the existing field value (as a prefix) or at the end of the existing field value (as a suffix).
Extract Text	You can keep part of the field value and delete the rest. For example, entering 2 and 5 for a 10-character word keeps characters 2 thru 5 and deletes the first character and characters 6 to 10.
Trim	You can delete characters from the beginning and end of field values. **Leading Spaces**

© N. Harris Computer Corporation

Option	Description
	deletes spaces at the beginning of the field value; **Trailing Spaces** deletes spaces at the end of the field value; **Leading and Trailing Spaces** combines the options.
Adjust	Changes a numeric field value by addition, subtraction, multiplication, or division. Date and time values can also be adjusted by incrementing or decrementing by, for example, days, hours, or minutes.

5. Click **Update**.

Deleting an entity

You can remove records from the database either individually or in batches. When you delete an entity, all links to that entity are deleted too. Deleting a record might cause an alert to be raised.

About this task

Exactly what you can delete, or even whether you can delete records at all, depends on the permissions that are granted to you by your system administrator. You might also need to check with the owner of the record. Depending on how your database is configured, you might have soft delete enabled. Soft deletion prevents records appearing in search results; however, the deleted records can still be restored until they have been deliberately purged.

Attention: If soft delete is not enabled, then deleting records will be a permanent and irreversible operation.

Procedure

1. Find the entity and open the record.
2. Depending on your organization, you may need to check with the owner of the record before deleting the entity. If you need to do this, the owner's username will be displayed as part of the entity details.
3. In the Show dialog, click **Delete** to remove both the entity and its links (you see a confirmation dialog before the deletion completes). The dialog closes after the deletion.

Links

The modeling and analysis facilities in iBase are based on the concepts of entities and links. Links represent relationships between entities, and the characteristics of those relationships.

For example:

- People can be linked to organizations as shareholders, directors, or employees.
- A pair of telephones can be linked by a specific call that is made between them.

You can add links to the database:

- On an individual basis
- By importing them

© N. Harris Computer Corporation

- By using a custom data sheet, in which case you can enter an entity and its links in one operation.

Generally, you add entities and then the links between them. Each link is a separate record in the database, and has a unique record number.

Note: A link can also have a direction, as shown by an arrow, and a line style that indicates the strength, such as whether it is confirmed or unconfirmed. In the **Database Explorer**, you might be able to get more information about a link type by moving the pointer over it to display a tooltip. For information on which semantic types are assigned to the link types in your database, run the **Database Design** report.

To list the link types available to you, click the plus sign to the left of the word Links. Select a link type to display the fields for that link type in the Explorer Detail window.

To find out which combinations of entities and links can be used together, right-click on a link type, and from the menu, selecting **End Types**.

Creating links

After you create the entities at each end of the link, you can add one or more links between them. The link represents the relationship between the entities.

Procedure

To create a link between two entities:

1. Select **New** > **Link** > **`link type`**.
2. Enter the link details.
3. Select the two entities to be connected. For each entity:
   a. Click **Select**. You might find the options are restricted. For example, when you select a record for a link end, the options are restricted to the allowed entity types at the ends of the link type.
   b. Enter the values that you want to find. You do not need to enter the whole value.
   c. Click **Find**. The found records are then listed. Click each record in turn to check it.
   d. Select the entity that you want to use, then click **OK** to review its details by right-clicking on the icon and selecting from the menu.
4. **Optional:** Set the direction of the link by clicking the arrow on the line between the entities and from the **Direction** menu, selecting **Add Arrow**. You can reverse the direction of the link by selecting **Direction** > **Reverse Arrow**. Alternatively you can use the following keyboard shortcuts to set the link direction:
   - No link direction: Press Ctrl+N
   - Right: Press Ctrl+R
   - Left: Press Ctrl+L
   - Both: Press Ctrl+B
5. **Optional:** Set the strength of the link - this indicates whether the link is confirmed, unconfirmed, or tentative. Click the arrow on the line between the entities and select one of the options from the Strength menu. Alternatively you can use the following keyboard shortcuts to set the link strength:
   - Confirmed: Press Ctrl+O
   - Unconfirmed: Press Ctrl+U
   - Tentative: Press Ctrl+T

© N. Harris Computer Corporation

Creating multiple links from one entity

You can create a number of links from one entity, each to another entity, in a single operation. Each link will be the same link type and share the same field values.

About this task

Creating

Note: If two entities were selected, they are placed at either end of the link. With more than two entities selected, they are placed at the End 2 of the link, leaving you to specify the End 1 entity.

Procedure

To create multiple links:

1. Select **New** > **Multiple Links**.
2. Specify the End 1 Entity by clicking **Select** and then selecting the entity that you want to add.
3. Select multiple entities to which you want to create links in the End 2 Entities list:
   - To add entities, click **Add** and then select the entities that you want to add.
   - To remove entities from the list, click **Remove**.
4. If more than one link type is valid between the selected entity types, select the type of link from the list.
5. To add an arrow in the required directions, click the arrow on the line between the entities and select one of the **Direction** options. Alternatively use the following keyboard shortcuts:
   - No direction: Press Ctrl+N
   - Right: Press Ctrl+R
   - Left: Press Ctrl+L
   - Both: Press Ctrl+B
6. To set the strength, click the arrow on the line between the entities and select one of the **Strength** options. Alternatively use the following keyboard shortcuts:
   - Confirmed: Press Ctrl+O
   - Unconfirmed: Press Ctrl+U
   - Tentative: Press Ctrl+T

   Note: When creating links, the default value for link strength is Confirmed. However, you can set a different default value within a given iBase session.
7. Click **OK** to close the dialog and display the Link Details dialog.
8. Enter the details for the link, or links, in either the Link Details dialog (if you had more than one End 2 entity listed) or the New Link dialog (if you had only one End 2 Entity listed).

   Note: If there are no details required for the link type, you are prompted to confirm that you want to create the multiple links.

© N. Harris Computer Corporation

Creating links between open entities

Links can be created between entities that have a supported link type available. If you have two entities that you already have open, you can create a link between valid link end types by dragging one entity to the other.

Procedure

1. Ensure both the entities that you want to link are open.
2. Click the entity icon and hold down the left mouse button, then drag the cursor over the **Show** tab that shows the entity at the other end of the link.
3. Release the mouse button to create the link.

Copying a link

You can copy a link between two entities.

Procedure

1. Find the link that you want to copy. For example, you can use Find, Links, Records, or Browse.
2. Double-click the link.
3. Click Copy or press Ctrl+Y to copy the link details.
4. Click Save or press Ctrl+S to save the copy as a new record.

Using data sheets to add or edit links

Your database designer can set up data sheets to use instead of the standard forms for adding, editing, and deleting data. Data sheets are designed especially for your work, and show the fields that you need, arranged in groupings. Depending on the data sheet, you might be able to use it to enter data for related entities and links.

Depending on the type of data sheet, you can enter details of a main entity, add links and possibly create new link end entities. The fields in the lower part can be a mixture of the link fields and the entity fields.

To use a data sheet to create links between new or existing entities; Select **New** > **Datasheet** > **`datasheet name`**. To review or edit the links between entities: select the entity, right-click and from the menu, select **Show With** > **datasheet name**.

Note: Your database designer might decide to make the data sheet the default method for entering record data for a particular entity. When you want to show records you can choose whether to show the details with a data sheet. To make this choice, select one of the **Show With** options on the menu.

When you use the **Show With** and **New With** options on the menu, you are shown the name of the data sheet in the lower half of the menu. Depending on how the database designer has named a data sheet, it may have the same name as the main entity type. Executing

Showing Links

You often need to see the contents of an individual link, either to read the details, edit the information, or find out who owns the record. In the Show dialog, the link details are displayed in the upper part of the dialog, and the entities which it connects are displayed in the lower part. Showing a link might raise an alert.

To find the link record that you want to display in the Show dialog, you can, for example:

© N. Harris Computer Corporation

- List the link records for a specific link type using dialogs such as Records or Browse
- Search for the link using dialogs such as Find or Query
- List the links in a specific set or query using the Set or Query Records dialog

To list the links from a specific link end entity, use the Links dialog.

In the Show dialog, click **New** to start entering a new link. The New dialog is displayed in place of the Show dialog. If an existing link is displayed, you need to click either **Save** or **Cancel** to complete your work on the displayed record before you can click **New**.

Enter the link details and then select the two entities to be connected by clicking Select. If you were previously editing a link, you will find that the same link end entities are specified, so that you can create a new link between them (you can however change these using Select).

Showing contact details for the owner of the record

Depending on your organization, a record might have an "owner" that you should contact before editing or deleting the record. The user's name is shown in the "owner" (or similarly named) field, and their contact details in the User Information dialog.

There are two ways of displaying their contact details when using the Show dialog:

- Click the username.
- Click **Edit** and then double-click the username.

The record owner might be a different person to the user who created or updated the record. To find out who these users were, right-click the link record in any record list, and from the shortcut menu, select Properties.

Note: Contact details might not be available to you in this database, or might only be available on certain entity and link types.

Merging links

If you have two or more links that you decide relate to the same relationship or transaction, you can merge these links into a single link. However, when links are merged, you cannot split the links back into the original links.

About this task

You can merge links between the same two entities. Merging means that one link (the 'merge' link) inherits data from the other links before those links are deleted, listing the record they were merged into. Only links of the same type, with the same link ends can be merged. Optionally, you can:

- Use the data from the entities that are deleted to replace blank field values in the 'merge' entity.
- Add the data in any Multi-line Text (Append Only) fields to the end of the corresponding fields in the 'merge' entity.

The original creation dates and usernames on the merge link are preserved, and the Update User and Update Date and Time are modified. Where a change is made to a Multi-line Text (Append Only) field, the text MERGE is also inserted after the username.

Note: The record that is displayed in the **Select the link to merge into** section is the record that is kept, and the records that are selected in the **Select the links to be merged** section are deleted when they are merged.

© N. Harris Computer Corporation

Procedure

1. Select **Edit** > **Merge Links**.
2. Select a **Link Type**.
3. In the **Link Ends** section, Select the entities at the ends of the links that you are investigating. When the link ends are selected, a list of the available links is displayed.
4. In the **Select the link to merge into** section, select the 'merge' link.

   Tip: You can check that the correct link is selected by reviewing its details. For example, you can right-click on the label and from the menu, select **Show**.
5. In the **Select the links to be merged** section, select the links to be merged.
6. **Optional:** If you want to assign values from the links that are to be deleted to blank fields in the 'merge' link, turn on **Use the values from the records above to substitute blank values in the 'merge' record (and merge append-only fields)**. If there is more than one selected record in the **Select the links to be merged** section, then the value is taken from the first record in the list. Any values in Multi-line (Append Only) fields are updated with values from the links that are to be deleted.
7. Check that only the required links are selected.
8. Click **Merge** and then click **Yes** to confirm that you want to merge and delete the links in the **Select the links to be merged** section.

Editing links

If you have permission, you can edit link records to add new information or to update existing information. If necessary, check with the owner of the record before you make any changes. Editing a link might raise an alert.

Procedure

1. Find the record that you want to change:
   a. Select **Edit** > **Find Link** and select the required link type.
   b. Enter a piece of information about the link into the appropriate fields and click **Find**. Records that are found with this information are listed.
   c. Right-click on the required link, and from the menu, select **Show**.

   Note: Depending on your organization, you might need to check with the owner of the record before you edit the link. If you need to contact the owner, the owner's username is displayed as part of the record details.
2. Click **Edit** and all fields that you can change are displayed with a white background.

   Note: On some fields, your administrator might set permissions such that you cannot modify the current contents, but you can append text. These fields are typically displayed with a yellow background.
3. Make your changes and click **Save** to store the changes to the link in the database:
   - Set the direction of the link by selecting **Direction** > **Add Arrow**. When you add an arrow, you can reverse the direction of the link by selecting **Reverse Arrow** from the **Direction** menu. Alternatively you can use the following shortcuts to set the link direction:
     - No link direction: Press Ctrl+N
     - Right: Press Ctrl+R

© N. Harris Computer Corporation

- Left: Press Ctrl+L
- Both: Press Ctrl+B

- Set the strength of the link (whether the link is confirmed, unconfirmed, or tentative) by selecting one of the options from the **Strength** menu. Alternatively you can use the following shortcuts to set the link strength:
  - Confirmed: Press Ctrl+O
  - Unconfirmed: Press Ctrl+U
  - Tentative: Press Ctrl+T
- Change the entity at the end of a link.
  - By opening the link, and selecting a different entity.
  - By dragging and dropping a different entity onto the icon that is shown at the end of the link.

Note: Editing a link does not change the owner of the record.

Viewing the valid end types for a link

You can check the types of entity that are allowed at each end of a particular type of link. You cannot change the types of entity, as they are fixed in the definition of the link types in the database design.

About this task

Knowing the valid combinations of entities that can be added to a link is useful information. For example, it can prevent you trying to create invalid link type and end entity type combinations.

Note: It is the combination of End 1 and End 2 types that matters, not the ends they are at or the link direction. For example, there would be no difference between these two examples:

- An Account entity at End 1 and a Person entity at End 2
- A Person entity at End 1 and an Account entity at End 2

Procedure

To view the valid end types:

1. Locate the required link type.
2. Right-click the link type and select **End Types**.

Viewing the links for an entity

You can view the links of a particular entity. For example, you can find out how many links there are of the various link types, the details of the links, which entities the main entity is linked to, and their details.

About this task

The **Link End Summary** lists all the distinct link ends for the selected type, and shows the total number of links between the main entity and each link end. For non-directional links, the number of links is shown in the Count column. For directed links, you can also see the number of links to or from each link end in the To and From columns.

Note: The To and From columns are not displayed, if there are no directed links of the selected type. If you have added a directed link of a type which previously had no directed links in the current session, the To and From columns will not be displayed until you close and re-open the database.

© N. Harris Computer Corporation

When there is more than a single link end for the selected type, an "All" entry is displayed as the first entry in the list. Click the "All" entry to view all the links of the selected type in the Links area.

The format of the labels is determined by the current labeling scheme. Details about the link end record are shown in the Link End Details pane.

Note: If there are many links to load, you can press Esc or Pause to pause the loading of the records, and click **Resume** when you are ready to continue.

Procedure

1. To view the links for an entity, select the entity or link record that you want to investigate. Then right-click the record, and from the menu, select **Links**.
2. View the information available:
   - In the **Links Summary**, click on a link type to view all the current link ends for that type.
   - In the **Link End Summary**, select a link end to display the following information:
     - A summary view of the link end details.
     - All the links to or from this link end. For each link, you can see a summary of the field values. Individual links are identified by the label of the entity at the end of the link and the label of the link itself.
3. **Optional:** Modify the records:
   - To modify the main entity, click the toolbar buttons to:
     - Show, edit, or delete the main entity.
     - Show the main entity, its links, and linked entities on an iBase link chart.
     - Find out whether there are any other records in the database that match the main entity.
   - To modify one or several links:
     - In the **Links** area, select one or more link records, right-click, and select an option from the menu.
   - To modify a link end entity:
     - In **Link End Details**, click **Show Link End**. You can review, edit, or delete the entity.

Creating multiple links from an entity

You can create a number of links from an entity, each to another entity, in a single operation. Each link type is the same, and the links share field values. You specify the entities and the link type, and you are then prompted to enter the link details.

Procedure

To create multiple links:

1. Select **New** > **Multiple Links**.

   Note: Alternatively, in a list of records, select two or more entities, right-click and select **Create Link**. If two entities were selected, they are placed at either end of the link. With more than two entities selected, they are placed at the End 2 of the link, leaving you to specify the End 1 entity.
2. Specify the **End 1 Entity** by clicking **Select** and then selecting the entity that you want to add.
3. Confirm that the entities to which you want to create links are present in the **End 2 Entities** list.

© N. Harris Computer Corporation

4. If more than one link type is valid between the selected entity types, select the type of link from the list.
5. To add an arrow in the required directions, click the arrow on the Link line and select the **Direction**. Alternatively use the following shortcuts:
   - Right: Press Ctrl+R
   - Left: Press Ctrl+L
   - Both: Press Ctrl+B
6. To set the strength, click the arrow on the link line and select the **Strength**. Alternatively use the following shortcuts:
   - Confirmed: Press Ctrl+O
   - Unconfirmed: Press Ctrl+U
   - Tentative: Press Ctrl+T
7. Click **OK**.
8. Enter the details for the link, or links.

   Note: If there are no details required for the link type, you are prompted to confirm that you want to create the multiple links.

Deleting a link

Exactly what you can delete, or even whether you can delete links at all, depends on your permission level. You might also need to check with the owner of the record before you delete it.

About this task

Depending on how your database is configured, you might have soft delete enabled. If soft delete is enabled, deleted records do not appear in any search results; however, the deleted records can still be restored until they have been purged.

Procedure

To delete a link:

1. Find the link.
2. Depending on your organization, you might need to check with the owner of the record before you delete the link. If you need to do this, the owner's username will be displayed as part of the link details.
3. Click Delete or press Ctrl+D to remove the link (you see a confirmation dialog before the deletion completes).

Entering and editing data

You can enter, maintain, and find data in any screen that display records. How you work with the fields in a record, depends on the field type and where the record is being accessed.

Recognizing mandatory fields

When you enter information in a new record, you must complete all mandatory fields. Mandatory fields are shown in blue (either the field name or the text box itself).

© N. Harris Computer Corporation

Entering and editing data

- You can enter and change data in a field with a white background.
- You cannot edit a field that has a gray background.
- You might be able to make a gray field editable by clicking **Edit** or by selecting a checkbox, for example for date fields. However, some fields are read-only and can never be edited.

Note: You might be able to move your mouse over a field to display a tooltip that provides helpful information.

Entering Unicode characters

Depending on your database, you might be able to enter Unicode characters. To find out whether Unicode characters are supported:

- Select **File** > **Properties** > **Database Properties**.

If Unicode characters are allowed, then the **Use Unicode Data types** checkbox is selected.

Note: Unless your database supports Unicode characters, do not enter characters into a record that are not included in the character set for your language group.

Selecting the record owner

Depending on your organization, each type of entity and link record might have an "owner" field or something equivalent. This identifies the person to contact about the record. When adding a new entity or link, the field might display $ as the default value, which means that you will become the owner of the record when the record is saved.

To select a different user as the owner:

1. Type $ in the "owner" field. This indicates that you are looking for a username rather than a file.
2. Click the **Browse** button next to the "owner" field to display the list of users for this database. If you know the first few characters of the name, enter these first - this will then scroll the list to that position in the list.
3. Double-click on a name to select that person as the owner.

Finding out the field type

To find out the type of a field. For example, the type of a Grade field in a Document entity:

1. In the Database Explorer view, click the record's entity or link type.
2. In the Explorer detail window, look up the field type in the Type column.

Entering text

To enter new text or edit existing text, click the text box or in a multi-line text box, double-click to display a Memo Editor, with scroll bars.

About this task

There are three types of text box:

- Text - text boxes are limited to the number of characters they can contain; this maximum number is set in the database design for the field.
- Multiline Text - similar to text boxes except that you can start new lines by pressing the Enter key.

© N. Harris Computer Corporation

- Multiline Text (Append Only) - in fields of this type you can add new text, but you cannot edit existing text.

Procedure

To edit a Multiline Text (append only) field:

1. Double-click in the field to display the Memo Editor. The original (non-editable text) displays, and a line is added to the bottom that tags new text with the username, and the date and time.

   Note: If you are logged in to the database as the system administrator, you can edit the field without using the Memo Editor. In this case, any additional text you add is not tagged with your name and the date and time.
2. Enter text in the bottom pane. To start new lines, press the Enter key.
3. Click **OK** to save your changes.

Results

The text that you enter in the bottom pane is added to the existing text in the field; none of the text in the field can be edited or deleted.

Note: In multiline text boxes, you can set the number of rows to be displayed. Select **Tools** > **Options** and select the **Number of rows to be displayed**.

Checking the spelling on records

When you are entering text for records either directly, or that use a data sheet, you can check the spelling of the record details. Terms are checked against a standard dictionary for your selected language and any custom terms in your local dictionary.

Procedure

To check the text stored in records:

1. Open a record and ensure that it is in edit mode.
2. Select **Spellcheck**.

   Note: If you are checking the spelling on a data sheet, the main record and the open link will be checked. To ensure that spelling is correct on other linked record, you need to run the spell checker with each linked record open separately.
3. For each term that the spell check flags, decide how you would like to handle the spelling:

Option	Description
Ignore	To ignore this instance of the term.
Ignore All	To ignore all instances of the term in the current spelling session.
Add	To add the term to your custom dictionary, marking the spelling as correct.

© N. Harris Computer Corporation

Option	Description
Change	To change the flagged term to one of the suggested terms, select the correct term and click **Change**.
Change All	To change all instances of the flagged term to one of the suggested terms, select the correct term and click **Change All** .

4. Click **Close**.

Numbers and currencies

You can use a range of field types to enter and maintain numerical and financial data. To enter a numerical value, select the number or currency field and enter a value or edit an existing value. When you type numbers and currencies, the figures you enter might be converted to a different format for saving and display.

The different numerical field types are:

- **Calculated numbers** are whole numbers that are derived from other field values, such as the sum of two other numbers. It is not possible to edit calculated numbers.
- **Counting numbers** are used for whole numbers. You can only enter numbers and a period (.), used as a decimal point. On saving, a prefix and commas (,) can be added; for example, 1675 might become gm 1,675. Although you can enter a decimal point, the typed figure is rounded to the nearest whole number on saving. For example, typing 3.49 or 2.51 is always saved as 3.
- **Real Numbers** are used for numbers that might have fractions. You can enter numbers, periods (as a decimal point), and the characters defined by your Windows™-defined regional number options (perhaps commas (,) to separate thousands). On saving, a prefix might be added and decimal places rounded. For example, 83.47 might become mL 83.5.
- **Currency** is used for financial values. For the decimal point in a currency value, you use the character that is specified in the Windows regional currency options. On saving, the value might be rounded and a currency symbol added. For example, 4856.4872 might become $4856.49.

Note: Only numbers between -2147483648 and 2147483647 are accepted.

Dates

You can enter and maintain many dates on records, and also find and select records by using date information. You can enter and maintain many dates on records, and also find and select records by using date information.

About this task

You enter and edit dates, in Date type fields, in the format defined by the Short date Windows™ Regional settings.

To specify a blank date for the record (even if a date is already displayed), turn off the Date checkbox.

Procedure

To enter a date:

© N. Harris Computer Corporation

1. Turn on the **Date** checkbox.
2. Click the arrow to the right of the field to display the calendar.

   Tip: By default the calendar is displayed with today's date highlighted. To go to today's date from anywhere in the calendar, right-click and select **Go to today**.
3. Select the date that you would like to use.

Documents and hyperlinks

You can use document and hyperlink fields to include documents or links to other iBase records in your database. When you are editing records, you can load documents or add links to them. When you are not editing, you can access the documents or records.

About this task

Document fields are used to store documents in a record in the database. The document is represented on charts by its icon. You add a document to the database by loading it using a Document type field.

Common document types include:

Type	File extension
Microsoft Word document	.doc
File format that many word-processing programs understand	.rtf
Text document (no formatting)	.txt
Microsoft Excel spreadsheet	.xls

When a document is loaded, you can open the file, save a copy, or delete it from the record. Although the contents of documents can be added to the Search 360 index, you cannot search on Document type fields by using the Find or Select options.

Note: iBase prevents you from opening certain types of files that could contain malicious code. For more information, see Blocked file types.

To load a graphic to represent an entity on a chart, use a picture type field instead.

Procedure

Hyperlink fields are used to store links to documents or iBase records. To enter a hyperlink:

1. Browse for the required document, or click to type the target of the hyperlink.
2. To add another hyperlink, click **Add** to display a blank hyperlink field. For example:

Target	Format example
Web page	`http://www.example.com`
File	`C:\documents\report.doc`

© N. Harris Computer Corporation

Target	Format example
iBase record	#PER15 (where PER15 is the Record ID of an iBase record, and # identifies the text as a Record ID rather than a web page or file)

Field attachments

If you want to add documents or images that provide information about a specific field, but you don't need those documents to be indexed, you can use field attachments. Field attachments can only be used in SQL Server databases that are set up to use them.

Procedure

- To add an attachment to a field:
  1. Open a record and select **Edit**.
  2. Select the attachment icon (a paperclip) next to the field name.
  3. Browse and select the attachment to add.
- To save a local copy of an attachment:
  1. Open a record that contains an attachment.
  2. Select the attachment icon (a paperclip) next to the field name.
  3. Browse and select a location to save the file.

Blocked file types

To reduce the risk of opening a file that might contain malicious code, iBase uses a blocklist of file types that you cannot open or view in the application. If you need to, you can add file types to (or remove them from) the list.

By default, iBase blocks files of all the following types from being opened:

```
.exe  .scr  .vbs  .vb   .js   .xml  .docm
.xps  .iso  .img  .arj  .lzh  .r14  .r18
.r25  .tar  .ace  .zip  .jar  .com  .bat
.cmd  .msi  .ws   .wsf  .scf  .scr  .pif
```

If you attempt to open a file of a blocked type, iBase gives you the option to save the file to your workstation, where you can open it outside the application.

The blocklist is stored in the Windows registry. If you have the necessary permissions, you can edit the list in the following registry key:

`HKEY_CURRENT_USER\SOFTWARE\i2\iBase\UserSettings\BlockOpenFileTypes`

After you edit the list, you must restart iBase for the changes to take effect.

Icons and pictures

You use icon and pictures to represent records on iBase link charts, on Analyst's Notebook charts, and in iBase.

There are two field types:

Icon fields

© N. Harris Computer Corporation

Icon fields are used to select an icon to represent the record. Select an icon name from the list for the field. To use the standard icon for the entity type, select the blank option at the top of the list. The standard icon for each entity type is specified in iBase Designer. If there is no blank option, then the field is mandatory and you must choose an icon from the list.

To change the icon shading color, select **Edit** > **Assign Icons**.

Note: Not all entity types have an icon field. To change the icon for an entity that doesn't have an icon field, you can also use **Assign Icons**.

Picture fields

Picture fields are used to store graphics, such as photographs, in the record so that you can display the actual graphics on charts rather than using icons.

To add a graphic to a record, load it by using a picture type field. For more information, see Documents and hyperlinks.

A picture can be used instead of the icon to represent the record on charts:

- Select **Tools** > **Options** > **Charting** and turn on **Chart pictures to represent records instead of their icons**.

Yes or No fields

Click a Yes or No field to select the option or to deselect it.

The field is:

- Turned on when a check mark is shown in the box, which corresponds to Yes.
- Turned off when there is no check mark, which corresponds to No.

Note: When you import Yes or No fields from a text file into SQL Server databases, blank values are interpreted as 'No'. This is useful where a Yes or No field is left blank for 'No'.

In some situations, for example when you are trying to **Find** records, there is a third setting that means either.

Code lists

Code lists can be used to suggest or restrict the values that are added to fields. Ensuring that field values are aligned across a database can improve search results.

There are three different types of code list:

- **Selected from** - you can only use the items on the list.
- **Suggested from** - you can either use an item on this list or enter your own value.
- **Security Classification Code** - this type of code list controls access to records.

  Attention: You can select security classification codes that deny you further access to the record after you save it.

If you want to find a field value that is no longer included in a list, use a query rather than Find or Select. In a query, you are allowed a 'free entry' so you can specify non-code list values.

Note: If the list does not contain the item that you want, you can add it by selecting **Code Lists** from the Edit menu (provided you have the required permissions as a user).

© N. Harris Computer Corporation

Times and time zones

You enter new times by turning on the checkbox to edit the time in the format defined by your Windows™ Regional Options. Alternatively you can turn it off to specify a blank time (even though a time might be displayed).

Procedure

To edit a time:

1. Click a part of the time to edit it.
2. Enter the time by using the number keys or the up and down arrow keys. To edit AM or PM, click it and then press the A key to change to AM or P to change to PM.

   On saving, the time might be changed for storing and display. For example, 9:08 AM might become 09:08:00.

   Time zones are used along with date and time type fields.

Strength fields

Line strengths are a useful way of indicating the general status or quality of information that is represented by an item. For example, whether an association between two people is definite or speculative, or that an event frame contains information that is unconfirmed.

When you add or edit links, you can click a Strength field to select a strength value:

- Confirmed
- Unconfirmed
- Tentative

Note: The default link strength is Confirmed but you can set a session default of your own so that every link you create or import during that session uses your value.

Coordinates in iBase

To plot an entity or link on a map, you need to enter coordinate values in two fields that are set up for this purpose. Your GIS package is configured to interpret the values in these fields so that the data can be plotted in the correct location.

In iBase, you will also be able to store geographic data in a number of formats, which are then converted, either manually when you enter the record or automatically after an import or using a bulk conversion. You can also run coordinate queries.

Types of field

The fields used to contain the coordinate data must be defined as Real Number type fields. They can contain the following types of coordinates:

- Latitude and Longitude values, entered in decimal degrees
- Easting and Northing data, entered in meters

These fields will typically be called Latitude and Longitude or X and Y. If you are not sure which fields you need to use, move the pointer over the field name to see its tooltip, or speak to your database administrator.

You might also have a Coordinate type field.

© N. Harris Computer Corporation

Note: You can only use latitude and longitude (decimal degrees) if Coordinate type fields are used.

Converting coordinates to a standard format

When you convert coordinates, they are always converted to decimal degrees of latitude and longitude, using the WGS 1984 datum (a global standard for plotting geographic locations).

To convert coordinates, the entity type requires a Coordinate type field in addition to fields for the latitude and longitude. The Coordinate type field must be directly above the latitude and longitude fields. You enter the coordinates in the Coordinate type field and the coordinates are then automatically converted and displayed in the latitude and longitude field.

The original coordinate value is stored so that it can be searched for, and for audit purposes.

Note: The conversion process changes longitude values greater than 180 to their equivalent negative value in order that they can be plotted correctly.

System fields

Values in system fields are automatically added when the record is saved, and can be displayed by right-clicking on the record and selecting Properties from the menu. You cannot edit the values in a system field.

However, you can search using these values:

- Record ID - the record's ID.
- Create Date and Create User - the date and time the record was created, and the username of the person who added the record.
- Update Date and Update User - the date and time the record was last updated, and the username of the person who made the change.

Entering and reviewing data on a data sheet

You can use a data sheet to create, edit, and review entities and links. How you do this depends on how a data sheet is designed.

For example, you might be able to:

- Create an entity.
- Create an entity along with some links and link end entities.
- Create links and link end entities for an existing entity.

Before you edit or delete an entity and its links, you might need to check with the owner of the records. For more information, see Show an entity.

Starting with a blank data sheet

If the data sheet is already displayed for the correct entity type, click **New** or press `Ctrl+W`. If you are currently working on another record, then you must first save your changes or cancel them.

Editing existing data in a data sheet

If a blank data sheet is already displayed, click **Select** to find the entity you want to edit. If the entity is already displayed, for example as a result of showing it, click **Edit** or press `Ctrl+E` instead.

© N. Harris Computer Corporation

Working on the main entity

1. Use the upper part of the data sheet to enter, or edit, the details of the main entity. The information that is required for the main entity might be organized into a series of pages.

   Notice that:

   - Mandatory fields, those that you must complete, are displayed with a blue label.
   - Fields into which you can enter data are displayed with a white background. For more information about entering data into the different iBase field types, see Entering and editing data.
   - If the database designer has set this up, you can be able to get help on what to enter in each field by moving the mouse pointer over the field labels. This displays a tooltip that will tell you something about the type of data to enter in each field.

   If you start to enter an entity that matches one that is already in the database, then you might see a message similar to:

   ```
   1 record with matching discriminators has been found in the database.
   ```

   For more information, see Creating an entity.

2. Click **Save** or press `Ctrl+S` to save the details of the main entity. If you see a message similar to the following, you are trying to save an entity that matches one that is already in the database:

   ```
   WARNING: 1 matching record has been found in the database.
   ```

For more information, see Creating an entity.

Note: To continue working on a data sheet after you save it, click **Edit** or press `Ctrl+E`.

Working on the links and link end entities

Depending on the data sheet, you can define the entities that are linked to the main entity, and the relationship between them. There can be more than one link between a main entity and another entity.

1. In the lower area of the data sheet, go to the appropriate page by clicking its tab or hyperlink. This determines which entity and link types you can select or add. The page can be further subdivided with its own tabs or hyperlinks, and you might need to use the scroll bar in order to see all the fields.
2. Specify the entity or entities that are linked to the main entity. For example, you can:
   - Select an existing entity for the link end by clicking **Select**. Depending on the data sheet, you might then need to enter some additional information for the link between this entity and the main entity.
   - Enter the details of a new entity for the link end. This entity is saved when you save the data sheet.
   - Add a second entity or link by clicking **New** (or press `Ctrl+W`) or **Copy** and then entering the details. If you click **Copy**, you need to change some of the details in the copy otherwise you create a duplicate record.
   - Remove the displayed link from the data sheet by clicking **Remove**.
3. If required, change the line style of the link. You can choose between Confirmed, Unconfirmed, or Tentative.
4. You can review the linked entities by clicking '>' to step through all the link end entities, if there is more than one. The status of the link is shown. For example, `** New Link ** (Changed)` means that you have edited the link but not yet saved it.

© N. Harris Computer Corporation

5. Save your work by clicking **Save** or pressing `Ctrl+S`. To continue working on this data sheet, click **Edit** or press `Ctrl+E`.

Reviewing data using a data sheet

To review entity and link data in a data sheet:

1. Click **Select** to find the main entity that is the subject of the data sheet. Depending on the design of the data sheet, this will display the links from the entity and the other entities to which it is linked - the link and entity types will depend on the design of the data sheet.
2. Review the entity and link details on the main page. You can go to the other pages (if any) by clicking the tabs or hyperlinks.

   For the link end entities in the lower half of the data sheet, you can change between a summary list view and a more detailed record view by clicking these two buttons:

   -

     Record

     Display the details of the selected record. Click > to step through all the link end entities, if there is more than one. For example, to display record '1 of 2' and then record '2 of 2'.

     You can edit the selected record by clicking **Edit** or pressing `Ctrl+E`.

   -

     List

     View summary details of all the records.
3. To review other aspects of this data, for example, all the links from the main entity, right-click on the icon in the upper left of the dialog, and select from the shortcut menu.

Deleting the records in a data sheet

You can delete the main entity of a data sheet and its links by clicking Delete or pressing Ctrl+D in the Datasheet dialog. This will not delete the link end entities. To delete these:

1. Find the link end entity. For more information, see Finding records.
2. Right-click on the entity that you want to delete and from the shortcut menu, select **Show** to display the Show or Datasheet dialogs.
3. Click **Delete** to remove the link end entity.

Starting other operations with the entities in the data sheet

You can start other operations with the main entity and the linked entities shown on the data sheet.

For the main entity:

- Click the toolbar buttons at the top of the dialog:

© N. Harris Computer Corporation

- **Show links** - List full details of its links in the Links dialog.
- **Link chart** - View the entity and all its links on an iBase link chart.
- **Matching records** - Find out if other records in the database share common features with this record.

For a link end entity:

- Click the **Links** button to see full details of the links from this entity (in list form), or click the drop-down arrow and select from the shortcut menu.

Adding notes to folders

You can add notes to sets, queries, definitions, and import and export specifications. These notes can then be viewed later to provide more information.

Procedure

1. Click **Description**.
2. Enter the text for the description and click **OK**.

Results

You can view the text in the list of items, or by moving the mouse pointer over the name.

Note: You might need to scroll to the right to read the complete text.

Selecting the record owner

You can choose the owner of a record from the users with access to this database. When you are selecting an owner, you might see additional information for those users who enter their details.

To select a user as the owner of the current record:

- Select the user from the list and click **OK**.
- Double-click the user.

Note: If there is a long list of names, enter the first letter of the name to scroll to that position in the list.

Viewing the record history

In SQL Server databases, you might be able to view the history of the changes that are made to records in the database (provided your organization uses this feature). How far back the history goes depends on how frequently your system administrator archives this data.

Procedure

1. Select the records to view:
   - In any list of records, right-click on one or more selected records, and from the menu, select **Show History**.
   - With a record open, click **History**.

   Note: If Audit History is not configured, these menu commands are missing.
2. To select further records of interest, open the **Audit History Viewer**, and click **Select**.
3. Refine your selection:

© N. Harris Computer Corporation

- Filter the records that are displayed by user, entity and link type, and by time.
- To display all records touched by a specific user, in the **Records to display** area, select **All records used by** and select the username from the list. The selected username is displayed as a reminder.
- To filter by entity and link types, in the **Types to display** area, select the entities and link types.

  Note: Only types with records in your selection are listed, and the records for a selected type are displayed only when the appropriate checkbox is turned on.
- To filter by time, in the **Time period to display** area, select a time period.

Note: Depending on the database, the number of times the records have been viewed, but not edited, is displayed in the **Views** column. Viewing includes all the following activities: listing in a record list, opening a record, listing or viewing in the Audit History.

4. Select the information to display in the edit history:
   a. Make sure that **by all users** is selected from the **Show Edits** list.
   b. Select from the following options:
      - **Show Headers** - to hide or show the shaded line that displays either the date/time/username or the field name. You cannot expand and collapse when this option is turned off.
      - **Plus** - to expand the history.
      - **Minus** - to collapse the history.
      - **Audit** - groups the entries by the name of the user who worked on the record and when they were created, updated, or deleted.
      - **Field** - groups the entries by the data that has been added, updated, or deleted. Click again to sort in ascending or descending order by date edited.
      - **Edits** - displays a history of the changes to the record (only available if the database is set to audit level 5)
      - **Views** - displays a history of who viewed the record and when (only available if the database is set to audit level 5)
   c. To display the history for a specific user, from the **Show Edits** list, select the username.

Results

The history of edits area shows information on the changes that are made to the selected record:

Entry	Description
Field Name	The old and current values.
Edited by	The logon name of the user who made the change.
Date Edited	The date and time of the change.
Reason	If required by the database, the reason given by the user for making the change.

© N. Harris Computer Corporation

Entry	Description
OS User	The Windows™ name of the user made the change.
Machine Name	The machine that the user was working on.
Location	The location as entered in the User Information.
iBase Change	When is turned off, the update was made outside iBase.
Extra Detail	You might see an Extra Detail column that displays additional information for the current record.

More data might be shown for each record, including:

- The name of the icon if an alternative icon is assigned to the record.
- The icon color (which is blank if the standard icon color is used)
- The record status (applicable only if Soft Delete is used). The record status can be Soft Deleted, Normal (the record has been soft deleted and restored), and Purged.
- Security Classification, the old and new SC code (if this feature is used and if you have authority to view this information)

Some information can be displayed that you do not usually see, such as the date the record was created and the record ID.

Changes to Code Lists

You can view changes to a specific pick list, icon list, or SCC List. In the list, click **History**.

Note: There is no option to view the history if:

- The database is stored in Microsoft™ Access.
- The database is not set up to record audit history.
- You do not have access to the audit history.

Changes can include:

- Old and new values
- Old and new descriptions
- Old and new parent pick lists, for filtered pick lists

All the changes that are made in the same session are grouped by username, date, and time.

As there can be several pages of changes, you can print the list or save it as a spreadsheet or PDF file.

To find out whether the audit history is recorded, select **File** > **Properties** > **Database Properties** and check the setting of the **Audit History**.

© N. Harris Computer Corporation

Viewing charts

There are two ways of viewing a chart that is embedded in an entity. You can either view the chart in Analyst's Notebook, or you can use Chart Viewer.

Procedure

1. To view a chart:
   a. To view the chart in Analyst's Notebook: right-click on the chart, and select **View**.
   b. To view the chart in the iBase Chart Viewer: right-click on the chart, and select **Chart Viewer**.
2. Depending on the chart type, review the cover sheet and then click **Open** if you want to continue. Both Chart Viewer and Analyst's Notebook display the chart in the same way but changes cannot be saved in Chart Viewer.
3. Some chart items might have additional information that is stored on cards, or as part of their item properties. To display this information, right-click on the chart item, and from the menu, select **Edit Item Properties**.

Importing data

You can import data from iBase databases, other databases, excel spreadsheets, or text files. Before you can import any data, you define how the source data is to be interpreted during the import in an import specification. You can run the import specifications singly or in a batch (if you first set up an import batch specification).

Planning imports

Before you create an import specification, compare the data that you would like to import with the database structure to determine the item types to import. In addition, determine the fields that are mandatory, and the fields that can be used as identifiers (used to decide whether data matches records in the database).

Note: If you are importing entities or links that use multi-line text (append only) fields, you might want to test the import first. You can only add text to the end of fields of this type - you cannot delete or edit existing text without removing the record.

Validating and protecting the data

During the import, you can check that values imported into Selected from Code List type fields are valid; any invalid values are reported as errors during the import.

You can protect existing data by turning on the **Do not update existing field values with blank values** checkbox. This option prevents existing data from being overwritten by blank values in the source data.

If you set up comprehensive record matching, you can control how records are created or updated. For more information, see:

- Matching entities in importing
- Matching links in importing

Manipulating the data

When you import data, you can transform field values in source records before you assign them to iBase fields. You can:

- Copy a value to assign it to more than one field.

© N. Harris Computer Corporation

- Merge two or more values to assign them to a single field.
- Split a value to assign parts to several fields.
- Update specified values with new ones (for example where the source data uses a different code list) by creating a substitution file.
- Trim unwanted space from the start or end of a source field.

For more information, see Transforming source data.

Recording the results of the import

You can record the results of the import by saving new and modified records in a set.

If required, you can log the numbers of new and modified records to a file.

Note: If you use auditing with a Microsoft™ Access database, and the audit level is set to 4 then the audit log only records the start and end of the import. It does not log the individual records.

Handling errors

You can save the errors that might occur during the import to a file. You can then fix the problems with the source data by editing the error file, and reimport the remainder of the data by using the error file as the source for the import.

Bulk imports and importing XML data

A bulk import allows significantly faster importing and importing from XML files. It is also useful for importing large quantities of data without user intervention. System administrators can set up a bulk import, that uses an import specification, although there are a few minor differences between a standard and a bulk import.

Note: In a case-controlled database, you can only import data into the case in which you are currently working.

After importing large numbers of records, you might want to compact your database.

Importing data with an existing specification

How records are imported is defined in an import specification. Your system administrator might set up the import specifications for you, or you might need to create your own.

About this task

The import specification defines the following details of the import:

- Which entity type or link type is being imported.
- For links, which entity types are imported as the link ends.
- The path or file that contains the data to import.
- How values in the data correspond to fields in the database.
- If you want to check for potential duplicates, which fields are used to check whether the data matches any existing records, and what happens when a match is found.
- Whether a set is created to contain the imported data, and its name.

The import specification can be defined to prompt for confirmation before records are updated or created. If this prompt is enabled, you need to confirm whether:

© N. Harris Computer Corporation

- An existing record is updated with data from matching information in the import file.
- When there is no matching record, whether a new record is created.

Note: You cannot run import specifications that import entities or links for which you do not have write permission to all the fields, or if you do not have permission to create sets.

Procedure

To use an existing import specification:

1. Right-click on **Import Specifications** and from the menu, select **List**.
2. Double-click an existing import specification to load it.
3. Click **Next** to review the details of the import specification.
4. **Optional:** Click **Session Defaults** to define default values for the standard fields that are common to all entity and link types in the database and link strength. They are not saved as part of the import specification.
5. On the last page, click **Run** to start the import. You might be prompted to confirm the import of particular records.

   Tip: Click **Verify** to see any errors without importing the data.

   Attention: If the import takes too long to complete, you can stop it by clicking **Stop**. Stopping an import does not cancel the records that are already imported but no set is created.
6. When the import completes, you are shown a summary of the import.
7. The final stage of any import is to check to see that the records are imported as you expect. For example, examine the records in an import set or run a query.

Creating import specifications

An import specification defines the types of record that should be created from a set of data. You must use an import specification for each import that you want to complete.

About this task

Before you can import any data, you need to define:

- The type of data you want to import.
- The type and format of the source data
- How you want to manipulate the source data (optional)
- How you want to validate the source data and match it to existing records.
- How you want to record the results of the import (optional)

These definitions are created in an import specification that you can create by using the import wizard.

Regardless of your progress in creating an import specification, you can:

- Save the current specification.
- Add or view a description of the specification.

Note: To modify an existing import specification; on the first step of the import wizard, click **Load**.

© N. Harris Computer Corporation

Selecting the type of records to import

iBase can import data from a number of different sources. The first step in creating an import specification is identifying the structure and location of data that will be added to records.

Procedure

1. Select **File** > **Data** > **Import**.
2. Select the type of record that will be created. Each import specification can import data for single entity type or a link type and two supported link ends. Specify whether the data corresponds to Entities or Links, and select the type of record to create.
3. **Optional:** Select the options that apply to your import:
   - **Validate imported 'Selected from Code List' values** - If you want to verify that the values in the source data are valid for the code lists in the current database. Invalid values are reported as errors during an import.
   - **Do not update existing field values with blank values** - If you want to prevent blank fields in the source data from overwriting fields in existing iBase records that contain data, turn on this option. This option applies where the action on discovering a match with existing data is **Modify Record**.
4. Select the type of file that contains the data:
   - **Text File** - Data in a file.
   - **OLE DB Compliant Data Source** - Data is held, for example, in a Microsoft™ Access database, in a Microsoft SQL Server database, or in an Oracle database. For more information, see Importing from OLE DB data sources.
   - **XML (iBase Schema) File** - For more information, see Bulk importing.
   - **Microsoft Excel Worksheet** - Data in a .xls file. For more information, see Importing from Microsoft Excel worksheets.
   - **Folder Contents** - Data is held in a specified folder. For more information, see Importing files from a folder.
   - **The import source contains 'Record ID's that originate from this database** - The data that is being imported relates to updates to specific records that are already in this database. For more information, see Exporting and importing externally edited iBase data.
5. **Optional:** On Page 3 of the Import Wizard, you can specify that the import specification is intended for bulk import. In addition, you can select **Do not perform safety checks before importing** to prevent verification of the number of records in the database that would be updated by the import. The Bulk Import checkbox is unavailable if the database has not been activated to allow bulk import, or you are using an Access database.

Bulk importing

Bulk imports enable you to import data more quickly, and should be considered if you have large volumes of data to import or if you find the standard importer too slow.

You can only run bulk import on an SQL Server database. Bulk imports from XML files additionally require that the database supports Unicode. In addition, you can only run a bulk import from iBase Designer or the Scheduler utility. Use the Scheduler to run bulk imports at times when the database is not being used.

© N. Harris Computer Corporation

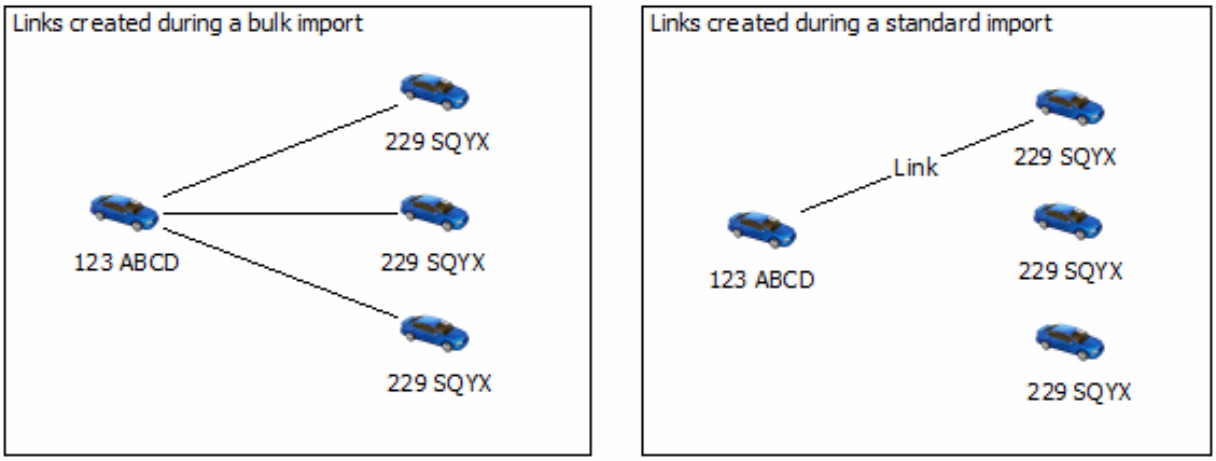
What is a bulk import?

A bulk import allows significantly faster importing, and is useful for importing large quantities of data without user intervention. You set up a bulk import in the same way as any other import, using an import specification, although there are a few minor differences between a standard and a bulk import (see the next section).

To define a bulk import specification:

- You need to be logged on as a database administrator.

Note that bulk importing has the potential to create more links than a standard import. In bulk importing, all specified links between matching link ends are created, in contrast, for standard imports only the first link between specified ends is created, see the example below:



A bulk import specification is the same as any other import specification, with the following limitations:

- You cannot import picture and document fields.
- There is no user action during the import to confirm matching records.

Differences between bulk imports and standard imports

Bulk imports have the following features:

- Bulk imports are not sensitive to trailing spaces.
- The order of importing elements can be different. When importing link with ends of the same type, bulk import will import all end 1 records before all end 2 records. If records are updated by both end 1 and end 2 data, end 2 updates will take precedence.
- Bulk imports are case sensitive when comparing the contents of Append Only fields.
- String comparisons take account of the locale.
- If no records are imported, an empty import set will be created, to identify the fact that the import took place.

Bulk import is incompatible with Audit Levels 4 and 5. At audit level 4 or 5 changes to individual records are audited, but when running a Bulk Import the creation or update of individual records is not audited.

© N. Harris Computer Corporation

Defining a bulk import specification

Bulk import specifications are defined, edited and saved in the same way as any other import specification. You can create a new specification from scratch, typically in iBase rather than iBase Designer, or load an existing one. For more information, see creating import specifications.

To mark the import specification as a bulk import, turn on the Bulk Import checkbox on Page 3 of the Import Wizard.

Note: The Bulk Import checkbox is unavailable if the database has not been activated to allow bulk import, or you are using an Access database.

Importing into a database with case control

If your database is case enabled, you have to specify the case into which you want to import the data when running the import.

When you run the bulk import, the Select Case dialog is displayed. Select a single case to which all the imported records will be added.

Note: You cannot run a bulk import into a case-enabled database using the Scheduler utility.

Specifying information about links

If you are importing link data, you need to specify the entity types that are at the ends of the link. In addition, the link direction and strength need to be set.

About this task

Procedure

1. Specify valid link ends for both **End (1)** and **End (2)**.

   Note: If you select **Fixed**, you can select a specific entity that is used as End (1) for all imported links.
2. Specify the link direction:
   - None - no arrows
   - End (1) to End (2) - an arrow points to End 2
   - End (2) to End (1) - an arrow points to End 1
   - Both - an arrowhead at each end
   - From Import Source - The direction of the link is derived from the data that is mapped to the direction field using the following code:
     - 0 = None
     - 1 = End (1) to End (2)
     - 2 = End (2) to End (1)
     - 3 = Both
3. Select a specified strength or click **From Import Source** to specify the strength based on the data that is mapped to the strength field that uses the following code:
   - 0 = Confirmed
   - 1 = Unconfirmed

© N. Harris Computer Corporation

- 2 = Tentative

Identifying records and fields in your data

Depending on the source of the data that you are importing, the process of identifying record and field specific data is different. To successfully import data, you must ensure that the characters used to separate records and the fields that are contained are identified.

About this task

The way that information is separated within the original data, must be specified so that the values can be correctly mapped to field types in iBase. Depending on your source, the steps to identify your data will differ.

- To identify records in an OLE DB datasource, see Importing from OLE DB data sources.
- To identify records in a Microsoft Excel spreadsheet, see Importing from Microsoft Excel worksheets.
- To identify records in files in a directory, see Importing files from a folder.

Procedure

To identify records in text files:

1. In the **File Name**, specify the file path and name for the file that contains your data.
2. To determine where to start the import, specify the record to start the import from, and whether the first record is a list of the field names.
3. Select the characteristics of the text:

Option	Description
Text Qualifier	The Character to use to identify the start of the data to import.
Auto Trim Leading and Trailing Spaces in Data	Removes any spaces at the beginning and end of a field value.
Record Delimiter	The character that is used to split the data into separate records.
Field Delimiter	The character that is used to split the data in a record into different fields.
Dates, Times & Numbers	The formats used to store different numerical values within the data.

© N. Harris Computer Corporation

Importing from OLE DB data sources

To import from an external data source, such as Oracle, Microsoft™ SQL Server, or Microsoft Access, iBase must connect to the data source, which it does through facilities in Microsoft Windows™.

About this task

The most efficient way to import from a data source is to use one of these providers:

- Microsoft Office 16.0 Access Database Engine OLE DB Provider (for Microsoft Access databases)
- Microsoft OLE DB Provider for Oracle
- Microsoft OLE DB Driver for SQL Server, or SQL Server Native Client

You can use Microsoft OLE DB Provider for ODBC Drivers, but this method is slower and less efficient, and is not described here.

You might need to ask your system administrator for information on the source database to set up a connection to it. For example, for SQL Server databases, you need to know:

- The server name.
- Whether Windows Integrated security is used or whether you must enter a username and password.
- Which method to use to select the database.

Note: When you import from a Microsoft Access database that contains linked tables, check that the database that contains the physical tables that are being linked is also available.

Importing from an Access database

Before you can import from a Microsoft Access database, you need to obtain the password from iBase Designer. From the iBase Designer **Tools** menu, click **Feature Availability > Options** and open the **Advanced** tab. The password is displayed in the **Database Password** field. Copy the password to the clipboard.

1. In iBase, click **File > Data > Import**. The import wizard opens.
2. In Step 1, select **OLE DB Compliant Data Source**. Click **Next**.
3. In the Connection area, click **OLE DB Data Source...**.
4. On the Provider page, select **Microsoft Office 16.0 Access Database Engine OLE DB**.
5. On the **All** tab of the **Data Link Properties** page, select **Jet OLEDB:Database Password**, click **Edit Value**, and enter the password that you copied from iBase Designer.
6. On the **Connection** tab, enter the location of the Microsoft Access database in the **Data Source** field. This is in the format of a file path ending with a `.mdb` file.
7. If required, enter the username and password for logging on to the Access database.
8. Click **Test Connection** to check that the connection is successful.
9. Click **OK**. The **OLE DB Data Source** field in Step 2, changes to bold to indicate a connection is added (although not that the connection is correct).
10. From the **Source** drop-down list, select the Microsoft Access table.
11. Click **Next** to continue in the usual way.

Changing the import timeout value

Imports from OLE DB data sources have a 30 second timeout by default. It is possible for your system administrator to change the timeout value by editing the registry, for example:

© N. Harris Computer Corporation

`HKEY_CURRENT_USER\SOFTWARE\i2\iBase\UserSettings\Importer:CommandTimeout`

`HKEY_LOCAL_MACHINE\SOFTWARE\i2\iBase\SystemSettings\Importer:CommandTimeout`

If registry key overrides are not present, your system administrator can update the configuration tables in the iBase database.

To add the setting to the main configuration table, run the following SQL command:

```
INSERT INTO _Configuration_Def ([Item], [Encrypted]) VALUES
 ('Importer:CommandTimeout', 0)
INSERT INTO _Configuration_Text ([Item], [Data]) VALUES
 ('Importer:CommandTimeout', '123')
```

Then, in replicated environments, the following SQL command is run in the `_GlobalConfiguration` table:

```
INSERT INTO _GlobalConfiguration ([Item], [Data]) VALUES
 ('Importer:CommandTimeout', '123')
```

If you need further assistance, contact i2 Support.

Bulk updating OLE DB data sources in import specifications

When you import from OLE DB data sources, iBase must connect to the data source using providers. Data sources might become invalid and cannot be used in import specifications, however you can fix your connection definitions in a single operation.

About this task

When an import specification contains invalid OLE DB connections, you are unable to import data. Sometimes OLE DB providers are not valid because they are unavailable on your computer or you might have switched to using 64-bit iBase. Similarly, the OLE DB data source location could be a physical file that is not found on your computer, or an invalid link to an SQL server. This can be particularly problematic if you use multiple specifications because the issue cannot be corrected from within the import wizard. You should use the OLE DB Import Specification Updater function to update your connections.

Procedure

The updater feature is available from both iBase User and iBase Designer. You need to login, however you do not need to have a database open to use the wizard. You must have **create** and **edit** permissions for import specifications.

1. From iBase User, select **Tools > OLE DB Import Specification Updater**. From iBase Designer, select **Tools > Database Administration > OLE DB Import Specification Updater**. The updater wizard opens.
2. Click the checkboxes to choose locations to load data sources from. Possible locations are:
   - Current database
   - Templates from your own templates folder
   - Templates from the Shared Workgroup folder

   Note: System administrators can select Database and Workgroup Templates. Other users with **create** and **edit** permissions should select personal templates folder.
3. When you have selected at least one suitable location, click **Next**. The wizard displays the OLE DB Connection String Property selection window.

© N. Harris Computer Corporation

4. Select a value for **Provider**, that is the OLE DB driver. The option to only include Providers not installed on your computer is selected by default. Alternatively, select a value for **Data Source**, that is the file path or server name that you want to connect to, and click **Next**.
5. The wizard displays available providers, or data sources, depending on your previous selection. Click the provider or data source that you want to update. A list of import specifications that include that value, is displayed.
6. Click the individual checkboxes for any import specifications that you want to update, or use the bottom checkbox to select all. Click **Next**.
7. Specify the replacement value for the OLE DB Connection String Property. If you selected **Provider**, a list of the installed OLE DB Providers is displayed. If you selected **Data Source**, a text field is provided for you to enter the location of the replacement data source.
8. Select the import specifications to update. You can test the specifications to determine if they will connect successfully. Click either **Test Selected Import Specifications** or **Test All Import Specifications**. Use the **Stop** button should you want to end the test process before completion.
9. Make any necessary changes based on the test results, and click **Update** to move to the **Update Summary** page.
10. The updater replaces the property values in your selected import specifications. The status of the update is displayed next to each specification. When complete, iBase informs you how many specifications were successfully updated.

What to do next

If your work with the updater process is complete, click **Close** to exit the wizard. If you want to continue working with your current selections, or start new selections, click **Start Over** to return to the first page of the wizard. Your completed updates are saved, you will not undo any progress so far by choosing to return to the start.

Importing from Microsoft Excel worksheets

You can import data from Microsoft™ Excel spreadsheets, one worksheet at a time. If you would like to import multiple files, or multiple worksheets from the same file, create multiple import specifications.

About this task

When you import from Microsoft Excel files, you must:

- Format the worksheet consistently because iBase derives the formats for dates, times, and numbers from the worksheet.
- The first row of the worksheet must contain the field names.
- The last field of every row in a Microsoft Excel worksheet must be populated with data. If necessary, create a dummy column and populate every cell in that column - you can ignore this row later when you assign the source data to iBase fields.

Procedure

1. In the Source area on the first page of the import wizard, click **Microsoft Excel Spreadsheet** and then click **Next**.
2. Enter the path of the spreadsheet or browse for it. The worksheets are loaded alphabetically.
3. Select the worksheet that you would like to use from the **Sheet** list.

© N. Harris Computer Corporation

Note: If you change the names or the order of the worksheets, click **Refresh** to update the list of worksheets in the import wizard.

4. Select the worksheet and then click **Next** to continue.

Importing files from a folder

You can import all the documents, charts, images, or text files from a folder. The text files must all have the same number and type of fields.

Procedure

To import files from a folder:

1. In the Type area on the first page of the import wizard, select a suitable entity or link type. For example, to import documents you need an entity or link type that has one or more Document fields.
2. In the Source area on the first page of the import wizard, click **Folder Contents** and then click **Next**.
3. Enter the path of the folder or browse for it.
4. If required, turn on **Include subfolders** to import files from any subfolders.
5. Select the type of file to import:
   - **Importing documents and charts from a folder**

     Imports all documents and charts. This option does not import any image files:

     a. Select **Documents** and click **Next** to continue.

     b. Assign the **Document** row in the **Source Field** column to document field.
   - **Importing pictures from a folder**

     Imports all pictures:

     a. Select **Pictures** and click **Next** to continue.

     b. Assign the **Picture** row in the **Source Field** column to a picture field.
   - **Importing text files from a folder**

     Imports all the text files from a folder although the files must all have the same data structure. You might want to keep text files in a separate folder from other file types.

     a. Select **Text files** and click **Next** to continue.

Assigning fields to data

You can map data into field types in iBase records. To help with this mapping, you can use actions to manipulate your data into a consistent format.

Procedure

1. Click **Auto Assign**. This checks the source fields for matches with iBase field types for the specified record type, and automatically assigns any matches. If the matches made are all incorrect, you can use the clear option to remove all the assignments.
2. Using the drop-downs in the **iBase Field** column, and using the **Import Data** column as a guide, ensure that the Source fields are mapped to the correct iBase fields.

© N. Harris Computer Corporation

3. **Optional:** If the data needs manipulation before the mapping can be applied, select **Field Actions** to transform the source data into a different format. For more information, see Transforming source data.
4. Use the **Identifier** checkbox to ensure that any fields are highlighted that can be used to identify unique records.

Transforming source data

When you import data, you can transform values in source records before you assign them to fields. If the data is being exported from a third-party application, transforming the data in the import specification allows you to change the format of data to match the iBase schema.

About this task

There are two main types of transformation that you can make:

You can create a new value by combining data from other fields in the source record.

For example, if you want to combine a given name with a surname to create a full name. To do create the full name, create a new field, choose the fields to combine, and specify how they are combined. The data is copied from the source data, so you can continue to use the original data for other fields that you want to import.

You can apply actions to the data in individual fields to change the format or value.

For example, you could prefix the text in a specific field with fixed text, suffix it with text from another column, replace it with a value retrieved from a substitution file such as Female for F, or remove ordinal suffixes such as st, nd, rd, and th from (English-language) dates. These transformations are referred to as Actions, and any number of them can be applied to individual fields.

The actions that you specify are saved as part of the import specification.

Note: If you are importing from OLE-DB data sources and change the source table after completing the field assignments, you might invalidate them. In this instance, the Import Wizard prompts you to review the field assignments. You have the option to clear all field assignments, keep the current assignments, or revert to the source prior to the changes being made.

Procedure

To specify one or more actions for a field:

1. Click a column in the preview table to select that field.
2. In the **Available Actions** list, double-click an action to apply it.
3. Complete the settings for the action.
4. **Optional:** By default, the action is applied to every row of data in the source file. If you want to apply the action under certain conditions only, you can set a condition.

Available actions

If the data that you are trying to import does not match the field structure of your database, you can add steps to transform the structure in the import specification. The actions that you can select are described.

Add Prefix

Adds text or values immediately before the import data. For example, Area Code: 01234, Area Code: 01235, Area Code: 01236 where the prefix is "Area Code: ".

© N. Harris Computer Corporation

To add a prefix to the values imported for a field, enter the text or values in the Prefix to be added before text box together with any additional spaces. For example, adding a prefix to a field that imports a date of birth of an entity, a prefix "DOB: " might be specified to import a date of birth in the format DOB: 20/11/58.

Add Suffix

Adds text or values immediately after the import data. For example, ABC Inc., BCD Inc., CDE Inc where Inc. is the suffix.

To add a suffix to the values imported for a field, enter the text or values in the Suffix to be added after text box together with any additional spaces. For example, adding a suffix to a field that imports a currency amount, a suffix of "USD" might be specified to import a currency value in the format 12383478 USD.

Change Capitalization

Changes the case of text. You can change case to lowercase or uppercase, capitalize initial letters, or reverse the current capitalization.

Compress Repeated Characters

Replaces repeated characters with a single instance of the character. If you are creating the source text files automatically using products such as spreadsheets or databases, you might find that they insert unnecessary characters into the data file. For example, if a field is defined to be 30 characters long and contains text that is 18 characters long, the text is printed followed by 12 space characters.

To complete the settings for an action to compress unwanted, repeated characters, specify the characters that you want to compress. If you compress alphabetic characters, you can also turn on **Ignore case of characters** to compress both uppercase and lowercase instances of these characters.

Copy Value from Previous Row

Replaces the value in a column with the value from the same column in the row above it.

Extract Portion of Text

Extracts a specific part of the data. For example, you might have a log of telephone calls of which only some have an area code, and you want to extract all the telephone numbers and not the area codes.

To extract a specific portion of text or data from a field, complete both the First Character and Last Character settings:

- First Character - The number of the first character to use. Select either From start or From end to specify whether to count from the start or end of the numbered characters in the field.
- Last Character - The number of the last character to use. Select either From start or From end to specify whether to count from the start or end of the numbered characters in the field.

: For example, in a list of telephone numbers, you might want to use the last six characters of the telephone number. Telephone numbers might appear as follows in the imported file:

01234 567890

0234567890

441234 567890

567890

© N. Harris Computer Corporation

If you set the following conditions:

- First character: 6 (from end)
- Last character: 1 (from end)

: You create an entity with the identity: 567890.

Find and Replace Text

Replaces text with different text. For example, you might remove ordinal suffixes (st, nd, rd, and th) from English-language dates by adding an action for each suffix that replaces it with an empty string.

Note: You might need to add extra actions to prevent (for example) the action for st also removing the last two letters of August, perhaps by replacing August with 08 first.

To complete the settings:

1. Type the text string in the **Match value** box for the text that you want to replace.
2. Turn on **Ignore case** if the value you want to replace appears in both lowercase and uppercase.
3. Type the text string that you want to use as the replacement text in the **Replace with** box.

Prefix with Another Column

Adds the values from another field immediately before the values imported from the selected field. For example, you can add a Date field to a Time field to import a combined date and time.

To complete the settings for an action to Prefix with Another Column:

1. Select a field from the **Select column to add** list.
2. Select the separator that you require to be inserted between the joined data columns. If you require a specific character or set of characters, for example " at " for [Date] at [Time], select Other separator and enter the characters and spaces in the adjacent box.

Remove Characters

Removes unwanted characters such as spaces, tabs, or specific characters. For example, the characters () can be removed from (863) 555 0140.

To complete the settings for an action to remove unwanted characters, turn on the checkboxes of the characters that you want to remove. These are tab characters, space characters, or other specific characters that you enter in the box provided.

Note: If you have turned on **Remove other characters listed below** and have entered alphabetic characters, you can also turn on **Ignore case of characters** to remove both uppercase and lowercase instances of these characters.

Remove Prefix

Removes unwanted text from the beginning of a field value. For example, MR(space) can be removed from MR SMITH.

To remove a prefix from the values imported for a field, type the text or values in the box provided together with any additional spaces.

Remove Suffix

Removes text from the end of a field value. For example, "Esq" can be removed from Andrew SMITH Esq.

© N. Harris Computer Corporation

To remove a suffix from the values imported for a field, type the text or values in the box provided together with any additional spaces.

Replace from Substitution File

Replaces a value in a field with values that are taken from a lookup table. The codes for marital status, for example, might be replaced with a term from a lookup table where S = Single, M = Married, D = Divorced.

To replace values in a field with values from a substitution file, a file must be created or available for selection.

Replace Value

Replaces a value with another value that you specify. For example, you want to replace the detailed confidential information displayed in a field with the value CONFIDENTIAL.

To complete the settings to create a Replace Value action:

1. Click the arrow on the **Match value** list, and select the data column value.
2. Turn on **Ignore case** if the value you want to replace appears in both lowercase and uppercase.
3. Enter the value that you want to use as the replacement value in the **Replace with** box.

Split Text

Divides the content of a column into two parts at the first occurrence of the split character. For example, a Full Name field "Andrew Smith" might be split into two fields: Name1 "Andrew" and Name2 "Smith".

To split the text that appears within a column:

1. Select the split character that you would like to use to divide the column. By default this uses a space, if you require a specific character, for example " | ", select Other and type the characters and spaces in the adjacent box.
2. Choose which side of the split you would like to use.

Suffix with Another Column

Adds the value from another field immediately after the value imported from the selected field. For example, you can add a Time field to a Date field to import a combined date and time.

To complete the settings for an action to Suffix with Another Column:

1. Select a field from the **Select column to add** list.
2. Select the separator that you require to be inserted between the joined data columns. If you require a specific character or set of characters, for example " at " for [Date] at [Time], select Other separator and type the characters and spaces in the adjacent box.

Trim Characters

Trim unwanted characters from the beginning and end of a field, such as spaces, tabs or specific characters. For example the character " can be removed from " Jelico's Restaurant".

To complete the settings for an action to trim unwanted characters, turn on the checkboxes of the characters that you want to trim. These are tab characters, space characters, or other specific characters that you enter in the box provided.

© N. Harris Computer Corporation

Note: If you have turned on **Trim other characters that are listed below** and have entered alphabetic characters in the box provided, you can also turn on **Ignore case of characters** to trim both uppercase and lowercase instances of these characters.

Applying conditions to actions

Transforming actions can be used to change the source data before it is added to field values. You can use conditions to apply one or more actions to a field, such as add a text prefix under specific circumstances.

Procedure

For each condition you want to apply to the field:

1. Select from the **Condition** list:

Option	Description
Contains	Where the field contains the exact portion of text, which can be at the start, middle or end of the field.
Does not contain	Where the field does not contain the exact portion of text.
Does not end with	When the field does not end with the entered value text.
Does not start with	When the field does not start with the entered value text.
Ends with	Where the field ends with the entered value text.
Is blank	Where the field does not contain any text.
Is not blank	Where the field contains text.
Length equal to	When the field width is equal to a specific number of characters.
Length greater than	When the field width is greater than a specific number of characters.
Length less than	When the field width is less than a specific number of characters.
Regular expression	When the data meets a general pattern set by the specified expression. For example, the expression [0-9] will match on 1, 2, 3, 4 up to 9.
Starts with	When the data starts with the entered value text.

© N. Harris Computer Corporation

2. Click in the **Value** column and enter the condition text.
3. If required turn on **Stop processing subsequent actions when conditions are met**. If selected, the processing of a particular action is stopped when the condition is met. The import wizard will then continue by processing the other actions in the **Actions Applied** list.

Information on formats and field types

The following considerations are specific to certain types.

Date, time, and number formats

You can control how dates, times, and numbers are formatted when you export data, or how they are interpreted when you import data. The default formats are determined by the Windows Regional Settings as set on your computer, if the default formats are unsuitable then you can configure the formats to use.

Note: The text qualifier, record, field, time, date, and number delimiters must generally be set to different characters.

Date formats

Date Order

The order in which to arrange the date parts. For example, Day, Month, Year, or Month, Day, Year.

Date Delimiter

The character that separates the date parts (day, month, year).

Month

A month can be formatted or interpreted:

- As a number. For example, months are specified by their number; January = 01, February = 02.
- Using an abbreviated name. For example: January = JAN, February = FEB.

Note: The month format always defaults to MM (such as 01) even if the format in the Regional Settings is set to MMM.

Four-digit years

A year can be formatted or interpreted as either a four-digit or a two-digit number. When you are importing:

- Turn off this option for two digits in the source data to be converted to four. The conversion is as determined by the Windows regional Date settings (specifically, the ‘When a two-digit year is entered, interpret as a year between’ setting). The digits that are used for conversion are the first (or only) two of the year in the source.
- Turn on, for all year digits in the source to be used, if present. If not present, then a conversion is made from the digits that are present, as detailed above for 'off'.

When you are exporting:

- Turn on for all four-year digits to be copied to the export file.
- Turn off for only the last two of the year digits to be copied to the export file.

Time formats

You can specify the format for times by changing the character (delimiter) that separates hours, minutes, and seconds. Valid characters include: colons (:), spaces, and periods (.). Do not use

© N. Harris Computer Corporation

alphanumeric characters. The order of the time parts (hours, minutes, seconds) is determined by your Windows Regional Settings.

When importing, this determines how times are to be interpreted in the source file; and when exporting, this determines how to specify times in the destination file.

Attention: You should use the **Auto Detect Formats** option carefully, and verify your final results. This is because the AM/PM setting is not always detected correctly when the regional settings for AM/PM are set to blank. Also, there can be differences in how operating systems process date and time formats, if you are using a version of iBase later than 9.2.1. For example, a date of 31st January 1933, expressed as 33/01/31, might see the month incorrectly interpreted as *33* rather than *01*. The wrong year might also be generated in your import because of how two-digit years are handled.

Number formats

You can specify the format for numbers by changing the character that is used as the decimal point (delimiter).

When importing, this determines which character is interpreted as being the decimal point in the source file. When exporting, this determines the character to be used as the decimal point in the destination file.

Note: In some regional settings, such as Russian, it is usual to use a non-break space character (<NBS>) as a delimiter for the thousands in a number format. This prevents the number from being broken up by a word wrap. When using the Import Wizard, the Date/Time & Number Formats derive the <NBS> character from the default locale settings of the Digit Grouping of the Regional and Language Options dialog.

If you change this character, iBase will not allow you to type it in (using the key combination Ctrl+Shift +Space). To re-enter this character, copy it from another application such as Microsoft Word, or from the Regional and Language Options dialog and then paste it into the Thousands separator box.

Time zones

If you want to import records that containing time zones from external data sources, or search for time zones, then you need to represent each time zone by the appropriate code. For example, in the import file, the time zone (GMT+00:00) Greenwich Mean Time: Edinburgh, London must be represented by 32.

If an entity or link type has at least one time zone field, then a Default Time Zone field is displayed on the final page of the import wizard.

When you export data the time zone is also represented by a code.

You can include a value for the field in the imported data or provide a default.

Time zones sorted by iBase code

The time zones in the following table are sorted in numerical order by their iBase code.

Code	Time difference	Name
1	(GMT+00:00)	Coordinated Universal Time
2	(GMT+04:30)	Kabul
3	(GMT-09:00)	Alaska
4	(GMT+03:00)	Kuwait, Riyadh

© N. Harris Computer Corporation

Code	Time difference	Name
5	(GMT+04:00)	Abu Dhabi, Muscat
6	(GMT+03:00)	Baghdad
7	(GMT-04:00)	Atlantic Time (Canada)
8	(GMT+09:30)	Darwin
9	(GMT+10:00)	Canberra, Melbourne, Sydney
10	(GMT-01:00)	Azores
11	(GMT-06:00)	Saskatchewan
12	(GMT-01:00)	Cape Verde Is.
13	(GMT+04:00)	Baku, Yerevan
14	(GMT+09:30)	Adelaide
15	(GMT-06:00)	Central America
16	(GMT+06:00)	Astana, Dhaka
17	(GMT+01:00)	Belgrade, Bratislava, Budapest, Ljubljana, Prague
18	(GMT+01:00)	Sarajevo, Skopje, Sofija, Warsaw, Zagreb
19	(GMT+11:00)	Magadan, Solomon Islands, New Caledonia
20	(GMT-06:00)	Central Time (North America)
21	(GMT+08:00)	Beijing, Chongqing, Urumqi
22	(GMT-12:00)	Eniwetok, Kwajalein
23	(GMT+03:00)	Nairobi
24	(GMT+10:00)	Brisbane
26	(GMT-03:00)	Brasilia
27	(GMT-05:00)	Eastern Time (North America)

© N. Harris Computer Corporation

Code	Time difference	Name
28	(GMT+02:00)	Cairo
29	(GMT+05:00)	Ekaterinburg
30	(GMT+12:00)	Fiji, Kamchatka, Marshall Is.
31	(GMT+02:00)	Helsinki, Riga, Tallinn, Vilnius
32	(GMT+00:00)	Greenwich Mean Time: Edinburgh, London
33	(GMT-03:00)	Greenland
34	(GMT+00:00)	Casablanca, Monrovia
35	(GMT+02:00)	Athens, Istanbul, Bucharest, Minsk
36	(GMT-10:00)	Hawaii
37	(GMT+05:30)	Calcutta, Chennai, Mumbai, New Delhi
38	(GMT+03:30)	Tehran
39	(GMT+02:00)	Jerusalem
40	(GMT+09:00)	Seoul
41	(GMT-06:00)	Guadalajara, Mexico City, Monterrey
42	(GMT-02:00)	Mid-Atlantic
43	(GMT-07:00)	Mountain Time (North America)
44	(GMT+06:30)	Yangon, Rangoon
45	(GMT+06:00)	Novosibirsk
46	(GMT+05:45)	Kathmandu
47	(GMT+12:00)	Auckland, Wellington
48	(GMT-03:30)	Newfoundland
49	(GMT+08:00)	Irkutsk, Ulaan Bataar

© N. Harris Computer Corporation

Code	Time difference	Name
50	(GMT+07:00)	Krasnoyarsk
51	(GMT-04:00)	Santiago
52	(GMT-08:00)	Pacific Time (North America)
53	(GMT+01:00)	Brussels, Copenhagen, Madrid, Paris
54	(GMT+03:00)	Moscow, St. Petersburg, Volgograd
55	(GMT-03:00)	Buenos Aires
56	(GMT-05:00)	Bogota, Lima, Quito
57	(GMT-04:00)	Caracas, La Paz
58	(GMT-11:00)	Midway Island, Samoa
59	(GMT+07:00)	Bangkok, Hanoi, Jakarta
60	(GMT+08:00)	Kuala Lumpur, Singapore
61	(GMT+02:00)	Harare, Pretoria
62	(GMT+06:00)	Sri Jayawardenepura
63	(GMT+08:00)	Taipei
64	(GMT+10:00)	Hobart
65	(GMT+09:00)	Osaka, Sapporo, Tokyo
66	(GMT+13:00)	Nuku'alofa
67	(GMT-05:00)	Indiana (East)
68	(GMT-07:00)	Arizona
69	(GMT+10:00)	Vladivostok
70	(GMT+08:00)	Perth
71	(GMT+01:00)	West Central Africa

© N. Harris Computer Corporation

Code	Time difference	Name
72	(GMT+01:00)	Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna
73	(GMT+05:00)	Islamabad, Karachi
74	(GMT+10:00)	Guam, Port Moresby
75	(GMT+09:00)	Yakutsk
76	(GMT-12:00)	Yankee (military)
77	(GMT-11:00)	X-ray (military)
78	(GMT-10:00)	Whiskey (military)
79	(GMT-09:00)	Victor (military)
80	(GMT-08:00)	Uniform (military)
81	(GMT-07:00)	Tango (military)
82	(GMT-06:00)	Sierra (military)
83	(GMT-05:00)	Romeo (military)
84	(GMT-04:00)	Quebec (military)
85	(GMT-03:00)	Papa (military)
86	(GMT-02:00)	Oscar (military)
87	(GMT-01:00)	November (military)
88	(GMT+00:00)	Zulu (military)
89	(GMT+01:00)	Alpha (military)
90	(GMT+02:00)	Bravo (military)
91	(GMT+03:00)	Charlie (military)
92	(GMT+04:00)	Delta (military)
93	(GMT+05:00)	Echo (military)
94	(GMT+06:00)	Foxtrot (military)

© N. Harris Computer Corporation

Code	Time difference	Name
95	(GMT+07:00)	Golf (military)
96	(GMT+08:00)	Hotel (military)
97	(GMT+09:00)	India (military)
98	(GMT+10:00)	Kilo (military)
99	(GMT+11:00)	Lima (military)
100	(GMT+12:00)	Mike (military)
101	(GMT+02:00)	Amman
102	(GMT+00:00)	Dublin
103	(GMT+00:00)	Lisbon
104	(GMT-06:00)	Galapagos
105	(GMT-06:00)	Easter Island (Chile)
106	(GMT-05:00)	Cuba
107	(GMT-04:00)	Falkland Islands
108	(GMT-04:00)	Paraguay
110	(GMT-03:00)	Eastern Brazil
111	(GMT-03:00)	Uruguay
112	(GMT-03:00)	French Guiana
113	(GMT-02:00)	Fernando de Noronha (Brazil)
114	(GMT+05:00)	Tashkent
115	(GMT+01:00)	Windhoek
116	(GMT-01:00)	Tunis
117	(GMT-04:00)	Tbilisi
118	(GMT-02:00)	Beirut
119	(GMT-06:00)	Almaty

© N. Harris Computer Corporation

Code	Time difference	Name
120	(GMT+07:00)	Chihuahua, La Paz, Mazatlan
121	(GMT-08:00)	Tijuana, Baja California
122	(GMT+04:00)	Manaus

Pictures and documents

A data source such as a text file can include references to picture and document files. You can also import pictures and documents directly.

You cannot use the iBase xml schema to export pictures and documents to XML to export to XML, use the Microsoft™ Rowset xml schema.

Importing pictures

When you are importing from text files, you can import referenced pictures if the picture file is accessible and you assign the file to a Picture field. For example, in the User Guide database, the Person entity has a suitable field.

When you are specifying picture files in a data file:

- If the picture file is in the same folder as the data file, use a name with the appropriate extension. For example, you can use: Hoffmann.bmp.
- If the picture file is stored in another location, use the full path. For example, assuming that the drives, folders, and files exist, you can use: `C:\photographs\smith.bmp` or `\\server-name \folder\CARTER.BMP`

You can import files in which only some of the records have a picture. Records without a picture need to include a pair of field delimiters with nothing in between to indicate the empty value.

Importing documents

You can import document files in a similar way to pictures by placing a reference to the file names in the data source and assigning the source field to an iBase field of type Document.

Exporting pictures

When you export data that contains pictures, each picture file is saved in the same folder as the export file, and in its original file format. The files take the same name as the export file.

Exporting documents

When you export data that contains documents, each document is saved in the same folder as the export file, in its original file format, with the same name as the export file.

Note: If your document or picture files are locked, they are not saved as part of the export process. The export completes with errors. Details of any files in use by another process can be found in the export file.

© N. Harris Computer Corporation

Checking for existing records

When data is added to iBase as a part of an import, it might match existing records. As a part of the import specification, you can choose how potential matches are handled.

About this task

Depending on whether you are importing entities or links, the options that you are provided have the same basic choices, but if you are importing links, you must make decisions not only for the link record, but also for the entities that are at either end of that link.

Procedure

For each type of record that you are creating, to check for matching records:

1. Select the option to check for matching records of this type using identifiers, and in the case of links, link ends.
2. Choose the method of handling potential matches:

Option	Description
Update it	Update the matching record with the supplied information.
Don't update it	Discard data that matches existing records in the database.
Confirm action	Each time matching data is identified, choose whether to update the record.

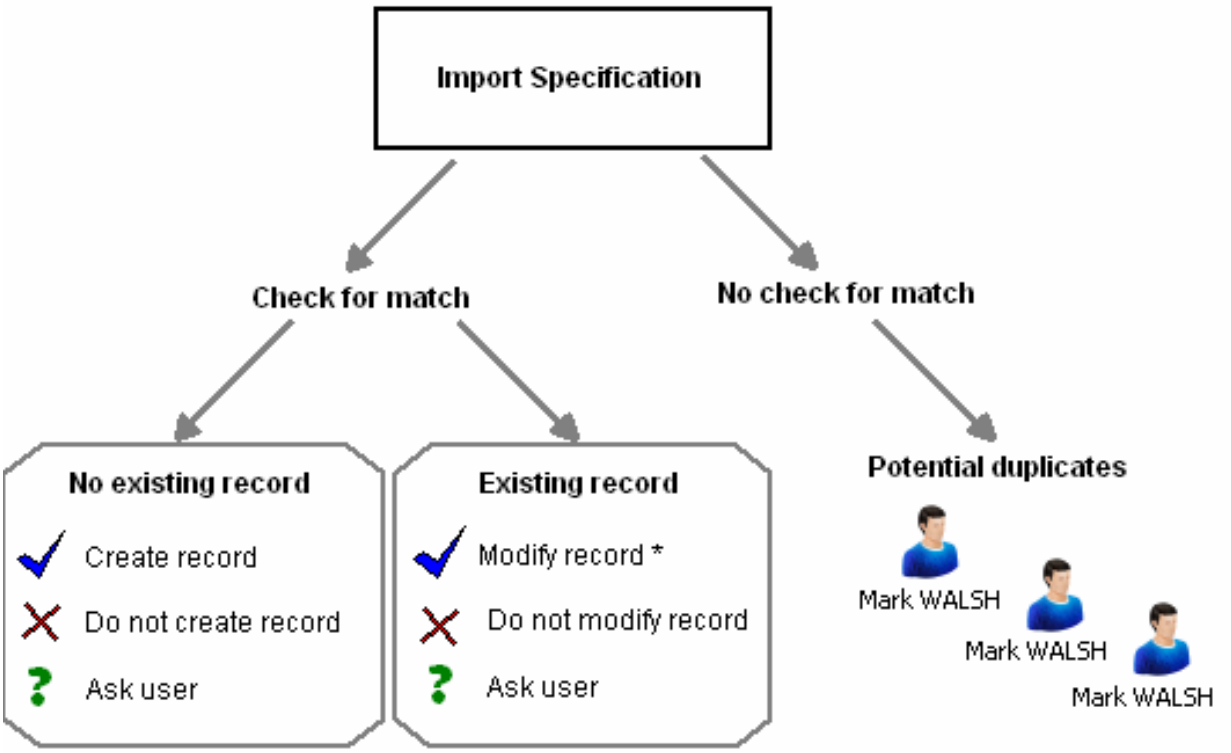
3. **Optional:** You can use the **Treat identifiers without values as matches** option to determine whether or not fields marked as identifiers should count as matches, if they do not contain values. The option is checked by default which means that identifiers that do not contain values are treated as a match.

   It is important to understand how this option can change matching behavior. In this example you are working with Person records and have First Name, Last Name and Date of Birth marked as identifers. You set matches to 'update if found', 'create if not found', and leave the identifiers without values option checked. When you run the import, records with matching identifiers are updated, even if the Date of Birth field, for example, is blank. If you then uncheck the identifiers without values option and run the import again, a new record is created even though the first and last names are a match. The importer regards the blank value in Date of Birth as not matching, and does not update the record.

© N. Harris Computer Corporation

Matching entities in importing

When you are importing entities, allow for the possibility that there might be a matching record already in the database. Similarly, the data source itself might contain repeated records. You can detect and handle these situations to suit different data and working practices.



Note: The result of using **Modify record** depends on how you choose to handle blank fields in the source data by turning on or off the **Do not update existing field values with blank values** option.

Matching entities have identical values in the identifier fields of the data source and an entity record already in the database.

Identifiers are suggested for each import specification, as either a combination of fields or a single field that is unique. The suggestion is based on the discriminator fields, which you can see in the database design report; you can choose different identifiers if appropriate. All identifier fields must match for there to be a matching record and you must be importing a field to use it as an identifier.

After a check has established whether or not there is a matching entity, there are two groups of options.

Note: In case-controlled databases, data can only be imported into the current case. Entities will only be tested for a match with records in the current case. Identical records may exist in other cases.

Handling a matching entity

Once a matching entity has been identified, the options for using the information in the data source are:

© N. Harris Computer Corporation

- Always update the existing entity record - You will usually want to do this if you know that the import data is reliable and up-to-date.

  Note: If you want to change the value of a field, that field must not be an identifier when you make the import. To avoid overwriting existing data with empty/blank fields from the source data, you can turn on Do not update existing field values with blank values when setting up the import specification.
- Always leave the record unchanged, that is, ignore the data source - You may want to do this if you think that the import data is older, less complete, or less reliable than information already in the database record.
- At the time of importing each record, ask the user which option to use

Handling a non-matching entity

If an existing entity has not been found, you have a different set of options for using the information in the data source:

- Always create a new entity record, known not to be a duplicate - Do this if you know that the import data is reliable and up to date.
- Never create a new record, that is, ignore the data source - Do this if you think that the import data is old, unreliable, or incomplete; or if working practices restrict how to create a new entity record.
- At the time of importing each record, ask the user which option to use

Creating a potential duplicate entity

In some situations, you may not be able to specify the best option beforehand or decide the best action at the time of importing data. If this is the case, you can choose not to check identifiers, which means that all records in the data source are used to create new records and may therefore produce matching or duplicate records. The import process will not inform you if there are duplicates, so you should make the detection and handling of duplicates part of your work practices following such data import.

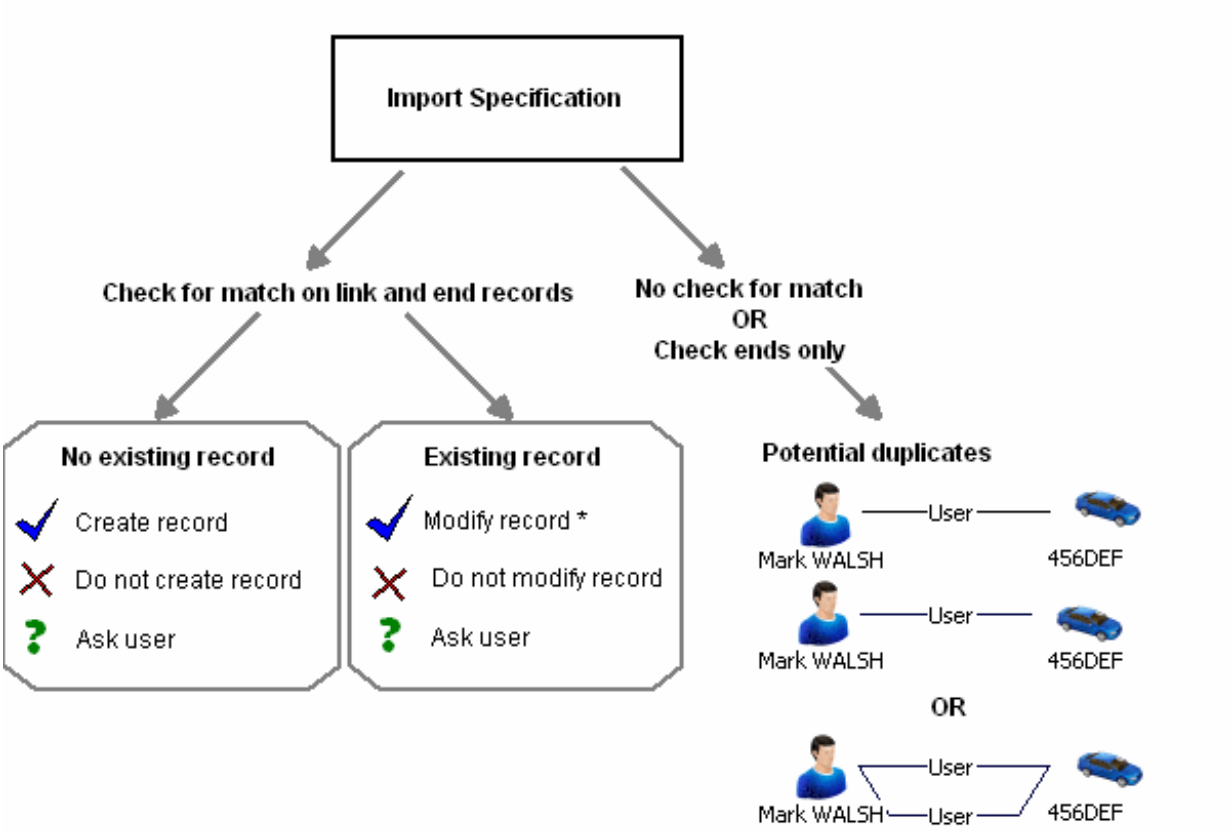
For example, you can check for duplicates using iBase tools such as the Matching Records or Duplicate Records Checker, then decide to merge any duplicate records or create links between the duplicates.

Matching links in importing

A check for a matching link uses information about the link and the two link end entities. You can check all these records to avoid any duplicates or check none, with results similar to checking an entity.

© N. Harris Computer Corporation

You can also make a partial check, on the end entities only, allowing potential duplicate links but not duplicate entities.



Note: The result of using **Modify record** depends on how you choose to handle blank fields in the source data by turning on or off the **Do not update existing field values with blank values** option.

Identifying a matching link

More conditions are checked when you match link records than entity records. For a matching link to be identified all these conditions must be satisfied:

- The link type must be the same.
- Any identifiers on the link must have identical values, as for entities.
- Both end entities must have matching records (in the database).
- The direction of the link must be the same.
- The strength of the link must be the same.

After a check establishes whether there is a matching link, there are two groups of options.

If an existing link is identified, the options are:

- Always update the existing link record.

© N. Harris Computer Corporation

Note: To avoid overwriting data with empty fields from the source data, you can turn on **Do not update existing field values with blank values** when setting up the import specification.

- Always leave the record unchanged, that is, ignore the data source.
- Ask the user which option to use, at the time of importing each record.

If an existing link is not found, you have a different set of options for using the information in the data source:

- Always create a new link record, which is known not to be a duplicate.
- Never create a new record, that is, ignore the data source.
- Ask the user which option to use, at the time of importing each record.
- Create the link only if the end entities exist in the database.

You might want to use this option when you know that the link information in the data source is reliable but the information about the end entities is unreliable or incomplete. (You implement this option by creating a specification that does not create end entities; without ends, a link cannot be created.)

You can choose to allow duplicate links in the same way as you can for entities, by turning off the check for identifiers. If you turn off the check for the link, you can choose separately whether to check the identifiers of the end entities. With links, often you require duplicates, for example it might be important to record each instance of an identical relationship link separately.

After any import of this type, detect and handle any duplicate records.

Completing and verifying the import specification

When you have completed the process of mapping source information to iBase field types, you can complete the import specification and verify the results before running the import.

Procedure

1. Check the **Session Defaults**.

   For specific types of iBase field, you can specify default values to be added if the source data does not contain a value. You should check that any existing defaults apply to your current import, and amend if necessary.
2. Select the overall import options:

Option	Description
Create an import set	You can create a set that contains records that were affected by the import.
Write the import statistics summary to a file	You can store the results of the import as a text file, and select either to overwrite any existing file, or produce a file identified using a timestamp.
Write all the records with errors to a file	You can create a report of any errors with the import, and select either to overwrite any existing file of the specified name, or produce a file identified using a timestamp.

© N. Harris Computer Corporation

Option	Description
Automatically stop the import if this number of errors occur	You can choose to stop the import if the number of errors hits a specified threshold.

Note: To analyze the incoming data and check for errors, you can instruct iBase to generate import logs by selecting the checkboxes for *both* "Write" options at this stage.

3. **Optional:** Click **Verify** to check the output of the import is as you expect, before making the changes in the database. This lists the time taken from **Start** to **Finish**, and the total **Duration** of the verification check. In addition, it displays the number of identified records of each type identified in the source data, along with the status of that information:
   - **Not found** - unique data that does not match existing records in the database.
   - **Different** - data that has matched records in the database using the identifiers provided, but has other values that differ from the existing records.
   - **Same** - data that has matched records in the database using the identifiers provided, and matches the other values present in the database.
   - **Errors** - any errors that were flagged when importing this data type.
4. Click **Run** to import the data.

Importing charts

You can import batches of Analyst's Notebook charts that can be viewed and searched like other iBase records.

In addition, if you have charts that contain information you would like to store within your iBase database, you can extract this data.

Loading charts

You can load charts into iBase if you have an entity type that has a document type field.

You load the charts by importing them individually using a standard import, or in batches from a folder. How to import all the charts in a folder, and its subfolders, is explained in Importing Files From a Folder.

Viewing charts

You can view charts without having to open Analyst's Notebook. Using the Chart Viewer, you can zoom in and out, find text, and examine chart item properties (such as cards), all as if you were working in Analyst's Notebook.

Finding text on charts

You can use Chart Viewer to find text on a specific chart.

In an SQL Server database, Search 360 searches for text on any chart, including data records and cards.

Extracting data into iBase

You can use the Chart Item Extractor to extract data from an Analyst's Notebook chart and store it in iBase. You can extract Analyst's Notebook data into iBase directly from Analyst's Notebook when connected to your iBase database or from iBase by opening the Chart Item Extractor and browsing to the required chart.

© N. Harris Computer Corporation

Importing errors

When data is imported into iBase, errors can occur. To ensure that the data is correctly imported, you can create a file of records that have experienced issues and use this information to correct problems in the source data.

About this task

Typical errors are:

- Invalid characters (it is not an error if the source file has a blank value).
- Incorrect number of fields in the record (perhaps a text qualifier or delimiter is missing).
- Data is too long for the field.

An error in any field prevents the iBase record from being created or updated. An error file is only generated if you chose to turn on **Write all records with errors to a file** on the last page of the import wizard. Whether the existing error file is overwritten, and how it is named, depends on options that are specified on the last page of the import wizard.

Note: In the import wizard, you can double-click an error message to display it. Only the first 10,000 errors are reported.

The error file contains one row for each source record that is in error, with tabbed columns of field values.

The first row contains the assigned iBase field names as headings for the columns, so that when reimporting from the error file (after you fix the problem), you can use **Auto Assign**.

In subsequent rows:

- The first column contains the source record number (for example its line number in the source text file; skip this field on reimporting).
- The remaining columns contain the source record field values (but only for fields that are assigned to iBase fields).

In addition to the file with the *. txt extension, there is a file with a further *.errl extension. This log file contains one line per source record that is in error, detailing the error.

Procedure

To correct errors and reimport the data:

1. Open the error file in a text editor.
2. Edit the text to correct the problem that is identified by the error messages displayed at the end of the import, and then save your changes.
3. In iBase, load the original import specification and then click **Next**.
4. In the **File name** box, enter the edited error file as the source data for the import and click **Next** .
5. Assign the source data to the iBase fields by clicking **Auto Assign**. Click **First** and **Next** to review the edited records in the source data.
6. Complete the import in the usual way.

© N. Harris Computer Corporation

Exporting and importing externally edited iBase data

You can export data from iBase for editing in a different application, and then reimport it. There are two ways of doing this, either by matching the information on import, or by exporting the record IDs, and by using the ID for record matching during the import.

About this task

Before you export the data, you need to display the record IDs for the entity or link types that you want to edit. This is done by a suitably qualified user in iBase Designer, by adding the record ID field to the entity or link types.

Once this has been done, you can export the records that you want to edit in the usual way. You should export the data to a text file.

When you edit the data:

- Do not change the values in the record ID field.
- Do not add new records - only existing records will be updated using this method.
- Take care not to edit or delete characters such as text qualifiers and field delimiters.

Procedure

To reimport iBase data:

1. In Step 1 of the import wizard, in the Source area, turn on **The import source contains 'Record ID's that originated from this database**. This option is unavailable if the entity type does not have visible record IDs. In this case, you will need to match the records carefully when you import them.
2. Click **Next** to display the Step 2 page.
3. Enter the file name. You do not need to set any of other options on this page. Click **Next**.
4. Click **Auto Assign**, to automatically assign fields in the source data to the iBase fields.
5. Check that the fields are assigned correctly. You must map the iBase Record ID type field to the import source field that contains the record ID.
6. You do not need to specify any identifier because iBase uses the record ID as the identifier. Click **Next**.
7. You do not need to decide whether to check for matching records because iBase will automatically check against the record ID and only update existing records. Any new record IDs will not be accepted; new records will not be created. Click **OK** to continue.
8. Complete the import in the usual way.

Managing import and export specifications

All the available import or export specifications that you can access, can be listed. The list displays key information about the specification such as the creation details and the access levels.

Procedure

1. In the database explorer window, right-click **Import Specifications** or **Export Specifications**, and select **List**.

   Tip: To add a specification, click **New**. Either the import wizard or the export wizard is displayed.
2. To administer the specifications for your database, right-click a specification and then select one of the following options.

© N. Harris Computer Corporation

Option	Description
Open	To start editing the specification, or to import or export data.
Rename	To change the name of the specification.
Save As	To save a copy of the selected specification under a new name. The name must be unique not just to the category folder but to the whole database.
Delete	To delete one or more selected specifications.
Categorize	To move one or more selected specifications, which might be in different categories, to another category.
Properties	Show system type properties for specification, such as its creation date and user. Only available when a single specification is selected.

Batch import or export

If you receive data from other databases, or you prepare data for others, you can set up batch specifications, that can run whenever new data is available. A batch specification is a list of preexisting import or export specifications that can be run with no further intervention required.

Procedure

To create a batch import specification:

1. Select **File** > **Data** and then either **Import Batch** or **Export Batch**.
2. Choose the specifications to include in the batch. The specifications are run sequentially in the same order as the list. To change the position of a specification in the Batch list, click the specification and then click the up and down arrows.
3. **Optional:** If you want the batch import or export to continue even if there is a problem with one of the specifications, turn on **Continue running** .
4. Click **Save** or **Save As** to save the batch specification. You might be prompted for a category and access control settings.
5. Click **Run** to run the batch specification.

© N. Harris Computer Corporation

Setting up the import and export specifications

The export and import specifications that can be run by the Import Batch or Export Batch specification are set up in the usual way. The only difference is in how you record any errors that might occur during imports and how you log the statistics at the end of each import.

About this task

If you would like to view details of any errors that occur during an batch run, you need to save individual error files. As a default, error files are overwritten when the batch import runs the next import specification. You do this on the last page of the import wizard by selecting **Append a timestamp to the file names each time the import is run**. The same applies if you want to save the statistics for each import.

Procedure

If you want a log of what went wrong with an import:

1. Turn on **Write all records with errors to a file** on the last page of the import wizard.
2. Enter the name of the error file, which will be saved in the same directory as the database. Ideally use a name that is unique to each import specification.
3. To save it in a different directory, browse for the folder in which to save the file and click **Save**. By default the file is overwritten each time you run the import. If you want to keep the error files, select **Append a timestamp to the file names each time the import is run,** also on the last page of the Import Wizard.

Running a batch import or export

A batch import and export can either be triggered to run manually, or by using the iBase Scheduler. If you are importing or exporting large volumes of data, you might want to run batch specifications when your system is less busy.

About this task

To schedule a batch import or export, you can use iBase Scheduler, which is available from the **All apps** (or **All Programs**) group of the Windows Start menu: **i2 iBase** > **Tools** > **iBase Scheduler Configuration**.

If a batch import fails with the message `Invalid procedure call or argument`, your account might not have the privileges needed to run the Scheduler service. In Windows Services, change the i2 iBase Scheduler service to **Log On as Administrator**. If you are still unable to run the service, ask your system administrator to grant the necessary user privileges.

Note: This message is associated with batch imports triggered by using Scheduler. Manually running batch specifications does not fail in the same way.

Procedure

To manually run an existing batch specification:

1. In the Database Explorer view, right-click either **Import Batch Specifications** or **Export Batch Specifications**.
2. From the menu, select **List**.
3. Load the required batch specification by double-clicking.

© N. Harris Computer Corporation

4. Review the details of the batch specification.
5. When you are ready to run the batch specification, click **Run**.

Managing batch specifications

Depending on whether you are importing or exporting, you can list all the batch specifications in the current category. If the All folder is selected, this is a complete list of all the batch specifications in the database.

Procedure

1. In the database explorer window, right-click **Import Batch Specifications** or **Export Batch Specifications**, and select **List**.

   Tip: To add a specification, click **New**.
2. To administer the specifications for your database, right-click a specification and then select one of the following options.

Option	Description
Open	To start editing the specification, or to import or export data.
Rename	To change the name of the specification.
Save As	To save a copy of the selected specification under a new name. The name must be unique not just to the category folder but to the whole database.
Delete	To delete one or more selected specifications.
Categorize	To move one or more selected specifications, which might be in different categories, to another category.
Properties	Show system type properties for specification, such as its creation date and user. Only available when a single specification is selected.

Listing and browsing

Many tasks require you to work through a list of entity or link records. You can select one or more records, right-click them and select further options from a menu.

There are two types of list that can be produced for an item type:

- Records, which displays all the fields.
- Browse, which displays selected fields.

You can also specify that a particular browse is run by default, by turning on **Autorun** on your browse definition. For example, a query might be set up to find all the records entered in the previous week.

© N. Harris Computer Corporation

The records that are listed depend on whether the database uses cases.

Listing records

You can list the records for a selected entity or link type so that you can review the data and take action as required. All the field values are shown, as well as the label for the entity or link type as defined in the default labeling scheme.

Procedure

1. In the left pane of the Database window, select an entity or link type.
2. Right-click and select **Records**.

   The records of that type are displayed. You can sort the records by the values in specific columns or change the order of the columns.
3. To change the number of records that are displayed, enter the number of records in the **Number of records to be displayed** box, and then click **Refresh**.

   Note: If there are a very large number of records, you may prefer to use a different method of viewing the records. For example, by using Find or Query in iBase.
4. To copy the information about records in the list, select the required records, and click **Copy to Clipboard**. You can then paste the records into another application, such as Microsoft Word or Microsoft Excel. Each record becomes a separate paragraph in a document or a separate row on a spreadsheet.

Browse records

You can generate a list of entity or link records in the database (or case) that shows just the record label and any specified fields. If required, you can save the field specification as a browse definition that you can use again later, or run automatically whenever you open the database.

About this task

Browse definitions are organized into categories (folders).

When you browse links, the browse definition contains three columns:

- Label for the link you are browsing.
- End1 label for any of the possible entities at one end of the link.
- End2 label for any of the possible entities at the other end of the link.

The position of the end entities (whether in the 'End 1 Label' and 'End 2 Label' column) does not indicate the direction of the link. To determine the direction, you need to open the link (right-click and select **Show**).

Procedure

1. Right-click on **Browse Definitions** and select **New**.

   Tip: You can open existing definitions by listing the browse definitions and double-clicking a browse definition to open and run it.
2. To change the browse definition, click **Configuration** to display the Configuration area.
3. From the Browse the item type list, select the type of entity or link.
4. Specify the source of the records:

© N. Harris Computer Corporation

- **Records** - browses the whole the database or case. You need to enter the maximum number of records to list.
- **Query** - browses the results of a query, which you need to select; you can only select from queries that output records of the specified type.
- **Set** - browses the records that are contained in a set. You need to select a set; you can only select from sets that contain records of the specified type.

5. Select the **Fields** to display, and the **Sorting** order of those fields.
6. Click **Browse** to list the current matching records.
7. **Optional:** To set up a browse definition to run automatically as soon as you open the database, select **Autorun**.
8. Click **Save** or **Save As** to save the browse definition. You might be prompted for a category and access control settings.

Viewing browse reports

You can generate a report on one or more records.

Procedure

You can either view this on screen or print it:

1. Run a browse by listing the browse definitions and double-clicking a browse definition, and select the required records.
2. Click the **Print** toolbar button.
3. Decide how you want to display the selected records. For example:

Option	Description
Header	The first label for the entity type will appear at the top of every page. To change this, edit the text in the Header box.
Footer	The name of the database will appear at the bottom of every page. To change this, edit the text in the Footer box.
Hide Record Label	The labels from the default labeling scheme will appear in the report (three labels if you are browsing link records). To hide these, turn on the Hide Record Label checkbox. You can only hide these if you have included one or more fields in the browse definition; see Setting up a browse definition for details.

4. Click **OK** to generate the report.

© N. Harris Computer Corporation

Autorunning browse definitions

You can specify one or more browse definitions that you want to automatically run whenever the database is opened. Each browse definition in the database is listed along with the entity or link type on which it is based.

Procedure

1. Select **Analysis** > **Browse** > **Autorun Definitions**.
2. Select the browse definitions you want to run.

Compare records

You can compare two or more records that are selected as a result of other operations, such as browsing, finding, and querying. Each record is identified by its label - the format of the label is defined in the current labeling scheme.

You can review the values of each record, and then open the selected record or use it in other operations, such as adding the entity or link to an Analyst's Notebook chart.

You can:

- Double-click to display a record directly or as a data sheet (depending on the default for the entity type).
- Copy the record details. For example, press `Ctrl+C` to copy the selected text, ready for pasting into another application.
- If you want to start other operations for one or more of the records, select one or more records, right-click, and select an action from the menu.

Viewing link charts

An iBase link chart can graphically represent the links and linked entities of an entity record, in a similar way to an Analyst's Notebook chart. The chart is centered on one entity (the one you opened the chart from), and shows this entity, along with its links and linked entities. There is only ever one 'level' of links shown.

You can rearrange the chart by clicking and dragging the objects, and you can simplify it by hiding specific entity and link types that you do not require.

Note: The labels on the chart are determined by the current labeling scheme.

Producing iBase Link Charts

1. Find the entity to use as the main subject of the chart.
2. Right-click the entity or link and select **iBase Link Chart**.
3. You can display a series of link charts by right-clicking on an entity or link in the chart and selecting **iBase Link Chart**.

Hiding and showing entity and link types

By default all the entity and link types are shown on the chart. To hide item types, click **Entity Types**or **Link Types** and then, in the list that is displayed, select required entity or link type.

Note: You cannot hide the entity from which you opened the chart.

© N. Harris Computer Corporation

Hiding and showing chart attributes

Any chart attributes that are set up for the entities may be shown on the chart underneath the entity labels. To hide or show chart attributes:

1. Select **Tools** > **Options** > **Charting**.
2. Turn on **Chart entity/link attributes**, and click **OK**.

   Note: The new setting takes effect only when you produce a new iBase link chart.

Starting other operations with the entities and links on the chart

You can work with the entities and links shown on the chart:

- To work with the main entity, click the toolbar buttons:
  - **Show** - View full details of the main entity.
  - **Links** - View a list of the links from the main entity, with full details of the links and the entities at the other end of the link.
  - **Matching Records** - Find out whether there are any other records in the database that share values in common with the main entity.
- To work with a link, in the chart area, click a link record, then right-click and select an action.
- To work with a link end entity, in the chart area, click a link end entity, then right-click and select an action.

Populating cards

Analyst's Notebook items can contain the details of an iBase record as a card. Adding iBase information to an Analyst's Notebook chart allows that information to be accessed by anyone who has access to the chart.

Procedure

To put the relevant entities onto a chart and populate the cards using data from the iBase records:

1. Add the required data to the Analyst's Notebook chart. For more information, see Searching iBase.
2. In Analyst's Notebook, select **Edit** > **Select All**.
3. On the **Selection** page of the **Data Sources Task Pane** , click **Populate Cards**.

   Tip: Alternatively, right-click on one of the selected chart items and from the shortcut menu, select **`database`** > **Populate Cards**, where `database` is the name of the iBase database.
4. Save the chart. You now have a saved chart, with populated cards.
5. To check that the cards are now populated:

   a. Right-click on one of the chart items and select **Edit Item Properties**.

   b. Select **Cards** > **Card1: `entity_name`** to display the contents of the first card.
6. When you are working on a chart where some chart items already have populated cards, you can choose to update existing cards with the new information or leave the cards unchanged.

   For each chart item with a populated card, you see a message similar to this:

   ```
   Record -'My record' has been modified since the chart was created.
   Do you still want to populate the card for this item?
   ```

   Click:

© N. Harris Computer Corporation

- **Yes** to update the existing card with new data from the iBase database
- **No** to leave the card unchanged
- **Cancel** to stop populating cards and leave all existing cards unchanged

Note: You will need to repopulate the card of any iBase chart item that you edit by using the **Show** command on the database menu as this action will remove all the card details.

Menus and record lists

In record lists and from the icons that are used in records, you can work on the selected records by right-clicking and selecting an action from the menu. The available commands depend on the record list and the current selection.

Show, Show With, Show Records

There are three ways of opening and viewing a record:

- **Show** - displays the selected record, either in the show record view, or the default data sheet.
- **Show With**
  - Select **Show With Show Record** to display the selected entity.
  - Select **Show With `datasheet name`** to use the indicated data sheet.
- **Show Records** - lists two or more records so that you can compare the selected records and browse their field values.

Show History

Displays the audit history so you can view the changes to the current records and find out who made those changes.

Note: Only available in databases that are set up to use this feature. See your system administrator.

Links

View the links and the link end entities for the selected record.

Matching Records

Finds any records that match the selected record, and then displays them. iBase searches for matching records using the fields that are defined as discriminators in the entity type.

Note: It might take a while to retrieve and display the records. To pause the retrieval, press the `Esc` key.

iBase Link Chart

Shows the links and link end entities for the record in an iBase Link Chart.

Add to Set

Adds the selected records to a new or existing set.

Set Membership

Lists the sets to which the record belongs.

Create Report

Sets up the report wizard to create a report on the selected record. It uses the default report definition for the entity type selected (if there is one).

© N. Harris Computer Corporation

Create Link

Create links between two or more selected records. Only available when you select multiple entity records:

- With two entities selected, the entities are placed at either end of the link.
- With more than two entities selected, the entities are placed at the End 2 of the link, leaving you to specify the End 1 entity.

Chart

Add the selected records to an Analyst's Notebook chart:

- **Chart > Add to Chart** - create chart items for the selected records.
- **Chart > Expand** - create chart items for the selected records and then expands them as specified in the Charting Settings dialog to add associated records to the chart.

Add Alert

Monitor activity on a record or changes to the results of a query by adding an alert.

Note: Only available in databases that have been set up to use this feature. See your system administrator.

Properties

Shows the properties for the selected record. These include the record's system properties such as its creation date, the name of the user who created it, and its record identifier (unique record number).

Sets

A set is a user-specified collection of records, possibly of different entity and link types. Entities and links can belong to more than one set, and you can use **Set Membership** to find out which sets a record belongs to.

You might want to use sets to:

- Keep entities that you found as the result of a query together as a group.
- Analyze only those records that were imported together.
- Define a group of records that you want to add to a chart.
- Group records together on a temporary basis; later you can analyze the set, or combine it with other sets to form new groupings.
- Analyze deleted records (only with Soft Delete).

You can create sets by:

- Adding records to an existing set or to a new one.
- Adding items on an Analyst's Notebook chart to an existing set or to a new one.
- Combining two sets to form a new one.

Sets are folder items that can be saved and stored in categories. A set does not store the *details* of the entities and links that it includes. When you select an entity or a link in a set, iBase displays its details from the current information in the database.

Notes:

© N. Harris Computer Corporation

- When you delete a *set*, you do not delete the records that it contains; you delete the definition of the set. iBase updates sets automatically when you delete *records*, but it does not delete an empty set when you delete the last record that it contains.
- You should delete sets that are no longer in use. An iBase database that contains a large number of sets can cause performance issues when using features such as Batch Edit.

Adding records to sets

There are various ways of adding records to a set. You can either use the commands described below, or you can drag records selected in other dialogs, such as the Browse or Records dialogs, or other Set dialogs, into a new or existing set.

Procedure

1. Select one or more records from a list.
2. Right-click and select **Add to Set**.
3. Choose whether to create a set or append the records to a set:

Option	Description
Create new set using the name	Enter the name of the set.
Append the records to the set	Select the set from the list.

4. Click **OK**.

Managing sets

The sets that are already in a database are organized into categories, and can be restricted using access control. You can manage the sets you have access to.

About this task

You can manage the sets in the database by using the options on the right-click menu, and you can also work with the records in an open set by selecting actions from the right-click menu.

Procedure

1. Select **Sets** > **All Sets** > **`Category`** to list the sets in the category.
2. Select a set in the list, right-click and select:

Option	Description
Open	Lists the records in the set.
Rename	Enter a new name for the set.
Save As	Save a copy of the set under a new name.
Delete	Delete the set.

© N. Harris Computer Corporation

Option	Description
Categorize	Move the set to other categories or set access control on them.
Properties	Show the properties of the set, such as its creation date, user, and access type.
Records	Lists the records in the set. There is a separate page for each entity and link type in the set. Click **Refresh** if you add or remove records in the set.

Note: You can delete or categorize multiple sets in the same action, by selecting multiple sets before right-clicking.

3. With an open set, you can:

Option	Description
Add	Add records to the set.
Remove	Remove records from the set
Save	Save your changes
Save As	Create a copy of the set with a different name

Tip: With two sets open, you can also drag records between sets.

Combining sets

You can combine the contents of two sets to isolate the records that you require. The resulting combination is placed into a new set.

About this task

For example you might have two sets, one called Prime Suspects and one set that comprises all the records that were added during a recent import session. You can easily discover which members of the Prime Suspects set were not added during the recent import.

Procedure

You can specify how the records in two sets are combined to form a new third set:

1. In a list of sets, select two sets, right-click and select **Combine**.
2. Select how to combine the sets from the list:

Option	Description
Include all records from both sets	Combine the full contents of the sets.

© N. Harris Computer Corporation

Option	Description
Include all records from set A only	Create a copy of set A.
Include all records from set B only	Create a copy of set B.
Include records from set A that are not also in set B	Create a subset of set A that excludes records that are also found in set B.
Include records from set B that are not also in set A	Create a subset of set B that excludes records that are also found in set A.
Include records from set A and set B that are not also in both	Create a set of the unique records in both sets.
Include records that are only in both set A and set B	Create a set of the common records between the sets.

Note: The Venn diagram illustration of the selected criterion helps you visualize the results.

3. Enter the set name in **Set C**.
4. Click **OK** to create the set. You might be prompted to specify a category and access control for the set.

Analyzing sets

You can find the links and entities that are common or uncommon to a group of sets. Looking for items in common provides information that might otherwise be overlooked.

About this task

If you have sets that contain information that is related, you can find records that match all of those criteria. For example, if you have sets of people:

- With connections to Boston
- Connected with the crime of arson
- Drivers of BMW cars

You can generate a list of all the records that share all three characteristics to provide suspects in an arson investigation.

Procedure

To analyze sets:

1. In a list of sets, select the sets to analyze, right-click and select **Analyze**.
2. In the Analysis Type area, select:
   - **Common Records**- find records that are in all the sets.
   - **Uncommon Records** - find records that are not in all the sets.
3. **Optional:** Change the list of sets to analyze by clicking **Add** and **Remove**.
4. Click **Results** to find the common or uncommon records.

© N. Harris Computer Corporation

5. To work on the records in the results list, select one or more records, then right-click and select an action.

Finding Out About Set Membership

Depending on the reason that sets have been created, set membership can provide additional information about a record. You can list the sets that a record is in that you have access to.

Procedure

1. Select a single record, for example in the Records or Browse dialog.
2. Right-click and select **Set Membership**.
3. To view the other contents of a set, right-click on the set and select **Records**.

Finding, checking, and searching records

You can find individual records, or groups of records, for adding to or editing the information in your database or to analyze data for a particular task. There are a number of ways that you can find records, and each is designed for a particular type of inquiry.

Feature	Description
Find	For locating one or more records of a specific type, such as person entities or phone call links, where you know some fragment of the information that you are seeking.
Queries	For discovering information about the contents of the database. Queries can be simple or complicated according to your needs. You can save queries for future use and in SQL Server databases, add alert definitions to them to notify you of changes to the records returned by the query.
Search 360	For a more flexible type of search across all entity and link types in the database. This type of search allows for variations, for example in spelling and word order. The results are scored depending on how closely the record matches your search criteria.

If you are working in a Microsoft™ Access database, then you can use these tools in place of Search 360:

Feature	Description
Word Search	For searching by exact word, by wildcard (basic or complex) or synonym matching, or by Soundex matching. Word Search highlights all found terms in context. You can also browse the index and find

© N. Harris Computer Corporation

Feature	Description
	words with a high or low frequency of occurrence or by their leading characters.
Scored matches	Very similar to queries but allow you to rank the results in order of likelihood that the results are exactly what you require. You can save scored matches for future use.
Matching Records	Allows you to discover if records in the database share common values with a single record that you are interested in.
Duplicate Records Checker	Allows you to search for records containing duplicate values. This is similar to Matching Records but it allows you to work with a set, a query or even the whole database.

The results returned by these search methods will depend on whether the database uses cases.

Finding records

You can search for particular types of records that you know, or suspect, are present in the database or case. When you find related records, you can work with them, for example add them to a chart.

About this task

When you use **Find** to search for information, the following rules apply to the search terms that you enter:

- You must enter at least one value, even if it is only a partial value, such as the first few characters of a name. For example, entering `John` might also find `Johns` and `Johnson`.
- Values are not case-sensitive; if `Johnson` is found, then `johnson` might also be found.
- **Yes/No** Checkboxes, are set to either value by default. By clicking the checkbox, you can cycle through the options and search for:
  - Records set to the No value.
  - Records set to the Yes value.
  - records set to either the Yes or No value.
- For fields that use pick lists, you can select the required value from the list. For filtered pick lists, selecting a value in one list can limit the values available in a subsequent list.
- Date fields are excluded unless you turn on the checkbox next to the date.
- By default, aside from numeric fields, any value you specify is assumed to end with the * wildcard character. This default means that values only need to start with the value to be a match. For example, entering `John` may also find `Johns` and `Johnson`.
- You can specify wildcards when you specify the values to search for. In this case, it is not assumed that value ends with the * character. This means that entering J*n will find John and Jon but not Johnson. To find Johnson as well, enter J*n*.

© N. Harris Computer Corporation

The results of your find are listed, but not ordered, by their iBase label, using the label format defined in the current labeling scheme.

Note: You cannot save the definition of a Find to be used in future searches directly. However, to prevent the need to re-enter the details each time, you can save the information either as a query, or as a new record.

Procedure

To find items:

1. In the Database Explorer window, right-click on an entity type or link type and select **Find**. You can change the entity type or link type by using the list provided.
2. **Optional:** To restrict the fields available for selection, turn on **Indexed fields only**.
3. Enter the values that you are interested in.
4. Click **Find**. The results are listed in the bottom pane.
5. You can check each record displayed in the record list by selecting it and viewing its details.
6. **Optional:** To work on the results, select the records, right-click, and select an option from the menu. For example, add the records to a set.
7. **Optional:** To refine your results further, select **Create Query** to add the values that you have entered to a query. For more information on creating a query, see Defining a query.
8. **Optional:** If you are searching for an entity, and no results are found, you can create a record that is populated with the values that you have provided.

   Note: Creating entity records from the values provided in a Find operation, always uses the Show record to display your values rather than a datasheet. This is to ensure that you can see all the fields that are pre-populated with values as the record is created.

Selecting records

You can find and select records as part of larger operations. You might be adding records to a chart, or adding a record to the end of a link, for example.

About this task

You start with a fragment of information, and you find the records in which it occurs. You might find all the records that include `Ken` in the **Given Name** field; the records for `Kenton SMITH` and `Kenneth BROWN`, for example, then select the appropriate one.

Procedure

To select one or more records:

1. Specify the entity type or link type of the records you want to select. You might find the options are restricted. For example, when you select a record for a link end, the options are restricted to the allowed entity types at the ends of the link type.
2. Specify the field values. You do not need to specify the whole value for a field, and you can use wildcards.
3. Click **Find**.
4. Review the records found. For example, in the record list, select each record to check its field values.
5. In the record list, select one or more records and click **OK** to confirm the selection.

© N. Harris Computer Corporation

Search 360

You can use Search 360 to find records that contain the same text as your search words. All records in the database are searched, including records that contain embedded documents and charts.

When you search for exact matches, the following allowances are made:

Variation	**Example**
Letter case	For example, `Peter`, `peter` and `PETER` are exact matches.
Punctuation and accents	`Francoise` is an exact match for `Françoise`
Word order	`SMITH, Fred` and `Fred SMITH` are exact matches.
Incidental words	"black pullover red baseball cap" is an exact match for a record with this text "wearing a black v-neck pullover and a red baseball cap"

You can also use Search 360 to find records that contain similar text to your search terms (a fuzzy match). In this type of search, the following allowances are made:

Variation	**Example**
Typing mistakes	Searching for `ROBERTSON` might also find `ROBETRSON`.
Missing spaces between words	Searching for `Daniel` might also find `Danielsmith`.
Spelling mistakes or sounds likes	Searching for PETERSON might also find PEDERSON.
Name variants	Searching for `MIKE` might also find `MICHAEL`.
Abbreviations	Searching for `road` might also find `rd`.

Note: The Search 360 file limit is 100Mb. Files larger than 100Mb are not indexed, and no search results are returned. You can compress large files to use with Search 360 and improve performance. Contact i2 Group Support if you need to increase the limit.

Searching for linked entities (related items)

You can also find information that is held in records that are directly connected to each other by a single link.

Searching for numbers, dates and time

You can search for numbers, dates, and times if the information was entered as text. The only recognized date and time separators are `/` (for example 2009/10/23) and `:` (for example, 15:43).

© N. Harris Computer Corporation

However, to take account of possible variations in number, date, and time formats you might need to search for spelling variations. For example,:

- An exact search for 10,000 does not find records where the number was entered as 10000.
- An exact search for 24.50 does not find records where the number was entered as 24.5.
- An exact search for the date 04/11/89 does not find records where the date was entered as 04/11/1989.

What you cannot search for

There are a few limitations on what you can search for:

- Values entered in number fields, date fields, or time fields
- If you are using spelling variations, words with fewer than 4 letters
- Common words, such as "the", "and" or any words in the stop list
- Punctuation

Note: Search 360 is only available in SQL Server databases that have been indexed. The date of the last index update is shown in the upper right area of the Search 360 pane.

Although the default setting for Search 360 indexing includes all available fields, your administrator might choose to reduce the scope of the index. Fields that are not added to the indexing process are be returned as part of the search results.

Searching for terms

When you use Search 360, each word that you enter is matched against all the terms in the database's index. The results can include matches in embedded charts, documents, and metadata.

Procedure

To start a search:

1. In the search bar, enter the words that you want to search for and press **Enter**.

   The results are displayed. The results with the strongest matches are listed first and the matching terms are highlighted in yellow. However, what is meant by "matching" depends on how the search options are set. Click **Advanced** to set the search options.

   For instance, here are the results of a search for "Michael Peterson", with the best match first:

Option	**Description**
Michael Peterson	Exact match on both words
Mike Peterson	Uses **Common name variations** to find "Mike"
Michael Pedersen	Uses **Similar sounding words** to find "Pedersen"
Mike Petresson	Uses **Include weak matches** and **Spelling variations** to find "Petresson"

2. **Optional:** Use the **Filter results** pane to filter the results to a specific item or property type.

© N. Harris Computer Corporation

3. Click a record to see the details.
4. To start another operation:
   a. Based on one or more results, right-click on the record and select an option from the menu.
   b. Based on all the results or all results of a specific type, in the Filter results pane, right-click on **All Types** or on a specific type, and select an option from the menu.

Results

The number of results and their type is shown.

Note: By default, Search results are limited to 100. To change the maximum number of records displayed, click **Advanced**, turn on **Limit number of results**, and then enter the required upper limit.

Exact searching

In an exact search, you are searching for records that are an exact match to your search terms. For example, an exact search for `Ford Thunderbirds` will not find results that contain `Ford Thunderbird` (singular).

Procedure

To do an exact search for one or more words:

1. In the Search 360 dialog, click the **Advanced** button.
2. Turn on **Exact matches** and turn off the options in the Include area.
3. Enter the exact words, and any related items, that you want to search for.
4. If your search is for two or more words, select one of the following:
   - **Strong matches only** to find the matches that contain all your search words (but not necessarily in the order you gave them)
   - **Include weak matches** to find partial matches, for example a search for `Ford thunderbird` might match on the word `Ford`, leading to results that include other types of Ford as well as unspecified Fords

     Tip: To search for both the single and plural form of a word, enter both forms of the word and select the Include weak matches option.

Fuzzy searching

In a fuzzy search, you use various techniques to find records that might match your search words. How useful the techniques are depends on the language of the text.

Procedure

To do a fuzzy search:

1. Select **Analysis** > **Search 360**.
2. Click **Advanced**.
3. Turn on the search methods that you want to use:
   - Exact Matches - direct matches to your search term
   - Spelling variations - include common typing mistakes or regional variants
   - Similar sounding words - include words that phonically match

© N. Harris Computer Corporation

- Synonyms - include words with the same meaning

4. Fuzzy searching is set to return only strong matches but you can select the **Include weak matches** option if required.
5. Enter the words, and any related items, that you want to search for and press `Enter`.

Searching for related (linked) items

You can extend a search to include related items. The main search is for the words entered in the search box or in the search bar. If any records are found with your words then the results are filtered to list only those that also contain the item entered in the Related item box. The results might be an entity record, a link record or both.

Procedure

To search for additional items:

1. Select **Analysis** > **Search 360**
2. Select the plus button next to the search to display the **Related item** search.
3. Enter the words you want to search for.
4. Click **Add** to insert the related item in the search box, separated with a semi-colon.
5. Repeat this step for any other related items you want to search for. These items must be directly connected to the first item listed in the search box.
6. Click **Search**.

   Note: You can enter a related item directly in the search box by entering your original search term followed by a semi-colon (;) and then the related item term (consisting of one or more words). Separate each related item with a semi-colon.

Word search

You can find the records anywhere in the database that contain specified text by using a word search. You do not need to know which field the text might occur in.

In addition to finding specific words you can also use the following to broaden your search:

Synonyms - words that have the same meaning

Synonyms are lists of words such that whenever you specify a particular word to search for, all the relevant words on the synonyms list are also searched for. The synonyms list might contain all the words that have the same meaning; for example, synonyms for Firearm might be: Firearm, Shotgun, Rifle, Hand gun, Revolver, Pistol. The lists are pre-defined in the database design, so you cannot change them here. However, you can see which synonyms are searched for.

Soundex - words that sound the same

Soundex means that words that sound the same as your specified word are also searched for. For example, using Soundex you might specify 'check' and find 'cheque'.

Any words added to the database since the database administrator last generated the Word Search index will not be found. The date when the index was last updated is shown. If you need to see details of how the index is defined, click **Index** on the Enter Words page.

Use the:

© N. Harris Computer Corporation

- Enter Words page if you are interested in specific words (or synonyms or similar sounding words) and where they appear.
- Word Index page if you are more interested in how frequently words occur in the database.

Finding records containing specific words

Use the Enter Words page if you are interested in specific words (or synonyms or similar sounding words) and where they appear:

1. Select **Analysis > Word Search**.
2. In the Word Search dialog, click the Enter Words tab to display the Enter words page.
3. Click the Search for box and enter one or more words to search for, separating words with spaces. You can use wildcards to broaden the search. The search ignores the lettercase. It might exclude certain other things, such as entirely numeric values. See 'What you can and cannot search' below for details.
4. In the Combine area, select one of the following:
   - And - the record must contain all your specified words or synonyms of those words if the User Defined checkbox is turned on in the Synonyms area.
   - Or - the record must contain at least one of your words, or one of the synonyms if the User Defined checkbox in the Synonyms area is turned on.
5. In the Type area, select either **Normal** or **Soundex** (includes similar-sounding words).

   Note: A list of words appears whenever any member of the list is specified in the Search for box. All of these words are searched for, in addition to the specified words.
6. If you want the search words to be highlighted in any records found by the word search, turn on the Highlight Words Found checkbox.
7. Click **Search**. Any records that contain the search words are then displayed. The records are identified by their label as defined in the current labeling scheme.

Finding by word frequency

Follow these steps if you are interested in how frequently words occur in the database:

1. In the Word Search dialog, click the Word Index tab to display the Word Index page.
2. Select Occurrences and then either Most or Least (frequent).
3. Specify how many words to list; use the upper button next to the number of words to increase it; use the lower button to decrease it. Alternatively, just click the box and type into it.
4. Click **Find**. The word list will show the most or least frequently occurring words. No records are found as a result of this step.

   Note: It is possible to exclude unwanted words such as "of", "for", and "from" from your search results. Ask your database administrator to set up an exclusion list.
5. In the list of words, select one of the words and then click Search for highlighted word to find the records that contain the selected word. The number of records that will be found is shown in the Count column.

Finding using the beginnings of words

Follow these steps to find records containing words that start with specific characters:

1. In the Word Search dialog, click the **Word Index**.

© N. Harris Computer Corporation

2. Select Beginning with if you want to find words that start with your specified characters.
3. Click the box and type in the starting characters. As you type, the word list shows the matching words and their frequency. No records are found as a result of this step.
4. In the list of words, select an entry and click Search for highlighted word to find the records that contain the selected word. The number of records that will be found is shown in the Count column.

What you can and cannot search

Using a Word Search, you can search fields of the following types:

- Hyperlink
- Multi-line Text and Multi-line Text (Append Only)
- Security Classification Code
- Selected from Code List
- Suggested from Code List
- Text

You cannot search:

- Document type fields - to search the text of embedded documents, use a fuzzy search (if available, SQL Server databases only).
- For punctuation, because punctuation is treated as a word break.
- For special characters, such as €, ~, <, +.
- For words over a certain length (the maximum length is set by the database administrator)
- For purely numeric values (unless your database administrator has chosen to use this option)

To find out the maximum word length or whether you can search for purely numeric values:

1. In the Word Search dialog, click Index. The Word Search Index Build dialog is displayed.
2. Click the Advanced tab to display the Advanced page. The page displays the maximum word length.
3. If you can search for purely numeric values then the Exclude numerics option will be turned off.

Note: By default, entirely numeric values are excluded from the Word Search index. Consider these examples:

Example	**Result if numerics are excluded**
BMW 320	320 is a numeric value, therefore BMW is indexed but 320 is excluded
BMW 320i	320i is not a completely numeric value, therefore both BMW and 320i are indexed
0012-3963	0012-3963 is indexed as a single non-numeric word
-3	Excluded because it is a numeric value
+3	Excluded because it is a numeric value

© N. Harris Computer Corporation

Full-Text Search

You can find records that contain specified text anywhere in the database by using a Full-Text Search. The results of the search depend on how up to date the index is. Your administrator can also choose to exclude certain words.

About this task

Full-Text Search is only available for SQL Server databases. If you use SQL Server 2005 or later, you might prefer to use Search 360.

Procedure

1. Select **Analysis** > **Full-Text Search**.
2. Select the type of search:

Option	Description
Word	Use the Word Search page to search for specific words.
Phrase	Use the Phrase Search page to search for specific phrases or inflections.
Fuzzy	If you do not know what exact combination of words or phrases to search for, use Fuzzy Search. For example, if you want to specify a sentence and find something with a similar meaning.

3. Enter one or more words to search for. Separate words with spaces. You can use the wildcard * or %, at the end of words, to match any number of other characters (including no characters).

   Note: You can search for words that contain punctuation and currency symbols if you use an exact match but not for words that start or end with these symbols.
4. In the Combine area, select either:
   - **And** (all your specified words must be present for a match)
   - **Or** (only one of your words need be present for a match).
5. In the Type area, select either:
   - **Normal** For a standard search.
   - **Soundex** To include similar sounding words.

   Note: Soundex does not operate on document type fields (embedded documents) or use synonyms.
6. **Optional:** In the Synonyms area, turn on:
   - **User Defined** - A list of words appears whenever any member of that list is specified in the search terms. All of these words are searched for, in addition to the specified words.
   - **Person Name Variants** - When searching for people in the database, you can search for different versions of common names. For example, a person in your database might have a given name

© N. Harris Computer Corporation

of Bob. Searching for "Robert" would also find "Bob", "Bobby", "Roberto", and "Rob" as all these names are defined as variants of each other.

Note: The list of name variants is fixed. Name variants are updated by your database administrator.

7. Select which fields you want to search:
   - **All** to search all the indexed fields in the database, and also all Document type fields.
   - **All Name Type Fields** to select all indexed fields that are assigned semantic types to indicate that they contain name information.
   - **Document** to search only Document type fields.
   - **Non Document** to search all indexed fields but no Document type fields.
   - **Selection** to search a selection of indexed fields that you define.

   Note: "Indexed fields" refers to fields that are indexed using Full-Text Search Indexing in iBase Designer rather than set as Indexed Fields.
8. **Optional:** To highlight the words in the record summary, turn on **Highlight Words Found**.

   Note: When the embedded document in a document type field contains the word being searched for and **Highlight Words Found** is turned on, the field name is displayed in Red.
9. Click **Search**.

Results

In the list of records found by the search, you can:

- Select each record to check the field values. The word or phrase will be highlighted if you turned on the Highlight Words Found checkbox before running the search. If the word occurs in an embedded document, then the name of the document field is highlighted in red (Word Search page only). If the text is in an embedded document, you need to show the record and then view the embedded document:
  1. Right-click the record, and select Show.
  2. Right-click the Document icon, and select View. The application for the document type starts and displays the document, you must have the appropriate application installed for the document type. For example, you cannot view an .xls file if you do not have Microsoft Excel installed.
- To work on one or more of the records, select the records, right-click and select an action. For example, it might be useful to add records to a set.

Finding matching records

You can check for matching records to discover if records in the database, or case, share common values with the single record that you are interested in. By default, two entities match if the values in their discriminator fields match, and two links match if the link end entities, link direction and link strength are the same.

About this task

Note: Matching Records is different from the Duplicate Records Checker where you can search for records that match the values in a set of records or the results of a query.

Matching records check discriminator fields by default, but you can match on other fields and the fields that you select are stored.

© N. Harris Computer Corporation

There are three types of entity match:

- Duplicate matching - where entities are matched if the discriminator fields are identical.
- Characteristic matching - where entities are matched if the characteristic fields are identical.
- User-defined matching - where you specify the fields to match.

You can choose to search for matches on any combination of fields. However, the combinations that are most likely to be used (the discriminator fields and the characteristic fields) are selectable as pre-defined combinations.

The choice of which fields are discriminators and which are characteristic is made as part of the database design. For example, if you were comparing crime entities, you can use:

- Duplicate matching to see whether records share a crime number.
- Characteristic matching to check for other crimes where the stolen item was the same as the crime you are interested in.

Matching of links can be much more precise since not only can the field values match, but also end entities, direction, and strength.

Link records directly match if they share the:

- Same field values (for the discriminator fields if any, otherwise for the fields that you select)
- Same link end entities
- Same direction and strength

Alternatively, link records can match if they share just the field values - the link end entities, direction, and strength do not have to match. To use this type of match, turn off **Same Link Ends** and select the fields that you want to match on.

Procedure

To find matching entity records:

1. Right-click the record in a record list and from the menu, select **Matching Records**.
2. Turn on the checkboxes next to the fields you want to check for matching values.
3. Click **Find** to start the search.

   Note: You can pause the search at any time by pressing the Esc key or clicking **Pause**.
4. When the search completes, select a record so that you can check the field values in the **Details** box.
5. To work further on the records, select one or more records, right-click and then select a menu command from the menu. For example, select **Add to Set** to add the records to a new or existing set.

What to do next

You can work with the records listed.

- For the main entity or link, select a toolbar option:
  - Show Records - to show, edit, or delete the main entity or link
  - Links - for entities only, list full details of the links from the main entity, and its link end entities
  - iBase Link Chart - for entities only, show the main entity, its links, and linked entities on an iBase link chart

© N. Harris Computer Corporation

- For a matching record, in the Matching Records area, select one or more records, right-click, and then select an option from the menu. For more information about the available options, see Menus and record lists.

Checking for duplicates

You can search for records that contain duplicate values. The results of this type of search are groups of records that have duplicate values in the fields that are used in the search.

About this task

When you use the Duplicate Records Checker:

- A duplicate entity record is one that has specific values in common with other records for the entity type. You choose which fields are checked.
- A duplicate link record is one where the link end entities, the direction, and the strength of the link are identical. If you want, you can also specify one or more fields to be checked. If required, you can turn off **Same Link Ends** in which case only the field values need to match (the link end entities, direction, and strength are ignored).

For example, if you search for vehicles by color and model, you obtain groups of records that are divided by color and model. Select the color and model combination you are interested in and then browse the records in the group. Color and model combinations that are unique to a single record are not shown in the results.

There are some similarities between **Duplicate Records Checker** and Matching Records because you use both of these features to discover which records in the database share common values. However, you use:

- **Matching Records** - to work with a single record.
- **Duplicate Records Checker** - to work with a set, a query, or even the whole database.

In both, you make the comparison against all the other records in the database. For example, you compare the values in the single record or in the set against the whole database.

Procedure

1. Select **Analysis** > **Duplicate Records Checker**. If the Duplicate Record Checker option is not shown, you need to activate the plug-in (**Tools** > **Plug-in Manager**).
2. In the Duplicate Records Checker, select the entity type or link type.
3. In the Source area, specify the records that you want to check the entity or link type against:
   - **All records**- check against any value in the database.
   - **Query**- check against the records included in the results for a specified query.
   - **Set**- check against the records included in a specified set.
4. In the Fields area, turn on the fields that you want to use in the comparison. You must select at least one field. Initially, the discriminator fields are selected but you can turn them off (and your selection will be remembered for the next time you use the Duplicate Records Checker).
5. If you are working on links, you can:
   - Turn on the **Same Link Ends** to search for links where the link end entities, the direction, and the strength of the link are identical.

© N. Harris Computer Corporation

- Turn off the **Same Link Ends** to search for links where only the field values in the link match (the link end entities, direction, and strength are ignored).

6. Click **Find**. The results of the check are shown on the right. Duplicate groups are listed and records with duplicate values are listed. The number of groups depends on the number of the combinations of duplicate values found. You can sort both the duplicate groups and the records by clicking the column headings.
7. Review the records within each duplicate group by selecting it in the top list.

Merging duplicate records

You can merge duplicate records that are found in the **Duplicate Records Checker** by clicking Merge or by dragging the records into Merge Entities.

Procedure

To merge duplicate records:

1. In the Records area of the Duplicate Records Checker, select the records that you want to merge.
2. Click **Merge** to display the **Merge Entities** with the selected records. The first record that is selected in the Records area of the Duplicate Records Checker is shown in the 'Merge the records below into this record' area of the dialog.
3. If the wrong record is displayed in the upper area of the dialog, select a different one from the lower area and click **Use**.

   Tip: To examine the records in a group of potential duplicates, or use them in a different operation, select the required records in the Records area of the Duplicate Records Checker, right-click and then selecting an option from the shortcut menu. For example, you might want to investigate a single matching value by selecting Matching Records.

Wildcard characters

Wildcard characters can be used to represent non-specific characters in searches. All searching aside from Word Search can use wildcards, words that include wildcard characters are not included in the Word Search index.

Wildcard characters can be included in search terms:

Characters	**Matching letter characters**
* or %	Any number of characters (including no characters) - so w*n matches win, won, and wooden and wn.
? or _	A single character - so w?n matches win, won, but not wooden.
[a-z]	A single letter character in the range between and including the two letters - so w[i-o]n matches win and won but not wan (not case-sensitive so wIn and wOn are also matches). To include accented characters in the search, you must specify the range correctly. To determine precisely which

© N. Harris Computer Corporation

Characters	Matching letter characters
	letters are included in the range, refer to the character map for the character set used in your locale. The character map lists the characters in the order used. Depending on your system, you might be able to obtain a character map for your character set by selecting from the Windows Start menu: **Programs > Windows Accessories > Character Map**.
[abz]	Matches any one of the letter characters - so [wb]ill matches will and bill but not 8ill or till (not case-sensitive so Will and Bill are also matches).
[!abz]	Matches any single letter character that is not in the list - so [!w]ill matches bill, but not will (not case-sensitive so Will is also not a match).

Wildcard numbers can be included in search terms:

Numbers	Matching numbers
[0-9]	Matches a single number in the range between and including the two numbers - so A[1-3] matches A1, A2 and A3, but not A4.
[129]	Matches any one of the number characters - so [12]999 matches 1999 and 2999 but not a999 or 3999.
[!129]	Matches any single number character that is not in the list - so [!12]999 matches 3999 and 9999 but not 1999 or 2999.

If you do not want a wildcard character to represent other characters, enclose it in square brackets. For example,:

`fin*me` finds `findme`

Whereas:

`fin[*]me` finds `fin*me` but not `findme`

Querying your data

You can use queries to run simple or complex searches on your database and display the results. A query is more powerful than a find in that you can locate several entities at the same time by using a much larger range of conditions.

For example, you can:

- Construct queries with specific entity types, link types, or a general Any entity type or Any link type.

© N. Harris Computer Corporation

- Restrict the fields that are searched by setting up one or more conditions on those fields.
- Output all the data that is found by the query or restrict output to specific entity or link types.
- Restrict the records that are queried to the contents of a set or found by a different query.
- Restrict the query to entities with a specific number of links.
- Use semantic types to search entity types, link types, and fields that contain the same sort of data (in SQL server databases only)

The results of a query depend on the records available at the time that you run the query, so you can get different results each time you run a query. If you need to save a static copy of the results of a query, create a set that contains all the results.

You can also define queries that when run, prompt you to enter specific values to be used in the query. This enables you to set up standard queries that can be used with different values.

Once you have created a query, you can use scored matching to find entities and links and place them in order of relevance.

You can also query your data to find, for example, the lowest or earliest values.

Existing queries

You can set up and use queries that cover regular searches. Using stored queries ensures that the same search parameters are used each time that a query is run.

Listing existing queries

To list queries, you can:

- Select **Analysis** > **Queries** > **List** .
- In the Database Explorer, right-click **Queries** and select **List**.

Double-click a query to open it. The Description column might contain notes to help you identify the purpose of the query.

Tip: You can also move the mouse pointer over a query to view a description of the query, if one is available.

Running an existing query

1. In the Database Explorer, right-click **Queries** and select **List**.
2. Double-click the query that you want to run.

   The query is shown as a diagram in the Structure area. The diagram contains icons for the entity types that you want to query and, if there are two or more entity types, links defining the relationship between the entities. The labels of the entities and links can contain information on the purpose of the query.

   Distinct counts are shown without an asterisk, as in (Count = n)
3. There might be some conditions (restrictions) on how the records for each entity and link type in the Structure area are searched. Click each entity and link type in turn to view their conditions. There can be one condition per entity and link type.
4. When you are satisfied that the query is correct, click **Results**.

© N. Harris Computer Corporation

All the records that meet the criteria that are specified in the query are listed on the Results page. The total number of records is displayed below the Results list. Notice that only the first 40 fields are displayed.

Note: If any of the entities or links have a parameterized condition, youare prompted to enter it.

Rerunning a query

To rerun a query, click **Refresh** on the Results page of the query.

If any of the entities or links have a parameterized condition, you can enter new values.

Results of a query

You can work with the records found by running a query:

1. On the Results page of the query, select one or more records.
2. Right-click and then select an action from the menu. For example, select Add to Set to save the results of the query. This is a convenient way of keeping a permanent record of the results of the query for this point in time.

   Note: From time to time, you may need to click **Refresh** to update the list of records. For example, until you refresh the list, a record that you have deleted will remain in the current list, or the old data for a record will be displayed.

Saving the results of a query

To keep a permanent record of the results of the query when run on a particular date and time:

1. On the Results page of the query, select one or more records.
2. Right-click and select **Add to Set**.

   Note: You can also run an existing query by right-clicking on the query and selecting **Records**.

Defining a query

A query is constructed in two parts, a structure that defines the record types of interest, and conditions that define exactly which records are retrieved. When you have defined a query, it can be saved for future use.

Procedure

To define a simple query, you need to draw a structure diagram containing the entity and link types that you want to query:

1. Select one of the entity types (or 'any type') in the Query palette and drag it into the empty pane on the right. The entity type will be shown with an output symbol next to it. This means that records for this entity type will be included in the results of the query. If the symbol is not displayed next to the entity type, you need to add it because at least one entity or link type must be set as output:

   a. Select the entity. The entity label is highlighted in blue.

   b. Click **Output Selected Item** to add it.
2. Select another entity type and drag it in to the pane on the right.

© N. Harris Computer Corporation

Note: Once you add an additional entity type to the query, you must link them. However, the link type drop-down list is only available if there is a link type defined for the two entity types in the drawing pane.

3. Add a link between the two entities:
   a. Select a link type from the drop-down list.
   b. Click **Add Link**.
   c. Hold down the left mouse button and drag a link from one of the entities to the other entity.
   d. If the link has a direction, right-click on it and from the shortcut menu, select **Add Arrow**. To reverse the direction of the arrow or remove it altogether, right-click on the link and select the appropriate command from the shortcut menu. When you run the query, iBase will only search for links with this link direction.
4. If required edit the labels on the entity and links. This serves a reminder of the purpose of the query, if you intend to save the query. To edit a label, click the label to highlight it and then click again. The label changes into a text box. Edit the label and then press the Enter key to apply your changes.
5. Click **Results** to run the query and display the results. For each item in the query, iBase will search all the records in the database. You can limit the search to a smaller number of records.
6. In the Results list, select each record to examine the values in its fields.

   Note: If necessary, you can refine the query by specifying query conditions.
7. Click **Back** and then **Save** to save your query so that you can run it again later.

Select the source records for a query

By default, when you run a query, iBase searches the whole database, or case, for records for each entity or link type in the query. However, you can restrict the records that are searched, for example, to a single record, to the records in a set or the results of another query.

To select the scope of the query:

1. Click the entity or link type in the structure diagram of the query and then click **Assign Source**.
2. Select one of the following options:
   - **All Records** - all records in the database or case are searched.
   - **This record only** - use a single record, which you select. Once selected, the boxes show the Record ID and label of the record.
   - **All except this record** - all records in the database except the one selected are searched. Once selected, the boxes show the Record ID and label of the record.
   - **Query** - search on the results of a query, which you select. You can only select from queries that find records of the appropriate type.
   - **All except the records in this query** - all records in the database except those resulting from the selected query are searched.
   - **Set** - search on the records in a set, which you select. You can only select from sets that contain records of the appropriate type.
   - **All except the records in this set** - all records in the database except those in the selected set are searched.

© N. Harris Computer Corporation

Specify query conditions

You can specify query conditions for each entity or link type in the query. The condition applies to a selected item type, and if there is more than one item type in the query there can be multiple conditions.

Each line specifies a condition, which is in the form of field name, operator and value.

The available operators vary according to the field type. For more information, see List of operators.

Note:

- Field values that are used in conditions are not case-sensitive, so entering '`findme`' finds '`Findme`'.
- Dates are entered in the format that is determined by your Windows regional settings.
- The @ character is a special character that is used to identify parameterized conditions. To find values that start with an @, you must prefix the entire value with an extra @. For example, if the value to match is @123 then you must enter `@@123`.
- For coordinate queries, the conditions are automatically created when you enter the coordinate data in the Coordinate Query Builder.

Add a condition to a query

1. In the Structure area, click the entity or link type for which you want to specify conditions.

   Note: You can only specify conditions on the 'Any Entity Type' item if you have defined standard fields or if the database is an SQL Server database and semantic types have been assigned.
2. In Conditions, click the first line of the table select a field name from the list.
3. Select an operator from the list.
4. Move to **Value 1**, and then either select a value from the list or enter a value using the keyboard. If required, you can use wildcards or parameters.
5. If you chose the between or outside operator, you will also need to enter a **Value 2** to specify the other end of the range.
6. You have now defined a condition for the entity or link type selected. Click **Results** to see the records found by this query.
7. If required, you can now add a second condition for the same entity or link:
   a. To start a new condition, click a blank row.
   b. Define the condition.
   c. In the first column of the condition (the column without a heading), specify how this condition is combined with the other conditions. Select:
      - **AND** - if the record must meet both search criteria, as defined on this condition and the first condition.
      - **OR** - if the record must meet either the criterion that is defined on this condition or the criterion that is defined in the first condition.

      Note: AND operators are evaluated before OR operators.

Using wildcards in conditions

You can use wildcards in field values, but only when using these operators:

- is like, isn't like
- contains, doesn't contain

© N. Harris Computer Corporation

- starts with, doesn't start with
- ends with, doesn't end with

For other operators, the characters are interpreted literally. For example, when using the 'equal to' operator `f*ndme` finds `f*ndme`, not `findme`.

Note: Using the 'ends with' operator and searching for `text`, is the same as using the 'is like' operator to search for `*text`.

Using parameters for usernames, dates and times

You can use the following parameters in query conditions:

Parameter	Represents...
@#USER	The logged-on username.
@#NOWDATE	The current date.
@#NOWDATE `+N/-N`	The current date. You can include `+N` which represents a date 'N' days in the future, or `-N` which represents a date 'N' days in the past.
@#NOWTIME	The current time.
@#NOWTIME `+N/-N`	The current time plus or minus a specific number of hours. You can include `+N` which represents a time 'N' hours in the future, or `-N` which represents a time 'N' hours in the past.

Editing, inserting, repeating, and deleting conditions

At any time you can click a box in the Conditions area to change its contents.

You can also move the rows in the Conditions area - this may have an effect on the results of the query as the operators will be evaluated in a different order.

You can also insert and delete rows:

1. Click the box at the left end of the row. An asterisk (*) appears in the box to indicate the current row.
2. You can then:
   - Click Repeat to copy the current row and paste it above the current row.
   - Click Insert to add an empty row above the current one.
   - Click Delete to delete the current row.

Specifying how multiple conditions are combined

To specify clearly how multiple conditions are combined, you must add brackets. Anything within brackets is interpreted as one part of the query, and evaluated before anything outside the brackets. AND operators are evaluated before OR operators.

Examples:

© N. Harris Computer Corporation

- (Sex is male and age is greater than 20) or eye color is blue - Finds all males over the age of 20 or anyone with blue eyes.
- Sex is male and (age is greater than 20 or eye color is blue) - Finds all males who are either over 20 or have blue eyes.
- Sex is male and age is greater than 20 or eye color is blue - This is the same as the first example because iBase evaluates AND operators first.

To add and remove brackets:

- To add a bracket, double-click the '(' or ')', on the row where you want the bracket to appear.
- Double-click a bracket to remove it.

Checking the position of brackets

To check the position of your brackets, you can select a bracket and press the F3 key. The block of text within the bracket and the other half of its pair is highlighted.

To remove the highlight, click anywhere in the grid.

Prompting for a field value when the query is run

You can parametrize the condition, for example, to prompt for a value when you run the query or to search for the current date.

Using semantic types in a query (SQL Server databases only)

You can use property (field) semantic types to search fields that are used in several entity types or link types and have different names, but store the same type of information. This feature is only available for SQL Server databases with semantic types assigned.

To use property semantic types:

1. In **Field**, select **Property Semantic Type**
2. Select the required property type.
3. Continue to construct the condition and then run the query.
4. You can review which particular fields have been searched after the query runs.

List of operators

Operators are available when you define the conditions for a query. The available operators depend on the type of field. For example, 'Yes or No' fields use a different range of operators to 'Text' fields.

Operators for text

You can use the following operators when you define conditions.

After; After or Equal to

If **Value 1** is set to YOUNG (a surname), the results exclude YOUNG but include YOUNGER. To include YOUNG in the results, use the operator **after or equal to**.

Before; Before or Equal to

To find all license plates that start with the numbers 1 - 3 (in this example, the license plate starts with a maximum of three numbers).

© N. Harris Computer Corporation

Enter either `before 4` or `before or equal to 399`.

Between

To find values that start with the letters `Sch` through `Tho`, you might enter `between Sch Tho` in **Operator**, **Value 1**, and **Value 2**. (The value in the Value 2 column is excluded from the results.)

Contains

To find records that contain the specific value.

For example, the word 'pistol' must be contained in a description of the 'Modus Operandi'. Use wildcards to find variations on the value.

Doesn't contain

To find records that exclude the specific value.

It is important to note that this operator does not return records where the specified field is blank. For example, if you specify that the description of the 'Modus Operandi' does not contain 'machete', the search returns only those records that contain a value but is not 'machete'. Records where the 'Modus Operandi' field is blank are not returned.

If you would like to return blank values as part of the results, add `OR Modus Operandi is blank` as an additional line to the query structure.

Ends with; Doesn't end with

For example, to find all license plates that end with the letters EW (the letter case is ignored):

```
ends with ew
```

To find values that start with a specific value and ending with the letters EW, you would need to use wildcards.

Equal to; Not equal to

Typically used to query values that were originally entered from a pick list - you select the value to query from the pick list. For example, use equal to when the field must exactly match the specified pick list value, such as the `Vehicle Style` must be `Sedan`.

Is blank; Isn't blank

Typically used to find values of fields that were not specified when the record was saved. Use **isn't blank** to search for records with any value in the specified field, or **is blank** to search for records that were left empty.

In list; Not in list

Finds values in a list of values you specify. For example:

- Directly enter a list of values, separating them with the pipe | character. For example: `US|UK|GB|`.
- Enter @ to be prompted for a list of values when the query is run.

You can also double-click **Value 1** to:

- Type or paste a list of value.
- Browse for a text file that contains the required list of values.

If you type in the path, then you must enclose it in braces {}.

© N. Harris Computer Corporation

Is like; Isn't like

Finds exact values (unless you enter a wildcard). For example, a query on the surname YOUNG finds records that contain YOUNG, Young, or young.

Typically used to query values in a field where there might be variations in the spelling. The field must contain a wildcard value in **Value 1**. For example, *Homicide* finds records with Homicide and Scene of Homicide.

Starts with; Doesn't start with

Finds all records that contain fields that start with a specified value.

To find values that start with a specific value and contain or end with other values, you need to use wildcards.

Operators for multi-line text

You can use all the operators that are listed in Operators for text, except for:

- After
- After or equal to
- Before
- Before or equal to
- Equal to
- In list

Operators for numbers

You can use the following operators when you define conditions.

Between

Finds all values between, and including, the figures you enter in the **Value 1** and **Value 2** columns.

Equal to; Not equal to

Finds records where the field exactly matches the value that is given in the **Value 1** column. Typically used to query values that are originally entered from a pick list.

Greater than; Greater than or equal to

Finds all values greater than the specified number. To include the specified number in the results, use the operator **greater than or equal to**.

Is blank; Isn't blank

Typically used to find values of fields that were not specified when the record was saved. Use **isn't blank** to search for records with any value in the specified field, or **is blank** to search for records that were left empty.

Less than; Less than or equal to

Finds all values less than the specified number. To include the specified number in the results, use the operator **less than or equal to**.

Outside

The opposite of the **between** operator. For example:

© N. Harris Computer Corporation

```
outside    10    50
```

finds all values less than 10 (exclusive) and more than 50 (inclusive).

Operators for dates, times, and time zones

You can use the following operators when you define conditions.

Note: When you define conditions that include time zones, you can use the equal to, or not equal to, and is blank, or not blank operators.

After

Finds all dates after the specified date (including the date you enter).

Note: When querying a date or time field, a comparison of **after** with the time portion of the date or time that is entered as 00:00:00 does not find any records for the specified date. You need to either enter the day before and exclude the time portion; or change the condition to **same as or after**; or set the time to 00:00:01.

Before

Finds all dates before the specified date (excluding the date you enter).

Between

Finds all dates between the two specified dates (including the dates you enter).

Day is; Day isn't

Finds all dates that fall on the specified day of the week (**day is**) or dates that fall on any day of the week other than the specified day (**day isn't**).

Different to

Finds records with any date or time value other than the entered value. This is equivalent to **not equal to**.

Is blank; Isn't blank

Finds records where the date or time zone was not specified when the record was saved (**is blank**) or where a date and time zone (*any* date and time zone) was specified (**isn't blank**).

Month is; Month isn't

Finds records with the specified month (**month is**) or any month other than the one specified (**month isn't**).

Outside

Finds records with a date or time outside the range of entered values (exclusive). This is the opposite of **between**.

Same as

Finds records with the specified date or time only. This is equivalent to **equal to**.

Same as or after

Finds records with the specified date or time, or later than the specified date or time.

Same as or before

Finds records with the specified date or time, or earlier than the specified date or time.

© N. Harris Computer Corporation

Year is; Year isn't

Finds records with the specified year (**year is**) or any year other than the one specified (**year isn't**).

Operators for yes or no fields

You can use the following operators when you define conditions.

Equal to

The field exactly matches the value that is given in the **Value 1** column.

Not equal to

The field contains any value other than the value given in the **Value 1** column.

Count conditions

Count conditions can be used in a query to find out information that applies to items that are linked to multiple records. For example, to find out which people are associated with more than two telephones, or which telephones are involved in multiple calls.

In the structure area, you can set a count condition on either a link or an entity. You can only set one count condition in the structure, and if the count condition is applied to an entity, it must have a single link. To set up a count condition, right-click an entity or link in the structure area and select **Count**.

Note: If you are using an SQL Server database, then you can use a distinct count when you want multiple links between two entities to contribute only 1 to the count.

Count conditions on entities

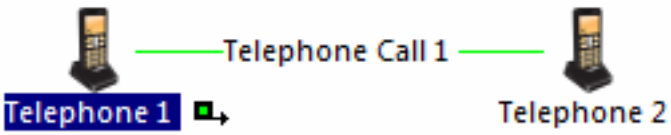
If you set the count condition 'more than N' on an entity in the query, then in the matching record group there will be in total more than 'N' links from all the entities that match that structure entity, to any one entity that matches the structure entity at the other end of the structure link.

Note: If you turn on **Distinct** (only available in SQL databases), then multiple links between the same entities only contribute 1 to the count. So if, in the example above, an account has 11 transactions all from the same person, then turning on **Distinct** will exclude it from the results.

Count conditions on links

If the count condition 'more than N', is set on a link in the query, then in the matching record group there will be more than 'N' links between two entities that match the link end structure entities.

For example, if your structure contains:



© N. Harris Computer Corporation

You might set 'more than 15' on the link to limit the results to telephones that have more than 15 calls between each other. Each telephone in the results is involved in at least 15 calls with one other telephone in the results.

Setting count conditions in a query

1. In the structure area of the query, select the entity or link.
2. Click **Set Count on Selected Item**.
3. Select **Condition** to set a count condition and select the type of count:
   - =

     Count must be equal to the number you enter. You can only enter 0 as the count number on entities or links that are not set as the output.
   - <>

     Count must be less than or greater than the number you enter.
   - <

     Count must be less than the number you enter. The count number must be 3 or higher.

     Note: This condition does not include 'equal to 0'. For example, if you had a structure Telephone - Call - Telephone, with less than 3 on one of the telephones, the results would not include telephones that have made no calls. To find telephones that have made no calls, you would must explicitly use the condition equal to 0.

     Because you cannot output the entity with the condition equal to 0, make sure you set the telephone without the condition as the output.
   - >

     Count must be greater than the number you enter.
4. Enter the actual count.
5. If you are using a SQL Server database, turn on **Distinct** where you want multiple links between the same two entities to only contribute 1 to the count. For details, see the next section.
6. Click **OK** to confirm the changes.

Distinct counts in SQL Server databases

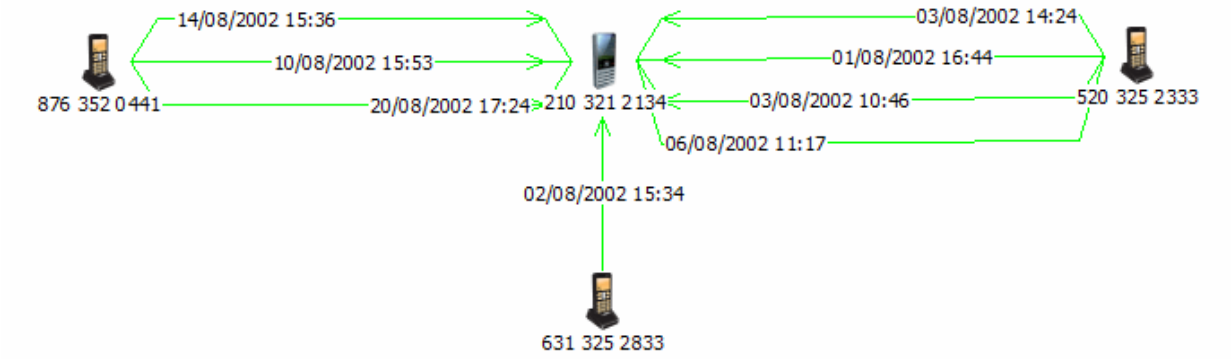
A distinct count is one that counts the number of different entities linked to a specific entity. In queries based on linked entities, a distinct count helps you determine how many different entities are linked to another.

As with other counts, you can look for relationships where the count is less than, equal to, or more than a number you specify.

For example, you may be interested in bank accounts where one account transfers money into only one other account, regardless of how many transactions take place. In another example, you might be interested in telephones that are used in calls with several other telephones, regardless of how many calls are made in total.

Consider this example:

© N. Harris Computer Corporation



In the example above, 210 321 2134 is linked to three different telephones, so it has a distinct count of three. All the other telephones have a distinct count of one. The table shows the distinct and non-distinct counts for comparison.

Telephone	Distinct count	Non-distinct count
210 321 2134	3	8
876 352 0441	1	3
631 325 2833	1	1
520 325 2333	1	4

With these counts, you can see the effect of non-distinct and distinct count queries:

- Working with an Access database, you can only specify a non-distinct count condition. A query based on searching for more than two telephones and counting repeated links to the same telephones is effectively a search for more than two telephone calls and would find 210 321 2134, 876 352 0441, and 520 325 2333.
- Working with a SQL Server database, you can specify a distinct count condition. A query based on looking for links to more than two different telephones would find only 210 321 2134.

Note: A non-distinct count is shown with an asterisk, as in Count*.

To create a query involving a distinct count, select the **Distinct** checkbox.

© N. Harris Computer Corporation

Using the query parameters list dialog

A parameterized query is one which has been defined to prompt you to enter the data that you want to use when querying the database. You use the Query Parameters (List) dialog to specify a list of values you want to use for the query.

Procedure

You can use a text file containing the values, or enter the values manually.

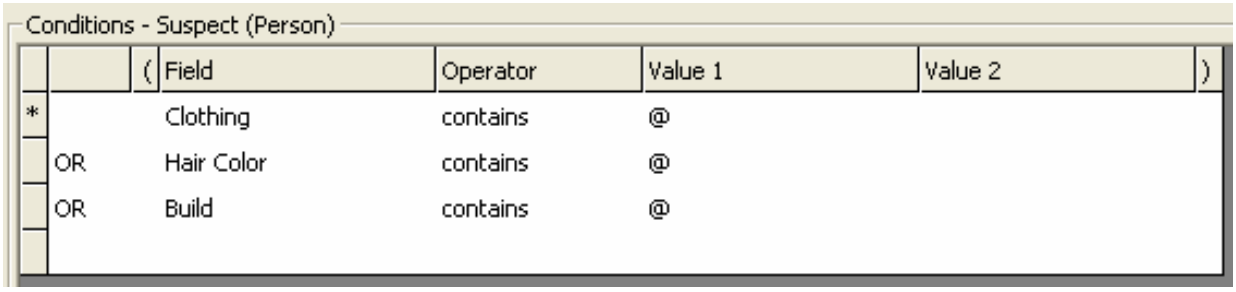
1. To specify a text file containing the list of values, select the first radio button, click `...` and browse for the required text file. To view the contents of the file, click `....`
2. To enter a list of values manually, select the second radio button. Type or paste the values into the box.
3. Click **OK** to run the query.
4. To re-run a parameterized query with values, click **Refresh** on the **Results** page. Enter the new values when prompted and click **OK**.

Queries that prompt

You can define a query that prompts you to enter the values for the search conditions when you run the query. This is useful if you have a standard query that you would like to repeat for different values. It is also possible to use this type of query to provide the source data for another query.

Example of a query with parameterized conditions

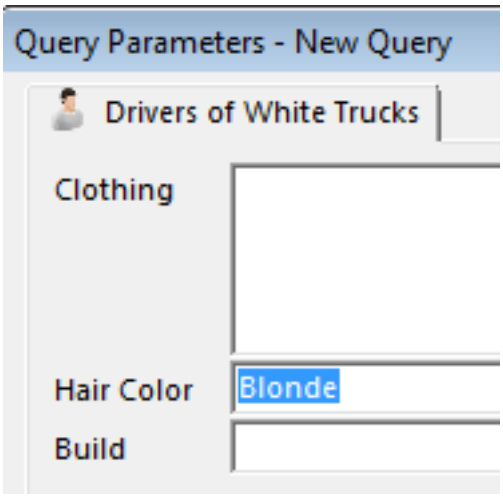
When you run a query with conditions defined like this:


Conditions - Suspect (Person)

		(	Field	Operator	Value 1	Value 2	)
*			Clothing	contains	@		
	OR		Hair Color	contains	@		
	OR		Build	contains	@		

You will be prompted to enter a value for each field set to the @ character:

© N. Harris Computer Corporation



To run the query, you must enter a value for all the parameterized conditions.

Setting up a parameterized query

When you define the conditions for the query, in the Value 1 column, enter one of the following:

- @ instead of the actual value.
- @ followed by a default value. For example, you might enter @Yes as the default for a Yes/No field.

You will enter the actual value when you run the query. Depending on the operator you used, you may need to enter @ in the Value 2 column as well.

Note: You may also find it helpful to give parameterized queries a name that indicates the purpose of the query, and that it is parameterized. For example, add a # at the front of the name to indicate parameterized queries.

Available parameters

You can use the following parameters:

Parameter	**Description**
@	There will be a prompt for this field value when the query is run.
@*default value*	There will be a prompt for this field value when the query is run. If you specify a *default value*, there will be no user input required to specify that value (apart from confirming it). For example '@BMW' specifies 'BMW' by default.

© N. Harris Computer Corporation

Running a parameterized query

When you click **Results** to run a query, a separate page for each item in query that has one or more parameterized conditions will appear. You must enter a value for all the parameterized conditions before the query can run.

Note: If you know which operator was used to define the condition, then you may be able to use a wildcard when entering the value. You cannot use wildcards with all the operators. .

Querying data using semantic types

Semantic types can be used to search for data in several different entity types or link types in a single query. You can also use semantic types to search a number of fields in different entity or link types that contain the same kind of information.

There are two ways to use semantic types in a query:

1. Entity and link semantic types. You can search across several entity or link types that all represent the same sort of information.
2. Property semantic types. You can search fields across different entity or link types which may be called different things but contain the same sort of information.

You will only be able to use semantic types in your query if the database is an SQL Server database and semantic types have been assigned to the data. If no semantic types have been assigned, none of the semantic options are available when you define a query. Speak to your database administrator for more information about the semantic types that are assigned in your database.

Constructing a query that contains semantic types

You construct a query that uses semantic types in the same way as you would create any other query. You can:

- Specify the semantic type for an entity or link type in the query structure area.
- Use a property semantic type for a condition in the Conditions area.

Viewing the scope of a query that contains semantic types

When you view the results of a query that includes semantic types it is useful to see exactly which entity types, link types and fields are searched. Unlike other queries, when you use a semantic type, you do not necessarily know before the query is run which types and fields are included in the query. You can use the scope to view a list of the searched items.

Semantic types in iBase

Semantic typing is a method of categorizing data to define how to interpret it. For example, the Person entity semantic type might be assigned to entity types such as Male, Victim and Witness. The semantic type indicates that each of those entity types are different ways of depicting people in the real world.

Semantic types can be assigned to each relevant entity type, link type, field, standard field, and icon. Semantic types are assigned by your database administrator using iBase Designer.

There are three different kinds of semantic type: Entity, Link, and Property (for entity and link type fields, including standard fields).

© N. Harris Computer Corporation

The i2 Semantic Type Library

The i2® Semantic Type Library contains semantic types that are assigned to data in your data sources. These semantic types identify the meaning of the data they represent, and are used by applications such as Analyst's Notebook to properly interpret and align the data from different data sources.

Each semantic type consists of the following elements:

- Name
- Data type, such as text or number
- Optional synonyms - alternative names that are used when you search for suitable semantic types.
- Description that provides guidance on how the type should be used.
- Notes

Depending on its location in the hierarchy of semantic types, the function of a semantic type is either general or specific. For example, Motor Vehicle is a specialized type of Transport, and Bus is a specialized type of Motor Vehicle.

Semantic types in queries (SQL Server databases only)

In SQL Server databases, semantic types can be used to search across several entity or link types that have been aligned based on the sort of data they contain. For example, your database may contain several entity types which are all different types of people: Victim, Offender, Officer, Suspect, and so on. All these types contain data about people. If you are looking for a person, you would search across all these types. Instead of running several queries, you can use a semantic type in the query.

This can be useful if you are not familiar with the database, and therefore do not know how the data in it has been structured - which types have been used, and how they are linked. It can also be useful if you know the Semantic Type library and can therefore construct the same query for use in different databases.

Unlike other queries, using a semantic type means that you do not necessarily know before the query is run which types and fields have been assigned the semantic type specified in the query. You can use the Scope dialog to view a list of what has been searched. The Scope dialog is accessible from the Query Results page.

Entity and link semantic types in queries

Semantic types can be used to search for data in several different entity types or link types in a single query. You can use semantic types in this way if your database is an SQL Server database and semantic types are assigned.

Procedure

To search entities that are assigned a particular semantic type:

1. List the available semantic types by either:
   - Drag the {Semantic Type} icon from the query palette onto the pane on the right.
   - Drag the {Any Entity Type} icon from the query palette. Then:
     - Click **Assign Semantic Type**.
     - Right-click on the icon, and select **Semantic Type**.
2. Select the required semantic type from the tree view or type the first few letters in **Search Available Semantic Types**.

© N. Harris Computer Corporation

3. Select the required option:

Option	Description
means	Finds all entity types that are assigned either the selected semantic type or one descended from it in the hierarchy.
is exactly	Finds all entity types that are assigned the selected semantic type, but not a semantic type descended from it in the hierarchy.
does not mean	Finds all entity types that are not the selected semantic type and are not descended from it in the hierarchy.
is not	Finds all entity types that are not the selected semantic type, but might be a semantic type descended from it in the hierarchy.

4. Click **OK**. The query pane is updated to show the selected semantic type and match option.
5. You can then set more specific conditions on the properties of the entity. In the Conditions area, click the first line of the table below the Field heading, and then select the field name from the list. The list is the same as for the {Any Entity Type} entry and contains:
   - Fields that are common to all entity types in your database, for example standard fields.
   - A <Semantic Type...> entry to search fields that are used by the selected entity semantic type.
6. Continue to construct the rest of your query as usual.

Link semantic types in queries

Semantic types can be used to search for data in several different link types in a single query. You can use semantic types in this way if your database is an SQL Server database and semantic types are assigned.

Procedure

To search links of the same semantic type:

1. Create a query by dragging two entity types into the query structure area on the right.
2. From the link type list, select {Semantic Type}. The available link types depend on the selected entity types. It is possible there might not be any link types available (if the two entity types are not valid ends for a link type). If so, {Semantic Type} is not available to select.
3. The pointer changes to a plus sign to show that you can add a link. Drag the link from one entity to the other.
4. Select the required semantic type from the tree view.
5. Select the required option from the Match Option list.
6. Continue to construct the rest of your query as usual.

© N. Harris Computer Corporation

Link semantic types between 'Any Entity Types'

You can use a link semantic type between two Any Entity Types to search for all link types that are assigned the selected semantic type, regardless of the link ends.

Procedure

1. Drag two {Any Entity Type} icons into the query structure area on the right.
2. From the link type list, select {Semantic Type}.
3. The pointer changes to a plus sign to show that you can add a link. Drag the link from one entity to the other.
4. Select the required semantic type from the tree view.
5. From the Match Option list, select the required option.
6. Continue to construct the rest of your query as usual.

Property semantic types in query

Assigning property (field) semantic types to fields in different entity or link types allows those fields to be searched together as if they were the same field. This typing can be used to group fields from various item types that contain the same information.

About this task

For example, Offenders might have a "Surname", Suspects a "Last Name" and Witnesses a "Family Name". All these fields contain the same sort of information. If these fields are assigned the Person Last Name semantic type, then you can search all the fields together by using the semantic type as a condition in the query.

You can search property semantic types in all the entity or link types in your database, or to a specific entity or link semantic type or specific entity or link type.

Note: To use semantic types in a query, your database must be SQL Server with semantic types assigned.

Procedure

To search fields of the same semantic type across all entity types:

1. Drag the {Any Entity Type} icon from the Structure area onto the pane on the right.
2. In the Conditions area, click the first line of the table below the Field heading and select <semantic type...>.

   All the property semantic types for semantic entity types are displayed.

   Some of the property semantic types that are listed might not be assigned in the database, but are the parents of types which are assigned. For example, Person First Name and Person Last Name may both be assigned, so Generic Name (their parent type) is included in the list.
3. Click a property semantic type to view more information about the selected type. If you know the name of the type you want, you can locate it quickly by typing the first few letters of any word in the property name in **Search Available Semantic Types**. As you type, possible matches are displayed. If the word matched is part of a semantic type name with several words, the full semantic type name is shown in brackets. Select the semantic type that you want in the list. It will also be selected in the main tree view.

© N. Harris Computer Corporation

4. To filter the list of properties to show only property semantic types that have been assigned to a field in every entity type being searched in the query, turn on **Show only common types**. This can result in an empty list.
5. Click **OK** to update the Field column of the Conditions area with the selected property. The field is identified as a semantic type as it is enclosed in curly brackets { }.
6. Specify the rest of the condition using the available operators.

   Note: Certain operators are not available when using property semantic types, including any that are negative.
7. Click **Results** to run the query.

Scope

The scope of a query is all the entity types, link types and fields that have been searched. Each entity type and link type, and any fields within that item type that were searched in the query can be shown.

You can use the Scope dialog to view a list of all the types and fields that have been searched when running a query. This is particularly useful when you run a query that uses semantic types, because you might not know beforehand which entity types, link types or fields have been assigned the semantic type selected in the query.

When you click **Results** to run the query, the Results screen is displayed and the Scope option is available. To open the Scope dialog, click the **Scope...** button.

Selecting an entity or link semantic type

Entity or link semantic types can be used in your query. When you choose a semantic type for {Any Entity Type} or {Any Link Type} in the query, then all the entity or link semantic types available in the database are shown.

About this task

When you choose a semantic type for a specific entity or link type, only those semantic types relevant to that entity or link type are displayed. This filtering is useful when the entity type uses different icons to represent specializations of the type. For example, if you select a Telephone entity type in the query, several Phone semantic types are available.

You might narrow your query to search just cellphones by selecting the Mobile Phone semantic type.

Note: All the ancestors of the semantic type are still displayed, indicating from which more general semantic types the Mobile Phone is type is derived.

Procedure

To select an entity or link semantic type:

1. In a query, right-click an item and select **Semantic Type**.
2. Select the required semantic type from the tree view.
3. You can widen your search by changing the text that is displayed in **Search Available Semantic Types**:

Option	Description
Shorten the displayed text	"Documents" to "Document"

© N. Harris Computer Corporation

Option	Description
Simplify the displayed text	"End date" to "date" or "end"
Consider alternative spellings	"tire" to " tyre"

4. If none of the semantic types in the Ordered Results area are suitable, you can browse the semantic types that are displayed in the tree view.
5. From the Match Option list select the required option:

Option	Description
means	Finds all entity or link types that are assigned either the selected semantic type or one descended from it in the hierarchy.
is exactly	Finds all entity or link types that are assigned the selected semantic type, but not a semantic type descended from it in the hierarchy.
does not mean	Finds all entity or link types that are not the selected semantic type and are not descended from it in the hierarchy.
is not	Finds all entity or link types that are not the selected semantic type, but might be a semantic type descended from it in the hierarchy.

6. Click **OK**. The query pane is updated to show the selected semantic type and match option.

Selecting a property semantic type

Select the required property (field) semantic type for use in your query. The available semantic types are shown as a tree view.

Procedure

To select a property semantic type:

1. Drag an entity type from the query palette.
2. In the Conditions area, select **Semantic Property**
3. Select the required semantic type from the tree view.
4. If you know the name of the type you want, you can locate it by typing the first few letters of any word in **Search Available Semantic Types**. The list of matching properties is updated in the Ordered Results area as you type more letters.
5. Click OK.

© N. Harris Computer Corporation

Managing queries

iBase can display a list of all the queries in a category. The full list of categories appears in a hierarchy that you can expand and collapse.

The list of categories displays information about when the queries were created and updated, and who created and updated them. You can sort the queries by clicking the required column heading.

To administer the queries for your database, right-click a query and then select:

Command	Description
Open	Run or edit the selected query.
Rename	Rename the query. Only available when a single query is selected.
Save As	Save a copy of the selected query under a new name. The name must be unique not just to the category but to the whole database.
Delete	Delete one or more selected queries. You are prompted to confirm the deletion.
Categorize	Move one or more selected queries, which can be in different categories, to another category, and set access control on them.
Properties	Show properties of the query, such as its creation date and user. Only available when a single query is selected.

Note: You should consider deleting queries that are no longer in use. An iBase database that contains a large number of queries can cause performance issues when using features such as Batch Edit.

Selecting a query or set

You can select an existing query or set, for example, as part of the process of adding records to the set.

Procedure

To select a query or set:

1. In the pane to the left, navigate to the category that contains the query or set you want to select.
2. In the window to the right, click to select a set or query.

   Only the sets or queries that contain the selected entity type are listed.
3. Click **OK**.

   Tip: Move the mouse pointer over the name of a query or set to view a tooltip containing more information about it, if a description for that query or set has been entered.

© N. Harris Computer Corporation

Scoring the results of queries

Scored matching provides a way of finding relevant records and listing them in order so that you can identify how well those records match a query. It is a useful analysis method where several fields and values might contribute to what makes the records interesting to you. The result of a scored match is always a list of entities of one type, which is listed in order of score.

About this task

Scored matching relies on you assigning a score, or weighting, to fields that match specified conditions. Those records with the highest combined score are listed first. You specify how the score is to be applied to individual fields. For example, you might be interested in dark haired people, so you score the Hair Color field: you weight the Black field value with 10, the Dark Brown value with 9, down to Fair with 1.

If you select a query as the source, the scored matching considers only the results of the query that are selected in the Entity Type to match list. Selecting a query makes it possible to score fields and values that relate to any of the other entities and links in the query.

For a query to be used as a source of data:

- It must be saved.
- The results must include the entity type that you want to see in the results of the scored match. You can include other entity and link types.
- If it has two outputs of the same entity type, the names used for the entities in the query structure should be meaningful. For example, you are unlikely to understand the significance of names such as Person 1 and Person 2 without examining the query.
- It should be as general as possible so that it does not exclude records that are unlikely matches but where you might want to define low scores.

Procedure

To use scored matching:

1. From the Entity Type to match list, select the type of entity you want to find (the 'match type').
2. In the Match on records from the following source area, select the scope of the search.

   If you select a query, you may see **Match to this item** if there is a choice of output entities for the 'match type'. Select the entity you want to match.

   Note: You cannot select a query that has a count condition on a link, such as count = 0, meaning, for example, 'give me all the people who have no address links'.
3. Set the minimum score you require in the Minimum score box.
4. Specify conditions and the scores to apply when the conditions are matched. In each line, specify the **Field**, **Operator**, **Value 1** and **Score** (for some operators **Value 2** is also required). The * character indicates the selected line.

   You can:

   - Click an empty row to start a new scored condition or click Insert. However, the order of the lines will have no effect on the results.
   - Use the same operators as in querying. Note: in list/not in list are not available for scored matching.
   - Use wildcards in field values.

© N. Harris Computer Corporation

- Click Repeat to copy the selected line.
- Click Delete to remove the selected line.

If you specify a query for the source, you can also specify conditions and scores for links and linked entities in the query. There are pages for each link and link end entity.

5. Click Results to list the records with their relevant scores on the Results page.

Finding ranges of values

You can perform calculations on groups of entity or link records containing numeric and date type fields, and report the results. The calculation is performed on the values of a particular field, across all the records in the group, which is either the whole database, a query or a set. For example, you could calculate the total of all the Goods Value fields of the Crime records in a Jewelry Theft set.

About this task

The calculations you can choose from depend on the type of the selected field

For Date or Time type fields, you can choose from:

- Earliest - the earliest date or time
- Latest - the latest date or time

For numeric type fields you can choose from:

- Lowest - the lowest numerical value
- Highest - the highest value
- Average - the average value
- Sum Of - the total value
- Standard Deviation - the standard deviation

Procedure

1. In the Field Calculator, specify an item type by selecting it from the list.
2. Specify the group of records from:
   - All records to work with all records in the database of the specified entity or link type.
   - Query to work with the results of a query, which you select. You can only select from queries that output records of the specified entity or link type.
   - Set to work with the records contained in a set, which you select. You can only select from sets that contain records of the specified entity or link type.
3. In the Fields list, click to select the field you want to perform calculations on. You can only select from numeric, date, or time type fields.
4. In the Functions list, select the calculation you want to perform.
5. Click **Calculate**.

© N. Harris Computer Corporation

Defining coordinate queries

Coordinate queries can find entities or links of a particular type within a defined geographic area or close to a location.

About this task

You can use coordinate queries to search entities or link types that have coordinate fields. You can only run coordinate queries on entity or link types that have a Coordinate type field, followed by two real number fields that contain coordinate data.

Procedure

1. Select **Analysis** > **Coordinate Query Builder**.
2. Select the entity or link type that contains the coordinate data.
3. Select the Coordinate type field. This will typically be selected automatically, as there is usually only a single Coordinate field for each entity or link type.
4. In the Source area, specify which records you want to include in the coordinate query.
5. In the Query Operator area, select the required operator:
   - Is near - finds records with coordinate data that is within a specified distance of a location you enter. Enter the location using coordinates and then specify the required distance and units.
   - Is between - finds records with coordinate data that falls within a rectangle whose corners you define. Enter the two sets of coordinates to form the corners of the search area.
6. In the Coordinates area, enter the coordinates

   If you selected 'Is near', enter one set of coordinates, then enter a value in the Tolerance box and the units for that value, such as Feet, Kilometers, Miles, Seconds, or Yards. This value is used to calculate the distance from the entered coordinates. The use of *seconds* as a unit is a way to specify small distances. Each degree is divided into 60 minutes, and each minute is divided into 60 seconds. If you, for example, set a tolerance of 30 seconds on a latitude of 38 degrees, the tolerance will be 30/3600 = 0.0083333 degrees.

   The Tolerance is calculated by adding the specified distance to both the horizontal (longitude) and vertical (latitude) part of the coordinates to create a square with the original coordinates in the middle. Any record with coordinates that are located within this square is found.

   If you selected 'Is between', enter two sets of coordinates. These coordinates form two corners of a rectangle. Records with coordinates that fall within the defined square are found.
7. Click **Next** to create the query.
8. Click **Results** to run the query.

Alerting

You can monitor records to detect when an item of interest changes or is viewed by someone else. To monitor items of interest, you add alert definitions to records (single or multiple) and to queries. When a change is detected, an alert is raised. On receiving an alert, you can drill down into the detail to find out what changed, who viewed the record, or mark the alert for follow up later.

Enabling alerting

Alerting is enabled in iBase Designer, and is only available for SQL Server databases. For more information, see Configuring alerting in the iBase Designer documentation.

© N. Harris Computer Corporation

Adding alert definitions

You add an alert definition so that alerts are raised when:

- A record is viewed or changed
- A record in the Query results is viewed or changed
- Additional records appear in, or no longer appear in, the Query results

Alert definitions are not linked to the queries or records they are based on. For example, modifying, copying, or deleting a Query does not affect any alert definitions based on it.

Where are alerts found?

Depending how alerting is set up, you might see one alert for each item of interest:

- In the alerting status bar
- In the alerting Inbox
- As an email message
- In your Windows™ system tray (most recent alert only)

Click **Refresh Settings** to set the frequency at which you receive new alerts. Your system administrator sets the frequency for email messages.

How are alerts followed up?

To view the item on which the alert was raised:

- Click the alert in the alerting status bar
- Click the alert in the alerting Inbox

Which types of action raise alerts?

There are four types of alert:

Record Viewed alerts

A Record Viewed alert is raised whenever the record is:

- Displayed in a record list, for example as a result of finding, browsing, or opening a set
- Displayed in Show or on a datasheet
- Displayed when soft deleted records are purged or restored
- Listed or viewed in Audit Viewer or the Audit History (but not when you are setting up alert definitions or viewing alerts)
- Listed as a link end record
- Viewed on an iBase link chart
- Exported or sent to an Analyst's Notebook chart

The alert is raised when the record is first shown or listed.

Record Changed alerts

A Record Changed alert on entities is raised when:

- Any entity fields are changed
- New links are added to the entity

© N. Harris Computer Corporation

- The strength or direction of any link to the entity is changed
- The entity is deleted
- Any links to the entity are deleted
- Entities or links are soft deleted or restored

Note: Changing a field on any links to the entity is not a change to the entity itself.

A Record Changed alert on links is raised when:

- Any link fields are changed
- The strength or direction is changed
- A link end entity is replaced by a different link end entity
- The link is deleted
- A link end entity is deleted causing the link to be deleted
- Link end entities or links are soft deleted or restored

Note: Changing a field on any link end entities is not a change to the link itself.

Records Added alerts

A Records Added alert is raised whenever an extra record is found that matches the selection criteria for the Query. This might be for the following reasons:

- New record added that matches the Query
- Changed so that it now matches the Query
- Restored (having previously been soft deleted)
- Changes to your permissions, which mean that you can now see more records

Records Removed alerts

A Records Removed alert is raised whenever a record that previously matched the selection criteria for the Query is no longer found. This removal can be for the following reasons:

- Changed so that it no longer matches the Query
- Deleted
- Changes to your permissions, which mean that you can now see fewer records

Note:

- Alerting is available in SQL Server databases only
- You can only add alert definitions if you are permitted to do so
- Email alerts can only be sent if your system administrator has enabled this feature

Further details for administrators are available in the Administration Center.

Alert definitions

An alert definition defines a set of conditions that when met generate an alert to specified users. You can add an alert definition that monitors access to specified records or the results of a Query.

Alert definitions are set up to generate alerts when specific criteria are met. For example, you might want to know if a record is updated with information, or an additional record is added to a set. Alert definitions have the following core components:

- Information on the records or query to be monitored

© N. Harris Computer Corporation

- Details of who to inform
- An expiry period if the alerts are only required for a set period of time

Note: Alert definitions are not directly linked to the original records or Queries, this means that:

- deleting a record does not delete any associated alert definitions
- modifying the Query does not update the alert definition, and deleting the Query does not delete it.

Adding alert definitions on records

You can set up an alert definition to generate alerts when changes to records are detected. Alerts can be generated when records are viewed, edited, or deleted.

Procedure

1. Find the record or records you are interested in.
2. Right-click on the records, and from the shortcut menu, select **Add Alert**.
3. Enter a unique name for the alert definition and an optional description. The name of the alert definition is used every time that an alert is sent to the subscribers of this alert.
4. Decide what the alert definition is monitoring by turning on the checkboxes in the **Alert when** area. These definitions raise alerts when:
   - Records are viewed
   - Records are edited
   - Records are no longer found
5. If required, set an expiry date. No further alerts are sent after the expiry date.
6. Click **Save**. The alert definition is listed in the **Alert Definitions** area of the **Database Explorer**.

   Note: As the owner of the alert definition, only you or the system administrator can edit or delete the alert definition. If other users do not want to receive alerts, they can unsubscribe.

Adding alert definitions on queries

You can set up an alert definition to generate alerts when changes to the results of a Query are detected. Alerts can be generated when the resulting records are viewed, edited, or deleted.

Procedure

1. Find the Query that you are interested in (it must not be a parameterized Query or a semantic Query).
2. Right-click on the Query, and select **New Alert**.
3. Enter a unique name and an optional description. The name of the alert definition is used every time that an alert is sent to the subscribers of this alert.
4. Click **Select Users** to add the subscribers to the alert.
5. Decide what the alert definition is monitoring by turning on the checkboxes in the **Alert when** area. These raise alerts when:
   - Records are viewed
   - Records are edited
   - Records are no longer found

© N. Harris Computer Corporation

- Extra records are found

6. If required, set an expiry date. No further alerts are sent after the expiry date.

   Note: As the owner of the alert definition, only you or the system administrator can edit or delete the alert definition. If other users do not want to receive alerts, they can unsubscribe.

   The alert definition will not be updated after a change to the Query on which it is based. To monitor a modified Query, you need to add an alert definition.

Adding the users who are alerted

Alert definitions can be used to generate alerts that are sent to multiple users. Adding multiple users to an alert definition reduces the need to set up extra alert definitions.

Procedure

1. In **New Alert Definition**, select **Select Users**. Any existing subscribers are listed on the right.
2. Select the users to receive the alerts.
3. Click **OK**.
4. Specify the type of alert that each subscriber receives. You can choose that the user is alerted when they open the database (an iBase alert), that they receive an email, or both. Both alert types also display in the computer's system tray.

Reviewing your subscriptions

Your alert subscriptions list the alert definitions that are currently generating alerts that you will be notified about. You can review your alert subscriptions to ensure that the alerts that are being generated still match your requirements.

To view the list of alert definitions that you are currently subscribed to:

- In the Database Explorer, expand **Alert Definitions**. The alert definitions that you subscribe to are listed with a blue icon.
- In the Alerting dialog, click **Subscriptions** to list the current alert definitions for which you are a subscriber.

To remove yourself from an alert definition, select the alert definition and click **Unsubscribe**. You will receive an alert that you have been unsubscribed but existing alerts will not be removed from your Inbox, and the detail of those alerts will remain available to you.

Reviewing expired alert definitions

Once an alert definition has expired, no additional alerts will be generated unless you modify the expiry date. You might want to reactivate an expired alert definition if the alerts are still required.

There are two ways of finding out which alert definitions have expired:

- In the **Database Explorer**, expand **Alert Definitions**. Expired alert definitions are listed with a gray icon.
- In the **Alerting** dialog, click **Expired Alert Definitions** to list the expired alert definitions that you own.

As the creator of these, you can do any of the following:

- Change the expiry date, for example to make an alert definition active again.

© N. Harris Computer Corporation

- Change the actions that are being monitored.
- Delete the alert definition - this will not delete the alerts that have already been sent (or that are in the process of being sent).

Editing and deleting your alert definitions

Alert definitions that you have created can be edited and removed. You can maintain alert defintions to ensure that the information remains relevant to your investigations.

About this task

After you have created an alert definition, you have permission to modify or remove that definition in the following ways:

- Add or remove subscribers - You can modify the users that are alerted
- Modify the expiry date - You can change the length of time that alerts are generated for
- Change the actions that are being monitored
- Delete the alert definition

Note:

- You cannot add or remove records from an alert definition or change the query saved as part of the alert definition. If you would like to change these options, create a new alert definition.
- Alert definitions are automatically deleted if a change to the entity and link types in the database makes the alert definition invalid. As the owner of the alert definition, you will receive an alert informing you of the deletion.

Procedure

1. In the Alerting dialog, select **Alert Definitions** to list the alert definitions that you own.
2. Click **Edit** to open the Alert Definition in an editable form.

   Note: Alternatively, click **Delete** to remove the alert definition entirely and prevent any further alerts from being generated.
3. Make your changes:
   - To modify the subscribed users, click **Select Users** and add or remove the users.
   - Use the **Duration** options to change the duration that the alert definition is active.
   - Modify the alert definition. For more information on the available options, see:
     - Adding alert definitions on records
     - Adding alert definitions on queries
4. Save your changes.

© N. Harris Computer Corporation

Clear alert data

If you have removed an alert definition, you might also want to clean up any alert data that has been generated since the alert was originally defined. You can use the **Clear Alert Data** option to remove alert information that is not connected to an alert definition.

About this task

When alerts are generated from an alert definition, details of the alert are stored in the database, and users are notified either within iBase or via email. When alert definitions are deleted, either directly, or automatically, the alert information is retained in the database, but can be removed when required.

For alerts that are no longer associated with an alert definition, using **Clear Alert Data** removes the following:

- iBase inbox notifications
- Information about the alert in the database

Procedure

1. In the Alerting dialog, select **Alert Definitions**.
2. Select **Clear Alert Data**. A message displays informing you of the consequences of this action.
3. Select **Yes** to confirm that you would like to proceed.

Receiving alerts

When an alert is generated for a Query or record that you are interested in, you are notified in the status bar. When you receive an alert, you can open the details, mark the alert to be followed up, or delete the information.

Alert status bar

When you have the alert status bar open, you are notified when an alert is detected.

Tip: To display the alert status bar, in the **Database Explorer**, right-click **Alert Definitions** > **Status Bar**.

The most recent alert is displayed in the status bar. You can click:

- The alert name to find out why the alert was raised.
- **Follow up** - to mark the alert for follow up in the Inbox.
- **Read** - to mark the alert as read in the Inbox.
- **Delete** - to move the alert to the Deleted Items folder in the Inbox.

To read or follow up the alerts later, click **Inbox**.

With the status bar open, you can set the frequency at which you receive new alerts. The alert frequency is initially the same as the server that manages alerting. You can change the setting to receive alerts more or less frequently, or immediately, if required.

Alerting inbox

To view all your alerts in the Alerting Inbox, in the Database Explorer, right-click **Alert Definitions** > **Inbox**. Alternatively, you can open the inbox from the Alerting status bar.

© N. Harris Computer Corporation

Alerts are either current and listed in the Inbox, or ready to be deleted and listed in the Deleted Items folder. Both views summarize what happened to raise the alert:

- Viewed - the number of times the records were viewed
- Edited - the number of times the records were opened, modified and then saved (rather than the number of changes made to the data in the records). Deleting a record also counts as an edit if the alert definition monitors edits only.
- Added - the number of new records or, if the alert was raised by a Query, that has been edited so that the records now match the conditions in the Query
- Removed - the number of records that were deleted or, if the alert was raised by a Query, that has been edited so that the record no longer matches the conditions in the Query
- Unsubscribed - as a result of security changes you are no longer subscribed to this alert definition or, if you are the owner of the alert definition, users that you added as subscribers are unsubscribed

Note: In a database that uses cases, the Inbox always lists all your alerts. In Multi-Case Analysis mode, you can view the detail of any of these alerts. When you are logged in to a single case, you can only view details of alerts that are relevant to the current case.

Details of alerts

When you are notified about an alert, you can open the alert details to discover more information. The records that triggered the alert are listed, along with the reason the alert was made.

Note: Security changes can mean that the records that are covered by an alert are no longer available to you. In this situation, the records are displayed without their icon, and no details are displayed.

The following details are available in the alert summary:

Views

A view is defined as an instance in which the records details are loaded. For example:

- Listing the record and then opening the details is counted as two views.
- Finding a record and then showing it counts as three views: it was viewed in the record list, again in the **Find** details pane, and again when the record is shown.

For details of who viewed a record, and when, select the record in the alert, and click **Views**.

Edits

An edit is defined as an instance in which the records details are changed. Notice that:

- Adding or removing a link counts as one edit.
- Editing several fields in the record in the same session counts as one edit.
- Clicking **Edit** and then canceling counts as a view and not an edit.

For details of who edited a record, select the record and then click **Edits**.

When you select an edited record, the details of the changes are displayed. If an entity or a link was added, the full record is shown, for edits, only the changed fields are shown.

Information shown	Description
Field Name	The old and current values for the fields in the entity or link.

© N. Harris Computer Corporation

Information shown	Description
New Value	The updated value.
Old Value	The previous value.
Edited by	The logon name of the user who made the change.
Date Edited	The date and time of the change (on the server).
Reason	If required by the database, the reason given by the user for making the change.
OS User	The Windows™ name of the user who made the change.
Machine Name	The machine that the user was working on.
Location	The location as entered in User Information, for example the team or department name.
iBase Change	This indicates whether the change was made within iBase or in an external system.
Extra Detail	Additional information for the current record.

Some information might be displayed that you do not usually see. In particular:

- Alternative icon representation shows the name of the new icon if a different icon is assigned to an entity type. This representation applies to entities that do not have an icon field.
- Icon Color shows the selected color shade if the default icon color was changed.
- Record Status is Normal unless the record is deleted (soft deleted and can therefore be restored) or Purged (permanently deleted).

When the change is a new, updated, or deleted link, the header, which is displayed in the blue band, summarizes the link details. For example:

- New Link: Shareholder[SIL31\GEN]
- Link Updated: VERMILLION Janet[PER:GEN\159]

For a new link, the New Value column lists:

- Entity type of the link end entities (Entity Type End 1 and 2)
- Link type
- Link direction and strength
- Record ID of the link
- Record IDs of the link end entities (Record ID End 1 and 2)

If a link is updated, only the changed fields are listed. For deleted links, only the changed status is listed.

© N. Harris Computer Corporation

Note: If the alert definition is monitoring a linked entity, changes to existing links are not reported because these changes are not directly changing to the entity.

Filtering by user

If several users edited or viewed the record within the time interval, then it can be useful to filter the details by user:

- From the **Show Edits** list, select the **`user name`**. Select by all users to display the full list again.

Reporting

Reporting collates the significant data in a database. The report can either be printed, copied to the clipboard, or output as a file.

The following file types are supported:

- HTML
- Rich Text Format (RTF)
- Microsoft™ Word document.
- Microsoft Access database.

There are two types of report that can be produced:

- Standard Reports - Each report covers a single entity type and can include none, one or several link types.
- Link Reports - Each report covers a single link type and can include none, one or several entity types.

Reports always show a snapshot of the data at the time the report is created.

Before you can produce a report, you need to define its contents and appearance in a report definition. Since you might need to create the same type of report many times, for example a weekly status report, report definitions can be saved and used to rerun reports as required. To speed up the preparation of reports, you can set a particular report definition as the default for a specific entity type.

Note: When data is added to report, it is no longer under any form of access control. It becomes your responsibility to ensure that access to any restricted or secret information is controlled in an appropriate way.

Producing reports

You can produce a report on entities and links in the database. Before you can produce a report, you might need to define its contents and appearance in a report definition.

About this task

When you produce the report, you specify the scope of the report and how the report is output. You can also define sets or queries first if you want to report on specific records, rather than on all records in the database. The available output formats for reports are:

Standard Report

Produces the report as a PDF document that opens in your PDF viewer.

Note: The PDF is a temporary file, it has no specified save location.

Microsoft Access

© N. Harris Computer Corporation

The report is saved in a Microsoft™ Access database.

Using this format, you can create reports for several different entity types, and then combine them in Access. Each iBase report creates a new table in Access; you need to use some of the facilities in Access to make your report more usable.

- **Update existing databases**

  Turn on **Update existing database** to update an existing Microsoft Access report (.mdb) file rather than create a new database. Browse for the existing database file. If you select this option, when you click **Finish** you can either:

  - **Refresh the data in the selected Microsoft Access report database with that generated by the report**

    The selected report database must contain all the necessary fields and tables for the report definition.

  - **Extend the Microsoft Access report database to include the report data**

    This option adds tables to the selected report database to generate reports that contain different but related data in a single report database.

Microsoft Word

The report is saved in a Microsoft Word document. The following options are applicable:

- **Open report when complete**

  Turn on to direct the document to open as soon as it is generated.

- **Use Styles in this template**

  To use the styles in a Microsoft Word document rather than the fonts defined in the report definition, turn on **Use Styles in this template**, and select a document from the list. When this checkbox is turned off, the report uses the formatting that is specified in the report definition.

- **Use template content to define report style**

  If you specify a Microsoft Word document or template, you can also turn on **Use template content to define report style** to apply the sections and table layouts in the template to the report.

Hyper Text Markup Language (HTML)

The report is saved as an HTML file (`.htm` file) and can be viewed in a web browser. It is also displayed in the iBase report viewer (but the viewer displays the path to the graphics file instead of the actual picture).

Rich Text Format (RTF)

The report is saved as a rich text format file (`.rtf` file). It can be displayed in the iBase report viewer, where you can print it or copy pages.

Note: This format includes icons but not pictures.

Procedure

1. Open the Report wizard:
   - In the toolbar, click **Create Report**.
   - In the **Database Explorer**, right-click an entity type, and from the menu, select **Report**.
   - In a record list, right-click a record, and from the menu, select **Create Report**.

© N. Harris Computer Corporation

2. Select the report definition and the data:
   - Select a report definition from the list. Some details of the report definition are displayed below the list:
     - The name of the main entity type, the subject of the report (you can only report on one entity type at a time)
     - Whether this is the default report definition for the main entity type
     - The number of link types included in the report definition (if any)
   - In the Source area, specify which main entity records you want to report on:
     - **All records** to report on all records for the main entity type.
     - **Single Record** to report on a single record
     - **Query** to report on the results of a selected query. You can apply query conditions and then filter the linked items. For example, your query links Accounts to People with an "Account Holder" link. Set a condition on the query that Person First Name starts with J. When you run the report with **Filter linked items using query** checked, only links that meet the condition are reported. You can uncheck the option to report any links associated with the main entity type.

       Note: You can only select from queries that output records of the main entity type.
     - **Set** to report on the records that are contained by a set, which you select. If you turn on **Only linked items contained within the set**, then those items are reported. Other items linked to the main entity type are not reported, unless they have already been added to the set. For example, if your set contains 5 people and you are reporting on people linked to accounts, the accounts are not reported unless they are part of the set.

       Note: You can only select from sets that contain records for the main entity type.
   - Click **Next** to continue.
3. Specify the title and output format:
   - The **Report Title** displays the default title as defined in the report definition.
   - Select the output format and enter a path and file name.
4. Click **Finish** to generate the report.

Reporting in HTML

You can share your iBase reports using your organization's intranet, or the internet. To do this, you need to have your data in HTML (HyperText Markup Language) format.

You can specify some limited formatting of the report data using the options available in the report definition.

If your report includes pictures, they are associated with the report using links; to make the report viewable on other computers (portable) you need to save the report using Internet Explorer, while iBase is still open, to establish the links correctly.

Note: To take full advantage of the features of HTML, you need to understand how to work with HTML using a specialist HTML editor or a text editor such as Notepad.

Creating a Report for the Web

1. On the first page of the report wizard, enter the details of the report in the usual way.

© N. Harris Computer Corporation

2. Click **Next** to display the second page and, in the Output to area, select Hypertext Markup Language (HTML) and then specify a file name and location.
3. Click **Finish**.
4. Click **Close** to close the Report Viewer, and **Close** to close the wizard. Do not close iBase at this stage.

You have created an HTML file containing an iBase report, and any icons and pictures have also been saved as separate graphic files. Using a text editor, or any specialist HTML editor, you can now add text and formatting using all the options that are available in HTML.

Note: To view any icons in the report, you must view the report on a machine on which iBase is installed. You cannot view any pictures. To establish links between the HTML file, the icons, and the pictures follow the steps below.

Making sure that icons and pictures remain linked to the HTML file

All your linked graphics are automatically saved in a separate `<filename>_files` folder, where `<filename>` is named after your HTML file. If you move or copy the HTML file, you must also move or copy the folder containing the graphics.

Reporting in Microsoft Access

You can create a report from your iBase database in a Microsoft™ Access database format. You can use Microsoft Access to create reports with more than one main entity type.

Why use a Microsoft Access database as a report format?

You might choose to report to a Microsoft Access database for any of the following reasons:

- iBase reports can be combined to allow reporting on more than one main entity. For example, a report on telephones, might also report on the subscribers of the phones involved.
- To allow iBase data to be combined with data from other systems; Microsoft Access can incorporate data from a wide variety of formats: Extensible Markup Language (XML), OLE DB, and Open Database Connectivity (ODBC).
- Microsoft Access data can be used by specialist database report tools.

When you choose a Microsoft Access database as the format for an iBase report, the output is as follows:

- Each main entity, link, or linked entity has a separate Microsoft Access database table.
- Each record that you include becomes a row in the relevant database table.
- Each field that you specify in the report definition becomes a column in the relevant database table.
- Any formatting that you specify as part of the report definition, for example fonts or highlighting, is ignored.
- Images and documents are not included in the output.

Note: When you export to Microsoft Access, any duplicate fields have a '2' added to the end of the field name. This occurs if there are duplicate fields in a link type and its link end entity type.

Creating a Microsoft Access database report

1. In the report wizard, enter the details of the report.

© N. Harris Computer Corporation

2. Click **Next** to display the second page and in the Output to area, select Microsoft Access and then specify a file name and location.
3. Click **Finish**. When the report is complete, click **Close**.

Viewing the Microsoft Access database

To view the database, start Microsoft Access and open the database that you created. Double-click the table that is named after the main entity in your report. For example, if you selected Telephone as your main entity, double-click Telephone_ to display the contents of the Telephone_ table.

Reporting on an extended selection of entity types

One of the reasons that you might choose to create reports in Microsoft Access is to extend the selection of the entities that you report upon, beyond one link distance. For example, in a report on calls by a particular telephone; being able to report on entities one further link away, means you might report on the subscribers too.

Reporting on different entity types requires that you create separate report definitions and use them to create reports, these reports can be generated into the same Microsoft Access database.

Note: Microsoft Access reports from iBase do not include all the relationships, these must be added manually.

Linking tables and creating relationships in Microsoft Access

1. Generate the first report and output to a new Microsoft Access database.
2. Generate the second report and output the report to the same Microsoft Access database, and select **Update existing database**.
3. Select **Extend the database** to include the report data.

   The Combined database now has one table for each of the entity types in your reports.
4. Create relationships between the tables that originated from the reports:
   a. Select **Tools** > **Relationships**. A relationship might exist between the MainEntity_ID fields in related tables.
   b. For each entity type, select a field relevant to your report in one table and drag it to the corresponding field in the related table. You can use the record IDs to help identify the relationships that need to be created.
   c. Click **Create** to make the relationship.

Defining a report

Before you can produce a report, you need to define the content and appearance of the report in a report definition. You select the report definition to produce the report.

Procedure

To define a report definition:

1. Select **Format** > **Report Definitions** > **New**.
2. Select the entity type.
3. Click a tab to define the report details and output format.
4. Save the report definition.

© N. Harris Computer Corporation

5. Click **Create Report** to test the report definition.

Page setup (General tab)

The General page of the Report.

Title

Click the Title box and enter, or edit, the text for the title. Click Font to select a different font for the title.

Header, Footer

Click the Header or Footer box and enter, or edit, the text for the header or footer. In the Header area, click Font to select a different font for both the header and the footer.

Footer Logo

Turn on Footer Logo if you want a graphic to print in the left end of each footer. Browse to the graphics file or enter the path.

Note: Graphics that do not have a 4:6 aspect ratio are stretched to fit a rectangle. To avoid this, edit the logo by adding a background of the correct size.

Truncate long text fields to

You can limit the values to a specific number of characters (in multiples of one hundred) by turning this on.

Default font

Displays a preview of the font for the main body of the report. To change this, click Font and select a font. When you output the report to Word, this font becomes the font for the Normal style.

Orientation

Set the page orientation by clicking Portrait or Landscape.

Page break after each entity record

Turn on if you want to start a new page for each section for the main entity.

Indent Links

To help the link records to stand out better, turn on **Indent Links**.

Default Microsoft Word template

Turn on **Default Microsoft Word** template and select a document (*.docx) for use as a template when outputting the report.

Only documents in the same folder as the database file, with a name starting with the database name are available. For example, if the status bar shows the open database as being C:\Databases\Vehicles\Vehicles.idb, then you could use C:\Databases\Vehicles\Vehicles Template 1.docx as a template.

Turn on **Auto-size List Tables (Microsoft Word)** to allow the width of the List table in the report to be automatically adjusted based on its contents, rather than use the table width set in the template.

© N. Harris Computer Corporation

Selecting entity fields

You can select the entity fields that you want to include and their order on the report. If required, you can highlight the column headings and the data from selected fields in order to make important fields easier to identify on the report.

Procedure

1. Click the Entity tab and then the Fields tab.
2. Select the fields that you want to include in the report, and whether there should be highlighting:
   a. Turn on the checkbox next to a field to include it on the report.
   b. While the field is still selected, turn on **Highlight Label** if you want to highlight the heading on the report (in bold for example).
   c. While the field is still selected, turn on **Highlight Data** if you want to highlight the field value.

   Note: You can change the highlight style if you output the report to Word.
3. If required, change the order in which the fields will appear in each record, printed on the report, by selecting one or more fields and then clicking the up or down arrow key buttons.

Selecting the link types and link end entities

Use the Links page to select the link types and link end entity types to include on the report. For each link type/end entity type combination that you select, you must include at least one field. In the report, these records appear in a sub-section of their main entity record section, with an optional title.

Procedure

To add a link type/link end entity type to the report definition:

1. Click the **Links** tab to display the Links page.
2. Click **Add** to add a link type/linked entity type combination. The Add Link dialog is displayed:
   - From the **Link** list, select a link type - only valid link types for the main entity type are listed.
   - From the **Link End** list, select a link end entity type - only valid entity types for the main entity type/link type combination are listed. This means that when you run the report only records for the selected main entity type, link type, and end entity type records will be added to the report.
3. If required, enter some text in the Title box to describe the relationship of the entity type records to the main entity.
4. Click **OK**. You are warned to select at least one field for the link type and end entity type combination that you have just added.
5. Set the sort order for the link type and link end entity type combination, and select the formatting style.

   Note: To indent the link records under the main entity record, go to the General page and turn on the Indent Links checkbox.

© N. Harris Computer Corporation

Sorting the records

Use the Sorting and Formatting page to specify how the records are sorted on the report - there are three levels of sorting.

About this task

This example shows sorting in ascending order (that is A before B):

Primary	Secondary	Tertiary
A	A	A
A	A	B
A	B	A
B	B	B

Procedure

To define how the data is sorted:

1. Click the Entity tab or the Links tab, and then the Sorting and Formatting tab to display the Sorting and Formatting page.
2. Turn on or off the sorting options:

   Sort by

   In the Sort by drop-down list, select the field that you want to sort by.

   Records in the report are sorted first according to their value in this field, in either ascending or descending order:

   - Click Ascending to sort a before z, and 1 before 9.
   - Click Descending to sort z before a, and 9 before 1.

   Then by

   In the Then by drop-down list, select the next field that you want to sort by. If there is no suitable field, select <None> from the drop-down list.

   Records that are placed in the same position by the primary sort are then sorted according to the field that you select here.

   Then by If required, in the second Then by drop-down list, select another field to further sort the records or select <None> from the drop-down list.

   Records placed in the same position by the primary and secondary sorts are then sorted according to the field that you select here.

© N. Harris Computer Corporation

Formatting the records

Use the Sorting and Formatting page to specify the layout of the individual records on the report.

Procedure

To define the format for individual records:

1. Click the Entity tab or the Links tab and then the Sorting and Formatting tab to display the Sorting and Formatting page.
2. For entities only - if the field details are insufficient to identify the entity, you might want to turn on **Show entity record labels** on the Entity page. This adds the record label as a title to the beginning of each section of the report. The record labels use the same font as the body of the report. To change this, click Font and select a font.
3. In the Format area, specify how records are to be laid out on the report:

Option	Description
Tabular	The tabular format uses a row for each record and a column for each field. This is useful when you are reporting on many entity records but only a few fields.
List	The list format uses a subsection per record and a row per field. This is useful when you are reporting on few entity records but many fields.
Include Icon	In the list format only, turn on this checkbox if you want to place the icon for a record at the left end of the record.
Include blank data values	In the list format only, turn on this checkbox if you want to print a row for a field, even if the value is blank. (Note: Does not apply to time-based fields.)
Include 'No' data values	In the list format only, turn on this checkbox if you want to print a row for a Yes or No type field, even if it has a No value.

Note: There are also some formatting options on the General page, for example to insert a page break before each entity record, or to indent the link records.

Default report definition for the entity type

To make the current report definition, the default report definition for the main entity type, turn on **Default report definition for this entity**.

If this option is turned on, you can quickly generate reports. For example, right-click an entity type and select **Create Report**.

© N. Harris Computer Corporation

Document templates

You can use templates to produce consistent reports in your chosen style. The easiest way to do this is to create a simple report and change the settings, formats, and styles for your requirements.

About this task

Template files can be saved as a Microsoft Word document or template, in the same folder as the iBase database. The file name must start with the name of the database for it to be recognized as valid.

You can use a template to create standard reports. If you want to make a change to the format of the standard report, simply change the document or template and all subsequent reports will incorporate the changes. Microsoft Word documents and templates can also include standard text and graphics.

Procedure

To create a template:

1. In the report wizard, enter the details of the report in the usual way. Set up a simple report based on a single entity type and the required link types.
2. On the second page, in the Output to area, select Microsoft Word and then specify a file name and location. You can choose any location and file name that has the .docx file extension.
3. Turn on **Open report when complete** to specify that you want to start Microsoft Word as soon as the report is ready.
4. Click **Finish**. Microsoft Word is started and the report is displayed. This report has very basic formatting but is a suitable basis for setting up the document for use as a template.
5. Copy the document into the same folder as the iBase database file (*.idb). Check that the file name starts with the database name.

Folder for documents used as templates

To be used as template, documents must be saved in the same folder as the database, with a file name that starts with the database name, and the .docx file extension. There is nothing to prevent you from saving the file with a .dot file extension but the report wizard uses only *.docx files.

For example, if the status bar shows the open database as being:

```
C:\Databases\Vehicles\Vehicles.idb
```

then, you might name a document:

```
C:\Databases\Vehicles\Vehicles Template 1.docx
```

Note: After a document is added as a template, close then re-open the report wizard or report definition to refresh the list.

Template styles

A Microsoft™ Word report is based on a number of special styles. You can modify each of the styles to change features such as fonts, paragraph spacing, and indentation. This gives you finer control over the styles than can be achieved with the normal report definition.

Styles are arranged in a hierarchy in iBase reports to allow you to make changes at the highest level possible so that changes are reflected in all lower levels. For example, the default font that is specified in the report definition is used to set the font for the Normal style. All other styles are based on Normal so that you can change the font for Normal and that change will be reflected throughout the report.

© N. Harris Computer Corporation

Changes can be made to all entity data or to all linked data, however you can also change the format of any individual component of the report.

iBase paragraph styles

iBase creates the following styles:

Style	Description
Normal	Setting the default font for the report, changing this style affects all the iBase styles in the report.
iBase Report Header	All header text
iBase Report Footer	All footer text
iBase Report Title	The title from the report definition
iBase Entity Record Label	Record labels, if included
iBase Entity Field Label Regular	Regular default field labels
iBase Entity Field Label Highlight	Highlighted field labels
iBase Entity Field Data Regular	Default field data
iBase Entity Field Data Highlight	Highlighted field data
iBase Link `N` Title	Title for link `N`
iBase Link `N` Field Label Regular	Default link field label for link `N`
iBase Link `N` Field Label Highlight	Highlighted link field label for link `N`
iBase Link `N` Data Regular	Default link field data for link `N`
iBase Link `N` Data Highlight	Highlighted link field data for link `N`

Where `N` replaces the number of the link. For example, the style for the title of the first link section is iBase Link 1 Title, for the second link that is specified, the title style is iBase Link 2 Title.

Special styles for link data

There are some other special styles that are not used directly but allow you to control the styles for link data at a level above the individual numbered styles. If for instance, you want to change the style of all the linked entity field labels, you can change the style iBase Link Field Label Regular, and the change will be applied to all of the iBase Link `N` Field Label Regular styles.

The special styles are:

© N. Harris Computer Corporation

Style	Description
iBase Link Field Label Regular	Default link field label for all links
iBase Link Field Label Highlight	Highlighted link field label for all links
iBase Link Field Data Regular	Default link field data for all links
iBase Link Field Data Highlight	Highlighted link field data for all links

Microsoft Word document properties

In some places in your report, you may want to use Microsoft Word Fields to include information about the report; the following information is included in the document properties automatically:

Property	Description
Title	The Report Title.
Subject	The Main Entity Type in the report.
Author	Your iBase login name.
Comments	Generated by iBase on dd mm YYYY using report definition 'name'.

Select the whole report and update fields in Microsoft Word to make sure that the information is up-to-date (press Ctrl and A, and then press the F9 key).

Formatting a Microsoft Word document as a template

How to format a Microsoft Word document as a template for use with iBase.

Procedure

1. In Microsoft Word, format the document as required. For example in Microsoft Word 2003, click anywhere in the paragraph and, from the Format menu, select Styles and Formatting. Modify the displayed style and apply the change.
2. If required, tidy up the content of the report. For example, you might want to remove details of specific records and add some text in its place to demonstrate each of the Microsoft Word styles used by iBase. Users are able to display this document in the report wizard.
3. From the File menu, select Save As and save the document in the same folder as your database. The file name should start with the name of the database.
4. Test that the Microsoft Word document functions correctly as a template by generating a report from it.
5. When you are satisfied that the document is correct, you can make it into the default Microsoft Word template for one or more report definitions. You do this by selecting it on the General page of the relevant report definitions.

© N. Harris Computer Corporation

Note: If you want different parts of the report to have a different layout, you can insert section breaks. This allows each section to have its own formatting and page setup; for example you can have some sections of the report with a portrait page layout and some sections in a landscape layout, if the data is particularly suited to such presentation.

Extra content in Microsoft Word reports

Documents used as templates can contain more content and styling in addition to the iBase paragraph styles.

Documents used as templates can contain:

- Logos and other graphics
- A title page
- Details of how the report was created.
- A table of contents, typically added to the title page.
- An end page
- Table formatting

The different parts of the report are defined using section breaks. This type of template is specific to a single report definition because, the number of sections must match the number of link types, and the selected link style (whether tabular or list).

Section breaks

Section breaks in the template are used to separate the parts of the report. The sections are as follows:

Section	Description
Title	Contains a piece of text in the Microsoft™ Word style iBase Report Title, plus any other items you want, such as a logo or a table of contents. This section is mandatory.
Main Entity	Repeated for each main entity in the report. The section starts with a table that contains the icon (if specified) and the entity label in the Microsoft Word style iBase Entity Record Label. It is followed by a table with the entity field contents. This section is mandatory.
First Linked Entity	There is one section for each of the linked entity definitions and these are repeated for each main entity. Not required if there are no linked entities.
Second Linked Entity	The same as for the First linked entity, for however many linked entities that you specify .

© N. Harris Computer Corporation

Section	Description
End	After the last section break there is an area where you can add extra text such as completion details or a description of how the report was produced.

There are three types of section break available in Microsoft Word and you can use any of these to separate the sections of your report:

- Next page inserts a section break and starts the next section on the next page.
- Continuous inserts a section break and starts the next section on the same page.
- Odd page or Even page inserts a section break and starts the next section on the next odd-numbered or even-numbered page.

Formatting a document as a template

The Microsoft Word document that you create should contain empty tables as placeholders for the data from iBase - all iBase data is displayed in Microsoft Word tables. For example, to force the entity record label to appear in the document, there should be a single cell table at the correct position that uses the Microsoft Word style iBase Entity Field Label Regular.

The simplest way to determine the correct position and style is to use an existing Microsoft Word report:

1. In iBase, generate a report from the required report definition. Set up a simple report based on a single entity type and the required link types.

   Note:

   - Select tabular or list style for the link data before setting up the document as a template - changing this setting requires you to set up a new document.
   - There is no need to enter a report title - the report will always use the title that is entered in the document.
2. In Microsoft Word, format the document as required.
3. Tidy up the tables in the report that will hold the data:
   - Clear the row and heading cells in the tables so that the cells are empty.
   - Check that the tables are wide enough for their eventual contents.
   - You only need one row per table if using the tabular style.
   - Separate the tables in the template with at least one paragraph.
   - Format the tables using the formatting facilities in Microsoft Word (from the Table menu, select Table Designer) to modify the properties.
4. From the File menu, select Save As and save the file in the same folder as your database. The file name should start with the name of the database, and have a .doc file extension (the file you have created is not a Microsoft Word Document Template).
5. Test the styles in the document or template by generating an iBase report from it.
6. The next step is to insert the section breaks, as described below.

Inserting section breaks

Before you can add additional content to a Microsoft Word document, you need to define the sections in the report by inserting section breaks:

© N. Harris Computer Corporation

1. In Microsoft Word, open the required document; it will be located in the same folder as your database.
2. Click the **Show** or **Hide** button on the Standard toolbar to display text markers and section breaks. Paragraph markers will be shown. From the View menu, select **Normal**.
3. Enter the section breaks.
4. Click on the paragraph marker at the position where you want to insert the section break - the break is inserted above the paragraph marker you select:

To insert a...	**Click on the...**
Title section	paragraph marker between the title and the first table
Main Entity section	paragraph marker below the table containing the entity data
First Linked Entity section	paragraph marker below the table containing the linked entity data
End section	This is created automatically when you insert the last section break.

5. From the Insert menu, select **Break**. The Break dialog is displayed.
6. In the Section break types area, select the type of page break and click **OK**. A section break is inserted.
7. Save the file.
8. Test the Microsoft Word document by producing an iBase report from it.

Adding material at the end of the report

In a Microsoft Word template with additional content, the End section of the report, the part that is after the final section break, does not contain any data from the report. You can use this area to add any details you want. For example, the document properties that are automatically generated when you produce a report:

1. In Microsoft Word, select **Insert > Field** from the Insert menu, select **Field**.
2. Select the Categories drop-down list, select Document Information.
3. Insert the fields which are updated by iBase as required:
   - Author
   - Subject
   - Comments

To locate the end section of the report, click in the paragraph below the final section break.

Using a document with extra content to define the report

1. In the report wizard, enter the details of the report in the usual way.

   Note: There is no need to enter a report title as this is taken from the Microsoft Word document.

© N. Harris Computer Corporation

2. In the Output to area, select Microsoft Word and then specify a file name and location. You can choose any name and location.
3. Turn on the Open report when complete checkbox to specify that you want to start Microsoft Word as soon as the report is ready.
4. Turn on the Use styles in this template checkbox and select the required Microsoft Word document from the drop-down list. This will apply the paragraph styles from the document.
5. Turn on Use template content to define report style to specify that you want to also use the contents of the document. This will apply the section breaks, table formatting, and content such as graphics.
6. Click **Finish**. A progress dialog is displayed, followed by a completion dialog.

   Microsoft Word is started and the report is displayed.
7. If the template contains any fields or a table of contents, you need to update them: Press `Ctrl + A` to select the whole report, and then press the `F9` key to update all fields.
8. Select Update entire table and click **OK**. All the fields are now updated.

Report definitions

The report definition list displays creation and update information. You can sort the report definitions by clicking the required column heading.

To add a report definition, click **New**.

To administer the report definitions for your database, right-click on a definition and then select:

Open

To edit the selected report definition; you can also double-click the definition.

Rename

To rename the selected report definition. If this report definition is set as the default for an entity type, then it remains the default even after you rename it.

Save As

To save a copy of the selected report definition under a new name. The name must be unique not just to the category folder but to the whole database.

Note: Saving a copy of a report definition that is set as the default for its entity type results in two defaults. The first report definition (in alphabetical order) is used. Therefore, you may want to turn off the Default report definition for this entity checkbox.

Delete

To delete one or more selected report definitions. There is no warning if you delete a default report definition.

Categorize

To move one or more selected report definitions, which might be in different categories, to another category, or set access controls on them.

Properties

Show 'system' type properties of the report definition, such as the date it was created and the name of the user who created it. Only available when a single report definition is selected.

Note: You can also list, and work with, report definitions in the Database Explorer detail window.

© N. Harris Computer Corporation

Link reports

Link reports are reports that are able to be sorted by using link fields, and can include the information that is entered on either linked entity. The difference between a link report and a standard report is that although standard report can include links, the report cannot be sorted by fields that are placed on the link.

As a default, link reporting is not turned on, but can be initialized by a database administrator if required.

Initializing the database for Link Reports

To Initialize the database for link reports, select **Tools** > **Initialize Database for Link Reports**.

Note:

- You must be logged in with an Administrator account to initialize the database for link reports.
- When the reports are enabled, **Initialize Database for Link Reports** is longer displayed under **Tools**.
- Link reports are supported on all database types.

Managing Link Report Definitions

The report definition list displays information on when the report definitions were created and updated. You can sort the report definitions by clicking the required column heading.

To add a report definition, click **New**.

To administer the report definitions for your database, right-click on a definition and then select one of the following options from the menu:

Option	**Description**
Open	To edit the selected report definition; you can also double-click on the definition.
Rename	To rename the selected report definition.
Save As	To save a copy of the selected report definition under a new name. The name must be unique not just to the category folder but to the whole database.
Delete	To delete one or more selected report definitions.
Categorize	To move one or more selected report definitions, which might be in different categories, to another category, or set access controls on them.
Properties	Show 'system' type properties of the report definition, such as the date it was created and the name of the user who created it. Only available when a single report definition is selected.

© N. Harris Computer Corporation

Note: You can also list, and work with, report definitions in the Database Explorer detail window; the lower half of the Database Explorer.

Exporting

You might need to share data from your iBase database with others. You can export entity and link records, in a format that meets the requirements of the target application.

You can export to any of the following file types and applications:

Text file (ASCII)

You use an export specification to define how the exported data is formatted. For example, you might define which entity or link type to export and how dates and numbers are formatted. If you regularly export data, you can save your settings in an export specification. You can also run a series of export specifications together by using an export batch specification to export a number of records, perhaps of differing entity or link types.

XML file

You use a database subset definition to define the entities and links to export.

Microsoft™ Excel

You export the selected data to a new spreadsheet that is automatically created for you. You can export multiple entity and link types.

Other Microsoft formats

You can use report definitions to produce reports in these formats:

- HTML (Hypertext Markup Language).
- Rich Text Format (RTF)
- Microsoft Word
- Microsoft Access

Exporting using an existing specification

You might need to share data with others, by exporting the data to a file that can then be imported into another application or into a different iBase database. The type of data to export is defined in an export specification. There is one export specification for each entity and link type that is involved in the export.

Procedure

1. In the Database Explorer view, right-click **Export Specifications** and select **List**.
2. Load the required export specification.
3. Click **Next** to review the contents of the export specification:
   a. Step 1, which entity or link type is exported.
   b. Step 2, the format of the export file, and in particular how dates, times, and numbers are handled.
   c. Step 3, which fields for the entity or link type are exported.
4. If the definition of the export specification is appropriate, select the data to export:
   - All Records - all records of the entity or link type are exported.
   - Query - the results of a query are exported.
   - Set - records that are contained in a set are exported.

© N. Harris Computer Corporation

5. Click **Run**. The number of records that are exported is displayed.

   Tip: You can then click **Back** to go back and export more data with the same export specification, for example from a different set or query.

Setting up for exporting data

Before you can export from iBase, you must define the type of records and their fields to export, and the format of the file to be created. You define this information by setting up an export specification, each export specification defines how to export data for a single entity or link type.

About this task

The export wizard is a series of pages that leads you through the process of creating an external file that contains data from the database.

Procedure

1. Select **File** > **Data** > **Export**.
2. If you want to create a new export specification, select the type of data to export:
   - To export entities, click **Entities** and from the Record Type list, select the entity type.
   - To export links, click **Links** and from the Record Type list, select the main link type. In the Link Ends area, select the link end entity types from the End (1) and End (2) lists.

   Note: You must export entities before links.
3. Click **Next** to continue. (In later steps you can click Back to return to previous steps if you want to change any settings.)
4. In **Export File Name**, enter the path of the export specification that you are creating or navigate to the folder, enter the file name and click **Save**.
5. Select **First Record Contains Field Names** to write field names into the first record in the export file.

   This is useful as a reminder of what the fields are when you browse the exported file. It also allows you to automatically assign source fields to iBase fields if you later import the file back into an iBase database.
6. Enter the character to use to enclose the field values in the export file in the **Text Qualifier**. Each value is bounded at each end by this character. If you do not want a text qualifier, delete the displayed character.

   When reimporting data, this prevents field values that contain field delimiters from being split into two.
7. In the Field Delimiter area, choose the character that is used to separate fields, or enter a different one in the Other box.

   Note: The record delimiter is an ASCII CR and LF character; this puts each new record on a new line.
8. The Dates, Times & Numbers area shows you how any dates, times and numbers will be exported. To change these formats, click Format.
9. Select the fields you want to export, and the order in which they appear in the export file.
   - **Entity and link fields**

     Entity and link fields are shown by their field name.

© N. Harris Computer Corporation

- **Fields for link end entities**

  Fields for link end entities are indicated by <(1)> for the End 1 entity and <(2)> for the End 2 entity.

- **<Direction>**

  If present, this stores the direction of the link, and is exported as codes:

  - 0 = no direction
  - 1 = End (1) to End (2)
  - 2 = End (2) to End (1)
  - 3 = Both

- **<Strength>**

  If present, this stores the strength of the link, and is exported as codes:

  - 0 = unconfirmed
  - 1 = confirmed
  - 2 = tentative

10. In the Records area, specify the source of the records to export, provided they are the appropriate type.

    Note: You must assign an *output* for queries that you want to export, or they are missed in the export specification. Entities and links that have an output assignment are flagged with a green square icon in the query structure. To assign an output, open your query and highlight an entity or link that does not have an output assignment. Right click the item and select **Output** from the available options. Repeat for every item that you want to export, and save the query. See steps 1 and 4 of the export wizard to check that your query is included in the export.

11. Click **Run**. The number of records that are exported is displayed.

    Tip: You can then click **Back** to go back and export more data with the same export specification, for example from a different set or query.

Exporting to Microsoft Excel

You may want to disseminate information from your database to colleagues who do not themselves have a database application, however they will often have access to Microsoft™ Excel.

Procedure

To export data to Microsoft Excel:

1. Select **File** > **Data** > **Export Data to Excel**.
2. Select an entity or link type from the list.
3. Specify which records to export.
4. In the **Fields to Export** area, turn on the checkboxes next to the fields that you want to export, turn off the checkboxes of fields that you do not want to export.
5. In the Totaling Functions area, turn on the checkbox next to the Microsoft Excel function that you want to apply to the data. The functions only apply to numerical fields. The functions are:

© N. Harris Computer Corporation

Function	For each number field returns...	Microsoft Excel function
Lowest	The lowest value	MIN
Highest	The highest value	MAX
Average	The average value	AVERAGE
Sum of	A total	SUM
Standard deviation	The standard deviation	STDEV
Mode	The most frequently occurring value	MODE

6. Click **Export**.

Exporting data as XML

You can export data to and from an XML file by using the iBase XML schema. The iBase XML schema describes the structure of your database.

About this task

There are two types of XML schema that can be used to export data:

- iBase specific schema - Allows database subsets to be exported. These subsets can contain any combination of entity or link types. Allows a single entity or link type to be imported from an XML file.
- MS Rowset schema - Allows a single entity or link type to be imported or exported as an XML file that contains the schema information. The exporting of database subsets, which can contain multiple entity or link types, is not supported.

The iBase XML schema is designed to correspond to the entities and links in your database and can be viewed by generating an XSD file from iBase Designer.

Use the iBase XML schema instead of the MS Rowset schema because an XSD file that describes the iBase XML schema can be generated using iBase Designer. The XSD file can be used to share XML information with 3rd parties and internally within your organization.

When exporting using an iBase XML schema, you can select the data to export using queries and sets and save the selection as a database subset definition. You base the export on this definition. This feature is not available when exporting to MS Rowset schema.

There are some restrictions on using an iBase XML schema:

- The database supports Unicode - to find out whether Unicode characters are supported, select **File > Properties > Database Properties**. The **Use Unicode Data types** checkbox is turned on if Unicode characters are supported.
- The database is initialized for database subsets - there is a folder for database subset definitions in the Database Explorer if the database has been initialized.
- You have selected the data to export by defining a database subset definition.

© N. Harris Computer Corporation

Note:

- You cannot export pictures or documents to XML using the iBase XML schema.
- When you export records from a case, the case name is exported as the SC code.

Procedure

1. Select **File > Data > Export Data to XML > Create XML File**.

   Note: The **Create XML File** menu option is only available when using a SQL Server Database.
2. Browse for the folder where the XML file will be created.
3. Browse for the database subset definition that defines the records you want to export.
4. Click **Export**.

Removing

Depending on your permissions as a user, you can remove entities and links from the database either individually or in batches. When you delete an entity, you also delete all links to that entity but you do not delete the link end entities. Deleting records is a permanent and irreversible operation unless soft delete is enabled for your database.

Depending on the way that your database is configured, you might have soft delete enabled. This means that when you delete records, individually or in batches, you have an opportunity to undo the deletion. Soft deleted records do not appear in search results or when you list and browse records. If soft delete is not enabled, then deleting records is a permanent and irreversible operation.

Finding out whether soft delete is in use

To check the setting of soft delete:

1. Select **File** > **Properties** > **Database Properties**.
2. Click the Advanced tab and view the setting of the Soft Delete checkbox (soft delete is in use if the checkbox is turned on).

Deleting single records

To remove an individual record, click Delete or press Ctrl+D. If you are deleting an entity,, then any links from that entity are also deleted.

Note: Your database designer might have restricted permissions for deletion.

Deleting batches of records

Batch Delete, if you have access to it, removes multiple records of a specific type, for example as held in a set or the results of a query. You can only undo a batch deletion if soft delete is enabled for your database.

Restoring and purging deleted records

You can undo the deletion of specific records by using Restore Deleted Records. This is only available if soft delete is enabled on your database. To make the deletion permanent, purge the deleted records.

Note: To undo a purge, you have to restore the records from a backup.

© N. Harris Computer Corporation

Deleting batches of records

You can use the Batch Delete function to delete multiple records of a specific entity or link type, or you can delete all the items types from a query or set.

About this task

For the specific entity or link type, you can delete:

- All records.
- All the records in a particular set (the set itself is not deleted).
- All the records of the specific type that are found by a particular query.

For queries and sets, you can choose to delete **All item types**. This can save you time if you have multiple item types in the query or set.

If soft delete is enabled on your database, then deleted records can be restored if necessary, that is until a purge operation makes the deletion permanent.

Attention: Deleting an entity also deletes any link that has that entity as one of its ends. However, if a link is subject to data access control or database restrictions, you might not be able to delete the entity at either end of that link. For more information, see the section on Batch Delete and Data Access Control in Batch delete.

Before you begin

Check that you have access to the Batch Delete command. If you do not have access, contact your system administrator.

Procedure

To delete batches of records:

1. Select **Edit** > **Batch Delete**.
2. Select the entity or link type of the records that you want to delete, or choose the **All item types** option to delete all the item types from a query or set.
3. In the Source area, specify the source of the specified entity or link type records.
4. Click **Delete**.
5. Click **Yes**.

   Tip: To cancel the delete, press Esc. The delete stops and a message displays how many records have already been deleted. The behavior when canceling a batch delete is different depending on the type of database you are using and the Audit Level set:

Database type and Audit level	Behavior when canceling delete
SQL Server - 1,2, or 3	No records are deleted.
SQL Server - 4 or 5	Records up to the point when the batch delete is canceled are deleted.
Microsoft Access - Any	Records up to the point when the batch delete is canceled are deleted.

© N. Harris Computer Corporation

If you are deleting a large number of records the deletion can take a while to complete. When the deletion completes, you are shown the number of records that have been deleted.

Restoring soft deleted records

You can undo the deletion of entities and links by restoring them. Restoring records reverses the deletion of records that have not been purged. **Restore Deleted Records** is only available if soft delete is enabled for your database, and if your system administrator has given you access to it.

Procedure

To restore deleted records:

1. **Edit** > **Restore Deleted Records**.
2. Select **Select a range of records** to search for records that were deleted between specific dates or by a specific user:

Option	Description
Deleted between	Set **Deleted between** to the date range you want to apply to the restore. These dates are inclusive; any records deleted on these dates are included.
Deleted by this user	Select a user from the Deleted by this user list to find the records deleted by a particular user.

   Note: Alternatively, select **Specify a Record ID** to restore a single record. However, you need to know the record ID.
3. Click **Next** to display the deleted records that match your criteria. These records are the entities that meet the search criteria (any associated, deleted links are not listed). Links are only listed separately if their link end entities are not in this list. Links will be listed if the entities still exist in the database, the entities were deleted by a different user, or within a different date range.

   Note: You do not need to restore all of these records: you can deselect the records that you do not want to restore.
4. Click **Next** to display a complete list of the entities and links to restore.
5. Click **Finish** to restore the records.

   Note: If the restore is taking too long, you can press the Esc key to stop the operation. This undoes any restore that is already completed (to avoid database inconsistencies).

Purging deleted records

You can purge selected soft deleted records to permanently remove the data from your database. Purged records cannot be restored except from a backup of the database.

About this task

Purge Deleted Records is only available if soft delete is enabled for your database, and if your system administrator has given you access to it.

© N. Harris Computer Corporation

Procedure

To permanently delete soft deleted records:

1. Select **Edit** > **Purge Deleted Records**.
2. Specify the age of the records that you want to purge by entering a cutoff date in **Deleted before this**.
3. To purge records deleted by a specific user, select the username from the **Deleted by this user** list.
4. To find out how many items will be purged from the total number of soft deleted records, turn on **Review number of records before purging** and click **Next** to see the statistics.
5. When you are ready permanently to delete these records, click **Purge**.

Databases

You must select the database to work with and provide credentials before you can access data.

Logging in to iBase

When you log in to iBase, you open a security file that defines the permissions for the user account to which you are logged on. The security file connection is closed when you log out.

About this task

When you are logged on to a security file, depending on the permissions of the user account, you can:

- Open one of the databases controlled by the current security file.
- Change your password.
- Create a database from a template.
- Change the local iBase instance, for example:
  - Plug-ins available.
  - Basic, charting, and advanced settings for using iBase.
  - Recently used databases listed on the **File** menu.

Procedure

1. Select **File** > **Logon**.
2. Browse for the security file to open.

   Note: The file name ends with `*.ids`.
3. Click **Open**.
4. If the Logon screen is displayed, enter your iBase username and password.
5. Click **OK**

© N. Harris Computer Corporation

Selecting a database

To use iBase, you need to log on and open a database. Logging on happens automatically if iBase is set up to use Windows™ authentication, alternatively, you are prompted to enter an iBase username and password.

About this task

You can only have one database open at a time in any iBase session. When a database is open, you have access to all of the actions that your access control settings allow.

Procedure

1. Select **File** > **Open Database**, and browse for the database (*.idb) to open.
2. Click **Open**.
3. If prompted, enter your iBase username and password (the password is displayed as asterisks (*) for added security).
4. Click **OK** to view the database summary.
5. Click **OK**.

Changing passwords

If you are using an iBase user account, you can change the password that you use to access iBase. The characteristics of the password, such as its length and the type of characters it requires, are controlled by your security administrator.

Procedure

To change your password:

1. Select **File** > **Change Password**.
2. Type your existing password in the **Current Password** box.
3. Type the new password in the **New Password** box.
4. Type the new password again in the **Confirm Password** box.

Creating a database from a template

Database templates contain standard components. Creating a database from a template reduces the time that is taken, and ensures that databases for a specific task are created consistently.

Procedure

To create a new database from a template:

1. Ensure that you are logged into iBase, but have no databases open.
2. Select **File > New Database**.
3. Click the **Template** tab.
4. Select a type from *Blank*, *User* (user created) or *Workgroup* (system created) templates. You have the option to create and delete templates from this tab. You can also click **View** if you want to see the entity types, link types, and fields in a selected template.
5. Click **Use Database** to create a database from an existing template. You can browse to any database accessible from the currently logged in security file. A temporary template is created

© N. Harris Computer Corporation

and given a type of *Database*. This template is automatically deleted if you change to a different template, create a database using the template, or close the New Database dialog.

Note: You can also create a template from a different database, and use that template instead. For more information, see Creating a template from an existing database.

6. Click the **Configuration** tab, and select the database type.
7. Click the **Details** tab, and enter the name of the database and some information about the purpose of the database or its contents.
8. Click the **Advanced** tab, and enter the details:
   - **Database Identifier**

     Optionally, enter a short string of text in the **Database Identifier** box. Do this if you want to identify entity and link records as belonging to this database. This database identifier is only necessary if you plan to perform operations outside iBase on records taken from different databases.

     Attention: The use of a database identifier has an impact on performance since the database identifier is appended to the record identifier on every record.
   - **Extra Detail Field for Audit Log**

     Type the name of a field (in this database) in the **Extra Detail Field for Audit Log** box if you want the audit log to record the value of this field when recording actions that affect records.
   - **Soft Delete**

     Turn on the **Soft Delete** checkbox if you want to use a two-stage process for deleting records. With **Soft Delete** turned off, all delete operations take place immediately. If the **Soft Delete** checkbox is turned on, all **Delete** commands mark records for deletion and make those records unavailable for most analysis, but do not delete the records.
   - **Read Only**

     Turn on the **Read Only** checkbox if you want to make the entire database read-only, and prevent any changes to records. Users can still create sets, queries, and other folder objects.
   - **Security Classification Codes / Case Control**

     Determines whether the database uses Standard Security Classifications or restricts information based on specific cases. If you select **Standard (SCC)**, you can additionally opt to **Restrict SCC lists to accessible items only**. Turn on this option to restrict any lists of Security Classification Codes to accessible ones only. This setting applies when you add or edit a record that includes an SCC list.
   - **First Day of Week**

     Displays the first day of the week as set for this database. This defaults to <System>, which is Sunday for Microsoft Access databases. For SQL databases, this is derived from the current locale as set on your machine or via the locale ID of the SQL Server machine.

     You should only need to change this if the locale on the SQL Server machine is different to your local machine or you are working with statistics and you want your week to start on a different day.

     Note: The start day of the week may affect calculations on dates and date parts.
9. Click **OK** to create the database with the settings you have chosen.

© N. Harris Computer Corporation

Editing the Most Recently Used list (MRU)

You can edit the most recently used list of databases (MRU list). The MRU list is the list of databases that have recently been accessed.

About this task

The MRU list is the list of databases at the end of the File menu; you select one of these to reopen a database quickly. Each time a different database is opened, an entry for it is placed at the top of the list.

You can delete entries. For example, if you have deleted the target database, perhaps from a network drive, so you want to avoid time being lost while the system tries to find it. You can change the order of entries. You might want the most used databases at the top of the list, for example.

Note: The database will be opened using the security that is in the same folder as the database file. It is possible that some of the databases no longer exist, for example, a user has deleted or moved the database or connection file. Some databases may be temporarily unavailable while a server machine is out of service. Such databases still appear as entries in this dialog but not in the File menu.

Procedure

1. Select **Tools** > **MRU List Manager**.
2. In the MRU list, click on a database to select it.
3. Change the position of a database by using the up or down arrows.
4. If required, delete a database from the list by clicking **Delete**.

User and database properties

Information about your user permissions and the database design are available for you to view.

The following permissions and design details are available:

User and database detail	Description
User - Permissions	Displays a list of the database activities you can perform, such as creating or deleting records. For more details about the user permissions, see User permissions.
User -- Group Membership	Displays the security groups to which you belong.
Security Design Report	Generates a report on the security aspects of the database design. Reporting on database security
Database Properties	Displays some details about the database and its purpose. Summary of the database properties
Database Statistics	Shows how many links and entities, and their types, are contained in the database.
Database Design Report	Generates a report on the database design, including semantic types.

© N. Harris Computer Corporation

You can also find out about the type of records in the database by using the Database Explorer.

User permissions

You can check the actions that you can perform in iBase. These permissions are part of the database design; they cannot be changed in iBase User.

To view your user permissions **File** > **Properties** > **User - Permissions**. If you are denied access to an action that you need, contact your system administrator.

Permission	When turned on	When turned off
Add Entity/Link Records	You can add new records to the database.	You can find, browse, and show the records in the database but you cannot add any new ones, either individually or by importing them.
Update Entity/Link Records	You can edit records that you added.	When you have added a new record, you cannot change it in any way. Includes batch editing, assigning new icons, and merging.
Delete Entity/Link Records	You can delete records that you added.	When you have added a new record, you cannot delete it, either individually or by using batch delete.
Update/Delete Entity/Link Records created by other users	You can edit and delete any record in the database.	You cannot edit or delete records that are created by other users.
Add Folder Objects	You can add new sets, and save queries, report definitions, and import specifications that you add yourself.	You can run queries, and reports either by using definitions created by other users or by using new definitions of your own. You cannot save your definitions.
Update Folder Objects	For folder objects created by you, you can edit existing queries, report definitions,and import specifications. You can also edit the contents of existing sets, including appending records to existing sets.	When you have added a new folder object, you cannot edit it.
Delete Folder Objects	You can delete folder objects that you added yourself.	When you have added a folder object, you cannot delete it.

© N. Harris Computer Corporation

Permission	When turned on	When turned off
Update/Delete Restricted Folder Objects created by other users	You can update and delete restricted folder objects that are created by other users.	You cannot update or delete restricted folder objects that are created by other users.
Update/Delete Public Folder Objects created by other users	You can update and delete public folder objects that are created by other users.	You cannot update or delete restricted folder objects that are created by other users.
Database Creator, Database Administrator, Security Administrator	System roles that are only relevant when you are using iBase Designer.	
Audit Administrator	You can view records that are displayed and modified by other users who are defined as having a restricted audit log.	

Note: The folder objects actions (as in Add Folder Objects, for example) apply to folder objects in general. There is also access control on individual folder objects based on the membership of Folder Object Control Groups.

Reporting on database security

The security design report provides details about the security groups, users, and their consequent permissions or restrictions that have been applied to the database. You can select the items you want to include in the report.

Before you begin

To view the details of the database security, you must be assigned the `SecurityAdministrator` role.

Procedure

1. In iBase User, Select **File** > **Properties** > **Security Design Report**.
2. Select the types of details to include in the report:
   - Groups - For each security group that has been defined for the database, you can list:
     - Permissions and restrictions - What members of the group can access, and what they are explicitly prevented from doing (for example, editing read-only items).
     - Users - The users that are currently members of the group.
     - Denied SCC items - The types of entities or link that have restrictions in place, to prevent members of this group from accessing particular records.
   - Users - For each user that can access a database, you can list:
     - User Information - The information entered centrally about the specified user.

       Note: This report only uses information added in iBase Designer.
     - Groups - The groups the user is a current member of.

© N. Harris Computer Corporation

- Permissions - The specific permissions for the user.

3. Once you have generate the report you can:
   - Browse through the pages
   - Refresh the information it contains
   - Print the report
   - Export the report as a spreadsheet, a PDF, or a Microsoft Word document.

Adding your contact information

Depending on your organization, you may need to enter contact details for queries about the data that you add to this database, or to help if people need to talk to you before editing, deleting, or merging records that you own. The contact details that are displayed are for the username that you use

About this task

You can add your own contact details if you have your own iBase user account rather than a shared group account.

Procedure

1. Select **File** > **Change User Information**.
2. Enter your full name, telephone number, e-mail address, and any notes.
3. Save your details by clicking **OK**.

Managing database connections

If you have the correct permissions, you can view details about all the current connections, to the database that you have open. **Current Connections** provides information on who is logged in to that database, and their current level of activity.

Before you begin

You must be a member of the Database Management System Administrators group, and be connected to a Microsoft™ SQL Server database. If you are a system administrator for the database, but can only view your own connection, your server permission to `View Server State` might need changing. To change user permissions:

1. Start SQL Server Management Studio.
2. Expand **Databases**, right-click the database, and then click **Properties**.
3. Click **Permissions**, and then click **View server permissions**.
4. In the **Logins or Roles** list, click the user to whom you want to grant the permission.
5. In the Explicit permissions for user list, click to select the **Grant** checkbox next to **View server state permission**.
6. Repeat step 4 through step 5 for each user to whom you want to grant the permission.

Note: Users with `View Server State` permission can view any databases they have access to. This permission can be revoked using `Deny View Database State`.

© N. Harris Computer Corporation

About this task

Database activity can be used to determine details about your database use. For example, the number of concurrent licenses that are needed, or the types of client that are currently open. In addition, as all connections must be closed before a database connection can be opened in iBase Designer, you can ensure that any connections that are not currently active can be closed.

Important: Ensuring a single open connection in iBase Designer prevents any potential data loss by administrator actions. Shutting down connections to the database logs out all active users, and any unsaved changes are lost. Try to ensure that all connections are not in use before you shut down those connections. Even after closing connections, you might encounter a “Database In Use” warning when you try to open the database in iBase Designer. This is because iBase Scheduler needs to be stopped. Check the iBase Scheduler status in Windows Services, and select Stop.

Procedure

1. To open the list of current connections, select **Tools** > **Current Connections**.

   For each active connection, you can view the following information:

Column	Description
Application Type - User	The type of application that is connected and the user that is logged in. **Note:** The user is displayed when the user information can be determined.
Machine Name	The machine that has the connection open.
Login Time	When the connection started.
Last Access Time	When the connection was last active.
Reads	The number of times information has been accessed from the database in this session.

2. **Optional:** To close all the active connections to the database, click **Close Connections**.

   When **Close Connections** is clicked, the list of active connections is updated automatically over a time until all connections are closed, apart from one remaining connection.

   This last connection is your own connection, which is closed when you close iBase User.

© N. Harris Computer Corporation

Auditing

iBase can be set up to log information about the actions you carry out on a database. The details that are captured in the audit log, depend on the auditing level that has been applied to the database.

Adding audit log journal entries

Journal entries are strings that can be used to provide context to the actions that you are carrying out in a session. If your database has been set up with an auditing level of 3 or higher, you can add journal entries from both iBase and Analyst's Notebook to your audit log.

About this task

A journal entry added to the audit log is a time stamped string from a particular user. These strings can be added at anytime to provide details of the work that is being carried out. For example: 'Searching for potential suspects in investigation A'.

Tip: You can check the current auditing level of your database in the auditing section of the database properties. To open the database properties, select **File** > **Properties** > **Database Properties**.

Procedure

1. Create a journal entry:
   - To add an entry from iBase User, select **New** > **Audit Log Journal Entry**.
   - To add an entry from Analyst's Notebook, with your database connected, from the Data Sources pane, select **Audit Log Journal Entry**.
2. Add the detail of your entry and press **OK**.

Viewing audit logs

You can use the Audit Viewer to view the audit entries for a database that has already been configured for auditing. The physical form and location of logs is different for security files and database files. In addition different options are available for Microsoft Access databases, and SQL Server databases.

To view audit entries, you must be a system administrator, a database administrator, or an audit administrator. However, it is important to note that not all entries might be accessible:

- Some users generate restricted audit log entries that you need the Audit Administrator role to view.
- Some audit log entries are hidden if SC codes are used (you can only view the entries for records that match your security classification).

The level of detail in the audit log is determined by the audit level set for the database within iBase Designer, and any changes to that audit level will affect the creation of future entries in the log, not previous actions that have already occurred. You can open multiple windows to inspect logs for several databases if those databases are managed through the same security file.

You can view the audit logs for security files and databases:

- Security file logs record the opening of databases, failed logon attempts, and a range of administrative actions such as creating templates, and managing users and groups.
- Database logs record all the requested actions within databases, and the closing of databases. Actions are recorded regardless of origin: users can request database actions from i2 iBase Designer, i2 iBase, i2 Analyst's Notebook, or third party mapping applications. You can inspect logs for several databases provided that those databases are managed through the same security file.

© N. Harris Computer Corporation

Viewing the audit logs allows you to monitor usage of iBase databases and commands. For example, you can find:

- Failed logons.
- Microsoft Access to databases by unexpected users or at unusual times.
- Use of commands that send data outside of iBase: to a printer, to a file, or to an external application such as Analyst's Notebook or a mapping application.
- History of changes to single iBase entity or link records and who made them (if you log historical information).
- Journal Entries detailing the specific comments of an individual user.

As audit logs are potentially very large files, which the viewer displays as a grid of rows and columns, where each row represents an action on a database or security file and each column provides a different piece of information about an action. Much of Audit Viewer is designed to provide ways to identify and arrange actions (the rows) that are interesting or related in some way.

You can:

- Print the displayed actions
- Export the actions to a file for further analysis using a spreadsheet, database, or other visualization tool
- Archive them to a standalone database file

Opening an audit log

When you log on to iBase Audit Viewer, you open a security file that defines the permissions for the user account to which you are logged on. When your credentials are recognized, you can select the type of log you would like to view.

Procedure

1. When you have started Audit Viewer, select **Logon** to log on to a security file without opening a log or archive.
2. Select one of the following commands from the **File** menu:

Option	Description
Open Database Log File	Displays the database log for viewing.
View Security Log	Displays the security log for viewing.
Open Archive File	Displays an archive of an audit log.
Open SQL Server Archive	Opens the SQL Server archive database. (**Note:** You must specify a valid archive database and not a standard iBase database.)

© N. Harris Computer Corporation

Results

The viewer opens each log or archive in a separate window with a title bar matching the title of the log or archive. You can maximize the window within the application window. On opening each log or archive, the viewer displays all of the logged actions today, meaning the day of viewing the log.

Note: Slight differences in the contents of the window occur depending on whether database logs, security file logs, or archives are viewed.

The grid displays the audit log data, one logged action on each row. For database logs, the columns in the grid are:

Date

The date and time of the action.

User

The iBase logon name of the user for the action.

Action

The action, such as Database Opened or Record Added. Double-click the action to display the detail.

Record ID

The iBase record identifier, if the action referred to an individual record, or blank if the action refers to multiple records as the result of a bulk import. If Audit History is turned on, double-clicking the record ID displays the history of that record.

Extra Detail

The contents of a field chosen by the database designer if the action referred to a specific record.

Location

The location of the user performing the action.

Note: The location for the user is recorded at the time the log entry is made. Subsequently, the location may have been edited, but the log reflects the correct location at the time of the action.

Filtering by location will not identify the records which a user owns (if you are using owner hyperlink fields), only those records that they create or update.

Network Login

The machine and user identifier of the user performing the action, as identified by Microsoft™ Windows™. This uniquely identifies users who log on using single sign-on rather than an iBase user account.

Note: The network login is displayed in both the **User** and **Network Login** columns for changes made outside of iBase as the machine name cannot be determined for this type of change.

Detail

Information, typically the item affected or setting changed by the action.

What to do next

You can change the rules used to display the log entries on the Selection Criteria and Actions pages. To apply your changes, click **Refresh**.

On the Selection Criteria page, for example:

© N. Harris Computer Corporation

- You can extend the date range if there is no data shown for today.
- You can view subsets of log data based on various criteria. For example, actions made at a time of day specified by start and end times, on particular days of the week, or by a specific user.

On the Actions page:

- You can filter the types of actions displayed. For example, you might want to know when a database is opened.
- You can change the type of actions that are available for display by selecting an audit level in **Display actions**

You can use wildcards to include or exclude specific log entries. For example:

- Entering [!user1] in Detail (contains) excludes log entries containing user1 in their Details field.
- Entering [user1] includes these log entries.

Logging on

To log on to the audit viewer:

Procedure

1. From the **File** menu, select **Logon** or **Logon As**. Select **Logon As** if you usually log on using your Windows user name and password but on this occasion want to log on using an iBase user name and password.
2. In the Security File dialog, browse for the security file to open. The file name will end with .ids.
3. Click **Open**. The Logon dialog may be displayed if you have an iBase user name and password. If you use your Windows user name and password, then the dialog is only displayed if you are able to log on as one of several iBase users.
4. If the Logon dialog is displayed:
   - either, enter your iBase user name and password
   - or, select the iBase user from the list

   Tip: to avoid this step in the future, turn on the Remember my selection checkbox
5. Click **OK** to open the security file.

Logging on as a different Windows user

Depending how Windows security is set up at your site, you may be prompted to select the user to log on as. To avoid repeating this step each time you log on, you may have turned on the **Remember my selection** checkbox in the Logon dialog.

Procedure

To cancel this selection:

1. Start iBase, not Audit Viewer.
2. Log on in the usual way (you do not need to open the database).
3. Select **Tools** > **Options**.
4. On the General page, turn off the **Remember user for Windows single sign-on** checkbox.
5. Log off.
6. Log on to Audit Viewer and you will then be prompted to select the iBase user log on as.

© N. Harris Computer Corporation

Audit log entries

The audit log can be sorted, searched, and filtered to make locating specific entries easier.

To navigate through the log entries displayed in the grid, you can:

- Use the scroll bars to scroll horizontally or vertically.
- Click any cell in the grid, then use the `Page Up` and `Page Down` keys.

To sort the log entries displayed in the grid, select the appropriate sort column from the **Sort Order** list.

Double-click anywhere on the row of the log entry to display details. For most entries, this provides the details of the action performed in a separate form. If you are accessing a log for a SQL Server database, a number of extra options are available:

- Double-clicking rows for actions that changed database records, shows the history of the changed record.
- Text can be copied from rows.
- Information can be appended to the details section of a row (like a Journal Entry, this is in the format of an additional string with the username and timestamp).

Exporting the log entries

You can select **Action > Export** to export the displayed log entries to a comma separated value text file (.csv file). This file can be read into a spreadsheet or other visualization tool.

Note: The audit history is not exported.

Printing the log entries

You can select **Action > Report** and click the printer icon from the toolbar (or select the printer icon from the Report toolbar), to print the log entries displayed in the grid.

Note:

- The audit history is not printed.
- Log entries are printed using Landscape layout.
- You might find that the contents of the Detail column wrap around to the beginning of the next line.

Deleting and archiving the log entries

The only way to delete selected data from the audit log is to create an archive. If you no longer require the data you placed in the archive, you can delete the archive file.

Note: For SQL Server databases, you can back up the whole archive database and then delete it. An empty archive database is created the next time a user performs an auditable action.

Changing the selection criteria

To reduce the scope of information that you are searching within an audit log, you can add selection criteria. This allows data to be manually checked, but also allows reports to be generated about specific date ranges, users or records.

Procedure

1. Open the audit log you are interested in.
2. On the Selection Criteria page, refine the audit entries based on criteria:

© N. Harris Computer Corporation

- **Between these dates**

  A date range to list activity between.

  Note: The date range must be valid, with the From date occurring before the To date.
- **Between these times**

  A time period within the date range to list activity. For example, activity that happened after specified office hours, or at lunch time, on days in the date range.
- **On these days**

  Select activity that occurs on a particular day of the week in your specified date and time range.
  - <All Days> - any activity
  - Weekdays - any activity that occurs Monday to Friday
  - Weekends - any activity that occurs Saturday or Sunday
  - Mondays - Sundays - any activity on the day specified
- **User**

  Activity logged for a specified User account within the date and time range.
- **Location**

  The registered geographic area that the activity happened within.
- **Network Login**

  The registered machine and logged in user that logged the activity.
- **Record ID**

  Activity for a specific database record.
- **Extra Detail**

  Searches for information added to the audit log from the field set up in the advanced database settings to add extra auditing information. For more information about extra details, see Adding extra details for auditing.
- **Detail (contains)**

  Allows you to search for text in the details of audit entries.

For information on how to refine the log based on types of auditable action, see Changing the types of actions.

3. Click **Refresh** to update the audit log view.

Changing the types of actions

When you start Audit Viewer, it lists entries for all possible actions at the current audit level. You can change the listed actions to match a different audit level, or select specific types of actions to list.

Procedure

To change the type of activity that is listed in the log:

1. Click the **Actions** tab to display the Actions page.
2. Select the actions that you would like to list.
   - Change the list of audit-able actions by choosing an audit level in the **Display actions** list.
   - Select the actions in the report by selecting them in the displayed list.

© N. Harris Computer Corporation

Note: If you select a higher level of audit than the current level set in the database, only actions that were carried out in a period when the database was also set to that higher level will display in the log.

3. Click **Refresh**.

Viewing audit histories

In SQL Server databases, you might be able to view the history of the changes made to records in the database provided your database is set up to use Audit History. How far back the history goes depends on how frequently your system administrator archives this data.

You can access the audit history of a record from:

Audit Viewer

To display the audit history of a record, double-click an audit log entry with a Record Modified action.

iBase User

To view the history of a record:

- From a list of records, right-click and select **Show History**.
- With a record open directly (not using a datasheet), select **History**.

Analyst's Notebook

With an item selected on the chart surface, in the Data Sources task pane, select **Show > History**.

With the Audit History open, you can now filter the records that are displayed by user, entity and link type, and by time, or you can extend the selection to include other records of interest.

Note: In a database set to audit level 5, the number of times the records have been viewed, but not edited, is displayed in the Views column.

Selecting records of interest

To select further records of interest, click **Select** in the top right of the Audit History.

To display all records touched by a specific user:

- In the Records to display area, select **All records used by** and select the user name from the list. The selected user name will be displayed as a reminder in the top left of the Audit History.

To filter by entity and link types:

- In the Types to display area, turn the entity or link type checkboxes 'on' or 'off'.

  Note: Only types with records in your selection are listed, and the records for a selected type are displayed only when the appropriate checkbox is turned on.

To filter by time:

- In the Time period to display area, select a time period.

Changing what's shown in the edit history area

The entries in the edit history area can be displayed in a variety of ways:

1. Make sure that `by all users` is selected in **Show Edits**.
2. Turn on or off the following options:

© N. Harris Computer Corporation

- **Show Headers** to hide or show the shaded line that displays either the date/time/user name or the field name. You cannot expand and collapse when this option is turned off.
- **Expand** to show the complete history.
- **Collapse** the complete history to just display the headers.
- **Audit** groups the entries by the name of the user who worked on the record and when they were created, updated or deleted.
- **Field** groups the entries by the data that has been added, updated or deleted. Click again to sort in ascending or descending order by date edited.
- **Edits** displays a history of the changes to the record (only available if the database is set to audit level 5).
- **Views** displays a history of who viewed the record and when (only available if the database is set to audit level 5).

In the edit history area, you can filter the edits to those made by a specific user. Filtering by user makes the other display options unavailable:

- Select **Show Edits > `user_name`**. The users on this list are selected by clicking **Select** in the top right of the Audit History dialog.

Copying the edit history

Click **Copy** to copy all the information shown in the History of edits area to the Windows clipboard.

Note: You cannot copy image and document fields.

Description of the columns in the history of edits area

The history of edits area shows information on the changes made to the selected record:

Information shown	Description
Field Name	The old and current values.
Edited by	The logon name of the user who made the change.
Date Edited	The date and time of the change.
Reason	If required by the database, the reason given by the user for making the change.
OS User	The Windows name of the user made the change.
Machine Name	The machine that the user was working on.
Location	The location as entered in the User Information dialog.

© N. Harris Computer Corporation

Information shown	Description
iBase Change	When this checkbox is turned off, the update was made outside iBase (and Audit Viewer might be unable to determine the machine name).
Extra Detail	You might see an additional Extra Detail column that displays additional information for the current record. (For information on how an administrator can set up the audit log to record extra detail, see Adding extra details for auditing) .

Additional historical data

Additional data may be shown for each record. This can include:

- the name of the icon if an alternate icon is assigned to the record.
- the icon color (which will be blank if the standard icon color is used).
- the record status (applicable only if Soft Delete is used). The record status might be Soft Deleted, Normal (because the record has been soft deleted or restored), and Purged.
- Security Classification, the old and new SC code (if this feature is used and if you have authority to view this information).

Some information might be displayed that you do not usually see, such as the date the record was created and the record ID.

Code list histories

Changes to pick lists, icon lists, or SCC lists are classed as **Code List Modified** actions in the Actions page.

Details could include:

- old and new values
- old and new descriptions
- old and new parent pick lists, for filtered pick lists

All the changes made in the same session are grouped together by user name, date and time. As there may be several pages of changes, you can print the list or save it as a Microsoft Excel spreadsheet or PDF file. How far back the history goes depends on how frequently your system administrator archives this data.

Note: Changes to code lists are only logged if the database is an SQL Server database and audit history is turned on. To find out whether the database logs audit history, check the Audit History setting in the Database Properties.

Filtering using sets

If you are only interested in events that affect a given set of records, you can filter the audit log entries based on set membership. This feature is only available for SQL Server databases.

Procedure

1. Click the **Set Membership** tab to display the Set Membership page. Any
2. Choose the sets of records used to filter audit log entries:

© N. Harris Computer Corporation

- To add a set - Click **Add**, and select the set to use.
- To remove a set - with the set selected in the list, click **Remove**.

3. Click **Refresh** to update the audit log with your selection.

   Note: Log entries that are not associated with a records, are not filtered by this feature.

Saving filters

If you have changed the types of entries listed in the audit log, you can save the criteria as a filter. This filter can then be reapplied should you want to investigate the same types of activity in the future.

Procedure

To set up a filter:

1. Choose the types of audit log entries to display. For more information on changing the types of audit log entries, see:
   - Changing the selection criteria
   - Changing the types of actions
2. On the Filters page, select **Save**.
3. Enter a name for your filter, and press **OK**.
4. Choose the category and access level used to store the filter, and press **OK**.

Results

You filter is saved. You can **Apply** your filter to use this set of criteria on the audit log in the future.

Creating an audit log report

You can take the audit log that you have open, and convert it into a report. The database audit log can be printed, or exported into a file format for your records.

About this task

Depending on the type of audit log that you have open, and the information that you are interested in, the audit log report will display different information.

Procedure

To create an audit report:

1. In the Audit Viewer, open an audit log and filter the log using the **Selection Criteria** and **Actions** to determine the contents of interest. If you regularly filter your audit information in the same way, you can save the filtering options as a **Filter**, that can be applied in later sessions.
2. To generate the report, click **Action** > **Report**.
3. With the report open, you can use the options in the toolbar to:
   - Refresh - check for new actions.
   - Print - send the report to a printer.
   - Export - save the report as a file compatible with Microsoft Excel (*.xlsx), Microsoft Word (*.docx), or a PDF document.

© N. Harris Computer Corporation

Example of using the audit viewer

You might want to discover if any user opens a database outside of normal working hours, here taken to mean between midnight and 09:00.

Procedure

1. Click the **Selection Criteria** tab to display the Selection Criteria page.
2. Set the date and weekday ranges, time of day, and User ID:
   - In the **Between these dates:** box, select an appropriate start date. For example, to set a date two months before today, click to highlight the month part of the date and press the down arrow key on your keyboard. In the to: box, leave the end date at today's date
   - In the **Between these times:** box, enter the time that you want logging to start. For example, the earliest time you can enter is midnight, which you must enter using the 24-hour system as 00:00:00.
   - From the **On these days:** list, select the appropriate days of the week. For example, to look at all days, leave the setting at its initial value of <All Days>.
   - In the **to:** time picker to the right, enter the time that you want logging to stop. For example, normal working hours may start at 9 am, which you must enter using the 24-hour system as 09:00:00.
   - You want to see all users who have opened databases outside of normal working hours so you can leave the value in the User box as <Any User>, which is the default. You could use wildcards in this and other text boxes.

   This set of rules will find all the users who have performed a logged action in the date and time ranges you have set. The grid does not change automatically.
3. Click **Refresh** to find the log entries matching these rules.
4. You can filter the log entries by selecting one or more actions. For example, you can only view the actions when a database is opened:
   - Click the **Actions** tab to display the Actions page. When you first open a log, this page lists all possible actions for the current audit level.
   - Click **Clear All** to turn off all actions.
   - Turn on only the **Database Opened** checkbox.

   This sets all the necessary rules. The grid does not change automatically.
5. Click **Refresh** to display actions based on all the rules that you have specified. The matching results are displayed in the grid.

Working with security logs

The security log lists the transactions of interest. To open a security log, select **File** > **View Security Log**.

There are two possibilities:

- The log is displayed immediately. (This is the log for the security file to which you are currently logged on.)
- A Security File browser is displayed where you can locate and open a security file. Audit Viewer opens a security log once you have successfully logged on to the security file.

Compared to working with a database log, there are some minor differences:

© N. Harris Computer Corporation

- The grid does not contain columns for record IDs or extra detail because these columns are relevant only to specific records, for which the security log does not record actions.
- The Selection Criteria page has fewer controls. The unavailable controls are those relevant only to specific records.
- The Actions page lists different actions and you cannot change the audit level. The extra actions are those relevant to database and security operations: compacting, converting, and upsizing databases; creating databases and templates; managing users and groups; and failed logon attempts.
- You can only sort by date (in the **Sort Order** list).

Viewing audit histories

In SQL Server databases, you might be able to view the history of the changes made to records in the database provided your database is set up to use Audit History. How far back the history goes depends on how frequently your system administrator archives this data.

You can access the audit history of a record from:

Audit Viewer

To display the audit history of a record, double-click an audit log entry with a Record Modified action.

iBase User

To view the history of a record:

- From a list of records, right-click and select **Show History**.
- With a record open directly (not using a datasheet), select **History**.

Analyst's Notebook

With an item selected on the chart surface, in the Data Sources task pane, select **Show > History**.

With the Audit History open, you can now filter the records that are displayed by user, entity and link type, and by time, or you can extend the selection to include other records of interest.

Note: In a database set to audit level 5, the number of times the records have been viewed, but not edited, is displayed in the Views column.

Selecting records of interest

To select further records of interest, click **Select** in the top right of the Audit History.

To display all records touched by a specific user:

- In the Records to display area, select **All records used by** and select the user name from the list. The selected user name will be displayed as a reminder in the top left of the Audit History.

To filter by entity and link types:

- In the Types to display area, turn the entity or link type checkboxes 'on' or 'off'.

  Note: Only types with records in your selection are listed, and the records for a selected type are displayed only when the appropriate checkbox is turned on.

To filter by time:

- In the Time period to display area, select a time period.

© N. Harris Computer Corporation

Changing what's shown in the edit history area

The entries in the edit history area can be displayed in a variety of ways:

1. Make sure that `by all users` is selected in **Show Edits**.
2. Turn on or off the following options:
   - **Show Headers** to hide or show the shaded line that displays either the date/time/user name or the field name. You cannot expand and collapse when this option is turned off.
   - **Expand** to show the complete history.
   - **Collapse** the complete history to just display the headers.
   - **Audit** groups the entries by the name of the user who worked on the record and when they were created, updated or deleted.
   - **Field** groups the entries by the data that has been added, updated or deleted. Click again to sort in ascending or descending order by date edited.
   - **Edits** displays a history of the changes to the record (only available if the database is set to audit level 5).
   - **Views** displays a history of who viewed the record and when (only available if the database is set to audit level 5).

In the edit history area, you can filter the edits to those made by a specific user. Filtering by user makes the other display options unavailable:

- Select **Show Edits > `user_name`**. The users on this list are selected by clicking **Select** in the top right of the Audit History dialog.

Copying the edit history

Click **Copy** to copy all the information shown in the History of edits area to the Windows clipboard.

Note: You cannot copy image and document fields.

Description of the columns in the history of edits area

The history of edits area shows information on the changes made to the selected record:

Information shown	Description
Field Name	The old and current values.
Edited by	The logon name of the user who made the change.
Date Edited	The date and time of the change.
Reason	If required by the database, the reason given by the user for making the change.
OS User	The Windows name of the user made the change.
Machine Name	The machine that the user was working on.

© N. Harris Computer Corporation

Information shown	Description
Location	The location as entered in the User Information dialog.
iBase Change	When this checkbox is turned off, the update was made outside iBase (and Audit Viewer might be unable to determine the machine name).
Extra Detail	You might see an additional Extra Detail column that displays additional information for the current record. (For information on how an administrator can set up the audit log to record extra detail, see Adding extra details for auditing) .

Additional historical data

Additional data may be shown for each record. This can include:

- the name of the icon if an alternate icon is assigned to the record.
- the icon color (which will be blank if the standard icon color is used).
- the record status (applicable only if Soft Delete is used). The record status might be Soft Deleted, Normal (because the record has been soft deleted or restored), and Purged.
- Security Classification, the old and new SC code (if this feature is used and if you have authority to view this information).

Some information might be displayed that you do not usually see, such as the date the record was created and the record ID.

Versions of records

If you have audit history enabled on your SQL Server database, the details of any record change are stored in the audit log. You can access a record of the changes made to records, and select a previous version of the record to be saved as the current state.

About this task

Records can only be reverted to versions that are stored in the audit history. If an earlier version was created before the audit history was enabled, you will not be able to revert to that record. In addition, you cannot revert the changes to a datasheet, but only to individual records.

Procedure

To work with previous versions of a record:

1. Open the record directly (not using a datasheet), and select **Revert**. The record will open with a slider bar at the top to navigate the versions of the record.
2. Use the slider, to move between different versions of the record. As details change between versions, the differences between field values when compared to the current version are highlighted in the display.
3. If you would like to revert to an earlier version of the record, select that version and click **Save**. A new version of the record will be saved with the same field values as the earlier version.

© N. Harris Computer Corporation

Database subsets

A database subset is a portion of records in the database that are copied into a separate database. This collection of records are selected by creating a database subset definition that consists of the results of queries and sets.

You might want to create a database subset for a number of reasons:

Creating an environment that matches your current production environment for testing or training.

Adding a smaller amount of real data from a production environment lets you test changes to the database, or train users in as close to the production environment as possible.

Working with a set of data that relates to a specific department or organization.

By creating an environment that only contains specified data allows sanctioned data to be shared.

A database subset can be created from a query at any time, unlike the information in a case, that is assigned as the data is added.

To create a database subset:

1. Define the records to include using a subset definition.
2. Create the database subset in either Microsoft™ Access or SQL Server.

The database subset can then be used independently, and if required, you can synchronize any changes with the original database.

Creating a database subset definition

The records in a database subset are selected by creating a database subset definition. When you have created the definition, you can use it to export the data you selected as XML, or you can create a database containing the selected records.

Procedure

To define the records in a database subset:

1. Log on as a user with permission to add folder objects, and open the database.
2. Select **File** > **Data** > **Database Subsets** > **Database Subset Definitions**.
3. Click **New**.
4. Select the records by adding queries and sets to the definition. The queries and sets form a part of the definition and deleting any of these sets or queries, as opposed to just removing them from the definition, invalidates the definition and any database subsets created from it.

   Note: If the subset definition is being used to create database subsets in Microsoft Access, you can use parameterized queries and the values required to run these queries are entered when the database subset is created (or synchronized). If you include parameterized queries, then you must enter values for them when creating database subsets (and when synchronizing). Advanced subsets cannot be created using subset definitions that include parameterized queries.
5. Click **Save** to save the definition.

To create a database subset from your definition:

1. Select the type of database storage to use for your subset:

© N. Harris Computer Corporation

- To create a subset in a Microsoft Access database, select **Create Subset**, and follow the instructions in Creating database subsets (Microsoft Access).
- To create a subset in a Microsoft SQL Server database, select **Create Advanced Subset**, and follow the instructions in Create advanced database subsets (SQL Server).

Results

The database subset definition is created.

At any stage, you can:

- Change the definition by adding new sets and queries or by removing them (during synchronization the database subset will be re-created).
- Rename and move the sets and queries that are listed in the definition (this updates the definition).
- Rename the definition.
- Move the definition to a different folder.

You can also delete the definition if it is:

- No longer required to create new database subsets.
- No longer required to update database subsets at the end of synchronization.

Creating database subsets (Microsoft Access)

You can create a database subset from the records that are included in the results of running queries or sets that are specified in a database subset definition. If you use the **Create Database subset** option, the subset database will be in Microsoft Access format.

Before you begin

Before you can create a database subset, you need to specify the records that you want to copy to the new database by creating a database subset definition.

Note: Only database administrators can initialize the database for database subsets.

Procedure

To create a database subset:

1. Log on as a user that has the Database Creator role.
2. Open the database from which you want to create the database subset.
3. Select **File** > **Data** > **Database Subsets** > **Create Database Subset**.
4. In the **Identifier** box, enter a unique ID for the database subset. The ID is up to five alphanumeric characters long. Previously-used identifiers are listed in the **Utilized Identifiers** list.
5. In the **Name** box, enter a name that will be used for both the subset security file and subset database.
6. A new user account with system administrator permissions will be created in the subset security file. Enter the username and password for this account. This account will be used to synchronize the database subset with the main database and to log on to the database subset if no other user accounts are added to the security file.

© N. Harris Computer Corporation

Note: Any records added to the database subset will have this user as their "Create User". You may therefore want to select a username that will be meaningful once these records are uploaded to the main database.

7. In **Destination folder**, browse to the folder where you want to create the subset security file and database. You can create a new folder if you have sufficient Windows permissions. The folder you use can contain only one iBase database and security file.
8. In **Subset Definition**, browse for the definition that defines the data to be copied to the new database. At this stage, it is not possible to know whether the definition is still valid or whether the total number of records exceeds 50,000 (the maximum allowed records).
9. Click **Create** to continue. You will be warned if the definition is invalid because it contains deleted queries or sets, or if the total number of records exceeds the 50,000 record limit.
10. Click **OK** to create the database subset. If the definition contains any parametrized queries then you will be prompted for the values. You can click **Cancel** but doing so will also cancel the creation of the database subset.

Synchronizing databases

Synchronizing databases, uploads the data from the database subset to the main database and downloads new and updated records in the subset definition to the database subset. You can update the database subset using the original subset definition or you can select a different subset definition.

About this task

A conflict occurs when an entity or link is changed in both the main database and the database subset. To resolve the conflict, you need to decide which record you want to keep. You can select:

- Discard the subset record changes - this means you keep the changes to the record in the main database and lose the information in the record from the database subset.
- Keep the subset record changes - this means you keep the information in the record in the database subset and overwrite the changes that are made to the record in the main database.

If the main record is deleted, then it is:

- Restored and updated to match the subset record if Soft Delete is in use.
- Re-created if the record is deleted or purged.

Restoring or re-creating a link always results in the link ends being restored or re-created if necessary. Restoring or re-creating an entity also restores or re-creates any associated links if the other end of the link is still active.

During synchronization, the following error messages might be displayed:

- The database subset has expired. - You cannot reuse an expired database subset. Re-create it from its database subset definition.
- The database subset has an incompatible schema. - The database subset is invalid because the schema of the main database was changed after the database subset was created. To fix this problem, use the **Database Schema Update** option in iBase Designer.
- The database subset is read-only. - Use iBase Designer to change the database properties of the database subset so that it is no longer read-only. Although you can change the Read-only property in an expired database, you cannot reuse it.
- This is not a valid database subset. - The selected database subset is either not a database subset or it might be a subset of a different database. You can set the database subset to expire if you do

© N. Harris Computer Corporation

not need it any longer. This deletes the contents of the database subset and mark it as read-only. The database subset can never be reused.

When you synchronize a database subset with the main database:

- Newly created entities and links in the database subset are added to the main database, with the same record identifier, create date or time, and create user.
- All (soft) deleted records in the database subset are ignored - they have no effect on the main database.
- Records in the main database are updated to match the changes in the database subset if there are no conflicts.
- If a record has changed in both the main database and database subset, since the last synchronization, then conflict resolution is applied. See below for details.

At the end of synchronization, you are informed of the changes made to the main database:

- The number of new records added to the main database.
- The number of records updated in the main database with changes made in the database subset
- If Soft Delete is used: the number of records restored as a result of conflict resolution
- If Soft Delete is not used: the number of records that are re-created as a result of conflict resolution
- The total number of conflicts resolved (at record level)

When synchronization is complete, an updated database subset, re-created using the latest version of the subset definition, is available for reuse in the field. Alternatively, the database subset is set to read-only if the database subset was set to expire.

Procedure

To upload the records in a database subset:

1. Back up the main database if it is in Microsoft Access format.
2. Log on using a user account that has the Database Administrator role and permission to add records, update records, delete records, and update or delete records created by other users.
3. Open the database from which the database subset was created.
4. Select **File** > **Data** > **Database Subsets** > **Synchronize**.
5. Browse for the database subset containing the records that you want to load.
6. Log on to the subset security file as a user with system administrator permissions. Typically, you will use the name and password specified when the subset security file and database was created.
7. Click **Connect**.
8. Decide what you want to do with the database subset:
   - To discard the subset after uploading the records to the main database, turn on **The database subset should expire after synchronization**.
   - To reuse the subset, browse for a subset definition (it does not have to be the original subset definition).

   Note: The Subset Definition box displays the name of the subset definition originally used to create the subset, and will be blank if the subset definition is no longer accessible or does not exist.
9. Decide how you want to handle any conflicts between changes made in the main database and those made in the database subset. By default, the dialog ensures that the synchronization will never overwrite changes in the main database.

© N. Harris Computer Corporation

Note: At this stage it is not possible to know whether there are actually any conflicts.

10. Click **Synchronize** and then **OK** to continue. If the subset definition contains any parameterized queries then you will be prompted for the values to use. If you cancel entry of the parameter values, you will also cancel the synchronization of the databases.

Create advanced database subsets (SQL Server)

You can create a database subset from the records that are included in the results of running queries or sets that are specified in a database subset definition. If you use the **Create Advanced Subset** option, the subset database will be in Microsoft SQL Server format.

Before you begin

Before you can create a database subset, you need to specify the records that you want to copy to the new database by creating a database subset definition.

Note: Only database administrators can initialize the database for database subsets.

Procedure

To create an advanced database subset:

1. Log on as a user that has the Database Creator role.
2. Open the database from which you want to create the database subset.
3. Select **File** > **Data** > **Database Subsets** > **Create Advanced Subset**.
4. In the **Name** box, enter a name that will be used for both the subset security file and database subset. This also generates the **Database Name** displayed in the **SQL Server (subset)** section.
5. The subset security file will be generated with the same users as the master database.
6. Enter the Server connection URL in the **Server** box, and enter your database credentials, these can be:
   - An exact duplicate of the credentials used to access the master database.
   - A specified user name and password
   - Windows Authentication

   Tip: Test your connection each time you change the server or the credentials used to access it.
7. In **Destination folder**, browse to the folder where you want to create the subset security file and database identifier. You can create a new folder if you have sufficient Windows permissions. The folder you use can contain only one iBase database identifier and security file.
8. In **Subset Definition**, browse for the definition that defines the data to be copied to the new database. At this stage, it is not possible to know whether the definition is valid.
9. Confirm the iBase password that was entered when you logged into the database.
10. Click **Create** to continue. You will be warned if the definition is invalid if it contains parameterized queries, deleted queries or sets, or if the total number of records exceeds the record limit (5 million).
11. Click **OK** to create the database subset.

© N. Harris Computer Corporation

Advanced synchronize

Synchronizing databases, uploads the data from the database subset to the main database and downloads new and updated records in the subset definition to the database subset. You can update the database subset using the original subset definition or you can select a different subset definition.

About this task

When you synchronize an advanced database subset, the records are compared, any records that have been updated either the main database or the database subset is updated in the other location.

A conflict occurs when an entity or link is changed in both the main database and the database subset. To resolve the conflict, you need to decide which record you want to keep. You can either:

- Discard the subset record changes, keeping the changes to the record in the main database and lose the information in the record from the database subset.
- Keep the subset record changes, keeping the information in the record in the database subset and overwriting the changes in the main database.

If the main record is deleted, then it is:

- Restored and updated to match the subset record if Soft Delete is in use.
- Re-created if the record is deleted or purged.

Restoring or re-creating a link always results in the link ends being restored or re-created if necessary. Restoring or re-creating an entity also restores or re-creates any associated links if the other end of the link is still active.

During synchronization, the following error messages might be displayed:

- The database subset has expired. - You cannot reuse an expired database subset. Re-create it from its database subset definition.
- The database subset has an incompatible schema. - The database subset is invalid because the schema of the main database was changed after the database subset was created. To fix this problem, use the **Database Schema Update** option in iBase Designer.
- The database subset is read-only. - Use iBase Designer to change the database properties of the database subset so that it is no longer read-only. Although you can change the Read-only property in an expired database, you cannot reuse it.
- This is not a valid database subset. - The selected database subset is either not a database subset or it might be a subset of a different database. You can set the database subset to expire if you do not need it any longer. This deletes the contents of the database subset and mark it as read-only. The database subset can never be reused.

When you synchronize an advanced database subset with the main database:

- Newly created entities and links in the database subset are added to the main database, with the same record identifier, create date or time, and create user.
- All (soft) deleted records in the database subset are ignored - they have no effect on the main database.
- Records in the main database are updated to match the changes in the database subset if there are no conflicts.
- If a record has changed in both the main database and database subset, since the last synchronization, then conflict resolution is applied. See below for details.

At the end of synchronization, you are informed of the changes made to the main database:

© N. Harris Computer Corporation

- The number of new records added to the main database.
- The number of records updated in the main database with changes made in the database subset
- If Soft Delete is used: the number of records restored as a result of conflict resolution
- If Soft Delete is not used: the number of records that are re-created as a result of conflict resolution
- The total number of conflicts resolved (at record level)

When synchronization is complete, an updated database subset, re-created using the latest version of the subset definition, is available for reuse in the field. Alternatively, the database subset is set to read-only if the database subset was set to expire.

Procedure

To synchronize an advanced database subset:

1. Log on using a user account that has the Database Administrator role and permission to add records, update records, delete records, and update or delete records created by other users.
2. Open the database from which the database subset was created.
3. Select **File** > **Data** > **Database Subsets** > **Advanced Synchronize (SQL Server)**.
4. Browse for the database subset containing the records that you want to load.
5. Enter the iBase username and password used to access the database subset.

   Note: This user account should also have the Database Administrator role and permission to add records, update records, delete records, and update or delete records created by other users.
6. **Optional:** Use the **Options** to determine whether field attachments and records that have been deleted are included in the synchronization.
7. Click **Next**.
8. Decide how you want to handle any conflicts between changes made in the main database and those made in the database subset. By default, synchronization will never overwrite changes in the main database.

   Note: At this stage it is not possible to know whether there are actually any conflicts.
9. Click **Next**.
10. View the list of changes and use **Ignore Selected** to stop selected records from being updated.
11. **Optional:** Decide whether to update field attachments where they differ using **Include field attachments when repopulating**.
12. **Optional:** To discard the subset after uploading the records to the main database, turn on **The database subset should expire after synchronization**.
13. Click **Syncronize**.

Cases

Data in your database might be organized into different cases. Each case contains records belonging to a particular investigation. You are assigned access to one or more cases by your database administrator.

Working on a single case allows you to focus only on those entities and links that relate to a particular investigation or series of investigations. You continue to receive alerts for records in the other cases to which you are assigned but you will not be able to view the details for those alerts. To view the details, you need to log on to the appropriate case.

© N. Harris Computer Corporation

You can be assigned to several cases, but to add or modify data in a case, you must select only that case when you open the database. You can log on to all the cases to which you have access (multi-case analysis mode). This enables you to view records in all the cases, but you will not be able to modify any data.

Open and closed cases

Records can only be added, edited or deleted when you are working in an open case.

Closed cases can be selected when opening the database but only in read-only mode. Closed cases are included in multi-case analysis mode.

Cases can be closed and re-opened multiple times.

Note: Only a user with both security administrator and database administrator permissions can close a case or re-open a closed case.

To change case, select **File** > **Change Case**. You can see the current case and its status in the application title bar. Cases are shown in square brackets after the database name.

Accessing a case

You are authorized to access cases by your security and database administrator.

If you are only assigned to a single case you will be connected to that case automatically when you log on, without being prompted to choose a case. When working in a single case, you can create new records as well as viewing existing data.

If you are authorized to access several cases, you can open a single case or all cases at once:

- If you open a single case, you can add or modify data.
- If you open all your cases in multi-case analysis mode, you cannot create new records.

Note: If you are not authorized to access any cases, then you will be unable to open the database. Refer to your database administrator for assistance on accessing cases.

Records in a case

- When a single case is selected, any queries that you run will return results based only on the records in the current case. This also applies to sets and reports; for example, a set will only list the records in the current case.
- Whenever you select "All records" when logged in to a single case, this refers to all the records in that case only.
- When several cases are selected in multi-case analysis mode, then "All records" applies to the records in all of the cases to which you have access.

Multi-case analysis mode

Multi-case analysis mode is useful when you want to query or report on data across several cases. In multi-case analysis mode, you can view records in all the cases (open and closed) to which you have access, but you cannot add, modify or delete any records in the database, or add alert definitions. You can also view the detail of all your alerts.

When you open a database, turn on **Multi-Case Analysis** to specify that you want to work in all the cases to which you have access. The cases are opened in read-only mode; no new data can be added.

© N. Harris Computer Corporation

When you select Multi-Case Analysis, the list of cases becomes unavailable, as you can no longer select a specific case to access.

How do I know which cases I am logged in to?

When you are logged in to a single case, the case name is displayed in brackets in the title bar of the main iBase application.

If the case you are logged in to is closed, this is indicated in the title bar.

When you are in multi-case analysis mode, the iBase application title bar shows "(Read Only)" after the database name to indicate that you cannot add or modify any records in the database.

To view a list of the cases you are logged in to, select **File** > **Properties - Database Statistics**, and click **Entity Types by Case**.

Note: A security administrator and database administrator sees all the cases in the database. If you do not have these roles, you will only see the cases to which you have been given access.

Note: If you only have access to a single case, or there is only a single case in the database, the Entity Types by Case tab is not displayed.

Case details

Each case has the following properties:

Property	Description
Name	The name given to a case when it is created. Case names must be unique across the entire database.
Date Created	Automatically captured when the case was first created.
Date Closed	Automatically captured when the status is set to closed.
Description	Used to provide more information about the case. Entered when the case is created or updated. Can be modified when required.

Note: Cases can only be used in SQL Server databases. Cases are set up by a security and database administrator using iBase Designer.

Note: Word search is unavailable if you have a case-controlled database.

Select a case

When you open a database that contains multiple cases that you can access, you are prompted to select the case to open. You only have access to cases that are authorized by your database administrator and security administrator.

To open a case, select the case that you want to use from the list and click **OK**.

© N. Harris Computer Corporation

To view a list of closed cases in this database, click the Closed tab. You can only work on a closed case in read-only mode (that is, you cannot add or edit any records). If you want to reopen a closed case, speak to your database administrator and security administrator.

Note: If you are only authorized to access a single case, you are connected to that case automatically.

Accessing all available cases

Turn on **Multi-Case Analysis** to specify that you want to work in all the cases (open and closed) to which you are assigned. The cases are opened in read-only mode. This means that:

- No new data can be added.
- Data cannot be modified or deleted.
- No alert definitions can be added.
- But you can view the detail of any of your alerts.

When you turn on **Multi-Case Analysis**, the list of cases becomes unavailable, as you can no longer select a specific case to access.

Create or edit a case

You can edit the general properties of a case, such as the description, and assign users and groups. You can only edit or create a case if you have both security administrator and database administrator permissions.

To create a case, select **New** > **Case**.

To edit a case, select **Edit** > **Case**.

Note: To change the name of an existing case or to delete a case, use iBase Designer.

If you do not have permission to create or edit a case, the menu commands are unavailable.

Setting general properties

Enter a **Description** for the case. This description is visible to the user when they select a case, when logging on to the database.

Specify whether the case is Open or Closed. Users are able to add data to an open case when they select only that case when logging on. Data in a closed case cannot be added, modified, or deleted by users.

Assigning users to the case

A list of all the users of this database is displayed on the left. This list is defined in iBase Designer. To add users to or remove users from the database, you need to start iBase Designer.

To assign a database user to this case, double-click their name.

Added users appear in the list on the right.

Assigning groups to the case

A list of all the groups of users (Data Access Control groups) for this database is displayed on the left. This list is defined in iBase Designer. To add or remove groups from the database, you need to start iBase Designer.

To assign a group to this case, double-click the group name.

© N. Harris Computer Corporation

Added groups appear in the list on the right.

Replication

iBase database replication is the process of automatically distributing copies of iBase data and database objects between SQL Server instances in different locations and keeping this data synchronized. The data is copied by use of SQL Server merge replication, using the standard tools provided in SQL Server.

iBase database replication provides more tools to manage the iBase database. All servers that are involved in replication must use the same SQL Server version.

In iBase database replication, one of the iBase database servers is configured as the Publisher, and empty iBase databases are created at the other locations.

Reviewing replication conflicts

Occasionally the same entity or link is updated or deleted by users in different databases in such a way that a conflict occurs between the different versions of the record. Such conflicts are automatically resolved by SQL Server when they are merged in the publication database. The conflict is automatically resolved either in favor of the change that is made at the Publisher or, if the changes were made in two subscription databases, in favor of the first version to merge with the Publisher.

About this task

It is important to check for and review any conflicts regularly. To begin with, a suitable interval might be hourly, then when you know how frequently conflicts occur you can adjust the interval, for example, to once a day.

Working in the publication database, you use the Conflict Viewer to review how the conflicts were resolved by SQL Server. If required, you can change the outcome of the conflict and even edit the record to combine information from the different versions. When you complete the review by clicking Apply, any changes to the outcome of the conflict are replicated to the other sites to produce a consistent view of the data across all the sites. If there are links involved, changing the outcome of the conflict can also restore or delete links depending on the type of conflict.

How the conflicts are displayed

In the Conflict Viewer, the tree view lists the entity and link types for which there are conflicts. To obtain a breakdown by type, expand the items in the tree view, and then click an entity or link type to list the actual conflicts. This record will be a modification unless DELETED or RESOLVED appears in the record label. The tree view always lists the winning version of the record. The shortcut menu that is available by right-clicking on each record, applies only to the winning version.

The Conflict Viewer does not display the total number of broken links until you click Broken Links or review the last conflict. This is because it might take some time to calculate which links are affected by the conflicts or broken as a result of changing the outcome of the conflicts.

Click Refresh or press the F5 key to update the list with any recent conflicts and to remove any resolved ones.

Procedure

To review conflicts:

1. Select **Tools** > **Replication** > **Conflict Viewer**.

© N. Harris Computer Corporation

Note: The **Replication** menu is only available when the database is initialized for replication, and you log on as a database or system administrator.

2. In the tree view, click a conflict.
3. Compare the two versions of the entity or link. The version that won when the conflict was automatically resolved is displayed as the winner, and the differences between the two versions are highlighted. To make it easier to identify the differences, turn on **Show Conflicting Fields Only**.

   If any links are going to be deleted or restored as a result of this conflict between entities, **Link(s) Affected** appears.

   The table below each version of the record shows additional information:

Option	**Description**
Field	Description
Record identifier	Double-click the identifier to display the winning record (you can never display the losing record). The identifier is the same for both the winning and losing versions of the record.
Created	Double-click to display the contact details for the user who originally added the record.
Last Updated	The Last Updated field is different in each version. It shows the date, time, and user who made the change or deletion. Double-click to display the user's contact details.
Description	Double-click to display a message about the databases in which the conflicting change was made. The format of the database name is: <SQL Server name>\SQL Server instance name>\database name>.

4. Resolve the conflict:
   - If you are satisfied that the conflict was resolved correctly, click **Apply** below the Winner area.
   - If the conflict was resolved incorrectly, click **Apply** below the Loser area. You will therefore resolve the conflict in favor of the version displayed in the Loser area.
   - To edit either version before resolving the issue, click **Edit** below the version that you want to modify and then make the required changes.

   Note: Although you can change any of the entity or link details, you cannot change the link end entities, and you cannot edit a version of the record that is set to be deleted.
5. Click **Apply**.
   - The preferred version of the record is replicated to all the subscription databases.
   - The rejected version of the record is permanently deleted in the publication database.

© N. Harris Computer Corporation

What to do next

Depending on the type of conflict, a broken link may be detected by the Conflict Viewer if someone adds a link to an entity that is deleted at a different site. The broken link feature enables you to view and then fix the problem by either restoring the link end entities or removing the broken link. Anything that is removed will be soft-deleted rather than purged. Fixing a broken link may also fix other related broken links.

To check for broken links, click on **Broken Links** in the tree view.

To fix any broken links that are found:

1. In the tree view, click on the broken link and then examine its details. You need to decide whether to delete the broken link or whether to keep it which will mean restoring the link end entity. You can use the commands on the shortcut menu to investigate the records further. You cannot edit any information for a broken link.
2. Click:
   - **Remove** to soft delete the broken link and leave the link end entities unchanged.
   - **Restore** to restore any deleted link end entities so that the link is no longer broken.

If you want to leave this broken link until later, click a different broken link in the tree view.

Load files for replication

Working in the publication database or in any of the subscription databases, you can load any type of file into the database for replication to the other sites. You can also add a description of the file.

Editing the description of a file, or deleting a file, also edits or deletes it at the other sites.

A loaded file has a unique identifier, this consists of a number that is appended with the database identifier for the database in which the file was first loaded. You can display the file ID by double-clicking the file. You cannot change any of the other file properties apart from its description.

Examples of file types that can be loaded into the database for replication to the other sites are:

- A database template (`.idt` file), extra database templates must be compressed or renamed with a different file extension
- Analyst's Notebook templates (`.ant` files)
- Archive log files (`.idla` files)
- Microsoft™ Word documents that are used as templates for iBase report definitions, if there are many documents, consider adding them to an archive file first (`.zip` file)

Note: After replication, copy these files to the correct place in the database folder or to the correct folder in the iBase installation.

In the File Manager:

- Click **Load** to add a file to the database. If you are loading a database template, you must first delete any existing database template that is listed in the File Manager. The addition will be replicated to the other sites. To refresh the list of loaded files, press the F5 key.
- Click **Edit** to enter notes or instructions on using the loaded file that will assist administrators at the other sites. The description, or any changes to the description, are replicated to the other sites.
- Click **Delete** to remove any unwanted files from the database. The deletion will be replicated to the other sites. This will not delete any files that have been saved to disk; you will need to ask the administrators at the other sites to do this manually.

© N. Harris Computer Corporation

Since any iBase administrator can delete a loaded file, and because most files are not usable until they have been saved to disk, save the file by clicking **Save to File**.

Updating the schema of a replicated database

Changing the schema of an iBase database requires the SQL Server administrator to stop replication for all sites while you apply an updated database template to the database at each site. Before replication is stopped, you need to prepare the database by reviewing all existing conflicts and then take the database offline. After the schema changes are applied at each site, the SQL Server administrator must reconfigure replication.

Before you begin

You need to prepare iBase so that replication can be stopped by the SQL Server administrator. This needs to be done in the publication database as the Conflict Viewer is only available at the Publisher:

1. Log on as a database or system administrator and open the publication database.
2. Using the File Manager, distribute the database template that contains the revised schema to the other sites. The template must remain loaded in the database.
3. Ask any users that are using the database to log off.

   Note: If you use iBase Scheduler, disable the Scheduler service, after this point, you must not allow any imports to run.
4. Check that all database connections are closed. For example, if you can log on to the remote servers, try to open the database in iBase Designer. If you can open iBase Designer, then you have exclusive access to the database at that site. If necessary, ask the SQL Server administrator to log out the remaining users.
5. Review the current conflicts in the Conflict Viewer. If you change any of the records that are involved in the conflicts make a note of the last record that you change and the nature of the change.
6. Check that any records modified as a result of using the Conflict Viewer are replicated to all the subscription databases.
7. Ask the SQL Server administrator to disable replication.
8. When the SQL Server administrator informs you that replication is disabled, open the publication database and run the Status report to confirm this. If replication is disabled, it reports `Publication not found`.

You are now ready to apply the schema change.

Warning: After replication is disabled, you must not make any changes to the data in any of the databases. The changes will not be replicated to the other sites when replication is reconfigured in SQL Server.

About this task

Once the SQL Server administrator informs you that replication is disabled, you can apply the new schema to each database by using **Update Database Schema** at each site. When you run this command, iBase closes and then reopens the database to gain exclusive access to it.

You must apply the same schema to all the databases. **Update Database Schema** displays the differences between the schema in the currently loaded template and the schema of the current database. The displayed additions, modifications, and deletions are applied to the current database.

© N. Harris Computer Corporation

Procedure

To apply a schema change at each site:

1. Ensure that you have a backup of the database.
2. Select **Tools** > **Replication** > **Update Database Schema**. iBase closes and reopens the database before displaying **Update Database Schema**. **Update Database Schema** is not displayed if you are a member of a Data Access Control group that denies access to any tables or fields in the database.

   Note: If you are warned that there is no database template, load the correct template by using the File Manager. However, do not load any other files into the database as all the files listed in the File Manager will be overwritten once replication is reconfigured.
3. If required, save a list of schema changes in a format that is useful for the SQL Server administrator at your site. You can print this file later.
4. Click **Update** to apply the changes and then, once this is finished, click **OK** to reopen the database.
5. At each site, check that the template has been applied successfully. Once all subscriber sites have applied the schema change, notify the SQL Server administrator that replication can be reconfigured. When the SQL Server administrator informs you that replication is running again for all the sites:
6. Verify that replication is running, for example, by running a status report. You may want to test that replication is running correctly.
7. Open the publication database and make the database available again to other users, select **Tools** > **Replication** > **Bring Online** . After a short interval, the status is replicated to the Subscribers.
8. Notify users that they can start work again.
9. If you use iBase Scheduler, restart the Scheduler service.

Shutting down replicated database

You can shut down a replicated database in order to gain exclusive access to the databases before you apply changes to the database schema or because the SQL Server administrator needs to stop replication for some other reason. Once replication is stopped, users must not add, edit or delete data and this command will prevent users from logging on until you choose to take the databases online.

About this task

To make it easier to gain exclusive access to the database, you can broadcast a message to all active users in iBase asking them to close the database. You can broadcast the standard message (which is `*** WARNING *** This database is now offline`) or add an additional message to provide users with some instructions or information. Existing users are not ejected from the databases by this message but new users (except for database administrators) cannot open the database.

Provided that there are no communication failures between any of the sites, taking one database offline, or online, will take all the databases offline, or online, as the command is replicated to all the sites.

Note: Shutting down will not deny database access to services such as the iBase Scheduler service. You must disable these manually. Disable a service rather than stopping it as this will prevent it from restarting if the server is rebooted.

Procedure

To send a message and take all the databases offline:

© N. Harris Computer Corporation

1. In the Take Offline dialog, enter an additional message (if required) and select the frequency with which the message is displayed. When selecting a frequency, you may need to leave time for the message to be replicated to the other sites.
2. Only users who are actively using iBase will see the message. For this reason, if users have not exited from the databases after a certain interval of time, ask the SQL Server administrator to log them out.

   Note: To change the frequency with which the message displays or to change the message, you need to select **Tools** > **Replication** > **Bring Online** and then reselect **Take Offline**.

What to do next

When the databases are offline, select **Tools** > **Replication** > **Bring Online** in order to bring all the databases back online thereby making them available to users wanting to open them.

You can bring the publication database online as soon as your SQL Server administrator tells you that replication is configured for the publisher site. The subscription databases will automatically come back online as the subscriber sites are configured for replication.

Warning: Do not bring the publication database online while replication is disabled. You should only add data (whether manually or by scheduled import), or edit or delete data while replication is running. This is because any changes that are made to the data while replication is disabled cannot be replicated to other sites even when replication is reconfigured.

Checking whether replication is running

To obtain a report on the replication status of the database, you should first open the publication database.

The following messages are reported for the current database:

Message	**Description**
Publication is OK	When run in the publication database, this indicates that the Publisher is configured for replication.
Publication not found	Replication is not running at the Publisher. If you run this report while in a subscription database then information on the Publisher, or whether replication is running, is not available.
The publication is invalid because it allows anonymous subscriptions	When run in the publication database, this tells you that replication is incorrectly configured in SQL Server. Contact your SQL Server administrator. (**Note:** You will not be able to review and resolve any conflicts until this problem is fixed.)
`<server>\<instance>:<SQL Server database name>`	When run in the publication database, this shows you the names of the subscriptions to this publication database.

© N. Harris Computer Corporation

Message	Description
No subscriptions	When run in the publication database, this indicates that the subscriber sites are not yet configured for replication.

Note: The Status report does not report any problems with the communication links between Subscribers and Publishers, or any problems with the replicated security connection file or audit log.

Working with other applications

In addition to the standard features that are present within iBase, integrating with other i2 products can be used to reveal relationships, patterns and trends within the data.

The following i2 products can integrate with iBase:

Analyst's Notebook

When you have both Analyst's Notebook and iBase, you can transfer information between them to visualize and store data. Data can be kept in a centralized location and shared between charts.

You can transfer data between iBase and Analyst's Notebook in three ways:

- Add data that is stored in iBase to an Analyst's Notebook chart.
- Add data to both a chart surface and to an iBase record simultaneously.
- Extract chart data from Analyst's Notebook items to be stored in iBase.

Note: Extracting data is available in the iBase Chart Item Extractor.

To reestablish the connection between an Analyst's Notebook chart and an iBase database, reopen the appropriate database. To reopen a database, select it in **Data Sources**, if required enter your iBase username and password, and click **OK**.

Charting iBase records

When working in Analyst's Notebook, you can chart iBase data provided that iBase is installed. If iBase is installed, then the Data menu will have an i2 iBase command.

Connecting to iBase as an Analyst's Notebook data source

Before you can chart with iBase data in Analyst's Notebook, you need to open the database that contains the required data. The database might be open if you have already added records from this iBase database to a chart, or if you have opened Analyst's Notebook from iBase.

About this task

You can open several iBase databases at the same time, but only if they share the same security file.

Procedure

To open an iBase database in Analyst's Notebook:

1. Start Analyst's Notebook.
2. Click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Open Database**.
3. Find the required database, and click **Open**.

© N. Harris Computer Corporation

4. Enter your iBase username and password, and click **OK**.

Creating association charts

By default, when charting from iBase an association chart is created with iBase entities represented as icons or pictures.

You can add iBase records to an Analyst's Notebook association chart by selecting, for example, **Find** on the New page of the Data Sources Task Pane.

When you are working with association charts, you need to set the Association or Custom Representation charting style. For more information, see Specifying general options.

Creating timeline charts

When charting iBase items you can choose to create timeline charts rather than association charts. On a timeline chart, entities are represented as event frames and entities without dates and times as theme lines.

To create a timeline chart, you could add the event frames to the chart first, and then expand to add the theme lines or vice versa. Alternatively, you may create the chart using the **Timeline Assistant** and then add further items to the resulting chart.

Before you can create a timeline chart, you require a charting scheme that at least defines how date and time properties are to be derived from iBase entities or links. For instance, if you specify that a Telephone Call link should use Start Date and Start Time as its date and time information then, when you chart telephone calls, the telephones will appear as theme lines and the telephone calls will be arranged in chronological order.

Note: Links are always charted using the From Database and Multiple options (in charting settings). This ensures that timed links are placed at their correct location and can be merged in the chart.

You can either create a timeline chart by using the Timeline Assistant or, if the charting settings are set up correctly in Analyst's Notebook, by using the standard **Add to Chart** and **Expand** options on the shortcut menu.

Using the Timeline Assistant

You can use the timeline assistant to specify the chronological order of events that are associated with items contained in a set or the results of a query. Using the timeline assistant, you define the timed item type, which forms the main subject of the timeline, and the charting scheme that will determine the details that are displayed on the chart surface.

Procedure

1. Create the set or query to define the records that you want to chart on a timeline.
2. Select **Analysis > Analyst's Notebook > Timeline Assistant**.
3. Select the entity or link type that will be main subject of the timeline by selecting from the **Timed Item Type** list.
   - If you select an entity type, the entities of this type will be represented by event frames on the chart. You can add linked items that will be represented as theme lines.
   - If you select a link type, the entities at the ends of the links are represented as theme lines on the chart.
4. In the Linked Items area, click **Add** and choose the link and link end combinations you want to represent as a theme line.

   Note: This is unavailable if the timed item type is a link type.

© N. Harris Computer Corporation

5. In the Records area, select **Query** or **Set** and choose a query or set to provide the records of the timed item type to add to the chart.
6. If you selected a query, you can turn on **Filter linked items using query** in order to use the selected query to filter the records for the chart. It is only valid if you have set up one or more link/link end combinations in the Linked Items area.

When turned off	When turned on
The queries you can choose from are restricted to those that output 'Timed Item Type' entities.	The queries you can choose from are further restricted to those where a part of the structure matches the 'timed item type - link type - linked entity type' combinations that you set up in this dialog, where 'timed item type' must be one of the query's output entities, and all the combinations involve that same entity.
The query is only used to select the timed entities, the link and linked entity part of the query structure is not used as a filter.	As well as the query selecting the timed entities, the associated part of the query structure (the link and end entity) is also used to filter the included link and linked entity records.

As an example, you might have a query structure part: `{Person entity} - {Involved link} - {Crime entity (condition 'type is Theft')}`

When turned off	When turned on
The chart would include all the people involved in a theft crime, along with all their crimes, whether they were thefts or not.	The chart would include the same people, but would now include only their theft crimes.

7. Select the charting scheme to use or click **New** to create a new scheme. As a minimum, the charting scheme must specify how the date and time are populated for the timed item type.
8. In the Options area, you may choose to **Separate simultaneous events by milliseconds** but note that this may mean certain analysis functions do not give the expected result. For detailed information on using this option, see Specifying general options.
9. Turn on **Add to current chart** to add the records in the set or query to the current chart. Turn off the checkbox to add them to a new chart.

What to do next

After creating a timeline chart, you can add other iBase items by selecting **Find** or one of the **Expand** options from the shortcut menu for the iBase database.

Note: When creating timeline charts in this way, you need to set the Timeline or Custom Representation style in the Charting Settings dialog. For details, see Specifying general options.

Closing a connection to an iBase data source

© N. Harris Computer Corporation

If you want to open an iBase database that uses a different security file, you must close the database connection. You can close the connection directly by using the **Data Sources Task Pane**.

Procedure

To close the iBase database, in the **Data Sources** page of the **Data Sources Task Pane**, close a database by clicking **Close** to the right of the database name.

Create iBase chart items

You can create chart items that contain iBase records. These records can contain information that you receive from other sources, or information that you assert based on your analysis.

When you create a chart item that contains an iBase record, the record exists in both the database and in the chart item on your chart.

iBase chart items are created using the New page of the Data Sources pane. To access the Data Sources pane, click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Datasources Pane**.

Note: The New page is only displayed when one or more databases are opened in the iBase session.

Selecting Association and Timeline option combinations

To get good results when charting iBase records, you need to define what aspects of the iBase record the chart items should reflect. You do this by specifying the charting style, charting scheme, and labeling scheme.

About this task

Charting Scheme

How information that is stored in iBase records is converted into Analyst's Notebook chart item properties.

Charting Style

The type of chart that you are trying to create.

Labeling Scheme

What information is used to construct the item labels.

You can store up to eight preset combinations, and if these combinations are set up, you can select a combination from the **Association & Timeline Options** list.

The name of the current combination is saved with the chart. When the chart is reopened, the combination is found and selected if possible.

Note: If iBase displays **Custom** in the Data Sources **Association & Timeline Options** list, and the **Edit** button is not enabled, this is because your current options differ from those previously selected. For example, you are working with "Combination 1" which has the charting scheme set to "Charting Scheme 1". You then chart some iBase items with a charting scheme of "Copy of Charting Scheme 1". This differs from the charting scheme set in the active options for "Combination 1", so the Data Sources pane displays **Custom** rather than "Combination 1" because a difference in charting scheme is detected.

Procedure

To edit a combination, click **Edit** to the right of the list. For more information about selecting a charting scheme, charting settings, and labeling schemes, see Editing iBase combinations.

© N. Harris Computer Corporation

Editing iBase combinations

iBase combinations can be used to group charting schemes, charting styles, labeling schemes, and connection multiplicity options in reusable combinations. These combinations can then be used when items are sent to a chart surface

Before you begin

Before you can specify an iBase option combination, the charting schemes and labeling schemes to select must be available:

- Creating charting schemes
- Labeling Schemes

Procedure

1. To access the Data Sources pane, click the **Home** tab, and then in the **Data Sources** group, select **Connect > Datasources Pane**.
2. On the New page of the Data Sources Task Pane, select a combination from the **Association & Timeline Options** list and click **Edit**.
3. Provide a **Preset Name** for the option combination.
4. Select the charting scheme.
5. From the **Charting Style** list, select a charting style:
   - **Association** - all entities are represented as icons.
   - **Timeline** - all entities with date and time settings in the charting scheme are represented as event frames and other entities are represented as theme lines.
   - **Custom Representation** - you can choose how each entity type is represented.

     To specify how entity types are represented in a custom representation:

     a. Click **Custom Representation**.

     b. Select an entity type in the **Type** column, and in the **Representation** column, select a representation from the list. This can be Icon, Theme Line or Event Frame.

     c. Click **OK**.

   Note: For timeline charts, if your data includes simultaneous items, you can turn on **Use Milliseconds to Separate Simultaneous Events**. See Separating simultaneous items.
6. Set the Entity and Link labeling schemes.

   Labeling schemes determine how the label that identifies and represents an entity or link is derived from the fields in the record. For example, person records may have a label comprising the Last Name field together with the First Name field. There will be one definition for the label to be used with iBase records, and one to be used when a record is converted to a chart item. For more information see Labeling Schemes.
7. Set the **Connection Multiplicity**:

Option	Description
Single	The connection between two entities is shown as a single link. This single link represents an association between two entities, which may in

© N. Harris Computer Corporation

Option	Description
	reality comprise many different instances of that association.
Directed	The connection between two entities is represented by one link for each possible direction. A directed connection has therefore a maximum of four links. Each of those links may represent several instances of the association, all in that same direction.
Multiple	Each link between two entities is displayed individually.

Note: If you subsequently add a link to a connection that has a connection style of Single, the style is automatically promoted. For example the connection style of the link will be updated from Single to Directed. For further information about adding a link to a connection, see the Online Help for Analyst's Notebook.

8. If you set the Connection Multiplicity to directed or single you can make the label be the count or sum of the labels of all the link records in the connection.

   - **From database**

     The label for the chart link is the chart label as specified in the default labeling scheme.

   - **Type name**

     The label for the chart link is the source iBase record link type.

   - **Occurrences**

     The label for the chart link is the number of iBase links that it represents.

     This is only relevant when Multiple Link Style is set to Single or Directed, when a chart link might represent several iBase links.

   - **Sum Numeric**

     If the iBase label for a link has a numerical component, then the values of this component, for all the iBase links represented by the chart link, are summed. The resulting number is used as the chart link label. The direction of links is taken into account; values are added if the links are in the same direction, or subtracted if they are in opposite directions.

     For example, Financial Transaction links might have an iBase label containing the value of the transaction, and there may be two links of this type between two bank account entities, one of $2000 and one of $1000. On the chart both of these links might be represented by one link, which will have a label of either $3000 (if both links are in the same direction), or $1000 if the links are in opposite directions.

     This option becomes relevant when Multiple Link Style is set to Single or Directed, when a chart link might represent several iBase links.

Separating simultaneous items

© N. Harris Computer Corporation

Analyst's Notebook can chart items with a date and time as controlling items. The horizontal position of controlling items on the chart is in relation to the flow of time from left to right.

If two or more items have the same date and time, they can display on top of each other. When you want to inspect these events visually, for instance looking for groups of such items close together, it is useful to separate them by a small time increment so that they are not displayed on top of each other.

If you turn on **Use Milliseconds to Separate Simultaneous Items**, the interface automatically adds a millisecond to the time of each chart item with an identical date and time. If there are several chart items with the same time the first item is unchanged, the second is one millisecond later, the third two milliseconds later. This ensures that all items are visible on the chart.

If you do not turn on**Use Milliseconds to Separate Simultaneous Items**, the items appear at the same horizontal location on the chart. This is indicated on the time bar by the marker symbol, which uses the overlap color to indicate that there is more than one item at this specific time location in the chart.

To view the time of events separated by milliseconds, your time format must be set to display seconds (for example, HH:mm: ss). However, the time bar shows the full time of events regardless of the time format setting.

Important: Adding milliseconds to the time might mean that certain analysis functions do not give the expected result. For example, if a millisecond is added to an event time of 4:00 PM then a search for events between 3:00 PM and 4:00 PM will not find the event.

Creating charting schemes

A charting scheme controls how the properties of items that are added to Analyst's Notebook charts that use iBase records are updated. Each charting scheme defines the chart item property mapping, the chart template, the labeling scheme, and whether to display attributes and pictures on the chart.

About this task

Charting schemes define the mapping that is used to convert property values on iBase records into a format that can be used within Analyst's Notebook. These property mappings can either be for any entity or link type in the database, or specific chart item properties for each entity or link type in the database. For example, you might want to display attributes and pictures on the chart for only a few of the entity types in the database.

In Analyst's Notebook, you can select which charting scheme is active at any time. You can also override the labeling scheme that is specified in the active charting scheme by selecting a different labeling scheme from Analyst's Notebook.

Procedure

To create a charting scheme:

1. In iBase User, select **Format > Charting Schemes > New**. A blank charting scheme is generated that lists the available system-wide defaults.
2. Add the charting scheme properties:

Option	Description
Labeling Scheme	The labeling scheme that defines how labels are displayed on an Analyst's Notebook chart. If you do not select a scheme, the iBase default is used.

© N. Harris Computer Corporation

Option	Description
Chart Template	The chart template on which to base new charts when charting from iBase, expanding to a new chart or when you use the Timeline Assistant. You can choose from templates that are stored in the same folder as the database file. The template filename must be prefixed with the `idb` filename, for example if the database file is `User Guide.idb`, the template filename must start with `User Guide`. If you do not specify a chart template, the Analyst's Notebook default template is used.
Restore Layout Settings	Whether any layout settings that are written to the chart are restored when iBase items are added to a chart.
Chart Attributes	Whether to create chart attributes for any of the entities or links that are sent to the chart. (The database design defines which fields, if any, can be sent as chart attributes). If set to Blank, the **Tools > Options** setting is used as the default.
Chart Pictures	Whether any entities that are sent to the chart are represented as pictures or icons. If set to Blank, the **Tools > Options** setting is used as the default.
Data Records	A data record displays the associated iBase record for the chart item as part of the chart item properties, including the semantic type.

3. **Optional:** For the other system-wide default mappings, specify the default behavior:
   - **Field** - the value is taken from a specified field value.
   - **Value** - a value is used.
   - **Blank** - no value is used.
4. **Optional:** For any item type-specific default mappings:
   a. Select the item type, and click **Add**.
   b. Select the property type and assign the default behavior:
      - **Field** - the value is taken from a specified field value.
      - **Value** - a value is used.
      - **Blank** - no value is used.

   Note: Any values added to a specific item type will be used instead of any system default for the same property.

© N. Harris Computer Corporation

5. **Optional:** For a specified item type, you can also select from the following to override the default setting:
   - **Chart Attributes** to choose whether chart attributes are created from iBase fields. Any entity or link types where the parameter is set to **No** are not sent to the chart. Change the value to **Yes** to create attributes for those fields, and send them to the chart.
   - **Chart Pictures** to choose whether to use pictures instead of entity icons.
   - **Data Records** to choose whether to create a data record.
6. Click **Save**. When prompted, enter a name and category for the charting scheme, and specify the access level.

Labeling Schemes

Labeling schemes determine how the label that identifies and represents a record is derived from the fields in the record. For example, person records label might comprise the surname together with the given name, while for vehicle records the label might comprise the manufacturer, model and registered number. You can also include 'free text' in the label, text that does not vary between labels.

About this task

For each item type in a labeling scheme, you can set the label to be used with iBase records, and the label to be used when records are converted to chart items.

Each labeling scheme covers all the entity and link types in the database. If there is more than one labeling scheme, it is because you might want different label formats at different times. You specify which scheme is in use by making it the 'default'.

Note: For certain entity types, Smart Matching in Analyst's Notebook assumes a property semantic type for the label value. Since the semantic type that is assumed cannot be changed, specify an entity type's label to be a property with a suitable semantic meaning. Consider that for some entity types, the assumed property semantic type is a Details kind. The label is then parsed during Smart Matching to locate the various parts of the textual value. Do not assign a Details kind of semantic type to a property that is used for a label. Instead, specify a property that is a part of a Details type, and assign the semantic type with the correct specific meaning. For example, for a Credit Card, specify the label to be the Card Number property, which is a part that is located in a Bank Card Details property. For more information, see the topic Assumed Semantic Types for Labels.

Procedure

To create a labeling scheme:

1. Select **Format** > **Labeling Schemes** > **New**.
2. Select the entity or link type to label.
3. Display the page for the type of label you want to define.
   - **Standard** for the label to be used for iBase records.
   - **Chart** for the label to be used for chart items.
4. Select fields and text you want to include in the label.

   Note: You can see a **Sample** label. Click **Next** and **Previous** to change which record the sample is based on.
5. Repeat these steps until you have defined standard and chart labels for all the entity and link types as required.

© N. Harris Computer Corporation

6. If you have more than one labeling scheme, you must set one to be the default, that is the scheme that is in use at any particular time. To select a labeling scheme as the default: In the Database Explorer window, right-click on the labeling scheme and select **Set as Default**.

Creating iBase entities

iBase entities store their important information in records. To create entities that contain records, create your entities using the Data Source Pane.

About this task

In a chart, the first time that you connect to an iBase database, a new palette is created that contains all of the types in your database. You can use the pallette to select an entity type to create. Alternatively, you can create entities using the **New Entity** option on the task pane.

When you create an iBase entity, that item contains an iBase record. You can enter values for some or all of the record properties, and you must enter values for mandatory properties.

Procedure

1. Open the Data Sources pane, click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Datasources Pane**.
2. Click **New Entity** and select the type of entity from the list provided.
3. Enter the information about the entity into the form provided.

Creating iBase links

iBase links store their important information in records. To create links that contain records, create your links using the Data Source Pane.

About this task

To create an iBase link, the entities that you connect must contain iBase records, and the link type that you use must be permitted between the two entities according to the rules of the database. For example, it might be permitted to connect two people by an association link but not by an owner link.

When a chart item contains multiple iBase records, such as when two iBase entities are merged, one of the records is assigned to be the lead record. When you create an iBase link that is connected to an iBase entity, the link record is connected to the entity's lead record.

Procedure

1. Open the Data Sources pane, click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Datasources Pane**.
2. Select the two entities to be linked on the chart surface, and click **New Link**.
3. Select a link type from the list of available link types that are relevant to the selection.
4. Enter the information about the link into the form provided.

Adding iBase items to a chart

You can add data from one or more iBase databases to an Analyst's Notebook chart.

You can:

- Find records in an iBase database.
- Add records from an iBase set.
- Add the results of iBase queries.

© N. Harris Computer Corporation

How the data is displayed depends on the current charting settings. For example, an entity might show as an icon, an event frame, or a theme line.

Each iBase item on a chart has a reference back to its source record. When the database is open, double-clicking the item displays an iBase dialog for viewing, editing, or deleting the item. When the database is closed, double-clicking a chart item, displays the Analyst's Notebook Edit dialog.

Searching iBase

If you would like to add items to your chart from an iBase database, you can search for the items directly from Analyst's Notebook.

Procedure

1. To access the Data Sources pane, click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Datasources Pane**.
2. Select the type of search to complete:

Option	Description
Find	Find one or more entities in iBase.
Search 360	Type the search terms that you would like to use in the textbox that is provided and press enter to search the database. (**Note:** Search 360 is only available in databases that have been configured to use it.)
Query Manager	Lists all the existing queries in the iBase database.
New Query	Define a query.
Set Manager	Lists the existing sets in the iBase database.
Timeline Assistant	Create, or add to, a timeline chart.

Adding the results of a query or set

The results of queries or items in sets in your iBase database can be added to your chart. You can open the set or query and modify the contents before you add the items to the chart.

Before you begin

You must have a relevant query or set available:

- For more information about creating a query, see Querying your data.
- For more information about creating sets, see Sets.

About this task

You can use the Query Manager and Set Manager options to interact with the queries and sets that are present in your database.

© N. Harris Computer Corporation

Procedure

1. To access the **Data Sources pane**, click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Datasources Pane**.
2. Select whether to add the results of a query, or the contents of a set:
   - To view the available queries, select **Query Manager**.
   - To view the available sets, set **Set Manager**.
3. Select the query or set to use and either review the items, or add them directly to the chart:
   - If you want to review the definition of the query, review the records found by the query, or see the contents of the set, click **Open**. Then, when you close the query or set, you are prompted to add the records to the current chart.
   - If you do not want to review the query or set, click **Chart**. The records are added to the current chart.

Expanding and finding related information in iBase

When iBase items are present on a chart, you can look for connected information in the database. You can also update the chart items to reflect any changes to the records in the database.

Procedure

1. To access the Data Sources pane, click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Datasources Pane**.
2. **Optional:** Set the item types that you are interested in retrieving from your database:
   a. At the top of the Charting section, click either **Filter Entity Types** or **Filter Link Types**. The lists display the entity or link types in the iBase database.
   b. Select the entity or link types that you to use on the chart.
   c. Click **OK**.
3. Select the items on your chart that you would like to investigate and select an option:
   - **Expand**

     Expand one or more iBase entities on an Analyst's Notebook chart in order to chart their links and link end entities. The number of links and linked entities that are shown (the expansion levels), and the types that are shown, is determined by the current charting settings. By default, only the links from the selected entity and their link end entities are shown.

     Click **Expand**. This adds the entities and links to the current chart. The entities and links that are added are determined by the current charting settings.
   - **Expand to New Chart**

     Expand the selected items onto a new chart. The entities and links that are added are determined by the current charting settings.
   - **Expand with Settings**

     Expand with custom settings, such as specifying the entity and link types to use, the expansion level, and link styles.
   - **Explore**

     Interactively browse through your database, expanding from one entity to the next. While you are exploring, you can choose to concentrate on specific entity and link types and ignore others. This can help to simplify the information you are trying to understand at any given time.

© N. Harris Computer Corporation

- **Find Path**

  Find the chain of intermediate links and entities that connect two entities on the chart, taking the link direction into account, and selecting the shortest path (depending how the charting settings are set). A path may not be found if the number of intermediate entities exceeds a preset maximum number.

  Note: If **Follow direction** is turned on, then select the entities in the correct order - the first entity you select is the 'from' entity, the second entity you select is the 'to' entity.

  If a path is found between the two entities, additional information regarding them is added to the chart. All the items are selected so you can tidy up the chart by dragging them to a new area of the chart.

- **Find Connecting Links**

  Find all the links that connect two or more entities on the chart. The number to the right of the **Find Connecting Links** displays the number of items on the chart that are already connected.

- **Find Common Neighbors**

  Find any entities that have connecting links to at least two of the selected chart items.

- **Find Common Neighbors Linked to All**

  Finds any entities that have connecting links to all of the selected chart items.

- **Populate Cards**

  Populate cards for items selected on the chart. For more information, see Populating cards.

Exploring items

When you **Explore** an item, a Link chart is displayed that shows any links to that item. This information can subsequently be added to the Analyst's Notebook chart.

Procedure

1. Select an item on your chart surface that contains iBase records.

   Note: If you select an item that has more than one record you will be prompted to select the record to use.

2. In the Data sources pane, on the **Selection** tab, select **Explore**. A link chart will display that contains all the linked records in the database.
3. Investigate the record to find related items of interest:
   - To filter the displayed items, select the item types to be displayed from the Entity Type and Link Type lists.
   - To explore a linked entity, select the entity in the explore window, this will change the focus of the exploration, expanding the links to the selected entity.
   - To add items to the chart, select them in the explore window and click **Add to chart**.

Mapping iBase items

If you have i2 iBase Geographic Information System (GIS) Interfaces configured to connect to mapping applications, you can map iBase records from Analyst's Notebook using your iBase mapping provider. This mapping is specific to the iBase records, and differs from the inbuilt Analyst's Notebook mapping options that are also available for items that contain iBase records.

Note: For information about the Analyst's Notebook mapping options, see Mapping chart items in the Analyst's Notebook documentation.

© N. Harris Computer Corporation

Mapping information held in iBase records using the iBase GIS Interfaces is very dependent on the mapping application that has been selected, and indeed the configuration that you use to display item information within that application. From Analyst's Notebook, there are two options for interacting with your mapping application:

- **Add to Map** - Checks the current selected items for valid coordinates, prompts you to select a mapping configuration, before displaying your items in the mapping application.
- **Select on Map** - Does the same validation as **Add to Map**, but changes the display of any items already added to the mapping application to indicate they are significant.

  Note: Any items that are selected that are not already available in the mapping application will be ignored along with any that do not contain coordinates.

Note: These options are present both in the Map section of the Selection tab of the Data Sources Pane, and from the shortcut menu if you have an item with valid coordinates selected.

For information about setting up configurations to mapping applications in iBase, see i2 iBase GIS Interfaces.

Updating iBase chart items

iBase records that are added to Analyst's Notebook chart are disconnected from the versions that are stored in the iBase database. This allows you to act on the data that was available at the time the item was added to the chart, but might lead to key future updates being overlooked. If necessary, you can refresh your iBase records, updating to later versions where applicable.

In addition to updating to the latest version of an iBase record, and editing the record directly, the following options that are available in Analyst's Notebook modify the records in the iBase database:

Add to Set

Adds the selected items to either a new or existing set. When you select **Add to Set**, you will be prompted to either:

- **Create new set using the name**, specifying the name to use for the set.
- **Append the records to the set**, specifying the set to add the records into.

The entities remain selected so that you can, for example, reposition them on the chart.

This command is available:

- from the **Database** section of the Selection tab of the Data Sources task pane.
- by right-clicking on a chart item, and from the shortcut menu, select **database > Add to Set**, where database is the name of the iBase database.

Merge in iBase

Merges iBase records in selected chart items in an iBase database.

Right-click on a chart item, and from the shortcut menu, select **database > Merge in iBase**, where database is the name of the iBase database.

Note: This is different from merging chart items. For information about merging chart items, see Resolve duplicate data that you identify.

Refreshing chart items

© N. Harris Computer Corporation

Analyst's Notebook charts contain a standalone copy of any iBase records rather than a connection to the link within the database. This means that changes that are made in the database will only be reflected in a chart when requested.

About this task

When you are currently connected to the iBase database that contains the records of interest, you can open items on your chart and edit the iBase records directly on an individual basis. You can also ensure that either all, or a specific selection of iBase records that are embedded in a chart are at the latest versions. This is a manual process to ensure that the data that you are using for your analysis doesn't change unexpectedly.

Procedure

To check for updated information in your iBase database:

1. To access the Data Sources pane, click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Datasources Pane**
2. From the Data Sources **Selection** > **Update Chart**. A list of all the items on the chart that have newer records in the connected iBase database are listed.
3. Select the items that you would like to refresh on the chart, any additional options that you would like to specify. By default, selected items with updates to the records will just receive the latest version of the record but in addition you can:
   a. Specify a different charting scheme to change the way iBase record values are used within the chart representation.
   b. Change the labeling scheme to update the chart labels with different field information.
   c. Specify the action to take for records that have been deleted in the iBase database:
      1. Remove - removes the iBase record from the chart item.
      2. Hide - Set the item to be no longer visible on the chart surface.
   d. Specify the action to take for iBase records that have been merged in the iBase database:
      1. Remove - removes the iBase record from the chart item.
      2. Hide - Set the chart item to be no longer visible on the chart surface.
      3. Update from new item - Replace the iBase record that is attached to the chart item with the iBase record that was updated in the merge operation.
4. Click **Update**.

Working with merged chart items

On a chart, you can merge two or more entities or links together, whether from the same or different databases. If you merge items from the *same* database, you can also choose to merge the records in iBase as a second step.

- Merging entities on the chart

  Select the entities that you want to merge, and select **Analyze > Merge and Combine > Merge Entities**. The merged records are shown on the chart as a single entity and treated as a single 'real-world' object, for example when finding the path between entities or expanding. Any links between the merged chart items are not displayed.
- Merging links on the chart

© N. Harris Computer Corporation

Select the links that you want to merge. Links must have the same ends. Select **Analyze > Merge and Combine > Merge Multiple Links**. The selected links are merged into one on the chart.

- Merging records in iBase

  With an item selected on the chart that contains at least two iBase records from the same database, go to the Selection tab of the Data Sources Task Pane and select **Merge in iBase**. Information from the records being merged is added to the primary record before the records are deleted. Any links will either reference the primary record, or if invalid, will be deleted.

  For links, follow the same process, select **Merge Links in iBase**, then define a record to be the target of the merge.

Merging items on the chart surface before merging the records in iBase keeps the records in the properties of the merged chart item, and preserves the record ID (database key).

Note: To see the record IDs of the merged chart item, right-click on the merged chart item and from the shortcut menu, select **Edit Item Properties**. The keys are shown in the Data Records folder for the database.

Behavior of merged chart items

Commands such as Expand, Find Path, and Find Common Neighbors behave as you would expect. For example, expanding a merged chart item will expand all the links and link end entities associated with all the entities in the merged chart item, and finding a path will take any additional links and link end entities into account.

All the entity types in a merged chart item are used when expanding, searching and so on, including any entity types that are excluded as part of the charting settings. For example, when finding a path between a Person entity and a merged Person/Vehicle entity, where the Vehicle entity type is excluded, the links belonging to the Vehicle entity are considered.

The Show and Explore commands are slightly different because they apply only to single iBase records:

Option	Description
Show	When you select Show for a merged chart entity, the Show List dialog is displayed. Select the record that you want to show.
Explore	When you select Explore, the Select a Record dialog is displayed. Select the record that you want to examine in the Explore window. This is a useful way of finding out which links are associated with which component of a merged chart item.

Note: You cannot update the chart when you edit the iBase record of a merged chart item.

Merging entities with different semantic types

You may want to merge entities that have different semantic types. For example, to associate an organization with its web address, or a person with a vehicle. In this case, the merged chart item is assigned the semantic type that is the lowest common parent of their semantic types. Depending on the

© N. Harris Computer Corporation

semantic types of the individual chart items, this may be an abstract semantic type, such as the Entity semantic type.

If the semantic type of the merged chart item is not useful to you, you can assign a different one as part of its properties. For example, you might choose to assign the Organization semantic type to the item created by merging Organization and Web Address entities.

Viewing item information

In the Selection tab of the Data Sources task pane, there is a View section. Use the commands in the View section, for example, to Show records, edit item properties, open hyperlinks and find matching records.

Select Items from Current Database

Selects chart items that are from the currently-connected database.

Filter Selection by Query

You can filter the currently selected chart entities or links using either an existing query or a new one that you define specifically for this task. After filtering, only those items within the current selection that match the query conditions will remain selected on the chart.

Click **Filter Selection by Query** to display the Filter by Query dialog the currently selected chart entities or links using either an existing query or a new one. Either double-click on an existing query to run it, or click **New Query** and start a new query.

Alternatively, select more than one chart item, and from the shortcut menu, select **database > Filter by Query**, where database is the name of the iBase database. The Filter by Query dialog is displayed. Either, double-click on an existing query to run it or click **New Query** and start a new query.

When you click **OK** or **Finish**, only the entities that are in both the chart selection and in the results for the query remain selected on the chart. If there is no match, nothing is selected on the chart.

Show

Click to display the Show dialog for a selected item, or the Show List dialog for more than one selected items.

Alternatively, right-click on the iBase entity and from the shortcut menu, select **database > Show** or select several chart items and select **database > Show List**, where database is the name of your iBase database.

Links

Displays the iBase Links dialog.

Alternatively, right-click on an entity on the chart, and from the shortcut menu, select **database > Show**, where database is the name of your iBase database. Then click the **Links** button to display the Links dialog.

Edit Item Properties *or* Combined Properties

Click **Edit Item Properties** to display the Item Properties dialog for the selected item, or Combined Properties to display the Combined Properties dialog for more than one selected item. These are Analyst's Notebook dialogs.

Alternatively, right-click on a chart item, and from the shortcut menu, select **Edit Item Properties**, or select several chart items, and from the shortcut menu, select **Combined Properties**.

Open Hyperlinks

© N. Harris Computer Corporation

If the entity type has a hyperlink field, then you can find any other entities or documents to which it is linked by a hyperlink. The hyperlink may be to a document stored on a file server or a website.

Alternatively, right-click on one of the selected entities and select **database > Hyperlinks**, where database is the name of the iBase database.

Matching Records

Finds any entities that have connecting links to two or more entities, or to all entities, on the chart (depending how the charting settings are set).

Alternatively, right-click on one of the selected entities and select **database > Matching Records**, where database is the name of the iBase database.

Specifying general options

You can define the default behavior to use when you are working with iBase chart items, on the **Options** page of the **Data Sources** pane.

General options

Select an option to specify what happens when you double-click items on the chart:

Option	Description
Open their Properties Dialogs	Opens the Analyst's Notebook item properties. The iBase record details are listed as an attached card.
Open their iBase Show dialogs	Opens the iBase record for a selected item, or the list of records for more than one selected items.
Expand the Items	Expands the selected items from the iBase database.

Session defaults

You can set a session default for any standard field or link strength to add the same value to a specified field for each record added during a session. For example, when you create an entity, you see values of the fields set to the default - you can override the default value.

- Select **Session Defaults for Standard Fields** to specify different values for the session: Setting session defaults.
- Turn on **Show Session Defaults Dialog at Logon** to display the current different values for the session.

Charting settings

Specify which items are selected on the chart, choose a labeling scheme, and set iBase options.

Specify the chart items that are selected and which of them are laid out when they are added to the chart:

- **Only Select Added Items** to select items that are added to the chart.
- **Only Layout Newly Added Items** to lay out only newly added chart items.

© N. Harris Computer Corporation

Select **iBase Charting Default Options** to specify some general settings for how you use iBase.

Specifying expansion settings

You can specify settings to use when chart items are expanded. The settings that you specify apply to all chart operations until you either change the settings or end your session. At any time, you can go back to the original settings, by clicking **Reset All Settings**.

Procedure

1. In Analyst's Notebook, select the items on the chart that you would like to expand.
2. To access the Data Sources pane, click the **Home** tab, and then in the **Data Sources** group, select **Connect** > **Datasources Pane**.
3. On the **Selection** page, select **Expand with Settings**.
4. On the general page, select the expand level and options:

Option	Description
Limit expansion level	Select to limit the expansion level, to the specified maximum level.
Include Connecting Links	Add any connecting links between the entities that are added as a result of the expand.
Include Common Neighbors	Add any entities that are connected to multiple entities added as a result of the expand.
Common Neighbors - Fully inclusive	Add entities that are linked to all the currently selected entities.

5. On the entity types, and link types pages, select the entity and link types to be included in expand.
6. On the link style page, choose how link labels are displayed:

Option	Description
From database	The label for the chart link is the chart label as specified in the default labeling scheme.
Type Name	The label for the chart link is the iBase link type.
Occurrences	The label for the chart link is the number of iBase links that it represents. This is only relevant for links that contain multiple links that are represented by a single line on the chart.
Sum numeric	If the iBase label for a link has a numerical component, then the values of this component, for all the iBase links represented by the chart link, are summed. The resulting number is used as the chart link label. The direction of links is used in the calculation; values are added if the

© N. Harris Computer Corporation

Option	Description
	links are in the same direction, or subtracted if they are in opposite directions. This is only relevant for links that contain multiple links that are represented by a single line on the chart.

As an example of the **Sum numeric** option, transaction links in iBase might have a label containing the value of the transaction. If two links of this type between two bank account entities - one of $2000 and one of $1000 - are added to a chart in a single link, the label value is either $3000 (if both links are in the same direction) or $1000 (if the links are in opposite directions).

7. On the link style page, choose how multiple links between the same two entities are represented:

Option	Description
Single	The connection between two entities is shown as a single link. This single link represents an association between two entities, which might comprise many different instances of that association.
Directed	The connection between two entities is represented by one link for each possible direction. Therefore, a directed connection has a maximum of four links. Each of those links can represent several instances of the association, all in that same direction.
Multiple	Each link between two entities is displayed individually.

Specifying charting settings

The Charting options control how iBase records from a connected datasource are handled on the chart. You can control the charting settings of each connected datasource separately.

Procedure

1. Click the **Home** tab, and then in the **Data Sources** group, select **Connected Sources** > **Charting Settings**.
2. Check the **General** settings match your requirements:
   - **Charting**

     In the charting section you can see:

     - Charting scheme
     - Chart Style - For Timeline charts you can select to separate simultaneous events by milliseconds to make them more visible, and apply this option to all chart items or just to those added after this point.
   - **Expand**

© N. Harris Computer Corporation

You can choose how many items to bring back when expanding items.

- **Common Neighbors**

  You can choose only to allow common neighbors that are neighbors of all the selected items.

- **Find Path**

  When looking for connections between items, you can restrict the found connections to a direction, specify whether to just return the shortest path, and how many items can be between the two items (up to a limit of 10).

For more information, see:

- Specifying expansion settings
- Expanding and finding related information in iBase

3. You can restrict the record types to be used in the chart using the lists in **Entity Types** and **Link Types**.
4. You can determine the link label and multiplicity behavior in **Link Style**.

Semantic Type Matching Behavior

Semantic types can be used to group items or properties that are related. You can use semantic types to help match entities.

Each property has a role to play in assessing the likelihood that two records of information describe the same entity. Some of the properties identify the entities, for example a national identifier for a person or a license plate number for a vehicle, while others contribute additional information. It is important to assign at least one identifying property to an entity in order for it to match.

Entity and property semantic types

This table lists the entity semantic types with matching behavior, and the property semantic types that are considered to identify them.

Note: All of the recognized property semantic types can contribute to the match for a pair of entities but they contribute less than the semantic types associated with the entity type. For example, a phone number identifies a phone and can contribute to a person match.

The '\' symbol in the table is used to indicate a child semantic type.

An item in **Bold** indicates that the property semantic type is an identifying type, or a component of an identifying type. You must assign at least one identifying type to an item for matching to be performed. For example, to match items assigned the Location entity type, the items must have a Full Address, or part of an address such as a Street Name or City Name. To match items assigned the Phone entity type, the items must have a Phone Number, a Local Number, or an Area Code.

In the **Culture** column of the following table:

- **Neutral**

  The matching behavior of these entity types is culturally independent. For example, Bank Card or Phone entity types match regardless of your locale.

- **Partial**

  The matching behavior of these entity types has some US/UK specific behavior, but matching these types is still effective in all locales. For example, Motor Vehicle, Organization and Event.

- **US/UK**

© N. Harris Computer Corporation

The matching behavior of these entity types depends on the US/UK English culture. For example, Person and Location entity types assume US/UK English culture for names and addresses. You can match these entity types outside of the US/UK English locale, however, the results may not be as expected.

Culture	**Entity Types**	**Specific Property Semantic Types**
Neutral	Bank Card	**Card Number**
	\Credit Card	Card Type
	\Debit Card	
US/UK	Bank Account	**Account Number**
		Account Routing Number (US); Bank Sort Code (UK)
Partial	Motor Vehicle	**VIN**
	\Bus	**Vehicle License Plate Number**
	\Car	Vehicle Model
	\Police Car	Vehicle Color
	\Motorcycle	Vehicle Make
	\Truck	Vehicle Body Style
		Vehicle Year
Neutral	Event	**Event Date & Time**
	\Meeting	**Event Date** or **Event Start Date**
	\Crime	**Event Time** or **Event Start Time**
		Event Title
		Event End Date
		Event End Time
US/UK	Person	**National Identifier**
	\Law Enforcement Officer	**Person Full Name**

© N. Harris Computer Corporation

Culture	**Entity Types**	**Specific Property Semantic Types**
	\Offender	**Person First Name**
	\Person Alias	**Person Middle Names**
		Person Last Name
		Email Address
		Date of Birth
		Person Title
		Person Suffix
		Gender
Partial	Organization	**Organization Name**
	\Company	
	\Bank	
	\Court	
	\Criminal Organization	
	\Government Agency	
	\Law Enforcement Agency	
	\Organization Name Variation	
Neutral	Phone	**Phone Number**
	\Cell Phone	**Local Number**
	\Fax Machine	**Area Code**
	\Pager	**International Code**
US/UK	Location	**ZIP Code**
	\ATM	**Full Address**
	\Mailing Address	**Address Line 1**

© N. Harris Computer Corporation

Culture	Entity Types	Specific Property Semantic Types
		Address Line 2
		Address Line 3
		Address Line 4
		Address Line 5
		Apartment Number
		Building Number/Name
		Street Name
		City Name
		State Name
		Post Office Box
		Country Name
		Country Code
Neutral	Website	**Web Address**
	\Web Page	

Link semantic types

Several link semantic types have matching behavior.

Is Equivalent To (and any type that is derived from this link semantic type)

Links assigned to this link semantic type (or its children) are recognized by Smart Matching, and contribute positively to the overall score for the match.

Does Not Match

This link semantic type is assigned to a link when you use the **Exclude** feature of Find Matching Entities.

Links assigned to this link semantic type are not recognized by Smart Matching. The link semantic type is used to indicate that the two entities are not representations of the same real-world object, even though there are similarities between the two entities.

Possible Match Between

This is the semantic type of the links added by Find Matching Entities when the user clicks **Link**. It means that the links are not then found again when **Ignore Previously Linked** is set.

© N. Harris Computer Corporation

These are the semantic types found by **Previously Linked Matches**.

Ignored by Matching

You can assign this link semantic type to any of the links on a chart. This does not prevent two items from matching, it is simply that the link is ignored as a reason for matching.

This semantic type is the parent of **Possible Match Between**.

Assumed semantic types for labels

You can use the text in the label of an item for matching in addition to any values in attributes or data records. Depending on the item type, a semantic type is assumed for the label value. It is not possible to alter which semantic type is used.

The table shows how the labels of entities are treated for matching purposes. It shows, for example, that labels on Location entity types are treated as the Location Name and not the address.

Semantic Type	Assumed Semantic Type for Label
Person	Person Details
Organization	Organization Name
Phone	Phone Number
Motor Vehicle	Motor Vehicle Details
Location	Location Name
Event	Event Title
Bank Account	Bank Account Details
Bank Card	Bank Card Details
Website	Web Address
Mailing Address	Full Address
Drivers license	Drivers license Number
Bank	Bank Name
Vehicle Registration	Vehicle license Plate Number

Some examples of how labels are used:

Person entity types

In Smart Matching, single-word names get a low score; they are not displayed as a Matched Set unless they have further information with matching behavior. For example, if a person with a single-word name has a date of birth, or has other properties that match, they are displayed as a Matched Set in Find Matching Entities.

© N. Harris Computer Corporation

Motor Vehicle entity types

In Smart Matching, labels on Motor Vehicle entity types are treated as Motor Vehicle Details and a match score might be generated if a label contains a license Plate Number. Other property semantic types might be recognized from a Motor Vehicle label but no match is found unless a license Plate Number is detected in the label, an attribute or a property.

Location entity types

In Smart Matching, places and businesses (bars, shops, and so on), are assigned the Location semantic type. The labels are treated as location names and not addresses. To match addresses the entities need to have further address information with matching behavior, for example, the first line of an address or a zip code. This information could be supplied as attributes or in data records.

Extracting charted items

Chart items in Analyst's Notebook can be added to an iBase database to allow them to be stored and searched. Although any Analyst's Notebook chart can be used to extract items, the closer the item properties match the record properties of your database, the easier the extraction is.

Creating an Analyst's Notebook template

Analyst's Notebook templates define the types of entities and links that you can create and display on a chart.

About this task

By creating an Analyst's Notebook template that maps directly to your database, you can ensure that:

- Only entity and link types that are present in your database are added to charts to be extracted.
- Mandatory fields are marked within Analyst's Notebook.
- Only key attributes are displayed on the chart surface.

Procedure

1. Select **File** > **Data** > **Chart Item Extractor** > **Create Analyst's Notebook Template**.
2. Select the user whose user permissions you would like to use to create the Analyst's Notebook template. If **Use this User** is turned off, your user permissions are used. Alternatively, you can use the permissions of another user by turning on **Use this User** . For more information, see Creating Analyst's Notebook templates by using a different user account.
3. Browse to the location where you would like to store your template.
4. Enter a file name and click **Save**.
5. Click **OK**.

   Important: When you create the Analyst's Notebook template, the security level of the logged in user is used to determine access to field information.

   When your create a template based on an iBase database, it is your responsibility to ensure that access to any restricted or secret information is controlled in an appropriate way.

   One method of controlling the access to data would be to set up user accounts with specific security access to allow specific template generation. See Creating Analyst's Notebook templates by using a different user account for further details.

© N. Harris Computer Corporation

Creating Analyst's Notebook templates by using a different user account

For databases that are used by organizations that have multiple security levels, access to certain item types or fields might be restricted to certain users. To avoid creating templates that contain restricted references, use user accounts with the same security permissions of the users that are creating the charts.

Procedure

For each type of template:

1. In iBase, select **File** > **Data** > **Chart Item Extractor** > **Create Analyst's Notebook Template**.
2. In the Analyst's Notebook Template Generator dialog, turn on **Use this User**.
3. Select the user.
4. Enter the password for the user.
5. Click **Connect**.

Enhancing Analyst's Notebook templates

An Analyst's Notebook template that matches the schema of your iBase database can be optimized for the extraction of data into iBase.

Setting up a template to be used by users

New charts that you create with Analyst's Notebook use templates to determine the types of items that are displayed.

You can provide Analyst's Notebook users access to the template, and to inform them that it is available for use. To make a template available for use, ensure that it is saved to the Local or Workgroup Template Folder that is specified within Analyst's Notebook.

Note: If you are creating charts that need to be extracted frequently into iBase, you can also set the template that you created in iBase to be the default template.

Controlling the display of attributes

You can control the number of attributes that are displayed on a chart for each item type. Controlling the display of attributes allows data to be entered that can be mapped to iBase fields without displaying it on your chart.

To select which attributes are displayed on a chart, open the template in Analyst's Notebook and edit the Attribute Classes within the Chart Properties.

Indicating which attributes relate to mandatory iBase fields

If your database contains mandatory fields for a particular item type, you can change how the attribute that relates to that information is displayed within Analyst's Notebook.

To change the appearance of attributes, open the template in Analyst's Notebook and edit the Attribute Classes within the Chart Properties.

© N. Harris Computer Corporation

Creating a chart

If you want to create a chart and you are planning to extract items into iBase, you can increase the amount of information that can be transferred to your database. Ensure that an iBase database template is used when the chart is created, and that the attributes you use on items relate to iBase fields.

If you create a chart using a template based on the iBase database that you are planning to extract into, the item types available will be mapped to iBase item types and recognized. This:

- Ensures that items added to a chart are of a type that will directly map to your iBase entities and links.
- Allows attributes to be added to items that will be recognized and added to iBase fields.

Adding attributes that can be matched to iBase fields means that:

- You can ensure all the information entered will be correctly stored within iBase.
- Any Suggested From Code List or Selected from Code List type fields will have their options available from within Analyst's Notebook.

Extracting chart items

If you have a chart that contains items you would like to add to your database, you can run the Chart Item Extractor. Chart Item Extractor identifies potential records and the properties that they contain.

Procedure

1. Open Chart Item Extractor.
   - In iBase, select **File** > **Data** > **Chart Item Extractor** > **Extract Chart Items**, browse to your chart, and click **OK**.
   - In Analyst's Notebook, with a connection to an iBase database, open the Data Sources pane, and select **Extract Chart Items**.
2. Configure the extraction.
3. Review the suggested alignment and make any changes necessary. For more information, see Changing the alignment.
4. View the validation results and make any changes.
5. Run the extraction, and review the results.

Configuring the extraction

When you extract data, you can determine whether to add a copy of the original data, whether to reference the chart, and whether to create a set of the extracted items.

Before you extract Analyst's Notebook data, you need to define:

- How you would like to store general information.
- Whether you would like to add a chart as an iBase entity in addition to the data.
- Whether you would like to create an extraction set that contains the results of the extraction.

Chart item data

When you extract chart items, you can extract data into standard fields in addition to mapping them to the correct field types.

Data that is held in the following places can be added to multi-line fields:

© N. Harris Computer Corporation

- Item data
- Attributes
- Data records
- Cards

Information that is held on cards is stored as a multi-line field. As a default option these will be added to a single multi-line text field that you can select using the configuration page. If you would like this information to be separated based on type, use the **Advanced** options.

Note: When you are working with records from Analyst Notebook Premium, system attribute data is not extracted from chart items. For example, your have Person records that contain system-governed Date of Birth and Gender properties. When you run chart extractor, these are not extracted and the **Attribute Data** field in iBase is empty. When you are working specifically with Information Store records, properties are not extracted from any chart attributes, they are extracted from database properties.

Information Store data limitations

When you are working specifically with Information Store records, properties are not extracted from any chart attributes, they are extracted from database properties. Also the following property values cannot be extracted into iBase:

- Co-ordinates / Geospatial properties.
- Timezone (seen in the Record Inspector as a combined property that has Date, Time & Timezone).
- Numbers of type **Double**, **Integer** or **Decimal** can only be mapped to iBase field types of **Currency** or **Real Number**, there is no direct equivalent.
- Selected **From** values might not match those in the **Select From** list in iBase, for a given entity or link type. In Chart Item Extractor, you should turn off the **Validate Selected from Code List** values to bring that data into iBase.

Creating a source document

The Create Source Document options allow you to add an Analyst's Notebook chart as an iBase entity and if required link chart entities to the source document using links or hyperlinks. To create a source document:

1. Turn on the Create Source Document checkbox.
2. Select the Source Document Entity Type.
3. If required, select the Create Source Document Links option. This links created items to the Analyst's Notebook chart they were extracted from.
    - To link entities to the source document, select a valid Source Document Link Type.
    - To link chart links to the source document, select a valid Source Document Hyperlink Field.
4. If required, select Include source records in extraction set to add the source document to the extraction set.

Note: In order to import charts within the application, you need to have an entity type that has a Document Field Type.

© N. Harris Computer Corporation

Creating an extraction set

As a record of this extraction you can create a new set. This might be useful later to identify the data extracted from this data source and for performing analysis on it. The checkboxes will be unavailable if you do not have permission to create sets.

To create an extraction set:

1. Turn on **Create extraction set** to create a new set containing the records that are added to the database.
2. Enter the **Extraction Set Name**.

   Note: You must enable the **Create source document** and the **Include source records in extraction set** to include source records in an import set.

Separating data into separate fields

To create records that can be searched and used in iBase, the information that is stored in the chart item needs to be mapped to specific properties. In addition, you can store a full copy of the chart information that is taken from item data, attached cards, item attributes, and data records for auditing purposes.

Procedure

1. On the **Chart Item Extractor Configuration** page, select the **Extract chart item data into multi-line text (standard) fields** option and click **Advanced**.

   Note: You can only select **Advanced** options if the **Extract all chart items into a single, multiline text field** option remains turned off.
2. In the Advanced Options, set the multi-line text field type that you want to use to store each type of information:

Field	Description
Item Data Field	Extracts the fields Label, Identity, Date, Time, Description of Date & Time, Time Zone, Source Reference, Source Type, Grade 1, Grade 2, and Grade 3 from the Chart Item
Cards Field	Extracts all information that is held on the Entity's Cards.
Attributes Field	Extracts all the Attribute information for the Entity.
Data Records Field	Extracts information from any source database.

3. Click **OK** to return to the main configuration dialog.

Resolving item labels

Text that is contained within a chart item's label can often relate to multiple fields within your database. Using the **Advanced** option on the **Change Alignment** page you can set up some rules for splitting up the label text.

Depending on the information available in the items label, you might need to:

© N. Harris Computer Corporation

Select the label fields

Label fields are used to store the extracted label text, they correspond to the iBase item that is selected.

To choose label fields:

1. Select a field from the list of Available Fields.

   The fields available are multi-line text Standard fields.
2. Click the right arrow button.

   The field is moved to the Label fields list.

   Note: To remove a label field click the left arrow button.

Select a separator

To determine how the data in the label is divided, you must chose a separator. This character will be used by the extraction to break up the label into fields.

Select the split character that you would like to use to divide the column. By default this uses a space, if you require a specific character, for example " | ", select Other and type the characters and spaces in the adjacent box.

Change the order of label fields

To reorder the fields that you have added, select the field and move it using the navigation controls.

Ignore parts of the label

If there is information contained within the label that does not match an iBase field, you can click Ignore to allow that information to be ignored.

Changing the alignment

Chart items will be automatically aligned based on the detected chart item properties. You can modify the alignment and extract the chart item label text if required.

Procedure

1. Select one or more items from the Alignment Results list.
2. Click **Change Alignment**.
3. Select the iBase Type that you would like to align to.
4. **Optional:** To extract the label, turn on the **Extract the label into a single field** and choose the field type.

   Note: To split the label into multiple fields, set the Advanced options (see Resolving item labels).

Removing the alignment

Chart items will be automatically aligned based on the detected chart item properties. If you would like to remove the alignment for a particular item, press **Remove Alignment**.

Note: Items that are not aligned to iBase types will not be extracted into your database.

Managing alignment results

© N. Harris Computer Corporation

When Analyst's Notebook chart items are extracted into iBase, they are scanned for information that can be aligned to entity types and link types. If the automatic alignment for an entity or link type is incorrect, you can modify or remove it.

Types of Alignment

Type	Method	Description	Example
Name	Automatic	The property name matches an item type or field name.	A person attribute of surname matches exactly a field called family name.
Semantic	Automatic	The property semantic type matches an entity, link, or field semantic type.	A chart item 'Villain' being of semantic type 'Person' would match to an iBase entity type 'Criminals' also having 'Person' as the assigned semantic type.
System	Automatic	The system looks for specific property types and assigns certain field types automatically.	A Person entity that is set to use a female icon when the gender field is set to female.
User	Manual	These are manually assigned as part of the extraction process.	An Officer entity using a specific icon being matched to a Person entity type.

Note: A combination of types may be used to determine an alignment, in which case all that apply will be listed (for example Name and Semantic).

To view the current alignment for an item, ensure it is selected within the **Alignment Results**, any chart item properties that have been aligned will be displayed.

Note: You can select multiple items within the alignment results. This allows item type alignment to be changed, but will not display the individual alignment details.

You can sort by clicking on the Alignment Type column header to group alignment types.

Validating the extraction

To preserve the quality of the information that is stored within iBase, you can ensure that items are only extracted that match the rules set in iBase Designer. For example, item extraction is prevented if mandatory information is missing, data exceeds field type limits, or maps to fields that require specific values that aren't specified.

Procedure

1. On the **Chart Item Extractor Validation Results** page, click **Options** and with **Extract chart item label** turned on, click **Advanced**.

© N. Harris Computer Corporation

2. To add these items that do not match your database validation rules to your database, select **Include in extraction**. There are four options that you can use to clean the data and prevent issues:
   - **Allow duplicates (entities only)**

     Allows the import of duplicate entity records.
   - **Validate 'Selected from Code List' values**

     Verifies that the values in the source data are valid for the code lists in the current database. Invalid values are reported as errors during an extraction.
   - **Auto-populate blank mandatory fields** Allows records to be created without all the mandatory fields being mapped. To allow this, a default character is inserted into each mandatory field that is blank. This will continue to use the default value if one has been specified within the field definition or will populate the field with:
     - '-' for text fields
     - '0' for number fields
     - the current date for date fields
     - the current time for time fields
   - **Truncating text field overflows**

     Truncates fields that exceed the maximum field length. If this option is not selected, records with fields that exceed the maximum field length will be marked with validation errors.
3. To exclude items from your extraction, select the **Exclude from extraction**.

   Note: These rules can be set up to be run on a selection of items by highlighting the items within the Validation Results and selecting the **Apply to selected items only**. To extend the rule to all items, select **Apply to all items**.

Viewing the validation results

When the alignment is complete, the items are checked to ensure that they are ready to be extracted.

Items are marked in one of the following possible states:

-

  Success

  The item is ready to be extracted.
-

  Warning

  The item is ready to be extracted but might match another item that is either included in the extraction, or an existing record.

  Note: Any duplicates that are detected can be merged after extraction.
-

  Error

  An issue with the item must be addressed or the item cannot be extracted.

The following messages indicate an issue that prevented an item from being imported:

© N. Harris Computer Corporation

- `Excluded`

  The item is manually set to be excluded in the options.
- `Not aligned`

  The item is not matched to an iBase type.
- `No aligned fields`

  The item is matched to an iBase type, but no fields match.
- `One or both link ends are not valid`

  The link ends are not matched to valid entity types.
- `The length of the <field name> field must not exceed <field length> characters`

  The item has a field that contains more data than can be stored in the field type so cannot be included in the extraction.

  Note: You can use the validation options to truncate the field.
- `No value was specified for the mandatory field - <field name>`

  The item is missing mandatory information and so cannot be included in the extraction.

  Note: You can use the validation options to auto-populate the mandatory field.

  Auto-populating the field uses the default value if one is specified within the field definition, or populates the field with:

  - '-' for text fields
  - '0' for number fields
  - The current date for date fields
  - The current time for time fields
- `<invalid value> is not in the defined list of values for this field`

  The item has information that matches a code list field type but the value does not match a code list option.

  Note: You can turn off the validation of Selected from Code List value by deselecting the option in the Validation options dialog.
- `Invalid date/time entered for <field name>: <invalid value>`

  A field is aligned with an iBase date field but does not contain a recognized date format.
- `<invalid value> is not a valid value for <field name>`

  A field is aligned with an iBase field but does not match the expected data format.

Managing identifiers

When iBase checks for matching records during the extraction, it evaluates each source record by using identifier fields. If the values in the source record match the values in an iBase record, then the extraction will suggest that the item is a duplicate.

About this task

Although suggested identifiers are automatically selected for each entity type, you can modify the selection to allow better matching for each entity type.

© N. Harris Computer Corporation

Procedure

To change the identifiers used to check extracted entities:

1. Click **Identifiers**.
2. Select the entity type you would like to identify.
3. Select the identifying fields.
4. Click **OK**, the extracted items will be rechecked for duplicates.

   Note: If no identifiers are selected, the imported item will not be checked for duplicate items.

Viewing the extraction results and checking for duplicates

As part of the validation process, chart items are checked for duplicates both within the extraction and against existing iBase records. After the items are extracted, the **Duplicates** page lists the potential duplicates.

Procedure

To manage potential duplicates:

1. Select an extracted chart item from the list.
2. Review the potential duplicates in the duplicates summary pane.
   a. If the duplicate is valid, click **Merge** to open the **Merge Entities** dialog.
   b. If the duplicate is not valid, you can change the **Identifiers** to ensure that the correct information is being used.
3. After you have resolved any duplicates, click **Close**.

Analyst's Notebook and iBase

When you use i2 Analyst's Notebook connected to i2 Analyze, and i2 iBase together, you gain significant extra functionality compared to a deployment of iBase alone. In particular, you can enrich your investigations with records from external data sources.

With i2 Analyze support, Analyst's Notebook gives you the ability to:

- Use External Searches in the Analyst's Notebook desktop client to search for information in data sources that are connected through i2 Connect.
- Visualize records from both iBase and i2 Connect data sources on charts in the Analyst's Notebook desktop client.
- Use iBase records as the starting point for external searches. For example, you might look for a record that shares field values with an iBase record.
- If you have the Export-to-iBase plug-in, export records from connected sources from the Analyst's Notebook desktop directly into iBase.

  Note: In order to export these records from the Analyst's Notebook desktop client, you must have write access to the target iBase database.

© N. Harris Computer Corporation

Using i2 Analyst's Notebook with iBase

This section contains details of the tasks that you can perform using i2 Analyst's Notebook, connected to i2 Analyze, with iBase.

You must first connect to an i2 Analyze server. If you do not already know the URL and login details, contact your system administrator. Click **Change Connection Details** in the upper-right corner of the application window.

Enter the URL of the i2 Analyze server and click Connect. If you are prompted, enter your user name and password and click **Log in**.

To use Analyst's Notebook with iBase, you must also connect to your data source using the **Connect** option in the **Data Sources** group. Select your iBase database from the **Data Sources** pane.

Searching external sources

In the Analyst's Notebook desktop client, the **External Searches** window contains everything that you need in order to understand, configure, and run the queries that search external data sources.

1. On the Home tab of the ribbon, click **External Searches**.

   The External Searches window opens, and the queries that your system administrator defined are listed on the Queries page. The list of available queries is displayed in up to three sections: Available with the current chart selection; Available with or without a chart selection; and Not available with the current chart selection. Sections are not displayed if there are no queries available.
2. **Optional:** Use the Filter these queries field to enter text that filters the list of queries by name.
3. Click a query to select it and display more information about it.

   The pane on the right of the **Queries** page lists the actions that you can perform with the selected query. The same pane also describes what types of information the query can return, indicates whether the query supports or requires seeds, and might include a longer explanation of its behavior.
4. Unless the query requires seeds that you have not selected on the chart surface, you can click **Open**.

What happens next depends on whether the query supports parameters:

- If the query does not support parameters, it runs immediately and you see a new page that contains results from the external data source.
- If the query does support parameters, you see a new page where you can enter values for those parameters before you run the search.

  Note: Queries that have no mandatory parameters have an extra action available: **Copy to chart**. This action copies any results straight to the chart without first showing them in a list. Long running queries do not gain the extra action.

After the query runs, the **External Searches** window gains a **Results** view that contains the search results. From here, you can select results and copy them to the chart surface.

Depending on the configuration of the Export-to-iBase plug-in, the **Export to iBase** icon might be activated. If it is, you can select records from the search results and export them directly to iBase.

Tip: You can use text in the **Search within results** field to filter the results.

© N. Harris Computer Corporation

Running seeded searches

In Analyst's Notebook, you can use records that you retrieve from external searches as the seeds for further searches. iBase records can also be used as the starting point for external searches.

To run a query that supports seeds against an external data source, you must first understand the kinds of seeds that it requires, and select some on an Analyst's Notebook chart. When you satisfy the requirements, the query becomes available for use.

About this task

When a connector writer enables a seeded query on an external data source, they can place restrictions on the types, number, and origin of the seeds that the search requires. Before you can open a seeded query, you must select chart items that contain records that obey its restrictions, as indicated in the Queries tab of the **External Queries** window.

Note: iBase link records cannot be used to seed external searches.

Procedure

1. Select an iBase record on the chart.
2. On the **Home** tab of the ribbon, click **External Searches**. The External Searches window opens, and the queries that you can use are listed on the Queries page. The queries that are available with your current chart selection are shown in the first section.
3. Optionally, use the **Filter these queries** field to enter text that filters the list of queries by name.
4. Click a seeded query to select it and display more information about it.

   The pane on the right of the **Queries** page lists the actions that you can perform with the selected query. The same pane also describes what types of information the query can return, indicates whether the query supports or requires seeds, and might include a longer explanation of its behavior.

   Seed type constraints

   The types of seeds that the query uses, and the number of each type.

   Seed count limits

   Lower and upper boundaries (if any) on the total number of seeds.

   Seed data sources

   The data sources from which seeds must originate.

   If one of these headings is not in the list, then the selected query has no requirements of that kind.

Exporting records from external searches to iBase

The Export-to-iBase plug-in makes it possible to export the records that you find in external searches from the Analyst's Notebook desktop client, and store them in your iBase database.

The behavior of these export operations is controlled by export settings and specifications that were put in place by your system administrator.

- If you can log in to iBase as a user with the security administrator role, then you can edit the settings and specifications as well as exporting records to iBase.
- If you do not have the security administrator role, but you see the **Export to iBase** section on the **Home** tab of the application ribbon in the Analyst's Notebook desktop client, then you can export records according to the settings and specifications that already exist.

© N. Harris Computer Corporation

To change the specifications in this scenario, contact your system administrator.

Configuring export settings and specifications

When you export records from external searches to iBase, iBase compares the incoming records with records that it already contains. The export specification that you apply determines how iBase performs that comparison, and what happens when it finds matching records.

More specifically, export specifications control:

- Whether you can export records directly from search results, or from the chart surface, or both
- Whether records must have values for *all* the discriminator fields in order to be exported
- Whether exported records can update the information in matching iBase records, and whether they should update all such records
- Whether an exported record that doesn't match any existing records should create an iBase record
- Whether iBase records that matched or were updated or created by exported records should be added to the chart

In addition, you can use an export specification to edit the mappings between the properties of records from external searches and the fields of iBase records. You can also override iBase's definition of which fields are discriminators and which are not.

Creating an export specification

Users with the Export-to-iBase plug-in and administrator privileges can create export specifications through the i2 Analyst's Notebook desktop client.

About this task

You need an export specification in order to export records to iBase. Export specifications contain all the necessary information for creating iBase records from the records that you find with external searches.

You can arrange for the specification to use different configurations for specific connectors. Any connector without its own configuration then gets the default configuration, which is also used for records that you create from the palette, and for any i2 Analyze records that TextChart creates.

Procedure

1. Select **Export** from the **Export to iBase** menu. The **Export to iBase** window opens and your connected database shows as the **Active iBase Database**. You can select a different database from the drop-down menu.
2. Click the **New** button. The **i2 Analyze to iBase Export Specification** window opens.
3. Type a name for your specification into the **Name** field. If you do not choose a name, your specification is assigned the default name of "New specification".
4. Use the **Description** field to add details about your export specification. You should include a general description, and information to help analysts decide if this specification meets their requirements.
5. Select the **Details** tab to choose the options for your new specification.
6. Use the descriptions of the options in Export specifications to configure the new specification for your needs.
7. To improve the alignment between records from external searches and iBase records, or to override iBase's discriminator settings for exported records, populate the **Mappings** window.

   For more information, see Export mappings.

© N. Harris Computer Corporation

To save your changes, click **Close** from the **Mapping Specification** tab, or **Save** from the **Details** tab. Your specification is saved to the iBase database. Saved specifications are available for other analysts to use.

Editing an export specification

Users with the Export-to-iBase plug-in and administrator privileges can edit and delete export specifications through the i2 Analyst's Notebook desktop client.

About this task

You need an export specification in order to export records to iBase. If you are able to log in to the connected iBase database as a user with the security administrator role, you can edit and delete export specifications. If you have write access to the iBase database, you can use the export specifications defined by an administrator, but you cannot modify them.

Note: You cannot edit an existing specification to use as the basis for a new specification. When you edit a specification, the changes are saved to the iBase database. If you change the **Name** field, you will overwrite the existing specification. A new specification is not created.

Procedure

1. Select **Export** from the **Export to iBase** menu. The **Export to iBase** window opens and your connected database shows as the **Active iBase Database**.
2. Select the specification you want to edit and click the **Edit** button. The **i2 Analyze to iBase Export Specification** window opens.
3. Edit the options on the **Specification** and **Details** pages. For more information, see Export specifications.
4. To save your changes to the iBase database, click **Save**.

To delete a specification, select it from the drop-down menu in the **Export to iBase** window. Click **Delete** > **OK** to confirm deletion or click **Cancel** to return.

Export specifications

In the Export-to-iBase plug-in, export specifications contain the rules that govern how iBase records are created from the records you find in external sources. This topic describes how export specifications work by explaining the options that are available through the Export to iBase dialog.

Enabling and disabling Export to iBase

When a user with write access to iBase clicks **Export to iBase** in the Home tab of the ribbon in the Analyst's Notebook Premium desktop client, the Export to iBase dialog appears.

The first page of the dialog allows the user to choose the **Target database** and the **Export specification** that they want to use. The user can examine the details of the selected export specification by clicking **Edit**.

When an iBase user with the security administrator role clicks the same ribbon button, they see two additional settings:

- **Enable export from search results** allows all users to export records in the results view directly to iBase without charting them first.
- **Enable export from the chart surface** allows all users to export records from selected items on the chart surface to iBase.

© N. Harris Computer Corporation

If neither checkbox is selected, users cannot export i2 Analyze records to iBase. Do not select either checkbox until at least one (default) export specification exists.

Configuring export specifications

When a user with write access to iBase clicks **Edit** to examine an export specification, the resulting dialog contains a single **Specification** tab, in which they can view the name and description of the specification.

For an iBase user with the security administrator role, the dialog contains an additional **Details** tab that provides a range of settings for configuring the behavior of export operations on a per-connector basis.

- **Connector**

  Unless otherwise specified, all the settings on the **Details** tab apply to records from all connectors, as indicated by the label `(Default)` in the **Connector** field.

  Note: The default settings also apply to records that users create from the `Gateway` palette, and to any other i2 Analyze records that use the same schema.

  To give different behaviors to records from different connectors, click **Add** to create a separate group of settings for a connector with a specific identifier. You can then use the drop-down to switch between connectors, or click **Remove** to return a connector to the default settings.

- **Export records with missing discriminators**

  The iBase database defines which fields of its records have values that discriminate them from other records.

  Optionally, you can override the iBase definitions and specify a different set of discriminator fields on a per item type basis. (See the topic about Mappings.)

  Regardless of how the discriminators are defined, however, a record being exported from the Analyst's Notebook desktop client might not have values for all of those fields.

  When this option is selected (which is the default), records are exported from the Analyst's Notebook desktop client even when they have missing discriminator values. When this option is not selected, records that do not have values for *all* the discriminating fields are *not* exported.

- **Chart records as they are added/updated in iBase**

  This setting controls what happens on the chart when a record that a user exports from the Analyst's Notebook desktop client causes a record to be created or updated in iBase.

  When the option is selected (which is the default), iBase immediately sends the affected record back to the chart. If the record was exported from the chart, the chart record is updated with data from iBase. If the record was exported from a results list, a new record is added to the chart.

- **Update iBase records that match exported records**

  When users export records from the Analyst's Notebook desktop client, iBase uses their discriminator values to determine whether there is already a matching record in the database.

  This setting controls what happens when iBase finds a match. When it's selected (which is the default), the existing iBase record is updated with information from the exported record. When it's not selected, the existing record is left unchanged.

  When **Update iBase records that match exported records** is selected, you also need to specify what happens if iBase finds more than one match for an exported record, through the **If an exported record matches several iBase records** setting:

© N. Harris Computer Corporation

- **Update the first matching record** means that the first matching record is updated, but all other matching records are left unchanged.
- **Update all matching records** means that all matching records are updated with information from the exported record.
- **Do not update records** means that all matching records are left unchanged.

- **Create iBase records for unmatched exported records**

  This setting controls what happens when iBase does not find a match for an exported record. When it's selected (which is the default), an iBase record is created from the exported record. When it's not selected, the exported record is ignored.

- **Mappings**

  The **Mappings** area of the **Details** tab allows you to fine-tune the process of populating the fields of iBase records from the properties of i2 Analyze records. For more information, see Export mappings.

Sharing export specifications

Clicking **Save** stores export specifications to the iBase database, from where they are available to all users of that database.

If you want to use the same specification for a different database, or to store it in a version control system, you can save it in a distributable form by clicking **Write specification to file**. To load a specification that someone has provided to you, click **Read specification from file**.

Export mappings

In the Export-to-iBase plug-in, the **Mappings** area of the **Details** tab allows you to fine-tune the process of populating the fields of iBase records from the properties of i2 Analyze records in two different ways.

Field alignment

Generating the gateway schema from the iBase schema results in a pair of schemas that are highly compatible with each other. Most fields are then populated automatically. However, there are occasions when you might need to provide a custom mapping, especially if the value types are significantly different.

Note: You cannot create a mapping for a link that has no properties.

You can type custom mappings directly into the multi-line text area. Each line represents a different mapping, in the following format:

```
[iBase field]=[i2 Analyze property]
```

Each entry takes the form `[Entity/Link Type Name].[Field/Property Name]`. For example:

```
Person.FirstName=Person.FirstName
Person.MiddleInitial=Person.MiddleName
Person.LastName=Person.LastName
Person.DOB=Person.YearOfBirth
Location.Latitude=Location.Coordinates,1
Location.Longitude=Location.Coordinates,2
```

In the entries for the `Location` type, the comma tells the plug-in how to process comma-separated i2 Analyze property values (such as coordinates). The number indicates the position of each parsed value - for example, 1 for before the comma and 2 for after it.

© N. Harris Computer Corporation

Note: For a gateway schema that was generated from the iBase schema, mappings such as the first three in the above example take place automatically and you do not need to specify them in the **Mappings** area. Similarly, comma-separated values are mapped directly to the iBase schema.

Discriminator fields

You can use the same multi-line text area to override iBase's definition of the discriminator fields for a particular item type. For example, imagine that iBase defines the discriminators of its `Person` type as `FirstName`, `LastName`, and `SSN`.

If you know that a connector returns `Person` records that never have social security numbers, but you want to require that exported records must have first and last name values, then you can do this:

```
*Person.FirstName=Person.FirstName
*Person.LastName=Person.LastName
```

When you mark an entry with an asterisk, you specify that you want to use the iBase field as one of the discriminators for that type, instead of the discriminators that iBase defined.

Exporting records from search results to iBase

You can export records from sources connected to Analyst's Notebook to iBase in a single operation.

About this task

When you run an external search against a connected source, and provided that your system administrator has given you permission to do so, you can select records from the results list and export them directly to iBase.

When records from an external search are exported to iBase, they become iBase records and gain an iBase key in the database. Depending on the i2 Analyze to iBase export specification, record matching runs during the export process and matches are updated with the new iBase record. If no matches are found, new iBase records are created.

Procedure

1. On the **Home** tab of the ribbon, in the **Export to iBase** section, click **Export**. The **Export to iBase** window opens.
2. Check that you are connected to the iBase database that you want to export your data to, and that you have selected the export specification that you want to use. Your selections persist across export operations.
3. On the **Home** tab of the ribbon, click **External Searches**.
4. Choose your query and run the external search.
5. View the returned search results. Refine your search if required, and select one or more records to export to iBase. The **Export** button is activated.
6. Click **Export**. The **Progress** tab in the **Export to iBase** window updates with information about the export, including the number of records that were added or updated in iBase, and the number of errors.

   For detailed information, click the **Log** tab, and use the **Find** bar to narrow your search. For example, iBase truncates string values that exceed the supported length. When this occurs, a warning is displayed in the log.

The chosen records are exported to iBase. If the **Chart items** option is selected in the export specification, the records are also added to the chart.

© N. Harris Computer Corporation

Note: Any links that you export from external search results are converted into iBase links when you export them.

Exporting records from the chart surface to iBase

You can select records from external searches that you've added to a chart in the Analyst's Notebook desktop client, and export them to iBase.

About this task

Export behavior depends on how the i2 Analyze to iBase export specification is configured by you or your systems administrator.

Note: If you want to represent a relationship between an iBase record and a record from an external search in the data that you export, you must connect the records with an iBase link. The iBase plug-in creates the link based on the label name, so do not change the link label. Analyst's Notebook links are exported to iBase if the link type between the end records is valid, and has no mandatory properties.

Procedure

1. Select the records that you want to export to iBase on the chart.
2. On the **Home** tab of the ribbon, in the **Export to iBase** section, click **Export**. The **Export to iBase** window opens.
3. Click **Export**. The **Progress** tab in the **Export to iBase** window updates with information about the export, including the number of records that were added or updated in iBase, the number of errors, and the number of warnings.

   For detailed information, click the **Log** tab, and use the **Find** bar to narrow your search. For example, iBase truncates string values that exceed the supported length. When this occurs, a warning is displayed in the log.

The chosen records are exported to iBase.

Note: Any links that you export from external search results are converted into iBase links when the records are exported, provided they have a link type name in the description.

Performing analysis with i2 Analyst's Notebook

As an iBase user, the additional functionality offered by i2 Analyst's Notebook connected to i2 Analyze, can enrich your investigations.

This example is to give you an idea of how analysts with different permissions might use the features. The connectors used in this example are for demonstation purposes.

For the purposes of this example, the investigation centers around one main person of interest, known as "Samuel Steele". The analyst needs to demonstrate that Sam was present at the scene of the crime.

Our investigating analyst, who has security permissions on the iBase database, is working in Analyst's Notebook Premium and logged into the i2 Analyze server. They use the Connect option in the Data Sources group, and select their iBase database from the Data Sources pane.

They search for Samuel Steele in iBase with the **Find** feature and add his iBase record to the chart. They then take Samuel's iBase record as the starting point for an external search. The iBase record is used to seed a search for vehicle owners across the Vehicle Registration Database connector. After running the query, the returned results contain linked records of Person – Owned by – Vehicle.

Our analyst then expands the Vehicle records to reveal links to previous owners, and adds the i2 Analyze records to the chart. The chart contains a selection of records that could be narrowed down

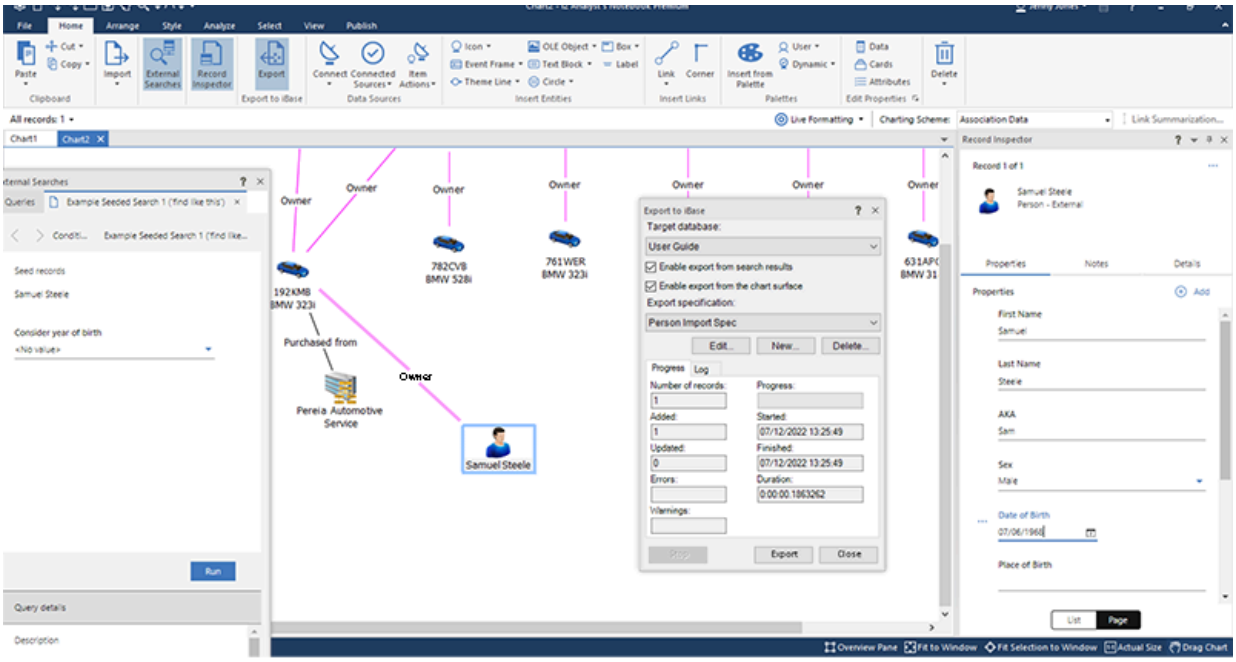
© N. Harris Computer Corporation

to an interesting subset. Our analyst looks at the vehicles that Sam had access to over a certain time range and considers other Analyst's Notebook features, such as **Bar Charts & Histograms** or **Time Wheel** to perform some temporal analysis.

They export the i2 Analyze records from the chart to iBase, using the default discriminator settings. The records gain an iBase key. They then notice that a company is linked to Sam's vehicle as a previous owner, and decide to investigate the company with a form-based external search on the Companies House connector. This time, the results contain employee records that have Government IDs but no passport number information. They copy the records to the chart.

As an analyst able to edit the export specification, they are able to specify that Government IDs should be used as a discriminator for the Companies House connector, in place of passport numbers. They keep the usual discriminators of "first name", "last name", and "date of birth". At this point an analyst who is unable to edit specifications could ask a system administrator to create a specification that meets their needs.

The employee records are exported from the chart to iBase, and our analyst observes the matching behavior when matches with iBase records are found.



i2 Explore for iBase

i2 Explore adds significant new functionality to an existing deployment of i2 iBase. In particular, it enables analysts to enrich their investigations by finding, understanding, and following connections in your organization's data.

The documentation guides you through installing and deploying i2 Explore, and how to configure highlight queries. The advanced section provides additional information on understanding the deployment process, prerequisites, and more detailed configuration tasks.

For information on how to use i2 Explore, see the i2 Explore User Guide.

© N. Harris Computer Corporation

Installing i2 Explore for iBase

You can install i2 Explore for iBase using an Installation Manager.

Before you begin

i2 Explore for iBase requires the iBase database and security database that it is connecting to be "upsized" to SQL server. In addition to this, the machine on which you are installing i2 Explore for iBase must have either SQL server or SQL server client tools installed.

i2 Explore for iBase uses the iBase search 360 feature - this capability must be enabled on the iBase database that you are connecting to before starting the installation process.

i2 Explore for iBase only supports iBase installations with users and groups configured in iBase, it does not currently support iBase installations that use Active Directory or Entra ID.

An internet connection is preferred but not required to install i2 Explore for iBase. If you do not have an internet connection, you will need to supply other prerequisites detailed in the Installing in an offline environment section.

Installation

The installer will collect information from you and install the i2 Explore for iBase deployment tooling on to disk. After this, you will follow Starting the application to deploy the i2 Explore for iBase application.

To use i2 Explore for iBase a new database, called "ISTORE", needs to be created in the iBase SQL Server instance that hosts your iBase database. There are two options available for creating this database:

1. Supply the user name and password for an administrator SQL Server login that has permission to create a new database and create a login and user in the iBase SQL Server instance. The deploy process creates the ISTORE database and views, and adds appropriate users to this database.
2. Do not supply a user name and password, and run the scripts to create the ISTORE database, logins and users manually. This should be done when the installer has completed. See the **Manually creating the ISTORE database** section in Starting the application for information on how to do this.

Unpack the `i2 Explore For iBase v1.2.zip` file to a location of your choice. This will create the following files:

- `Setup.exe`: The installer for the i2 Explore for iBase deployment tooling.
- `i2 Explore for iBase.msi`: The MSI installer file.

Run the `Setup.exe` file to start the installation process.

The installer will prompt you for the following information:

- **Installation location**: The directory where the Explore for iBase application will be installed.
  - The default location is `C:\i2`.
- **The database admin user name**: The user name for the administrator user in the iBase SQL Server instance. This is required to create the ISTORE database and views, and add appropriate users to this database.
- **iBase admin user password**: The password for the administrator user in the iBase SQL Server instance. This is required to create the ISTORE database and views, and add appropriate users to this database.
  - This value can be changed in the following file at any time after this installation has completed:

© N. Harris Computer Corporation

- `<installation_location>/explore-for-ibase/analyze-deployment-tooling/configs/explore-for-ibase/configuration/secrets/sqlserver/admin_PASSWORD`

Highlight queries

You can configure highlight queries to work with your iBase database. Highlight queries are used to highlight specific data in the i2 Explore for iBase user interface, such as entities, links, and attributes. This allows users to quickly identify important information in their data.

To configure highlight queries to work with your iBase database, you first need to run the deploy process to deploy the application. This generates a highlight queries XSD file, based on the schema in the connected iBase database, to the following location:

- `<installation_location>/explore-for-ibase/analyze-deployment-tooling/configs/explore-for-ibase/configuration/highlight-queries-configuration.xsd`

When the XSD file has been generated, you can create a highlight query file that conforms to the XSD schema. For help designing and implementing highlight queries, refer to the following documentation:

- Highlight query structure
- Highlight query file
- Highlight query example

When you have created the highlight query xml file, place it in the following location:

- `<installation_location>/explore-for-ibase/analyze-deployment-tooling/configs/explore-for-ibase/configuration/highlight-queries-configuration.xml`

When the file is in place, run the deploy process to deploy the file to the application.

Starting the application

This section describes the actions you need to take to successfully deploy the i2 Explore for iBase application.

NOTE: If you have elected to create the ISTORE database manually, this must be done *before* running the deploy script. Follow the instructions in **Manually creating the ISTORE database** below.

The application is deployed using the `setup` script in the `analyze-deployment-tooling` directory. You can double-click the `setup.exe` file directly. If you have a more complex deployment, you can run the script from the command line. Navigate to the `analyze-deployment-tooling` directory in a terminal and run the following command:

- `setup -t deploy`

The deploy script will perform the following tasks:

- Download the required prerequisites if they are not already present in the `pre-reqs` directory.
- Generate i2 Explore configuration from your iBase database.
- Create the ISTORE database in the iBase SQL Server instance.
- Deploy configuration files to the Liberty application server, Solr and Zookeeper.
- Start the Liberty application server, Solr and Zookeeper services.

Manually creating the ISTORE database

© N. Harris Computer Corporation

If you did not supply a user name and password during the installation process, you need to manually create the ISTORE database in the iBase SQL Server instance *before* you can run the deploy process.

When manually creating the ISTORE database, it is necessary to lookup the collation of the iBase database and use the same collation for the ISTORE database.

To do this:

- Run the following query on the iBase database:

```
USE master;
SELECT collation_name
FROM sys.databases
WHERE name = '<IBASE_DB_N#ME>'
```

Replace `<IBASE_DB_N#ME>` with the name of your iBase database.

- Set the `Collation` property in the following file to the value returned from the query above:
  - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/configs/explore-for-ibase/configuration/InfoStoreNamesSQLServer.properties`

To generate the database scripts required to create the ISTORE database, logins and users:

- Navigate to the `analyze-deployment-tooling` directory in a terminal.
- Run the following command: `- setup -t generateDatabaseScripts`

You need to run these scripts on your SQL Server instance to create the ISTORE database and users. Note that `generateDatabaseScripts` also generates scripts to remove the ISTORE database, logins and users; these can be used later if you want to remove the ISTORE database and users.

Deployment Tasks

The following toolkit options and tasks are available to use with the setup script in the deployment tooling.

usage: `setup [-t TASK] [-v | --verbose] [-h | --help]`

Argument name	**Argument description**
-t, --task TASK	Specifies the task to perform
-v, --verbose	Enable verbose logging.
-h, --help	Shows this help message, and exits

The default task is `deploy`, which will run if no task is specified.

The following deployment and setup tasks are available

Name	**Description**
deploy	Deploy the Explore for iBase environment (default).
generateDatabaseScripts	Generates the database scripts required for manual creation of the ISTORE database.

© N. Harris Computer Corporation

Name	Description
clean	Stops all services, removes installed binaries, Solr & Zookeeper data, and removes the ISTORE database.
start	Starts all i2 Explore services.
stop	Stops all i2 Explore services.
startLiberty	Starts Liberty.
stopLiberty	Stops Liberty.
startZkHosts	Starts ZooKeeper hosts.
stopZkHosts	Stops ZooKeeper hosts.
startSolrNodes	Starts the Solr nodes.
stopSolrNodes	Stops the Solr nodes.
startSolrAndZk	Starts the Solr nodes and ZooKeeper hosts.
stopSolrAndZk	Stops the Solr nodes and ZooKeeper hosts.

Exploring the deployment

When the installer has completed, the following directory structure will be created in the `<installation_location>/explore-for-ibase` directory:

```
explore-for-ibase
### analyze-deployment-tooling
#   ### configs
#   #   ### explore-for-ibase
#   #   #   ### configuration
#   #   #   #   ### secrets
#   #   #   #       ### sqlserver
#   #   #   #       #   ### admin_PASSWORD
#   #   #   ### utils
#   #   #   #   ### variables.conf
|   |## environment-secrets
#   #   #   ### provided-secrets
#   #   #   #   ### certificates
#   #   #   #   |   |## externalCA
#   #   #   #   |   |   |## CA.cer
#   #   #   #   |   |## <HOST_FQDN_I2ANALYZE_SERVICE>
#   #   #   #   |   |   ### server.cer
#   #   #   #   |   |   ### server.key
#   ### pre-reqs
#   #   ### jdbc-drivers
#   #   ### jdk.zip
#   #   ### solr.tgz
#   #   ### wlp.zip
#   #   ### zookeeper.tar.gz
```

© N. Harris Computer Corporation

```
#   ### Setup.exe
### deploy
#   ### jdk
#   ### solr
#   ### wlp
#   ### zookeeper
### data
### logs
```

Deploy directory

The `deploy` directory contains the deployed binaries for the i2 Explore for iBase application.

Data directory

The `data` directory contains temporary data files for the i2 Explore for iBase application.

Logs directory

The logs directory contains the logs for the i2 Explore for iBase application. The logs are generated by the Liberty application server, Solr and Zookeeper services and also the deployment tooling. These can be used by the i2 support team to diagnose issues with the i2 Explore for iBase application.

Making changes to the deployment

You can make changes to the iBase database or ports, after deployment.

iBase database changes

iBase Designer requires exclusive access to the iBase database. Because of this, it might be necessary to stop the i2 Explore services before launching iBase designer. You can do this by running the following command:

- `setup -t stop`

When changes are made to the schema of the iBase database, these need to be reflected in the i2 Explore application. Certain changes, like removing item types from the iBase database, can cause errors for users of i2 Explore, but will be rectified once the i2 Explore application has been updated.

The update to i2 Explore is done by running the `deploy` task of the `setup` script:

- `setup -t deploy`

Changing ports on an existing deployment

If you need to change any of the Liberty, Solr or Zookeeper ports on an existing deployment, you will need to follow these steps:

1. Run the `clean` task
   - `setup -t clean`
2. Update the `variables.conf` file with the new port values
   - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/configs/explore-for-ibase/utils/variables.conf`
3. Run the `deploy` task
   - `setup -t deploy`

© N. Harris Computer Corporation

Advanced information

This section provides advanced information about the i2 Explore for iBase application, including its deployment architecture, manual configuration, TLS connection, and offline installation.

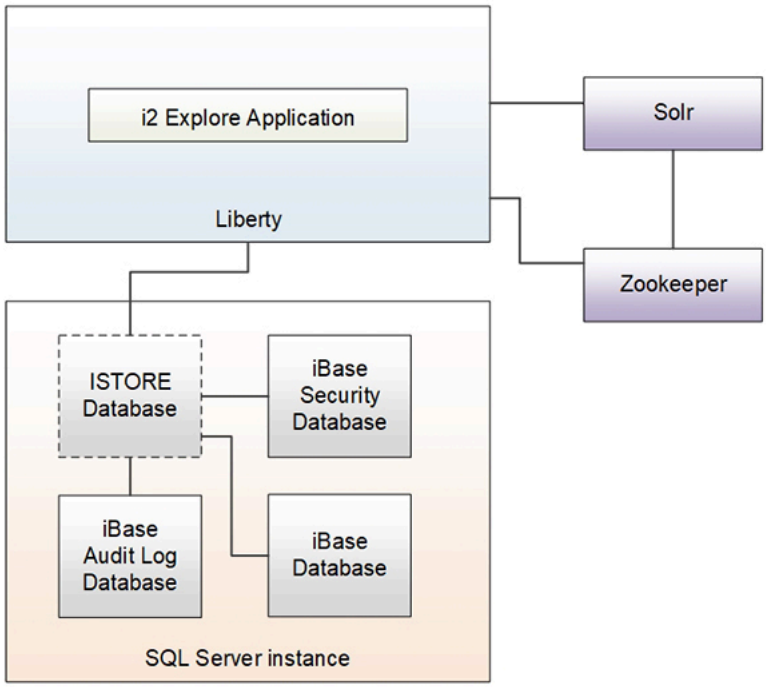
Understanding the deployment

i2 Explore for iBase is a web-based application that is deployed on a Liberty application server.

Deployment architecture

The Liberty application requires a connection to an iBase database and can be accessed through a web browser. The Liberty application also uses a Solr and Zookeeper service to provide text search capabilities.

The Liberty application requires a new "ISTORE" database to be created in the same SQL Server instance as the iBase database. The ISTORE database contains a series of views through which the Explore for iBase application can access the iBase data.



Updating the configuration manually

Details of the properties that are set during the installation of the Explore for iBase application, and how to change them after the installation has completed.

Properties that are set in `variables.conf`

The following properties can be set in the installer, and/or changed after the installation has completed, by modifying the appropriate property in the following file:

© N. Harris Computer Corporation

- `<installation_location>/explore-for-ibase/analyze-deployment-tooling/configs/explore-for-ibase/utils/variables.conf`

Properties that will be set in the `variables.conf` file during the installation process include:

- **iBase admin user name**: The user name for the administrator user in the iBase SQL Server instance. This is required to create the ISTORE database and views, and add appropriate users to this database.
  - The value will be set on the following property in `variables.conf`:
    - `DB_ADMIN_USER`
- **The iBase database name**: The name of the iBase database that the Explore for iBase application will connect to.
  - The value will be set on the following property in `variables.conf`:
    - `IBASE_DB_N#ME`
  - This is the SQL Server database name, which can be found in the database properties dialog in iBase designer.
- **The iBase security database name**: The name of the iBase security database that the Explore for iBase application will connect to.
  - The value will be set on the following property in `variables.conf`:
    - `IBASE_SEC_DB_N#ME`
  - This is the SQL Server database name, which can be found in SQL Server Management Studio (SSMS).
- **The iBase database port**: The port number that the iBase database is listening on.
  - The value will be set on the following property in `variables.conf`:
    - `HOST_PORT_DB`
    - Default value: `1433`
- **The fully qualified domain name of the iBase database server**: The host name or IP address of the server that is running the iBase database.
  - The value will be set on the following property in `variables.conf`:
    - `DB_FQDN`
- **The fully qualified domain name of the i2 Explore host server**: The fully qualified domain name (FQDN) of the server that is running the i2 Analyze application.
  - The value will be set on the following property in `variables.conf`:
    - `I2ANALYZE_SERVICE_FQDN`
- **The i2 Explore service port**: The port number that the i2 Analyze service will listen on.
  - The value will be set on the following property in `variables.conf`:
    - `HOST_PORT_I2ANALYZE_SERVICE`
    - Default value: `9443`
  - Changing this port on a existing deployment requires a specific process described in Starting the deployment.
- **The Solr service port**: The port number that the Solr service will listen on.

© N. Harris Computer Corporation

    - The value will be set on the following property in `variables.conf`:
        - `HOST_PORT_SOLR`
        - Default value: `8983`
    - Changing this port on a existing deployment requires a specific process described in Starting the deployment.
- **The Zookeeper service port**: The port number that the Zookeeper service will listen on.
    - The value will be set on the following property in `variables.conf`:
        - `ZK_CLIENT_PORT`
        - Default value: `2181`
    - Changing this port on a existing deployment requires a specific process described in Starting the deployment.
- **The SQL Server install location**: The location of the SQL Server installation, or the SQL server client tools on the host machine (e.g. "C:/Program Files/Microsoft SQL Server/Client SDK/ODBC/170").
    - The value will be set on the following property in `variables.conf`:
        - `DB_INSTALL_DIR`
- **The iBase database location**: The location of the iBase database data files. This may be on a different machine from the i2 Explore server (e.g. "C:/Program Files/Microsoft SQL Server/MSSQL16.MSSQLSERVER/MSSQL/DATA").
    - The value will be set on the following property in `variables.conf`:
        - `DB_LOCATION_DIR`
- **Operating System that the iBase database is on**: Valid values are `UNIX` or `WIN`.
    - The value will be set on the following property in `variables.conf`:
        - `DB_OS_TYPE`
        - Default value: `WIN`

If you want to change any of these settings after the initial deployment, you need to run the process described in Starting the deployment for the changes to take effect.

Configuring TLS connection to iBase

Details about the certificates you need to provide for securing the i2 Explore service, and installing the generated certificate.

To configure the TLS connection between the i2 Explore service and the web browser, you will need to provide the CA trust certificate for certificates that are received from the Liberty application server.

You must also obtain a signed certificate for the server where you are deploying i2 Explore for iBase. You will be prompted to provide the certificate and the private key during the installation process. These can also be provided at a later time by copying the certificates to the locations outlined below.

These certificates may be supplied at a later point, after the installer has completed. The i2 Explore for iBase services may be started without them, in which case certificates will be generated for you. **The generated certificates should not be used in production**, and may be replaced with your own certificates by providing them in the locations outlined below and running the deploy process.

© N. Harris Computer Corporation

- **CA trust certificate for certificates that are received from the i2 Explore server**: This is the certificate that is used to verify the identity of the i2 Explore server.
  - This certificate will be placed into the following location and renamed to `CA.cer`:
    - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/environment-secrets/provided-secrets/certificate/externalCA/CA.cer`
- **Signed certificate for i2 Explore host machine**: This is the certificate that is used to secure the connection to the i2 Explore service.
  - This certificate will be placed into the following location and renamed to `server.cer`:
    - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/environment-secrets/provided-secrets/certificates/<I2ANALYZE_SERVICE_FQDN>/server.cer`
- **Private key for i2 Explore host machine**: This is the private key that is used to secure the connection to the i2 Explore service.
  - This private key will be placed into the following location and renamed to `server.key`:
    - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/environment-secrets/provided-secrets/certificates/<I2ANALYZE_SERVICE_FQDN>/server.key`

The certificate generated for the host machine must be created using the host name of the machine that matches exactly that supplied in the `I2ANALYZE_SERVICE_FQDN` property, such that the certificate contains:

```
Owner: CN=<I2ANALYZE_SERVICE_FQDN>
SubjectAlternativeName [
  DNSName: <I2ANALYZE_SERVICE_FQDN>
]
```

Installing the generated certificate

If you have not supplied your own certificates, the deployment tooling will generate certificates for you. To access the i2 Explore application, the machine that you are connecting from must trust the certificate that it receives from the Liberty server.

To enable trust, install the `<installation_location>/explore-for-ibase/analyze-deployment-tooling/environment-secrets/generated-secrets/certificates/externalCA/CA.cer` certificate as a trusted root certificate authority in your browser and operating system's certificate store.

For information about installing the certificate, see:

- Windows: Install Certificates with the Microsoft Management Console
- MacOS:
  - Add certificates to a keychain using Keychain Access on Mac
  - Change the trust settings to Always Trust
- FireFox:
  - In the settings menu, type View Certificates and open it. Then, click Import and locate the CA.cer file.

If you are using a Mac and want to access the deployment from a Windows VM, you must install the certificate in your Windows VM.

© N. Harris Computer Corporation

Installing in an offline environment

If the machine on which you run the deploy process has an internet connection, the deployment tooling will automatically download the required prerequisites for you. If the machine does not have an internet connection, you will need to supply the required prerequisites manually.

i2 Explore for iBase requires the following prerequisites to be available on the host machine:

- **Java Development Kit (JDK)**: The JDK is required to run the i2 Explore for iBase application.
  - The JDK can be downloaded from the Adoptium website.
  - The JDK must be placed in the following location:
    - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/pre-reqs/jdk.zip`
  - Supported version: 17.0.11
- **Liberty application server**: The Liberty application server is required to run the i2 Explore for iBase application.
  - The Liberty application server can be downloaded from the IBM website.
  - The Liberty application server must be placed in the following location:
    - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/pre-reqs/wlp.zip`
  - Supported version: 25.0.0.2
- **Solr**: The Solr service is required to provide text search capabilities for the i2 Explore for iBase application.
  - The Solr service can be downloaded from the Apache Solr website.
  - The Solr service must be placed in the following location:
    - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/pre-reqs/solr.tgz`
  - Supported version: 9.8.1
- **Zookeeper**: The Zookeeper service is required to provide coordination for the Solr service.
  - The Zookeeper service can be downloaded from the Apache Zookeeper website.
  - The Zookeeper service must be placed in the following location:
    - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/pre-reqs/zookeeper.tar.gz`
  - Supported version: 3.9.3
- **SQL Server JDBC driver**: The SQL Server JDBC driver is required to connect to the iBase database.
  - The SQL Server JDBC driver can be downloaded from the Microsoft website.
  - The SQL Server JDBC driver must be placed in the following location:
    - `<installation_location>/explore-for-ibase/analyze-deployment-tooling/pre-reqs/jdbc-drivers/`
  - Supported version: 12.6.0.jre11

© N. Harris Computer Corporation

Updating the prerequisites

Occasionally, it might be necessary to supply updated versions of the prerequisites listed above (if a critical security patch is required, for example). In this instance, you can supply the updated prerequisite in the locations described above, and then do the following from the `analyze-deployment-tooling` directory:

- `setup -t clean`
- `setup -t deploy`

Uninstalling i2 Explore for iBase

This process describes how to uninstall the i2 Explore for iBase application.

The process for uninstalling i2 Explore for iBase is as follows:

- `setup -t clean`
  - This will stop all services, remove installed binaries, Solr and Zookeeper data, and remove the ISTORE database.
- Run the installer Setup.exe file again and select the "Uninstall" option.

i2 iBase Plate Analysis

i2 iBase Plate Analysis is a dedicated analysis environment designed to make it easy to manage and analyze Automatic License Plate Recognition (ALPR) data. iBase Plate Analysis is an addition to i2 iBase so that you also have access to all the powerful functions offered by those products to work with ALPR data and use it in combination with other data stored in your database.

Introduction to iBase Plate Analysis

Automatic license plate recognition (ALPR) is a mass surveillance method that uses optical character recognition on images to read the license plates on vehicles and store the Vehicle Registration Marks (VRMs) in a database. ALPR systems can use existing closed-circuit television or road-rule enforcement cameras, or ones specifically designed for the task.

iBase Plate Analysis is designed to analyze ALPR data stored in an iBase database, and use it in the context of other relevant data, to maximize its potential to provide actionable intelligence. The main functions are shown in the table below:

Function	Used for
Geographical Profiling	Compares ALPR reads around the location of specified incidents or events, and discovers vehicles that have been seen in the vicinity.
Convoy Analysis	Compares a target list of vehicles with Plate Analysis records to discover co-occurrences of vehicles traveling together.
Generating VRM Lists	Enables you to generate lists of VRMs from a set of Plate Analysis records. The lists can be used in iBase queries or in other applications.

© N. Harris Computer Corporation

Function	Used for
Finding Common Vehicles	Finds vehicles that occur in two or more specified sets.
Combining Vehicle Data Sets	Combines multiple sets easily and quickly.
Retrieving Images	Allows you to view images that are linked to ALPR reads.
Configuring iBase Plate Analysis	Allows you to specify the entities and fields that are used for Plate Analysis. You will only see the configuration options if you are an iBase database administrator.

To start the Plate Analysis Task Manager, from where all the Plate Analysis functions can be accessed, click the Plate Analysis Task Manager icon on the iBase toolbar. The Task Manager is displayed with the options displayed on the left. Click the name of the function that you want to use.

Note: Plate analysis is only supported in SQL Server databases, and is not supported in case-enabled iBase databases.

Geographical Profiling

Geographical profiling enables you to find vehicles captured by an Automatic License Plate Recognition (ALPR) system, that were detected in the vicinity of an event. This gives you the opportunity to carry out searches for vehicles, captured by a camera, traveling in the area, around a specified date and time.

You can compare vehicles recorded near a series of events to establish if any of these vehicles are common to more than one of the events.

The process of geographical profiling comprises the following steps:

1. Select the events to be analyzed
2. Specify the dates and times of interest
3. Specify the geographical area of interest
4. Examine the number of Plate Analysis reads that are found
5. Analyze the vehicles that are associated with two or more events

There are a number of ways that the results of the analysis can be stored for further examination.

Geographical profiling has multiple screens, click **Next** or **Back** to move between them.

Select Events

The source for geographical profiling is one or more events; for example a series of related crimes or incidents. The source records must be event entities as specified in the Plate Analysis configuration. You can specify the source records in one of three ways:

- specify a query that returns event records in the results
- specify a set that contains event entities
- specify event entities individually

In each case, all the event entities must include valid location information in the form of coordinate data.

© N. Harris Computer Corporation

Specify Dates and Times

When you have selected the events of interest, you need to set the dates and times, relative to the date and time of each incident, that you want to examine. You do this by specifying offsets from the date and time of the event. This can be done individually for each event, or can be set for all events that you select. In this way for example, you can specify that you want to analyze all ALPR reads that were recorded up to one hour before an event and up to five hours after. Alternatively, you can specify the time before and after individual events down to the nearest minute.

Note: If one of the dates for an event is blank then Plate Analysis will use the same already populated value for both start and finish dates. If both dates are blank then both fields will be left blank, and the results will be sorted at the top of the display.

When offsets are set for each event, you can adjust the offsets, individually or collectively, for even finer control of your analysis. For example, if you have set up different offsets for each event, you can select them all and add an extra ten minutes to the time after the event.

Offsets can be specified in days, hours and minutes, and can be positive or negative. The specified offset is displayed in the list of events.

Setting Geographical Area

The location of the Plate Analysis reads to be analyzed can be either relative to the locations of the events, or can be defined as a specific area.

You can set a geographical tolerance around the coordinates in terms of the distance North, South, East, and West of each event. As before, you can do this for each record individually, or select more than one event and set the tolerance for all the selected items.

If appropriate, you can specify any camera location for one or more events by choosing a location inside a rectangle defined by specified pairs of coordinates.

Note: If any of the specified events do not have coordinate data, you see a message telling you how many of the records do not have coordinates. These events are not analyzed.

Examine Number of ALPR Reads

The first result you see is a list of events with the number of associated ALPR reads that have been detected within your specified date, time, and area. At this stage you can examine the Plate Analysis records for each event or add them to a new or existing set for further analysis.

Analyze Vehicles Associated With Events

The final result of Geographical Profiling is a list of Vehicle Registration Marks (VRMs) that are associated with two or more events. By selecting each VRM in turn, you can see which events are associated with each one. You can select one or more of the VRMs and copy the list to the clipboard or into a text file; the list can then be used for further analysis, for example as input to an iBase query using an in list parameter.

Convoy Analysis

Convoy Analysis allows you to search for vehicle co-occurrences. You can detect one or more unknown vehicles traveling in convoy with known target vehicles. This can help detect:

- vehicles following cash in transit vehicles
- vehicles that were previously unknown, but have been identified traveling with known or suspected criminals

© N. Harris Computer Corporation

- vehicles traveling with stolen vehicles that are subsequently used in criminal activity
- compromised covert operations

Source target vehicles can be entered individually or you can use a list stored in a text file (perhaps generated as a result of Geographical Profiling for example).

Select the date and time boundaries for the Plate Analysis records to be searched. With this done, the time between the target vehicle passing a camera, and any other co-occurring vehicles passing the same camera, can be set in the time window along with the minimum number of co-occurrences. The results can be further restricted by choosing to return only the vehicles that co-occur most frequently.

Note: If you are looking for co-occurrences on a particular day, you could retrieve records for co-occurring vehicles that were actually recorded on a preceding or succeeding day. The time you set relates only to the target vehicle. For example, if a target vehicle is recorded at one minute to midnight, and the time window is specified as five minutes, you could detect co-occurring vehicles at up to four minutes past midnight.

The target vehicles and their co-occurrences are clearly displayed and any co-occurrence can be selected for a more detailed view by simply double clicking the item.

Convoy Analysis has a number of screens, click **Next** or **Back** to move between them.

Specify Parameters

Enter the Vehicle Registration Marks (VRMs) of the target vehicles that you want to analyze; they can be entered individually or you can use a list stored in a text file (perhaps generated as a result of Geographical Profiling for example).

Enter the VRMs in the **Target VRMs** box, one VRM per line separated by a carriage return. Alternatively click the **Browse** button to retrieve VRMs from a text file.

Select the date and time boundaries for the Plate Analysis records to be searched. The date and time range relates to the times that target vehicles are recorded in an ALPR read. Set a start and end date and time.

If you only want to consider a certain period each day, for example you may want to analyze Plate Analysis records between 09:00 and 09:30 each day, turn on the **Include records only within these times** checkbox and set the From and To times below.

Finally set the following options:

Time Window

Set the time window that defines a co-occurrence. For example, if you set a time window of 10 seconds (the default), it means that vehicles recorded as passing an ALPR camera within plus or minus 10 seconds of a target vehicle are considered to be in convoy. A range of values, up to 5 minutes is available. The larger the time window, the more results you are likely to get.

Minimum Co-occurrences

Set the minimum number of co-occurrences that you want to consider for each target vehicle. For example, setting the number to 2 means that you only want results where the vehicle has been in convoy with a target vehicle on more than one occasion. You can set any number between 1 and 10000, although a higher setting can reduce the number of results.

Ranked VRMs

© N. Harris Computer Corporation

You can choose to see all results by selecting **All**, or you can specify a Number of results to see. For example if you set the number to 10, you will only see the top ten ranked co-occurring vehicles, sorted by the number of co-occurrences.

Examine Co-Occurrences

This screen allows you to see how many co-occurrences have been detected; each co-occurring vehicle is listed along with the number of co-occurrences. If you have no results, or too many, you can go back and change the parameters by clicking the Back button.

You can see the Plate Analysis records relating to the convoy by selecting one or more groups of results, and clicking the **Show Records** button.

Note: If you select a result with five co-occurrences for example, when you show the records you will see ten records; for each co-occurrence you will see the Plate Analysis records for both the target and co-occurring vehicle.

Select one or more of the co-occurrences that you want to analyze in the grid view. When you have selected the results, click **Next** to go to the grid view.

Create a Grid Display

Co-occurrences for each target vehicle are displayed on a grid. The best way to understand the display is to consider this example. There are three target vehicles with the VRMs **P2 DCM**, **F576 SQZ**, and **HH18 CLK**. Each target has a tabbed page showing all the co-occurrences that relate to that vehicle. The table below represents the tabbed page for **P2 DCM**.

P2 DCM					
	P2 DCM	MJ75 LRV	**F576 SQZ**	**HH18 CLK**	PJ80 LJW
31 Jul 2022 23:59:59	**Camera1 23:59:59**	Camera1 +1	-	**Camera1 +9**	Camera1 +41
01 Sep 2022 07:59:59	**Camera1 07:59:59**	Camera1 +60	-	-	-
01 Sep 2022 13:30:10	**Camera1 13:30:10**	Camera1 +49	-	-	-
01 Sep 2022 19:41:10	**Camera1 19:41:10**	-	-	-	-
01 Sep 2022 23:57:00	**Camera1 23:57:00**	Camera1 +30	-	-	-
01 Oct 2022	**Camera1 00:00:00**	Camera1 +3	**Camera1 +10**	-	Camera1 +20

The first column of the grid shows the date and time that the target vehicle was recorded by an ALPR camera (this only shows the times that the target vehicle was recorded as a part of a co-occurrence, the vehicle may have been detected on many other occasions, but only co-occurrences are shown here).

© N. Harris Computer Corporation

The second column, headed with the target vehicle VRM, shows the name of the camera that recorded the read and shows the time that the target vehicle passed the camera.

The third column, headed with 'MJ75 LRV', shows the camera name that recorded a co-occurrence and the time (in seconds) relative to the target vehicle. The first entry in the third column shows that MJ75 LRV passed Camera 1 one second after the target vehicle. (If the time had been shown as -1, it would show that the target vehicle was recorded after MJ75 LRV.)

Each co-occurring vehicle has its own column; those shown with a bold heading, 'F576 SQZ' and 'HH18 CLK' in the example above, are themselves target vehicles; and they have their own tabbed pages.

You can copy the results for the current grid to the clipboard by clicking the Copy to clipboard button. Alternatively you can copy the results for all grids by turning on the **Copy all grids** checkbox before clicking the Copy to clipboard button.

From the clipboard, you can paste the results into another application such as a spreadsheet or text editor.

Generating VRM Lists

You can generate a list of VRMs from the Plate Analysis entities that are included in one or more iBase sets. VRM lists can be particularly useful in comparing vehicles against hotlists of vehicles of interest, or for retrieving details of vehicles from national databases.

About this task

When the list of VRMs is displayed, you can select some or all of the VRMs and copy them to the clipboard, or save them to a text file. For example, if you wanted to use some of the VRMs as target vehicles for Convoy Analysis, you would select the VRMs of interest, copy them to the clipboard, and then paste them into the Convoy Analysis dialog. Saving the VRMs to a text file would allow you to use the list as input to another application for further analysis.

Procedure

1. Click **Add**, and select the set or sets you want to analyze. You can select several sets at once, they do not have to be added individually. Click **OK** to select the sets and close the Select Set dialog.
2. Click **Generate** to create a list of VRMs

   Note: If you want to remove sets from the selection before analysis, select the required set or sets and click **Remove**. To go back to the set selection step, click **Back**.
3. Select one or more of the displayed VRMs and save them in a text file, or copy them to the clipboard for use elsewhere.

Finding Common Vehicles

The Find Common Vehicles function allows you quickly to discover which VRMs are common to specified sets, allowing you to establish connections. The results are displayed such that you can quickly see which sets contain each common VRM.

About this task

When the results are displayed, you can select some or all of the VRMs and copy them to the clipboard or save them to a text file. For example, if you wanted to use some of the VRMs as target vehicles for Convoy Analysis, you would select the VRMs of interest, copy them to the clipboard, and then paste

© N. Harris Computer Corporation

them into the Convoy Analysis dialog. Saving the VRMs to a text file would allow you to use the list as input to another application for further analysis.

Procedure

1. Click **Add** and, in the Select Set dialog, select the sets you want to analyze. You can select several sets at once, they do not have to be added individually. Click **OK** to select the sets and close the Select Set dialog.
2. Click **Find** to display a list of VRMs that are included in two or more sets.
3. Select each VRM in turn to see which sets it is included in. The sets that include the selected VRM are displayed in bold text with a green tick. When the results are displayed, you can select some or all of the VRMs and copy them to the clipboard or save them to a text file.

   Note: If you want to remove sets from the selection before analysis, select the required set or sets and click Remove.

Combining Vehicle Data Sets

Plate Analysis can often involve the creation of a large number of iBase sets. Combining Vehicle Data Sets allows you to combine multiple sets in a single operation. The combination is additive only; all of the records in all of the specified sets are added to a new set, with a specified name, and duplicates are removed from the set automatically.

About this task

The resulting set can be used for analysis using iBase or the other functions of iBase Plate Analysis. The original sets are not affected and remain available for further analysis, or for use in other combinations.

Note: Removing duplicates only relates to set membership, the original database records remain unchanged.

Procedure

1. Click **Add** and, in the Select Set dialog, select the sets you want to combine. You can select several sets at the same time, they do not have to be added individually. Click **OK** to select the sets and close the Select Set dialog.
2. Click **Combine** to put all the records into a new set. You need to specify a new and unique name for the combined set, a category in the Categorize dialog, then click **OK** to create the set. A message is = displayed to confirm that the sets have been successfully combined.

   Note: If you want to remove sets from the selection before creating the combined set, select the required set or sets and click **Remove**.

Retrieving Images

You can retrive images stored outside of iBase, using an interface between Plate Analysis and an external system.

Sometimes you might want to see camera images that are associated with Plate Analysis records, for example to confirm that a VRM has been correctly interpreted, or that a vehicle matches its description. Images can be stored as a part of the iBase record, and viewed by showing the record as usual. However, due to the large number of images that are collected by the ALPR process, your system

© N. Harris Computer Corporation

administrator might choose to retain these images in an external ALPR system and not load them into the iBase database.

For images that are stored outside iBase, an interface is provided through which an external system can receive requests from iBase Plate Analysis to supply an image to the iBase user. The image request asks for either the number plate image, or the full image of the vehicle. The picture that is displayed is read only, and there is no facility to save it to an iBase record using this route.

Note: You are only able to retrieve and view images stored outside iBase if your system administrator has set up a link to an external system that stores ALPR records. To check whether your system has access to images, contact your system administrator.

Viewing Images

When records are displayed, for example in a record list, you right-click on a record and select **Retrieve Image** from the shortcut menu. You can then choose to view either the number plate image or the full image of the vehicle. You have the option to view an image of the number plate (Plate Patch), or the whole image (Overview Image).

Note: If your system has not been configured to show images stored in external systems, an error message is displayed.

Configuring Image Retrieval

To allow images from an external system to be displayed, you need a developer to implement an interface from a type library. Full details of the configuration are shown in a separate help topic called Implementing Image Retrieval.

Implementing Image Retrieval

iBase has the ability to store images as part of a record. In situations where the iBase records are not populated with an image, you can develop your own code that sits between iBase and the source system.

About this task

This topic is intended for administrators and developers who want to develop code to implement iBasePlate Analysis Image Retrieval from the ALPR source system. This code implements a type library interface that allows iBase to provide instructions detailing the image and type of image (either number plate or entire vehicle) that the iBase user wants to see. When the interface receives the image request, its behavior and further actions are controlled by the development carried out by you. Normally this involves retrieving the image from the ALPR source system and displaying it.

Important: This topic contains specific information regarding editing the Windows Registry. You should always back up the registry before you edit it. If you alter the registry, you could cause your computer to stop functioning. i2 provides this information "as is", without representation or warranty of any kind, express or implied, including without limitation any warranty concerning the accuracy, adequacy, or completeness of such information contained herein. i2 does not assume responsibility for the use or inability to use the software product as a result of providing this information.

To create the code that works with iBase Plate Analysis Image Retrieval, the developer must implement the IImageRetrieval interface from the ANPRExternalInterfaces.tlb type library that is installed as part of the general iBase Plate Analysis installation. A description of the ANPRExternalInterfaces.tlb type library is given below.

```
<<interface>>ANPRExternalInterfaces.tlb::IImageRetrieval
```

© N. Harris Computer Corporation

```
+ShowImage(inVRM : String,
 in ImageDate : String,
 in ImageTime : String,
 in ReadID : String,
 in PartialOnly : Boolean)
```

Procedure

This example program accepts the image retrieval parameters from iBase and displays them to the user in a modal dialog:

1. Create a new VB ActiveX DLL project. This creates a project called Project1 containing a class module with the name Class1.
2. Add a reference to ANPRExternalInterfaces to your Project.
3. Add a Form to the project.
4. On the form, create the following controls, and set the design-time properties shown.

```
Control         Name        Property
 ------------------------------------------
 Label           Label1      Caption="VRM"
 Label           Label2      Caption="Date"
 Label           Label3      Caption="Time"
 Label           Label4      Caption="Read ID"
 Label           Label5      Caption="Type"

 Command Button  Command1    Caption="OK"
```

5. Add the following code to the Form:

```
Option Explicit
   Public Sub ShowImageDetails(ByVal VRM As String, _
                               ByVal ImageDate As String, _
                               ByVal ImageTime As String, _
                               ByVal ReadID As String, _
                               ByVal PartialOnly As Boolean)
       Label1.Caption = VRM
       Label2.Caption = ImageDate
       Label3.Caption = ImageTime
       Label4.Caption = ReadID
       If PartialOnly Then
           Label5.Caption = "Partial Image"
       Else
           Label5.Caption = "Full Image"
       End If
   End Sub
   Private Sub Command1_Click()
       Unload Me
   End Sub
```

6. Add the following code to your ClassModule:

```
   Option Explicit
   Implements ANPRExternalInterfaces.IImageRetrieval
   Private Sub IImageRetrieval_ShowImage(ByVal VRM As String, _
                                         ByVal ImageDate As String, _
                                         ByVal ImageTime As String, _
                                         ByVal ReadID As String, _
                                         ByVal PartialOnly As Boolean)
       Dim ImageDisplay As Form1
       On Error GoTo ErrorSub
       ' Code to retrieve the image from the source system and
```

© N. Harris Computer Corporation

```
        ' display it to the user goes here
        Set ImageDisplay = New Form1
        ImageDisplay.ShowImageDetails VRM, _
                                      ImageDate, _
                                      ImageTime, _
                                      ReadID, _
                                      PartialOnly

        ImageDisplay.Show vbModal
        Exit Sub
    ErrorSub:
        MsgBox "Error:" & CStr(Err.Number) & ":" & Err.Description
    End Sub
```

7. Add the ProgID of your class module to the registry. The following three lines might be saved to a file with the .reg extension and imported into the registry through the RegEdit program:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\i2\iBase\8\ANPR]

"IR_ProgID"="iBaseANPR_IR.iBaseANPR_IR"
```

8. Compile your project.

Configuring iBase Plate Analysis

iBase database administrators can configure iBase Plate Analysis to ensure that the data is processed correctly by the iBase Plate Analysis.

To access the Configuration options you must be logged into iBase as a database administrator.

Important: Some of the configuration actions can cause disruption to other users of the database. If possible, perform the configuration when other users are not using the database.

Configuration involves four distinct steps:

1. Select the coordinate system and datum.
2. Select the entities that represent ALPR reads and events, and map their fields to ensure that the data is processed correctly by the iBase Plate Analysis functions.
3. Apply the configuration to save your settings.
4. Optimize the database by generating an index in the database.

In addition, you should select an appropriate datum for the specified coordinate system.

Select a Coordinate System

You can select one of two coordinate systems to be used when running iBase Plate Analysis.

Note: In all cases the coordinate system used for Plate Analysis records must be the same as that used by iBase event entities.

The following coordinate systems are supported by iBase Plate Analysis:

- BNG -- British National Grid (Easting and Northing)
- Geodetic - Decimal Degrees (Latitude and Longitude)

© N. Harris Computer Corporation

Select and Map the Required Entities

The purpose of the mapping process is to allow you to match the appropriate fields within iBase to the fields used for specific functions within iBase Plate Analysis. By providing a specific Plate Analysis configuration, you are not restricted to a rigid schema for your Plate Analysis entity. With the exception of the data fields that must be present to allow iBase Plate Analysis to work, you are free to add or remove fields in the Plate Analysis entity as required.

There are two distinct data areas from which the fields need to be selected:

- Data specific to the selected Plate Analysis entity
- Data specific to the entities that can be used as events for Geographical Profiling, appropriate entities could be crimes or incidents, for example

Plate Analysis Entity

The first task is to select an entity to represent ALPR reads. This must have fields for the following items:

- VRM
- Date
- Time
- Camera ID
- Capture ID (optional unless you want to use Retrieving Images)
- Location coordinates (two fields for Easting and Northing, or Latitude and Longitude, depending on the selected coordinate system).

Plate Analysis Entity Data

ALPR specific data is:

Field	**Used for**
Vehicle Registration Mark (VRM)	The number or license plate value captured by the ALPR camera
Date	The date of the ALPR camera read
Time	The time of the ALPR camera read
Camera ID	An identifier for the ALPR camera that captured the VRM
Capture ID	A unique identifier for the specific ALPR read
X coordinate	The X coordinate of the camera location
Y coordinate	The Y coordinate of the camera location

Note: Because of the potentially high number of Plate Analysis records, and the limited number of Plate Analysis specific fields, the X and Y coordinates for the Plate Analysis entity must be held in the

© N. Harris Computer Corporation

Plate Analysis entity itself, and not in a separate dedicated location entity. This is not the case for event entities; events can have the location stored in a specified linked entity.

Event Entity

You can map more than one entity as an event to be used for Geographical Profiling. The appropriate entity can be selected by the user when running the Geographical Profiling tool.

Note: If you do not specify any event entities, you will not be able to use Geographical Profiling, however the other iBase Plate Analysis functions will work as normal.

Event Entity Data

Events must have the following fields mapped:

Field	**Used for**
From Date	The start date of the event
From Time	The start time of the event
To Date	The end date of the event
To Time	The end time of the event
X coordinate	The X coordinate of the event. This location can be on another iBase entity such as a dedicated location entity
Y coordinate	The Y coordinate of the event. This location can be on another iBase entity such as a dedicated location entity

Note: The X and Y coordinates of the event can be taken either from the event entity itself or from a linked location entity. Many iBase schemas are designed to hold all location/address data, regardless of the originating entity type, in a single location entity. An example of this would be a particular entity that is designed to hold personal addresses, business addresses, and event addresses.

Apply the Configuration

When you have selected a coordinate system and mapped all the required fields, click **Apply** to save the configuration.

Optimize the Database for Plate Analysis

The final step in configuration is to click **Optimize**. This creates an index in the database that can be used to quickly locate data. This index is used for both Convoy Analysis and Geographical Profiling and improves the performance of both functions.

Note: The time taken to create the index will depend heavily on the amount of data; more data will take longer to index.

When you have optimized the database, you will not be able to remove the Plate Analysis entity from the database schema directly in iBase Designer. In the unlikely event that you do need to remove the

© N. Harris Computer Corporation

Plate Analysis entity from the database schema, you will need to remove the associated date field first, and then delete the entity. Leave the entity configured as the Plate Analysis entity until after it has been deleted.

i2 iBase GIS Interfaces

You can use i2® iBase Geographic Information System Interfaces (GIS) to connect to mapping applications. You can plot entities and links to maps and then analyze them visually using the analysis tools in the mapping application.

Sets and queries can be passed back to iBase for further analysis, and entities and links can be sent to and from i2 Analyst's Notebook.

Entities and links are plotted on the map by using geocode data, in the form of x and y coordinates. Extra data can be sent to the mapping application, and data can be displayed using icons or symbols, and labels.

Note: This option is only available if you install the i2 iBase Geographic Information System Interfaces (GIS) extended option. In addition, your database administrator might need to initialize your database before you can send data to a map or use a mapping configuration.

When you have iBase GIS Interfaces installed, and configured you can send and select data on maps from various places in iBase and Analyst's Notebook. Whenever you add data to a map or select data on a map, you need to select a mapping configuration.

The following options are available:

Adding new data to a map

Use **Add to Map** to send records to a map, whether selected from an iBase list, set, or in the results of an iBase query, or on an Analyst's Notebook chart. If the records are merged in Analyst's Notebook, all of the merged records are plotted individually.

Selecting existing items on a map

To select items that have already been sent to a map, select the entities or links in iBase, for example by creating a set or a query, and then use **Select on Map**. All of the selected entities and links represented by map items are selected on the map. If the entity you select is a merged chart item, all of the merged records are selected on the map.

Entities and links without geocode can also be selected, but since they are not plotted to the map, they do not appear selected. To view them,display the mapping application data table.

When the iBase GIS Interfaces are installed, you can access the mapping features from the following places in iBase:

- The **Add to Map** and **Select on Map** commands on the menu for entity and link records, queries, and sets.

  Note: Some mapping applications do not support the Select on Map command.
- The **Analysis > Mapping** menu.
- The **Add to Map** toolbar button.
- The **Functions > Mapping** in the database Explorer.

Mapping commands are also available in i2 Analyst's Notebook.

All supported mapping applications provide a set of iBase functions that:

- Show iBase records for map items.

© N. Harris Computer Corporation

- Create iBase sets.
- Create iBase queries based on the entities and links within the selected area of the map.
- Add or expand map items to an Analyst's Notebook chart.

In addition, in some mapping applications, you are able to display a density grid to highlight concentrations of items on the map.

The iBase GIS Interface for Esri ArcGIS Pro uses the built in capabilities of ArcGIS Pro to calculate the density grid and class divisions. This calculation is done differently to the other iBase GIS Interfaces, and therefore produces results that should not be directly compared with those from other GIS interfaces.

Note: `iBase GIS Interfaces Log.txt` records the number of records that are processed by iBase GIS Interfaces, how many were plotted on a map, and how many were ignored because of missing data. The file is created in the folder: `C:\Users\<your username>\AppData\Local\Temp`, and is overwritten each time you send data to the mapping application.

Setting up mapping configurations

You can configure how entities and links from iBase are to be plotted to a map by defining a mapping configuration. You can create many different mapping configurations to reflect the different ways you want to plot your data. Mapping configurations are stored with your iBase database.

About this task

Before you can send data to a map, you must select a mapping configuration. A mapping configuration defines how entities and links from iBase are to be plotted to a map. They define:

- Some details of the map you are using, such as the units, and minimum and maximum coordinates.
- How geocode data is provided. For more information, see Geocode data.
- How map items are displayed (either as icons or points).
- Whether labels are required and which labeling scheme to use.
- Whether extra data is required by the mapping application.

Procedure

To set up a mapping configuration:

1. Select **Format** > **Mapping Configurations**.
2. On the General page, select the mapping application and enter details of the map.
   a. From the **Mapping application** list, select the mapping application to which you want to send your data.
   b. Define the coordinates of the map that you want to use:

Option	Description
Offset	An offset is the difference between the origin (0, 0) position of the x and y coordinates in your geocode data and the origin used by the map. For example, if your coordinates are based on an origin of Washington but the map uses an origin of New York, then the offset is

© N. Harris Computer Corporation

Option	Description
	the difference between Washington and New York. Offsets must be specified in the units that are used in your geocode data and are multiplied by the factor. The default offset is 0.
Factor	Where the units in your geocode data differ from the units that are used by the map a factor is applied to scale the x and y coordinates up or down. For example, if your coordinates are in kilometers but the map is in meters enter a factor of 1000. The default factor is 1.
Min.	The minimum x and y coordinates specify the minimum values for the x and y coordinates. Any entities or links with coordinates outside the minimum values are not sent to the map. The default minimum value for x and y coordinates is -999999999.
Max.	Similarly, the maximum x and y coordinates specify the maximum values for the x and y coordinates. Any entities or links with coordinates inside the maximum values are sent to the map. The default maximum value for x and y coordinates is 999999999.

c. To display item labels, turn on **Display labels**. The default is not to display labels. For more information about selecting a labeling scheme, see Changing the display of map items.

3. Define where coordinates are going to be stored:

- If coordinates are stored directly in a record, no further action is required.
- If coordinates are calculated by converting address details in the mapping application, turn on **Send non-geocoded data**.
- If coordinates are held in a separate database rather than as part of the record details, click the **Geocoding** tab to enter its details. See Storing coordinates in a geocoding database.

4. On the Entities tab, set the entity types that contain mapping information, and how to handle that information. For more information, see Configuring how entities are plotted on maps.

5. On the Links as Points tab, set the link types that contain mapping information, and how to handle that information. For more information, see Configuring how links are plotted on maps.

6. Click **Save**.

© N. Harris Computer Corporation

Configuring how entities are plotted on maps

Entities can be added to maps. Before you can add entity information to a map, you must define how the mapping application handles that entity type.

Procedure

1. Select the entity types to plot.
2. To define the source of the geocoding data, click the name of the entity type and select
   - If your coordinates are stored in the entity, click **Location Fields** and then select the fields that provide the x and y coordinates.
   - If your coordinates are stored in a linked entity, click **Location Entity** and then select the link to the link type that contains the x and y coordinates.
   - If a geocoding database provides the coordinates, click **Geocoding Fields** and then select the fields in the entity or link that are used to match the records to the geocoding database.
   - If the mapping application provides the geocode data, you don't need to specify anything.

   For more information, see Geocode data.
3. To display links between plotted entities, turn on **Display links**. For links to be displayed on a map, they must exist in the database and must be included in the selected data, for example a query or set.

   Note: Displaying links between plotted entities is not the same as plotting links on a map as points. To display links between plotted entities no coordinates need to be provided because the entities themselves provide the locations of the link ends. Links plotted as points, however, require geocode data to locate them.
4. Click **Plot Style** to configure the display of entities as icons or points, and to select a labeling scheme. For more information, see Changing the display of map items.
5. If you need to send additional data to the map, click Extra Fields to select the required fields. For more information, see Sending extra data to the mapping application.

Configuring how links are plotted on maps

Links are configured in the same way as entities, except that you cannot plot links by using a linked entity to provide the geocode data.

About this task

Coordinates can be added to links from:

- Coordinate fields stored in the link.
- A geocoding database.
- Geocoding defined in the mapping application.

Procedure

To configure how links are plotted on maps:

1. Click the **Links as Points** tab to display the Links as Points page. All of the link types available in the database are displayed.
2. Select the link types that you want to plot.

© N. Harris Computer Corporation

3. If you need to define the source of the geocoding data, click the link type and then select:
   - **Location Fields** - If your coordinates are stored in a link, and then select the fields that provide the x and y coordinates.
   - **Geocoding Fields** - If your coordinates are provided by a geocoding database, and then select the fields in the entity or link that match the records to the geocoding database.
4. Click **Plot Style** to configure the display of links as icons or points, and to select a labeling scheme. See Changing the display of map items.
5. If you need to send more data to the map, click **Extra Fields** to select the required fields. See Sending extra data to the mapping application.

Storing coordinates in a geocoding database

You can use a geocoding database to store coordinate data that map to address data in your database. The geocoding database can be stored either in an iBase instance, or Microsoft Access. This enables you to plot entities and links without having to manually define coordinates in your database.

Procedure

To specify a geocoding database:

1. In the **DB path\name** box specify the database that contains the geocode data or browse for it. The geocoding database must be either an iBase database or a Microsoft Access database.
2. From the **Table name** drop-down list, select the name of the database table that contains the geocode data.
3. From the **X** drop-down list, select the field in the geocoding table containing the x coordinates to be copied to the corresponding field for matching entities and links.
4. From the **Y** drop-down list, select the field in the geocoding table containing the y coordinates to be copied to the corresponding field for matching entities and links
5. Specify fields to accurately locate an entity or link. Typically two fields are required for this. If necessary, fields 3, 4, and 5 can be used to select other geocoding table fields to match with entity or link fields:
   a. From the **Field 1** drop-down list, select the first field to be used to match the entity or link field to the corresponding field in the geocoding table.
   b. From the **Field 2** drop-down list, select the second field to be used to match the entity or link field to the corresponding field in the geocoding table.

Geocode data

Geocode data can be stored in a number of locations. To ensure that coordinates are detected and processed correctly, they need to be provided in a supported format.

There are four methods of providing geocode data (x and y coordinates):

- Two fields that contain x and y coordinates can be stored in the entity or link.
- Geocode data can be stored in a linked entity. For example, a person entity that has no geocode data can be linked to a crime entity that does contain x and y coordinates.
- Geocode data can be located in a geocoding database. For example, an entity or link might contain fields for the town and street name. When these fields are matched in a geocoding database, the x and y coordinates are supplied, enabling the entity or link to be plotted to the map.

© N. Harris Computer Corporation

- Non-geocoded data can be sent to a map and then positioned by using a geocoding database set up in the mapping application. For more information, see Setting up mapping configurations for details.

You can use a mixture of these methods.

Storing coordinates in records

If the geocode data for your entities or links are stored in the entity or link record, you need to set the fields that store coordinates in the mapping configuration.

Procedure

1. In the mapping configuration, click the **Entities** or **Links as Points** tab to display the appropriate page.
2. Click the name of the entity or link type and then click **Location Fields**.
3. From the **X** and **Y** lists, select the fields in the entity or link.

   Note: Ignore the **Address** and **Extra** boxes. These fields are provided for compatibility with earlier versions of iBase. If you need to supply extra data to the mapping application, use the extra fields. For more information, see Sending extra data to the mapping application.
4. Click **OK** to continue.

Sending extra data to the mapping application

Extra fields can be used in the mapping application, for example to apply different labels and symbols to map items or as address data for geocoding. You can send up to 10 extra fields for each entity or link type.

Procedure

To configure entity and link types to send extra fields of data from your database:

1. Select **Format** > **Mapping Configurations**.
2. Select the mapping configuration. See Setting up mapping configurations.
3. Click the **Entities** or **Links as Point**s tab to display the appropriate page.
4. Click the entity or link type that contains the required data.
5. Click **Extra Fields**.
6. To add an extra field, select the field from the Available fields list and click **>** to move it to the Selected fields list.

Results

The fields are displayed in the mapping application in the order in which they are listed.

Storing coordinates in linked entities

If the geocode data for your entities or links are stored in a linked entity record, you need to specify the record type that store coordinates in the mapping configuration.

Procedure

1. In the mapping configuration, click the **Entities** tab to display the Entities page.
2. Click the name of the entity type and then click **Location Entity**.
3. From the list, select the link type to the entity type that contains the geocode data.

© N. Harris Computer Corporation

4. Select the entity type at the link end that contains the geocode data.

   Note: To identify the coordinates, ensure that this entity type is added to the mapping configuration. For more information, see Storing coordinates in records.

5. Click **OK**.

Storing data in a geocoding database

You can store coordinate data in a separate geocoding database.

Procedure

1. In the mapping configuration, click the **Entities** or **Links as Points** tab to display the appropriate page.
2. Click the name of the entity type and then click **Geocoding Fields**.

   Note: To use geocoding fields, you must configure a geocoding database in the mapping configuration. For more information, see Storing coordinates in a geocoding database.

3. From the lists, select the fields in your database to match entities and links to the geocoding database fields.
4. Click **OK** to continue.

Changing the display of map items

You can configure whether map items display as icons or points and specify the labeling scheme. Each entity or link type can use different style and labeling settings.

About this task

You cannot change the style of associated links that are plotted as a result of turning on **Display Links** on the Entities page of the mapping configuration. This type of link is always plotted as a simple line.

Procedure

To configure the display options for mapped items:

1. Click the **Entities** or **Links as Points** tab, then click **Plot Style**.
2. To configure the style of map items:
   a. Click the Style tab, and select either **Point** or **Icon**. The default is **Icon**.
      - **Icon** - To plot entities, and links plotted as points, that use the default iBase icon for the entity type. When you are plotting links as points, you need to select an icon from the list. Each link type can use a different icon.
      - **Point** - To use a symbol to plot entities and links.

   b. If you select **Point**, select the type of symbol and its color.
   c. Select a size for the icon or point from the **Size** list. The default is 15.

   Note: To enhance performance, mapping applications cache icon files on startup. Therefore, to view any newly configured icons the mapping application must be restarted.

3. To configure the labeling of map items:
   a. Click **Label**, and from the **Labeling scheme** list, select the labeling scheme to use. The default is the iBase default labeling scheme.

© N. Harris Computer Corporation

b. Select the type of label:

- **Standard label** - To use the label style used within iBase.
- **Chart label** - To use the label style that is defined for use in Analyst's Notebook.

Note: By default labels are not displayed on the map. To turn on the display of labels, click the General tab in the mapping configuration and select **Display Labels**.

Identifying records without geocode data

When an iBase record has no geocode data, it can be sent to the mapping application, but it cannot be plotted on a map. If these items are identified, you can determine whether the mapping configuration is incorrect, or if the items are being sent in error.

Procedure

To identify the iBase records that are missing geocode data:

1. Select the mapping configuration. For more information, see Setting up mapping configurations.
2. Click the **General** tab, turn on **Send non-geocoded data**, and then click **Save**.
3. Send the data to the mapping application. A log file is displayed listing how many records were ignored.
4. To locate these items, look in the mapping application data table.

Removing temporary mapping files

Depending on the amount of data, and how frequently you send data to mapping applications, a large amount of disk space may be used by data mapping files. You should remove the data mapping files that accumulate in your temporary directory .

Procedure

Select **Analysis** > **Mapping** > **Clean Up**.

The **Clean Up** option only removes the files generated by the interfaces that are installed on your machine.

Coordinates in iBase

To plot an entity or link on a map, you need to enter coordinate values in two fields that have been set up for this purpose. Your GIS package is configured to interpret the values in these fields so that the data can be plotted in the correct location.

In iBase, you will also be able to store geographic data in a number of formats, which are then converted, either manually when you enter the record or automatically after an import or using a bulk conversion. You can also run coordinate queries.

Note: When you add a coordinate type field with two real number fields to make the combination for automatic conversion, the field order is important. The coordinate field must be followed by the latitude field and then the longitude field.

Types of field

The fields used to contain the coordinate data must be defined as Real Number type fields. They might contain the following types of coordinates:

© N. Harris Computer Corporation

- Latitude and Longitude values, entered in decimal degrees
- Easting/Northing data, entered in meters

These fields will typically be called Latitude and Longitude or X and Y. If you are not sure which fields you need to use, move the pointer over the field name to see its tooltip, or speak to your database administrator.

About converting coordinates to a standard format

When you convert coordinates, they are always converted to decimal degrees of latitude and longitude, using the WGS 1984 datum (a global standard for plotting geographic locations).

When you add a coordinate type field with two real number fields to make the combination for automatic conversion, the field order is important. The coordinate field *must* be followed by the latitude field and then the longitude field.

The original coordinate value is stored so that it can be searched for, and for audit purposes.

Note: The conversion process will change longitude values greater than 180 to their equivalent negative value in order that they can be plotted correctly.

Converting single coordinates

Depending on your data sources, you may need to convert coordinates into a standard format so the records can be plotted on a map. Converted coordinates are always stored as decimal degrees against the WGS 1984 datum.

Use the:

- Coordinate Conversion dialog (described below) to convert the coordinates in a single record
- Bulk Coordinate Conversion dialog (described in Converting Coordinates in bulk) to convert the coordinates in a database, set or query

Note:

You can only convert coordinates if there is a Coordinate type field followed directly by two other fields for the latitude and longitude (Real Number fields). These two fields will contain the converted coordinate values. The first field, stores the original coordinate before conversion.

See Coordinates in iBase for further background information.

See Supported coordinate systems for further information on converting data to the coordinate system standard to your organization. This topic also describes the rules that are applied when converting coordinates.

Converting coordinates directly

You can enter the coordinates and convert them immediately by:

- Typing the value in the Coordinate type field and then pressing the Tab key twice to move to the next field. The conversion takes place automatically.

If the value is not recognized, you will see an error message. In this case:

1. Click

© N. Harris Computer Corporation

to the right of the Coordinate type field to display the Coordinate Conversion dialog. If the format is recognized, the correct coordinate system is automatically selected.

2. Select the datum of the original coordinates if they do not use the WGS 1984 standard.
3. Click **Preview**. The converted coordinates are displayed, in the Standard Format area, in decimal degrees using the WGS 1984 standard.
4. Click **OK** to accept the values.

Entering coordinates using the Coordinate Conversion dialog

1. Click the

button to the right of the Coordinate type field to display the Coordinate Conversion dialog.

2. Select the required coordinate system.

   The boxes in the middle of the dialog will be updated to reflect the kind of data that the selected format requires.

3. Enter the values in the relevant boxes, and select any options that are required to identify the location, for example Northern Hemisphere or Southern Hemisphere or N/S and E/W.

   Note: You can also click Clear to delete any entered values and start again.

4. Select the datum of the original coordinates if they do not use the WGS 1984 standard.
5. Click Preview to display the converted coordinates in decimal degrees in the Standard Format area.

   If the value is not recognized, or in the incorrect format, an error message is displayed.

6. Click OK to accept the values.

If you do not know which coordinate system to use

If you have some coordinates but are unable to enter them directly because you do not know which coordinate system to use:

1. Click the

button to the right of the Coordinate type field (above the two coordinate value fields). The Coordinate Conversion dialog is displayed.

2. From the **Coordinate System** list, select **Auto Detect**.
3. In the **Coordinates box**, enter the coordinate values.
4. Click **Preview**. If the values are in a legitimate format, the coordinate system will be automatically selected.
5. Click **OK** to accept the values.

© N. Harris Computer Corporation

Converting Coordinates in bulk

To ensure that all the records that store coordinate data have a complete set of coordinates in a consistent format, you use the Bulk Coordinate Converter dialog.

All coordinates are converted to decimal degrees against the WGS 1984 standard. Converting your coordinates enables you to:

- Use the data in the Coordinate field to insert or update latitudes and longitudes (as decimal degrees). This allows records to be plotted on maps or for records to be included in coordinate queries.
- Use the data in the latitude and longitude fields to insert or update the coordinate field value. You may want to do this to ensure that your data is complete.

You can specify how the update is applied. You can also save a list of records that fail to update, for example because of insufficient data, in a set for review later.

Note: You can only convert coordinates if there is a Coordinate type field followed directly by two other fields for the latitude and longitude (Real Number fields). These two fields will contain the converted coordinate values. The first field, stores the original coordinate before it is converted.

See Coordinates in iBase for further background information.

Note: See Supported coordinate systems for further information on converting data to the coordinate system standard to your organization. This topic also describes the rules that are applied when converting coordinates.

To convert coordinates

1. Decide on the scope of the bulk conversion. For example, you can convert all the records in the database or you can restrict the conversion to the records in a query or set. You will convert only the records that you have access to.
2. To track which records converted successfully, you can create sets that you can review later:
   - To obtain a list of failures, turn on the **Add records that failed to update** checkbox, and enter the details of the set.
   - To obtain a list of successes, turn on the **Add successfully updated records** checkbox, and enter the details of the set.
3. Click **Next** to continue.
4. In the Update area, specify whether you are updating latitudes and longitudes or the coordinate field value as explained in detail below.
5. Select the datum of the original coordinates if they do not use WGS 1984. The datum you select will be remembered for the next time you use the Bulk Coordinate Converter dialog.
6. Click **Next** to continue.
7. Depending on the coordinate system:
   a. For UTM or UPS coordinates where the hemisphere is not specified, select **North** or **South**.
   b. For BNG coordinates, select the precision that you want to use (see Supported coordinate systems for details).
8. Click **Convert** to apply the conversion.

© N. Harris Computer Corporation

Updating latitudes and longitudes

You can automatically update latitudes and longitudes if the records have a value in the Coordinate type field, which will enable you to plot these records on maps.

Select the Update latitude and longitude field values option and then decide on the scope of the update:

Option	Update scope
Only if both the fields are blank	Select this option to update only those records that are missing both the longitude and latitude.
Only if either one or both the fields are blank	Select this option to update any records with a missing latitude, missing longitude or both.
Always	Select this option to update all records, including records that already have a latitude and longitude.

Updating the coordinate field value

You can automatically update the Coordinate type field if the records have latitudes and longitudes, which were entered either manually or by importing.

Select the **Update coordinate field values** option and then decide on the scope of the update:

Option	Update scope
Only if the field is blank	Select this option to update only records without a value in the Coordinate type field.
Always	Select this option to update all records, including those that already have a value in the Coordinate type field. For example, you might want to do this after editing a series of latitudes and longitudes.

Supported coordinate systems

There is a wide range of formats in which you can enter coordinate data. Be aware of the following points when you are using any of the following coordinate systems.

Military Grid Reference System

MGRS coordinates with less that the prescribed five-digit northing and easting values are accepted by iBase, but these low-precision values represent a large square surface area. For conversion purposes, the upper left corner of their effective area is used.

For example, 40UCE11 and BCE11 are interpreted, for conversion purposes, as being identical to 40UCE1000010000 and BCE1000010000.

British National Grid

BNG coordinates with better than 1-meter accuracy are not supported when automatically converting coordinates in bulk. You can choose one of the following options:

© N. Harris Computer Corporation

- Treat as conversion failure: the conversion is skipped so that you can review the record and update the coordinates as required.
- Round to nearest meter: this conversion automatically rounds the coordinate down to the nearest meter.

Due to the potential for overlap with Degrees, BNG coordinates that fall within 0, 0 and 360, 360 are not recognized. If you want to enter coordinates in this area, use a zone letter. For example, SV0030000300.

Decimal Degrees

Latitude and longitude must be within the range 90 - 90 and 180 - 360.

The flags N, S, E, and W can be replaced by words (North, South, East, and West) when this format is used: 01.00°X, and 02.00°Y. These values are not case-sensitive.

Decimal Minutes

Decimal minutes is not a natively supported system, so all decimal minutes formats are converted to decimal degrees and stored in the decimal degrees format.

Latitude and longitude must be within the range 90°S to 90°N and 180°W to 360°E.

The flags N, S, E, W can be replaced by words (North, South, East, and West) for the following formats (these values are not case-sensitive):

- 01° 02.00'X, and 03° 04.00'Y
- 01°02.00'X, and 03°04.00'Y

The characters that are assigned as the degree and minutes representations must remain constant for a single set of coordinates. For example:

- 56°45'N 32°14'W is valid.
- 56°45'N 32D14MW is not valid.

As a minimum, there must be a single character between the degrees and minutes if you omit the degree representation for the following formats:

- -01°02.00', and -03°04.00'
- 01°02.00'X, and 03°04.00'Y
- X01°02.00', and Y03°04.00'

If not using the degree representation, use a space instead. For example:

- -1234 8221.4 is not valid.
- -12 34 82 21.4 is valid.

Important: Decimal minutes formatted as minutes are not supported. For example, it is not valid to format 03° 04.00'Y as 184.00'Y.

Degrees Minutes Seconds

Latitude and longitude must be within the range 90°S to 90°N, and 180°W to 360°E.

The flags N, S, E, and W can be replaced by the words (North, South, East, and West) for the following formats (these values are not case-sensitive):

- 01° 02' 03.00"X, and 04° 05' 06.00"Y

© N. Harris Computer Corporation

- 01°02'03.00"X, and 04°05'06.00"Y

As a minimum, there must be a single character between the degrees and minutes if you omit the degree or minute representations for the following formats:

- -01°02'03.00", and -04°05'06.00"
- 01°02'03.00"X, and 04°05'06.00"Y
- X01°02'03.00", and Y04°05'06.00"

If not using the degree or minute representation, use a space instead. For example:

- 123443.6 822113.8 is not valid.
- -12 34 43.6 82 21 13.8 is valid.

Universal Polar Stereographic

The easting and northing values are interchangeable if E or N is used. If neither E or N is used, then the first number is assumed to be the easting value. For example, the following are all valid and represent the same point:

- 2,500,000mE 1,850,000mN
- 1,850,000 2,500,000mE
- 2500000 1850000

If easting and northing are swapped over, the final character on E must be "E" and the final character on N must be "N". Neither of these values are case-sensitive.

Configuring the iBase Environment

This documentation provides information about how to configure your iBase environment.

Setting session defaults

Your database designer can add standard fields that are present on all the entities and links in your database. You can set up default values for standard fields that will be used when items are created within the current session.

About this task

If you are creating multiple items that contain the same information, configuring a default value for your current session can reduce the time taken to enter that data manually. For example, if you are adding data that relates to a particular case, or from a specific source.

Note: If you are creating items that consistently contain the same information across several users, or a prolonged period of time, it might be more efficient to set up a field default. For more information, see Creating a field.

You can set up session based defaults for fields that meet the following criteria:

- must be set up as a standard field (available on all item types)
- cannot be set to read-only (for example system fields, ones you do not have security to write to, or fields containing calculated values
- are of the following types:
  - Coordinates
  - Hyperlinks

© N. Harris Computer Corporation

- Link Strength
- Multi-line Text
- Multi-line Text (Append Only)
- Security Classification Codes
- Selected From List
- Suggested From List
- Text

When you set up a session default, you add the specified value for the field to each record added during a particular session, whether manually or when importing. When you create a new record, you see the fields' values already set to the default, which you can override if you want to.

Note: If you leave iBase and restart, the session is complete and the session default values are reset to blank.

Procedure

1. In iBase User, open **Tools** > **Session Defaults**.
2. Select the cell in the Value column for the field you want to set.
3. Enter or edit the default value, or select a value from the list.
4. Click **OK** to set the session defaults.

Code Lists

Code lists are the lists that are available when you need to choose from certain predefined values. For example, for a Person entity type, this could be a list of values for hair or eye color.

There are three types of code list:

- Pick lists (drop-down lists) provide textual values for Selected from Code List or Suggested from Code List fields.

Pick lists can be filtered, so that the value selected in one list controls the values available in the list below it.

- Icon lists provide access to icons, typically an appropriate subset of the icons supplied with iBase.
- SCC lists for use only with the Security Classification Code field type.

Code lists are created by database designers but you may be able to change their contents.

Editing a Pick List

Pick lists, or drop-down lists, determine the values available for selection when using a Suggested From Code List type field or a Selected From Code List type field. Your system administrator may restrict who is allowed to edit pick lists.

About this task

Pick lists, and the fields they apply to, are defined in iBase Designer. Filtered lists, where one list is assigned as the parent of another, are also set up in iBase Designer.

There are two versions of the Pick List dialog, depending on whether the list is filtered or not. Filtered lists allow you to assign a group of values to one or more values in a parent list.

© N. Harris Computer Corporation

Note: Some similar fields may use different code lists. The name of a list can indicate what fields it is used by. For example, a Hair Color list might be used for a Hair Color field in Person entity type records, and a Vehicle Color list might be used for the Vehicle Color field of a Vehicle entity type. Although both lists contain lists of colors they are separate pick lists.

When you rename a pick list item in the Pick List dialog, any record containing a field for which that value had been selected keeps the existing value for the field, that is, keeps the value that was selected at the time.

If you subsequently edit a record containing a field for which the renamed value had been selected:

- Selected From Code List field types: the old value is still selected and appears at the bottom of the pick list. The renamed value is also available in the list. Either value can be selected.
- Suggested From Code List field types: the old value is still selected, but does not appear in the list. Any value can be entered.

Procedure

1. Display the Pick List dialog for the pick list you want to edit. From the Edit menu, select Code Lists, then Pick Lists, then select the list you want to edit.
2. In the Items list, select an item.
3. To edit the item name, click in the Value column and enter the name. You can press the **Esc** key to reset the item to its previous value.
4. To edit the item description, click in the Description column and enter the description. Item descriptions are optional but can help you to choose the right value when entering data in a record.
5. To add a new item, do one of the following:
   - Click the row below where you want the new item, and then click **Insert Row**. Enter the name and description in the new blank row.
   - Click in the blank row at the bottom of the list. This adds a new item at the end of the list. You can then use the Move buttons to move the item to where you want it in the list (unless you are viewing all the items, in which case you cannot reposition the item manually).
6. To delete an item, select the row and click **Delete**. You can also right-click, and from the shortcut menu, select **Delete** or press the **Delete** key. If you delete an item that is assigned to multiple items, you will be given the option of removing the assignments but leaving the item in the list, or deleting the item altogether.
7. Click **OK** to confirm your changes and close the dialog.

Editing an Icon List

Icon lists determine the icons available for selection when using an Icon type field. The left list contains all the available icons from the Icon List file as set up by the system administrator.

About this task

The right list contains the icons in this icon list. Your system administrator may restrict who is allowed to edit icon lists.

- Not all icon fields use the same list. The name of a list can indicate what field(s) it is used by. For example, the Crime Icon list might be the list for the Icon field of Crime entity type records.
- The icon lists, and the fields they apply to, are defined in the database design.
- The Name may only be changed in iBase Designer.

© N. Harris Computer Corporation

- Icons can have a different shading color applied on a record by record basis. The icon shading shown in this dialog is the standard (default) color, which will be used in iBase unless you change it.

Procedure

1. In the Icon List dialog, click the Description box and enter or edit the description. For example, use this to give a useful hint on where the list is used.
2. Click to select an icon in the left Items list and note its preview to the right of this list. If you want it to be available in the icon list, click **Add** to move it to the right list.
3. Click to select an icon in the right list and note its preview to the left of the list. It you want to remove it from this icon list, click **Remove** to move it to the left list.
4. Click **OK**.

Editing an SCC List

Security Classification Code lists determine the options you can select when using a Security Classification Code type field. Your system administrator may restrict who is allowed to edit SCC lists.

About this task

Note:

- Not all fields of this type use the same list. The name of a list may indicate what fields it is used by.
- The Security Classification Code lists, and the fields they apply to, are defined in the database design.
- The name may only be changed in iBase Designer.

Procedure

To edit SCC lists:

1. Click the **Description** box and enter or edit the description. For example, use this to give a useful hint on where the list is used.
2. In the **Items** list, click to select an item:

   a. To start editing the text, click again.

   b. To delete the item, right-click and select **Delete Item**.

   Note: Do not edit or delete items that are used in data records.
3. Add a new item using the blank item at the bottom of the list.
4. Click **OK** to confirm your changes and close the dialog. Other users who are logged on to the database will continue to see the old list until they close and reopen the database, or display the SCC List dialog and then click **OK** to close it (which updates the contents of all the lists).

Filtered Pick Lists

Filtered pick lists aid data entry by creating a parent-child relationship between two consecutive pick lists so that the selection of a value in the parent list limits the values available in the child list to only those that are relevant or suitable. This can speed up data entry and ensure consistency in the database.

There are two main reasons for using a filtered pick list:

- It shortens a list that you would otherwise find difficult to use because of the number of items on it.

© N. Harris Computer Corporation

- It prevents incorrect values from being selected---selecting the value from a first list restricts the values on a second list to only the values appropriate for the field based on the selected parent entry.

The child list can be thought of as an amalgamation of several sublists, each one relevant for a single item in the parent.

Filtered pick lists can be arranged into a hierarchy of any number of levels, such that the value selected in the first list filters the available values in the second list, and the selected value in the second list then filters the values available in the third, and so on.

Note:

A filtered pick list may be used elsewhere as an ordinary pick list, when it is not directly beneath its assigned parent list. In this case, all the values in the list will be available for selection, sorted alphabetically.

Two consecutive pick lists will not function as parent and child, if they have not been set up to behave in that way in iBase Designer. In the example above, there are several consecutive drop-down lists for fields -- Vehicle Type, Make, Model, Color, Vehicle Style -- but only Make and Model are set up as a parent-child pick list.

The parent-child relationship between two pick lists is defined in iBase Designer. Groups of items in a child list are then assigned to an item in the parent list. These assignments can be set up in iBase Designer but can also be created and edited in iBase by any user with sufficient permissions.

A pick list can only be assigned to a parent list in iBase Designer. Items in the child list may already have been assigned to parent items in iBase Designer, but you may have permission to do all the following: reassign items from one parent item to another, remove items altogether and add new items to both the parent and child list.

Editing Filtered Pick Lists

Pick lists, or drop-down lists, determine the values available for selection when using a Suggested From Code List type field or a Selected From Code List type field.

About this task

Pick lists can be filtered, such that each value in one pick list can be assigned to one or more values in a parent list. This improves data entry as it allows available values to be filtered.

Procedure

1. In the Pick List dialog select an item from the Items list.
2. To edit the item name, click in the Value column and enter the name. You can press the Esc key to reset the item to its previous value.
3. To edit the item description, click in the Description column and enter the description. Item descriptions are optional but can help you to choose the right value when entering data in a record.
4. To add a new item, do one of the following:
   - Click on the row below where you want the new item, and then click Insert Row. Enter the name and description in the new blank row.
   - Click in the blank row at the bottom of the list. This adds a new item at the end of the list. You can then use the Move buttons to move the item to where you want it in the list (unless you are viewing all the items, in which case you cannot reposition the item manually).

© N. Harris Computer Corporation

5. To delete an item, select the row and click Delete. You can also right-click, and from the shortcut menu, select Delete or press the Delete key. If you delete an item that is assigned to multiple items, you will be given the option of removing the assignments but leaving the item in the list, or deleting the item altogether.
6. Click OK to confirm your changes and close the dialog.

Using Filtered Pick Lists to Enter Data

Filtered pick lists contain a list of available values determined by the selection of a value in the pick list above. When you are entering data in a record or datasheet and there are two consecutive pick lists, the value entered in the first pick list may control the values available for selection in the one below it.

The relationship between a parent and child pick list is defined in iBase Designer. Only a pick list which has been assigned a parent pick list and is also directly below its parent pick list in the data record or datasheet is filtered.

Note: A filtered pick list may be used elsewhere as an ordinary pick list, when it is not directly below its assigned parent list.

Filtered lists can be of two types, depending on the field data type:

- Suggested From Code List pick lists allow you to enter any value
- Selected From Code List pick lists force you to choose a value in the list

Categorizing queries, sets, and definitions

Queries, sets, report definitions and browse definitions, provide a means of grouping the entities and links that are stored in your database. You can manage these objects by storing them in folders that are labeled based on specific categories.

By default, iBase presents sets, queries, and definitions in single general lists. If your lists are long, you can use categories to make them easier to work with. For example, if some of your sets group suspects of particular types of crime, you can create a category that contains only those sets.

Category folders behave like standard folders, with the following exceptions:

- Category folders cannot be renamed.
- Category folders are not displayed if they are empty.
- Access to the contents of some category folders might only be available if you are a member of a particular user group.

Setting the access on folder objects

1. When you modify multiple folder objects that currently have different access restrictions, a Change Access checkbox will be displayed in the Categorize dialog. You first need to turn on this checkbox before modifying the access restrictions.
2. Select one of the following:
   - Public

     Any user can access the folder objects.
   - Private

     Only the person who flagged the folder object as private and the system administrator can see it. For example, unless you are logged on as a system administrator, private report definitions belonging to others will not appear in the All Report Definitions folder.

© N. Harris Computer Corporation

Note: If you are a member of a folder object control group, membership of this group may prevent you from setting the access on folder objects to private. Access to the object will always be set to the folder object group. For detailed information, see below About folder object control groups.

- Restricted to groups

  Only users who are in the group(s) can see these folder objects. With Restricted to groups selected, check the Folder Object Control group(s) that you want to have access - no other groups have access. You only see the Folder Object Control groups of which you are a member. For detailed information, see the About folder object control groups section.

3. Click **OK** to confirm your changes and close the dialog.

About folder object control groups

A folder object control group is the most restrictive of the three types of folder access: public, private, and group. Folder object control groups are defined in iBase Designer using the Security Manager but their usage is defined by the users who belong to the group, and the following two settings made in the Options dialog:

- Default to 'Public' Access
- Prompt for category when saving

How these Option dialog settings affect the use of folder object control groups is summarized below.

Category prompt	**Default access type**	**Result when you save a folder object**
ON	Private	You are prompted to select the access type for the folder object. The default access type is the folder object control group to which you belong. If you belong to several groups, the access type defaults to private.
ON	Public	You are prompted to select the access type for the folder object - the default access type is public.
OFF	Private	The folder object is automatically saved in the folder object control group to which you belong. If you belong to several groups, you are prompted to select one or more groups, or to change the access type.
OFF	Public	The folder object is automatically saved with public access - overriding the membership of the Folder Object Control group. If you belong to several groups,

© N. Harris Computer Corporation

Category prompt	Default access type	Result when you save a folder object
		you are prompted to select one or more groups, or to change the access type.

Setting default categories

When you save queries, sets, and definitions, you add them to specific category folders. You can specify the default behavior to use when you create items of these types.

Procedure

To modify the default options:

1. Select **Tools** > **Options** > **General**.
2. Select the options to modify:

Option	Description
Default Category Name	To avoid the need to reselect a different category each time, you can set a default category.
Default to 'Public' access	Access to items in a category can be either public, or private a specific user. Private items cannot be listed or viewed by any other user (apart from the system administrator).
Prompt for category when saving folder objects	Queries, sets, and definitions are always saved in a category. However, by turning on this checkbox you can select the category and set the access type.

Saving items into categories

You can save a query, set, or definition into specific categories to sort them and control access.

Procedure

To save an item in an existing category:

1. Click **Save**. You are then prompted for the item name.
2. Enter the name for the item, which must be unique to the database, and then click **OK**.

   The default category is displayed in the Selected Category box. If the default category is empty (does not exist yet), it is displayed in brackets next to the text Selected Category.

   Note: If you are not prompted to select a category after clicking Save, then you need to follow the steps in Moving items between categories.
3. Select the category in which to save the item.

© N. Harris Computer Corporation

Note:

To create a new category, simply enter the names of the categories you want, separated with a backslash (\. For example: `Operation Crest\Unit B\Vehicle Owners`

In this example, both the categories for `Operation Crest` and `Unit B` will be created for you if they do not exist.

4. **Optional:** Restrict who can access the item by clicking Restricted to groups and then selecting the groups who can access the item:

   - **Public**

     Any user can access the items.

   - **Private**

     Only the person who flagged the folder object as private and the system administrator can see it. For example, unless you are logged on as a system administrator, private report definitions belonging to others will not appear in the All Report Definitions folder.

     Note: If you are a member of a folder object control group, membership of this group may prevent you from setting the access on folder objects to private. Access to the object will always be set to the folder object group. For detailed information, see below About folder object control groups.

   - **Restricted to groups**

     Only users who are in the groups can see these items.

     With **Restricted to groups** selected, check the groups that you want to have access.

     Note: You only see the groups of which you are a member. Group access to categories

Moving items between categories

You can move queries, sets, and definitions into different categories as required.

Procedure

To move one or more items to a different category:

1. Select the item or items that you want to move.

   Note: To select items from several categories, you need to open the All folder for the item type, for example All Report Definitions.

2. Right-click on one of the items and from the menu, select **Categorize**.
3. If you are moving multiple items that are currently in different categories, **Change Category** is displayed. Turn on this checkbox to indicate that you want to continue.

   Note: You cannot change the category of an item from a public folder to a restricted folder while another user has that item open on their screen.

4. Specify the category that you want to use.
5. If you do not want to set access control on the items, click **OK**. If you want to restrict who can access these items, seeGroup access to categories.

Group access to categories

A folder object control group is the most restrictive of the three types of category folder access: public, private, and group. Folder object control groups are defined in iBase Designer using the Security

© N. Harris Computer Corporation

Manager but their usage is defined by the users who belong to the group, and the defined category settings.

The following options can be found in **Tools** > **Options** > **General**:

- Prompt for category when saving
- Default to 'Public' Access

How these settings affect the use of folder object control groups is summarized below:

Category prompt	**Default access type**	**Result when you save a folder object**
ON	Private	You are prompted to select the access type. The default access type is the folder object control group to which you belong. If you belong to several groups, the access type defaults to private.
ON	Public	You are prompted to select the access type for the folder object, the default access type is public.
OFF	Private	The folder object is automatically saved in the folder object control group to which you belong. If you belong to several groups, you are prompted to select one or more groups, or to change the access type.
OFF	Public	The folder object is automatically saved with public access.

iBase Settings

You can specify general, charting, and advanced settings in the iBase **Options** dialog.

Click the **Tools** tab on the iBase menu, and select **Options**. The **Options** dialog opens.

- Click the **General** tab to specify basic settings, for example, how you use categories. These are personal settings that do not affect other users.
- Click the **Charting** tab to set defaults used when charting in Analyst's Notebook, unless specified in a charting scheme or the **Charting Settings** dialog. These are personal settings that do not affect any other users.
- Click the **Advanced** tab to set the User Settings and the Local Machine Settings, for example, the location of the templates folders.

General settings

Default Category Name

© N. Harris Computer Corporation

Choose a default category name to use when you save a new folder object, such as a set. The default folder name is set to 'General'.

This applies to all database users when no default category is specified in iBase Designer. If default categories are defined for users in iBase Designer, that is the default folder where all objects are saved.

Prompt for Category when Saving Folder Objects

Displays a prompt for a category when you save a new folder, such as a set.

If you deselect this option, folder objects are automatically saved in the default category with default access type wherever possible. However, the Categorize dialog is always displayed if you belong to more than one folder object control group.

Default to 'Public' access

Determines whether access to a folder object is public, private to the user who flagged it as private, or restricted to members of a folder object control group. Private folder objects can only be listed and viewed by the user who flagged it as private, and the system administrator.

Maximum number of most recently opened databases to show in the file menu

Each time a new database is opened, an entry is placed in the Most Recently Used list. You can see the most recently used databases in the **File** menu. Selecting an entry is a quick means of re-opening the database.

This setting determines the maximum number of entries in the list. Once the limit is reached, new entries at the top push the bottom entries off the list.

Number of rows to be displayed in a multi-line text box

Determines the size of the box when entering or editing data in multi-line text type fields, in terms of the number of lines it can display.

Open last used database on start-up

Opens the last database used when you start iBase. A prompt gives you the option to cancel the open and select an alternative database.

Check for matching records whenever a discriminator field value changes (datasheets only)

Displays a warning about possible duplicate records when data entered in a discriminator field matches one or more existing records. Applies to datasheets only.

This option checks data at an earlier stage than the creation of matching records warning that occurs when you attempt to save the record.

Prompt to confirm creation of matching records

You are always warned when you attempt to save a record that creates a possible duplicate. However, you can specify an additional prompt that appears when you click Yes to create the record.

Remember user for Windows single sign-on

Depending how Windows security is set up at your site, you might be prompted to select the user to log on as. To avoid repeating this step each time, you can select the Remember my selection checkbox in the Logon dialog. You will then log on automatically in future sessions.

© N. Harris Computer Corporation

Deselect the 'Remember user for Windows single sign-on' checkbox when you need to log on as a different iBase user.

Display dialogs in tabs

Choose whether new dialogs appear as tabbed or independent windows. By default this is set as tabbed.

Use legacy icons

Change the appearance of entity icons to display the 8.1 or earlier icon set. By default the latest icon set is displayed.

Activate global shortcut keys

Choose whether to use and potentially customize the global shortcut keys. By default this is set to on.

Default charting settings

Use the Charting page of the Options dialog to set the basic options for charting in Analyst's Notebook. These settings can be changed in Analyst's Notebook for the duration of the session, using the Charting Settings dialog.

Chart entity/link attributes

When a record is added to a chart to become a chart item, whether chart attributes are added or not.

Only applies if:

- Chart attributes are defined for the entity or link type in the database design.
- Not overridden in the charting scheme by the Chart Attributes option settings (for the entity or link type or 'Defaults').

Chart pictures to represent entities instead of their icons

When an entity is added to a chart and has a Picture type field, whether the picture field value is used to represent the chart item instead of the entity's icon.

It only has an effect if there are no applicable Chart Pictures options settings in the charting scheme (for the entity type or 'Default') that have a non-'Blank' setting.

If there is more than one picture type field, the top field when the entity is open in a Show dialog is the one used. If you are in doubt, and your entities are not displayed in a Show dialog (they may open in a datasheet-based dialog), consult your system administrator.

Rearrange new items added to a chart (not the whole chart)

How much a chart is re-arranged to accommodate newly added items.

Default Link Label

The labels for chart links added from iBase.

The selected option becomes the default selection in the Charting Settings dialog.

Multiple Link Style

How iBase links added to charts are represented on the chart. The selected option is the default selection in the Charting Settings dialog.

© N. Harris Computer Corporation

Advanced settings

The options on the Advanced page in the User Settings area of the dialog affect just you. Because the options in the Local Machine Settings area affect all users of this computer, your system administrator may prevent you from changing these settings.

User Templates Folder

Path name of the folder containing templates for creating new databases. Any user can change this path.

Temporary Files Folder

Path name of the folder for temporary files.

These files are created when, for example, you use View to edit a document specified in a document type field.

Any user can change this path.

Command Group File

Path name of the Access database that organizes the iBase command access control.

Icon List File

Path name of the file that lists all the available icons when, for example, you are editing an Icon List code list.

Workgroup Templates Folder

Path name of the folder containing database templates for use by all users of this machine.

Graphics Folder

Path name of the folder containing the icons used in, for example, the database explorer and menu items (but not the entity, or entity type, icons).

Number of records to be displayed before auto-pausing

The number of records to be displayed before an automatic pause. You might see this, for example, when records are being loaded in a Browse dialog.

Specify '0' to disable auto-pausing.

Note: Contact your system administrator before disabling this, or setting it to a high number, as doing so may result in large numbers of alerts being raised.

Managing Plug-ins

A plug-in is a software component that extends the basic functionality of iBase.

You need to activate plug-ins before they can be used. You might want to deactivate any plug-ins not in use to simplify the user interface, and use less memory. Plug-ins that can be installed on your system include:

Plug-in	Description
Alerting Configuration	Used in iBase Designer to set up alerting.
Alerting Inbox	Used in iBase Designer to set up alerting.

© N. Harris Computer Corporation

Plug-in	**Description**
Audit History Viewer	In SQL Server databases, you can store the history of the changes that are made to records in the database.
Bulk Coordinate Validator	Ensures that coordinate formats are valid when converting coordinates in bulk.
Chart Importer	Allows the contents of ANB charts to be imported into an iBase database.
Composite Indexes	Allows the creation and management of SQL Server composite indexes from within iBase Designer.
Custom Terminology Builder	Provides the ability to customize the text displayed within iBase applications.
Database Subsetting	Manages the creation and synchronization of database subsets.
Duplicate Checker	Used to identify records that contain similar information.
Excel Interface	Used to export data to Microsoft Excel.
Field Attachments	Enables attaching of files to individual fields.
Field Security	Enables security classification codes to be applied to individual fields.
Full-Text Search	A method of searching SQL Server databases. In later versions, the method of searching SQL Server databases is Search 360.
Import and Export	Provides data importing and exporting.
Merge Links	Allows links of the same type to be merged into a single link.
Schema Update	Used in iBase Designer to manage changes to the schema.
Search 360	The search mechanism for Microsoft SQL Server databases.
Valid End Types Utility	Used in iBase Designer to set the valid record types to add to the ends of a link type.

© N. Harris Computer Corporation

Plug-in	Description
Word Search	The search mechanism for Microsoft Access databases.
XML Import and Export	Used in both iBase and iBase Designer to import and export data as XML.

After making changes, you need to restart iBase or iBase Designer for the changes to take effect.

To manage plug-ins:

1. In the Plug-in Manager dialog, turn on a checkbox to activate a plug-in or turn off a checkbox to deactivate a plug-in.
2. Click **OK** to confirm your changes and close the dialog.
3. Restart iBase or iBase Designer for the changes to take effect.

Defining Common Folder Objects

Defining common folder objects is part of the Schema Update process.

About this task

Schema Update uses a template generated from one database to update the schema of another. This is useful when you have a group of databases which you need to keep consistent in their design and content. Updating a database schema can only be done in iBase Designer using the Update Schema command. See the Designer online help for more details. The full process for updating a schema is as follows:

1. In iBase, define the group of folder objects in the Common Folder Objects dialog that you want as a core set of folder objects across a number of databases. You can, if required, view or edit an object (such as a query or browse definition) to test that it is the one required or update it, before making it common.

Sets cannot be defined as common folder objects as they refer to specific records in the database, which will not exist in other databases.

1. In iBase Designer, make any other changes to the schema that you want to apply to the databases.
2. Generate the template.
3. Use Schema Update to apply the template to each database in turn. The common folder objects will be synchronized between the source database and target database, that is, the result of the update will be that the same set of folder objects exists in both databases.

Common folder objects across databases:

- have identical names
- are in the same categories
- have an identical definition

Procedure

To define a folder object as a common folder object

© N. Harris Computer Corporation

1. From the Tools menu, select Common Folder Objects. The Common Folder Objects dialog is displayed. It lists all the folder objects in the database.
2. In the Types area, select the type of folder object that you want to review. Sets are excluded from the list because they are dependent on specific records existing in the database. Labelling schemes are also excluded. The hierarchy of folders and the objects in the All folder are listed on the right.
   - To include a folder object in the template, turn on its checkbox. When the template is used to update the schema of another (target) database, the object will be added to the group of common folder objects in the target database. If the object exists as a common folder object in the target database already, it will be updated so that it matches the object in the template.

     Categories are also copied across so that there is a consistent set for all common folder objects.
   - To exclude a folder object from the template, turn off its checkbox. The object will be removed from the template. When the template is used to update the schema of another (target) database, the object will be removed from the group of common folder objects in the target database, if it exists. The folder object will still exist in the source database, from which the template was generated.

     If the common folder object has been renamed since it was last applied to other databases as a common object, this may cause a clash with another object in the target database which has the new name. In this case, the existing object in the target database is renamed - an underscore is added as a prefix. During the update process a message will inform you of any renamed objects in the target database. Renamed objects are easy to identify as they appear at the top of the list due to the underscore prefix.
3. Click OK to update the common folder objects for this database. You will be warned if a common folder object:
   - is dependent on a set. Sets cannot be saved in a template so you must turn off the checkbox for any object that depends on a set.
   - is dependent on a folder object that has not been selected as a common folder object (or that has been deleted).

   The setting of these checkboxes is saved in the database and redisplaying the Common Folder Objects will display the common folder objects defined in the database.

   To make the folder objects common to a group of databases, you now need to create a template from this database, and apply it to each target database in turn using the Schema Update command in iBase Designer.

Upgrading iBase

Upgrading your system to a later version of iBase.

To upgrade iBase:

- You must have administrator privileges
- You should back up any databases before starting your upgrade
- You should ensure that your any prerequisites are still supported and if not, upgrade these prerequisites before starting your iBase upgrade
- If your existing deployment is earlier than iBase version 8.9.13, upgrading is a two-step process. You must first install iBase 8.9.13 and open all databases in iBase Designer, before you can proceed to the latest version.

© N. Harris Computer Corporation

Note: Your database is upgraded when it is first opened in iBase Designer, and not as a part of the software installation.

First, ensure that the server's hardware and software are compatible with both versions of iBase. See the iBase release notes listed on the i2 support site for details.

Important: Always back up iBase before you perform any maintenance work. If you need to roll back the upgrade, you can use the backup as a restore point. For more information, see Backing up the system.

Ensure there are no users connected to iBase User, iBase Designer, and any applications that might be connected to iBase, such as i2 Analyst's Notebook.

Ensure you have no scheduled imports or indexes set to run.

Download the latest version of iBase. Extract the installer and run it as indicated in Installing i2 iBase.

This upgrades your current version of iBase to the latest version.

Important: You must upgrade all of your iBase clients at the same time. After the database is updated, your older clients might not be able to work with the database.

When the client upgrade is complete, upgrade the iBase databases that you use.

If you use Search 360, you also need to install the newer version of the Search360indexer. This enables the indexer to work properly with the database. The indexing tool is listed under the options in the iBase installer.

When you have upgraded all your clients to the latest release, they can start using the database again.

Planning the upgrade

The upgrade process is designed to meet the needs of multiple types of environment. Depending on the setup within your organization, a number of deployment options are available to minimize the impact on day-to-day operations.

Before you upgrade iBase, consider the following areas:

Original system configuration and use

Every iBase deployment is configured to match the needs of different users, so it is important that you recognize which features are used, and ensure they are configured correctly after the upgrade.

Requirements of the system after the upgrade

Perform an analysis of the new features of iBase to ensure that all the features relevant to your organization are set up correctly.

Client location

The installation process will differ depending on the number of users and the method of connection used by iBase clients when connecting to the server. For deployments in organizations that only use locally hosted databases and clients, the upgrade can take place in a single phase. For deployments in organizations that include remote users, large numbers of clients, or who share data with other agencies, a phased approach to the upgrade might be more suitable. A phased approach gives all relevant parties a chance to upgrade their client software before the database upgrade takes place. Depending on the number of client machines involved, you can carry out deployment of the iBase clients manually, or schedule it for automatic deployment using enterprise systems management software.

Timescale for deployment

© N. Harris Computer Corporation

Creating a realistic timescale for deployment will allow you to plan the deployment around the peak times of your system use.

Best practices throughout the upgrade process

There are a number of practices that you can follow when upgrading your iBase system. If you follow best practice, you can reduce the likelihood and impact of issues.

Test the upgrade first

Perform an upgrade to iBase in a test environment before you perform it against a live system. The test environment you use should closely match your current system, allowing you to troubleshoot any issues that arise.

Communicate with all system users

Once the upgrade testing has been successfully completed, you should send a communication to all system users advising them of the planned work, timescales, and how this will affect them. You should continue this communication throughout the process, alerting users to the progress that has been made.

Back up your system

It is of utmost importance that you make a complete backup of your existing iBase configuration and databases and verify they are successfully backed up before and after you perform the upgrade.
To back up your system, follow the instructions in Database upgrades to ensure that the upgraded databases are successfully backed up and your changes stored.

Ensure that backups are stored in different secure locations to insure against any unforeseen issues and to prevent extra space being used on your Server.

Use supported prerequisites

iBase only supports the operating systems and supported software listed in the detailed system requirements. Ensure that existing deployments are upgraded to currently supported versions of all the software required. For more information about the software supported for the current release, see the system requirements.

Database Administration

Both before and after the database has been upgraded, a Schema Integrity Check should be carried out to ensure that the database is in a stable condition. The Wizard checks the schema, reporting any problems that were found and offering to fix those it can repair.

Additional actions for users of iBase Scheduler

When making any changes to the database, you should ensure that the iBase Scheduler has completed any outstanding import or export tasks, and has been turned off before the change takes place.

If Scheduler databases are on more than one SQL Server, an alternative to stopping the Scheduler service is to deactivate any Scheduler triggers that operate over the databases on the server you are upgrading. This will ensure that scheduled imports for databases on other servers continue unaffected.

When the modification has been completed, enable iBase Scheduler to perform its imports and exports by either starting the Scheduler service or reactivating the Scheduler triggers using the iBase Scheduler Configuration tool.

Additional actions for replicated databases

You must carefully plan the upgrade of replicated security files and databases. Data consistency issues can arise if changes are made to the security files or databases at multiple sites while the databases are not being replicated.

© N. Harris Computer Corporation

When making any changes to the database, you should ensure that database replication has been stopped by the SQL Server DBA before the changes are carried out. Replication of the audit log is optional, but if you have chosen to replicate it, the SQL Server DBA must stop replication on this database too.

The upgrade process

In order to minimize the disruption to end users, you can upgrade the database and clients separately, reducing the time that systems need to be continuously offline

To reduce issues, the databases should be upgraded before clients. In order to complete the upgrade, the database must be opened and closed in iBase Designer before use. After you have upgraded a database, you should not attempt to access that database with an earlier version of iBase Designer.

Database upgrades

Database upgrades change the structure of your database to add new features. Usually, you will upgrade all security files and databases, but there might be occasions when you need to keep some databases in an earlier iBase format.

Important: Between backing up and upgrading the databases, you must not allow any users to access iBase security files or databases. Locking users out of the database after the backup ensures that no data loss occurs in the unlikely event that you encounter problems during the upgrade.

If there's a reason why you cannot upgrade a particular database (for example, you might share it with another organization that is not in a position to upgrade), you can still selectively upgrade other databases. Be aware that when you upgrade a security file, you *must* also upgrade all databases secured by that security file. To upgrade a subset of databases that are secured by a single security file, you must copy the security file and move the database that you want to convert to a new location.

Note: Copying a security file that's in SQL Server format requires that you also copy the security database to a new instance of SQL Server.

To perform an upgrade of an iBase database and its security file:

1. If your security file is in SQL Server format, ask the database administrator to bring the security database online following the successful backup.
2. Open the security file in iBase Designer to start the upgrade.
3. Ask the database administrator to bring the log file and the main database online.
4. Open the main database in iBase Designer.
5. When prompted, select the Upgrade option to initiate the upgrade.

   During the upgrade process, iBase Designer displays a series of warning and status messages that you must acknowledge in order to continue.

   Note: If you don't see a prompt, a database upgrade is not required. However, the version number that you see might still change.
6. Ask the database administrator to change the upgraded database to multi-user mode, and allow the users back in to the database.

When the upgrade is complete, you can only roll it back by reverting to your backups of the connection files and SQL Server databases.

© N. Harris Computer Corporation

Additional actions for users of iBase Scheduler

If you're planning to perform a staggered upgrade, enable iBase Scheduler to perform its imports and exports by either starting the Scheduler service or reactivating the Scheduler triggers, using the iBase Scheduler Configuration tool.

Note: If multiple iBase databases are using the Scheduler, an alternative to stopping the service is to deactivate any Scheduler triggers that operate over the databases you are updating. This will ensure that scheduled imports for the other databases continue unaffected.

Additional actions for replicated databases

If you're planning to perform a staggered upgrade, ask your SQL Server database administrator to reconfigure replication between the servers before allowing any users to access the database again.

Note: When making any changes to the database, you should ensure that database replication has been stopped by the SQL Server DBA before the changes are carried out. Replication of the audit log is optional, but if you have chosen to replicate it, the SQL Server DBA must stop replication on this database too.

Client software upgrade

Depending on the volume and location of end users, you can update the iBase client software on a machine-by-machine basis, or deploy it across a network using Windows Installer and enterprise systems management software.

Whichever method you use to deploy the client software, the following actions need to be performed.

Backing up client data

There are a number of files that are deployed with the iBase client and may have been modified by end users. When the upgrade occurs, these will be overwritten with the later default versions. A full list of the files that may be affected can be found in Client file types. If these files have been modified, you must back them up before the upgrade takes place.

Upgrading the client software

iBase clients are only compatible with the database that they were released with or later versions. When the backup of client data is complete, you can begin the update of client software.

Note:

- You must upgrade your iBase databases before you upgrade iBase clients
- iBase Designer does not support backward compatibility. You should only use it with databases that have been upgraded. Users of iBase Designer should upgrade their installed software at the same time as the databases are upgraded.

Merging the client data changes

When the upgrade is complete, you will need to restore any client data that was included in the backup process to the client system.

You can copy the following files directly from the backup:

- `WSExclude.txt`
- `FTExclude.txt`

© N. Harris Computer Corporation

- `i2.wor`
- `i2.apr`

You must merge the following files from the backup with the new versions of each file:

- `commandgroups.mdb`
- `iconlist.txt`
- `i2.mxd`

For each of the above files:

1. Open the new version.
2. Merge in your changes.
3. Save the file in the installed location.

Upgrading the iBase Scheduler

In order to allow the scheduler to interact with clients that have been upgraded, you must also update iBase Scheduler. This method will retain all scheduler history in the Scheduler database:

1. Stop the Scheduler service.
2. Upgrade iBase on the machine running the Scheduler service.
3. Restart the Scheduler service.

Troubleshooting upgrade

The upgrade log provides details of the steps carried out in the upgrade and any issues that were encountered.

You can review the log to check, for example, if a large database is taking longer than expected, or if the upgrade has stopped for some reason. Any issues should be resolved before retrying the upgrade.

When an upgrade completes, your upgrade log is stored in the same location as the iBase database connection file (.idb). It contains information about the steps that were carried out, and any issues that were found in the upgrade process.

Reviewing the upgrade log

Upgrades to the database in iBase Designer create a log file in the same folder as the database .idb file. Log files are named in the format `iBase_upgrade_\<timestamp>.log`. Each time you upgrade the same database to a different version of iBase, a timestamped log file is created.

It might not be possible to access the log file when the upgrade is in progress. This can indicate that more time is required for the upgrade to complete. When you are able to access the file, look for entries that refer to `Link End Update`. Entries should have a start and corresponding end, for example:

```
Start Link End Update 1
End Link End Update 1
```

If an entry is missing the `End Link End Update`, this could mean that the upgrade is attempting to complete, or that it was interrupted. Specific errors are listed in the log file under the `Error during upgrade` heading.

When the upgrade is complete, the log file ends with the entry:

```
Got to end of Update about to update version numbers
```

If you are unable to interpret the contents of the upgrade log, you should contact i2 Support.

© N. Harris Computer Corporation

Rebuilding the Full Text Search Index

After an iBase database upgrade, iBase users may see the following error message when they try to rebuild the Full Text Search index:

```
File 'sysft_i2Catalog' cannot be reused until after the next BACKUP LOG
 operation.
```

To solve this problem:

1. Launch SQL Server Management Studio.
2. Right-click the iBase database, and select **Tasks** > **Backup** from the context menu.
3. For Backup type, select Transaction log.
4. Perform the backup.

After performing the backup, you can try again to rebuild the Full Text Search index.

Rolling back the upgrade

If the upgrade fails, to roll back the process:

1. Restore all SQL Server iBase databases, iBase audit log databases, and iBase security file databases overwriting the partially upgraded databases.
2. Restore all .ids, .idb, .idl and .idx files.

Locating connection files

The following list gives some reasons why you might not be able to locate connection files for all the databases on an SQL Server instance:

- The MRU list on a machine has been cleared, so the database was not reported.
- Searching the entire network for iBase databases is not feasible.
- A machine was overlooked during the machine identification phase of the upgrade.
- A machine that was used for iBase has been retired due to age or failure.
- The database still resides on a SQL Server machine even though the database connection file (.idb) is no longer in use (and may have been deleted).
- The original network share location for connecting files has been dropped or changed.

You can obtain the iBase version and location of the connection file for the iBase security file or database by examining the contents of the _Configuration table. (An iBase log database does not contain this table because all its details can be obtained from its associated database connection file.)

The _Configuration table has two columns: Item and Data. Item contains the name of the entry, while Data contains the value for the entry. The following table details the value of the Item column for the relevant entries:

	Security file	**Database**
Security file location	`SQLServer:ConnectionFile`	`SYS:SecurityFile`
Database location	Not applicable	`SQLServer:ConnectionFile`
Version	`SYS:Version`	`SYS:Version`

© N. Harris Computer Corporation

Download PDFs

iBase documentation is also available in PDF documents.

- i2 iBase

© N. Harris Computer Corporation