Easysoft® Data Access
Easysoft Data Access for ISAM

Installation Guide and User Manual

Version 13.

This manual documents version 1.3.n of Easysoft Data Access for ISAM.

Publisher: Easysoft Limited

Thorp Arch Grange

Thorp Arch

Wetherby

LS23 7BA

United Kingdom

CONTENTS

This page left blank intentionally

LIST OF FIGURES

PREFACE

About this manual

This manual is intended for use by anyone who wants to access ISAM application data, stored on a Windows or Unix machine, from an ODBC-compliant application.

Chapter Guide

- **Intended Audience**

- **Displaying the Manual**

- **Notational Conventions**

- **Typographical Conventions**

- **Contents**

- **Trademarks**

Intended Audience

The sections written for the Microsoft Windows platforms require some familiarity with the use of buttons, menus, icons and text boxes. If you have any experience of Apple Macintosh computers, Microsoft Windows or the X Window System, you will have no difficulty with these sections.

The Unix-based sections require that you are experienced at using a Unix shell, and can perform basic functions like editing a file. More complex activities are detailed more clearly and do not require any knowledge of specialist Unix shells.

Displaying the Manual

This manual is available in the following formats:

- Portable Document Format (PDF), which can be displayed and printed using the Acrobat Reader, available free from Adobe at **http://www.adobe.com**.

- HTML (the format Easysoft recommend for viewing onscreen).

Notational Conventions

Across the range of Easysoft manuals you will encounter passages that are emphasized with a box and a label.

A *note box* provides additional information that may further your understanding of a particular procedure or piece of information relating to a particular section of this manual:

NB	Note boxes often highlight information that you may need to be aware of when using a particular feature.

A *reference box* refers to resources external to the manual, such as a useful website or suggested reading:

REF	For more manuals that use this convention, see the rest of the Easysoft documentation.

A *platform note* provides platform-specific information for a particular procedure step:

Linux	In Linux you must log on as the `root` user in order to make many important changes.

A *caution box* is used to provide important information that you should check and understand, prior to starting a particular procedure or reading a particular section of this manual:

Caution!	Be sure to pay attention to these paragraphs because Caution boxes are important!

Information has also been grouped within some chapters into two broad classes of operating system, Windows and Unix, for which side tabs are used to help you turn to the section relevant to you.

Typographical Conventions

To avoid ambiguity, typographic effects have been applied to certain types of reference:

- User interface components such as icon names, menu names, buttons and selections are presented in bold, for example:

  Click **Next** to continue.

  Where there is a chain of submenus, the following convention is used:

  Choose **Start > Programs > Command Prompt**.

- Commands to be typed are presented using a `monotype` font, for example:

  At the command prompt type `admin`.

- Keyboard Commands

  It is assumed that all typed commands will be committed by pressing the *<Enter>* key, and as such this will not normally be indicated in this manual. Other key presses are italicized and enclosed by angle brackets, for example:

  Press *<F1>* for help.

- File listings and system names (such as file names, directories and database fields) are presented using the `monotype` plain text style.

Contents

- **Introduction**

  Introduces Easysoft Data Access for ISAM.

- **Installation**

  Explains the procedure for installing Easysoft Data Access for ISAM on Windows and Unix.

- **Configuration**

  Explains how to query and set up your own data sources.

- **Administration**

  Describes the administration of your data source, creating and granting user permissions via the Web Administrator.

- **Client Setup**

  Explains how to set up client machines to connect to ISAM data on the server.

- **Demonstration**

  Demonstrates how to connect to local ISAM application data from an ODBC-compliant application.

- Appendices

  Comprising a Technical Reference and a Glossary.

Trademarks

Throughout this manual, *Windows* refers generically to Microsoft Windows 7, 8, 2012 R2, 10, 2016, 2019 or 2022, which are trademarks of the Microsoft Corporation. The X Window system is specifically excluded from this and is referred to as *The X Window System* or just *X*.

Note also that although the name UNIX is a registered trademark of The Open Group, the term has come to encompass a whole range of UNIX-like operating systems, including the free, public Linux and even the proprietary Solaris. Easysoft use Unix (note the case) as a general term covering the wide range of Open and proprietary operating systems commonly understood to be Unix 'flavors'.

Easysoft and Easysoft Data Access are trademarks of Easysoft Limited.

INTRODUCTION

1

Introducing Easysoft Data Access for ISAM

This section explains the individual components that make up Easysoft Data Access for ISAM, what it does and how it works.

Chapter Guide

- **Easysoft Data Access for ISAM components**
- **Easysoft Data Access for ISAM features**
- **How Easysoft Data Access for ISAM works**
- **Overview of the installation and setup procedure**

Easysoft Data Access for ISAM components

Easysoft Data Access for ISAM is a complete solution, built from products within the Easysoft Data Access Range.

On its own, Easysoft Data Access for ISAM enables ISAM data to be accessed from an ODBC-compliant application *on the same machine*.

However, Easysoft Data Access for ISAM can also be used in conjunction with other Easysoft Data Access products to enable cross-platform data access:

ODBC ACCESS
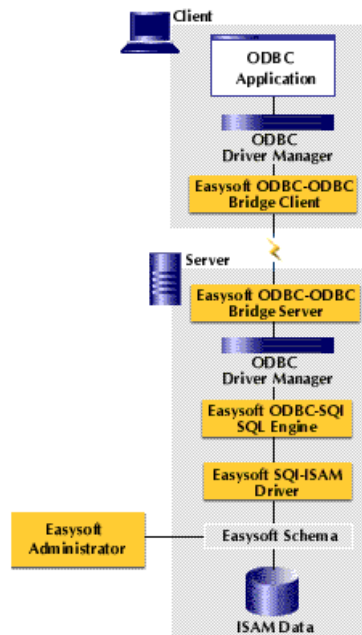
Figure 1: Cross-platform ODBC access to remote ISAM files

- The Easysoft ODBC-ODBC Bridge allows ODBC access to the Easysoft ODBC-SQI SQL Engine from remote client applications.

  For example, the Easysoft ODBC-ODBC Bridge allows access to ISAM data stored on a Unix machine from Microsoft Access running on a networked Windows PC.

- The Easysoft ODBC-SQI SQL Engine appears to ODBC enabled applications as an ODBC 3.5 driver providing access to all the expected API calls and features, reducing complex queries to a common low level API, the Simple Query Interface.

- The Easysoft SQI-ISAM Driver provides the interface between the data files and the Easysoft ODBC-SQI SQL Engine, to which it conforms to the Simple Query Interface.

JDBC ACCESS

- The Easysoft JDBC-ODBC Bridge enhances Easysoft Data Access for ISAM by replacing the Easysoft ODBC-ODBC Bridge, allowing JDBC access to the Easysoft ODBC-SQI SQL Engine from Java client applets or applications.

  For example, the Easysoft JDBC-ODBC Bridge allows access to remote ISAM data on a networked Windows NT machine from a Java Applet running on a separate machine.
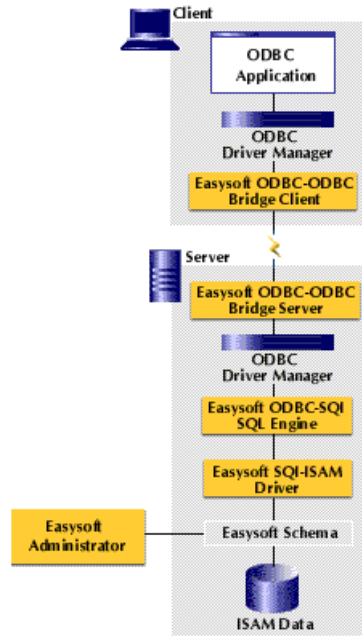
Figure 2: Cross-platform JDBC access to remote ISAM files

Easysoft Data Access for ISAM features

Easysoft Data Access for ISAM is an ODBC 3.5 driver which enables direct access to ISAM data using ODBC-compliant applications such as Microsoft Access.

This means that data can be viewed, added, modified and deleted by using a preferred ODBC-compliant application rather than the interface provided by an ISAM application.

For example, Easysoft Data Access for ISAM makes it possible to load data into Microsoft Excel for reporting or updating purposes.

Easysoft Data Access for ISAM provides:

*   direct support of a schema which is defined via a simple Windows-based graphical Administrator (for further details, see the separate Easysoft Administrator manual).

*   easy creation of users and their access rights to restrict access to the ISAM data via either the Easysoft Administrator or the in-built Web Administrator (see **"Using the Easysoft Web Administrator" on page 79**).

*   access to all the tables within an ISAM application and full support for its data types.

*   conformance to ODBC 3.5.

*   support for the ODBC minimum SQL grammar with the majority of SQL92 extensions.

How Easysoft Data Access for ISAM works

Each ISAM application requires a schema to define the structure of the target database.

Each function selected in an ODBC-compliant application which involves accessing ISAM data passes from the application into the ODBC Driver Manager, which then loads Easysoft Data Access for ISAM and passes the SQL query to it.

The schema is accessed to determine the structure of the data before the SQL statement is converted into commands that can query the data and return the results back to the application:
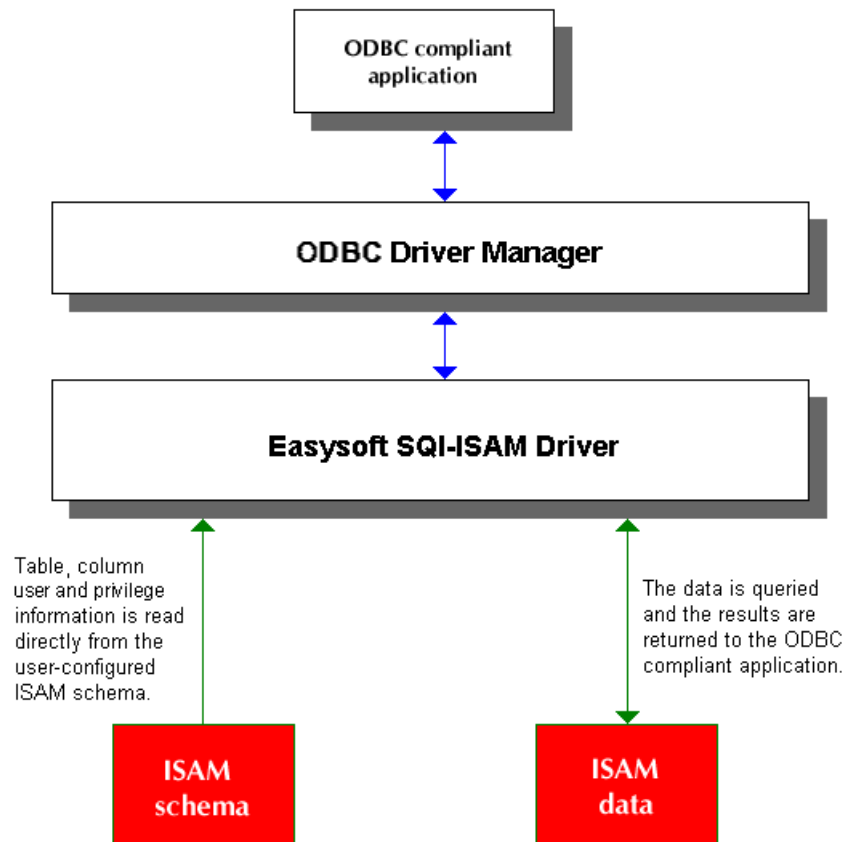
Figure 3: The Easysoft Data Access for ISAM single-platform solution

With a cross-platform solution, the Easysoft ODBC-ODBC Bridge (by transfering ODBC API calls to the server) or Easysoft JDBC-ODBC Bridge (by translating JDBC methods into ODBC API calls on the server) can be used to transfer the SQL query from a client platform onto a different server platform running Easysoft Data Access for ISAM, where the schema is accessed and the results returned to the application:
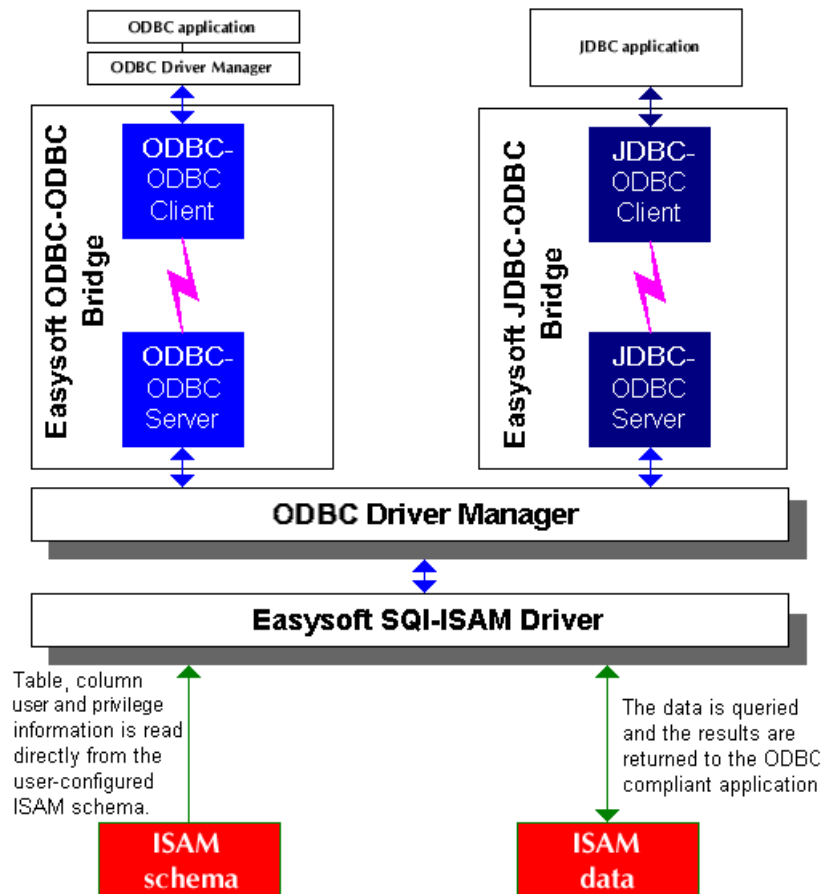
Figure 4: The Easysoft Data Access for ISAM cross-platform solution

In both the local and cross-platform solutions, a driver manager sits between the ODBC or JDBC application and Easysoft Data Access for ISAM.

A driver manager enables an application to use multiple ODBC drivers simultaneously without knowing what those drivers are and is responsible for loading the appropriate driver at runtime:

- On Windows machines, the Microsoft ODBC Driver Manager is provided by default.

- On Unix machines, the open source unixODBC Driver Manager is provided and can be set up and configured during the installation routine (see **"Installing on Unix" on page 47**).

Visit the Easysoft web site at **http://www.easysoft.com** for more information about the Easysoft ODBC-ODBC Bridge (**http://www.easysoft.com/products/2002/main.phtml**) or the Easysoft JDBC-ODBC Bridge (**http://www.easysoft.com/products/2003/main.phtml**).

Find out more information about unixODBC from the project web site at **http://www.unixodbc.org**.

Overview of the installation and setup procedure

To install and configure Easysoft Data Access for ISAM:

1. Install Easysoft Data Access for ISAM on the server (and optionally the Easysoft ODBC-ODBC Bridge server or Easysoft JDBC-ODBC Bridge server), which also optionally creates a data source on the server to point to your ISAM data (see **"What to install" on page 27**).

2. License Easysoft Data Access for ISAM, which also automatically licenses the Easysoft ODBC-ODBC Bridge (see **"Licensing on Windows" on page 38** or **"Licensing on Unix" on page 51**).

3. Enable users and grant them access to your ISAM data (see **"Administration" on page 79**).

4. Optionally, install the Easysoft ODBC-ODBC Bridge client on the machine of each ISAM user (see **"Installing the Easysoft ODBC-ODBC Bridge client" on page 90**).

5. Optionally, create a data source on each of your client machines to connect to your ISAM data via the Easysoft ODBC-ODBC Bridge (see **"Setting up a data source on your Windows client" on page 91**).

The installation of the Easysoft ODBC-ODBC Bridge and the Easysoft JDBC-ODBC Bridge are explained briefly in this manual, but for full details please refer to the product-specific documentation.

This page left blank intentionally
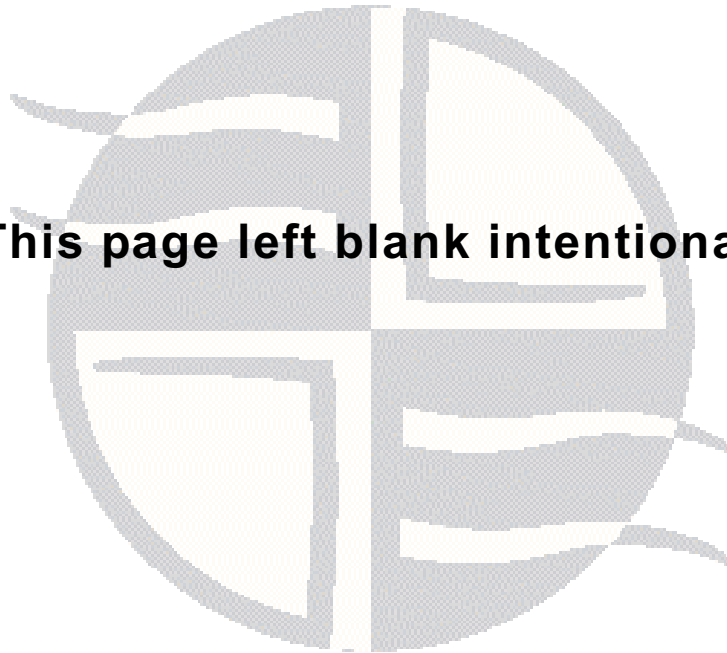
INSTALLATION

Installing Easysoft Data Access for ISAM

This section explains how to install, license and remove Easysoft Data Access for ISAM on supported Windows and Unix platforms.

The Windows installation can be carried out by anyone with local administrator privileges for the target machine.

The Unix installation assumes you are, or have available for consultation, a system administrator.

Chapter Guide

- **Obtaining Easysoft Data Access for ISAM**

- **What to install**

- **Installing on Windows**

- **Uninstalling on Windows**

- **Installing on Unix**

- **Uninstalling on Unix**

Obtaining Easysoft Data Access for ISAM

There are three ways to obtain Easysoft Data Access for ISAM:

• The Easysoft web site is available 24 hours a day at **http://www.easysoft.com** for downloads of definitive releases and documentation.

Select **Download** from the Easysoft Data Access for ISAM section of the website and then choose the platform release that you require.

First time visitors must complete the new user form and click **Register**. Note that your personal Internet options may require you to login and click **Continue** if you have previously registered.

• The Easysoft FTP server is available 24 hours a day at **ftp://ftp.easysoft.com**, containing free patches, upgrades, documentation and beta releases of Easysoft products, as well as definitive releases.

Change to the `pub/eda-isam` directory and then choose the platform release that you require.

• You can order Easysoft software on CD by email, telephone or post (see **Contact Details**).

What to install

The selection of components that you require to download in order to configure Easysoft Data Access for ISAM varies depending on the platforms on which you wish to run.

All installations must download the Easysoft Data Access for ISAM software itself.

The name of the Easysoft Data Access for ISAM install file varies from platform to platform, but is of the form:

- `eda-isam-x_y_z-platform.exe` (Windows)

– OR –

- `eda-isam-x.y.z-platform.tar` (Unix)

where "x" is the major version number, "y" is the minor version number and "z" is the build index, which is incremented when minor changes are made.

"$platform$" will vary depending on the operating system distribution you require and you may come across files of the form:

- `eda-isam-x.y.z-platform-variation.tar`

within specific Unix platforms, where "$platform\text{-}variation$" refers to alternative versions available for a single platform.

NB	Select the highest release available for your platform within your licensed major version number (installing software of a different major version number requires a new Easysoft license).

Unix filenames may also be suffixed with `.gz` for a "`gzipped`" archive, `.bz2` for a "`bzip2ed`" archive, or `.Z` for a "`compressed`" archive.

NB	If you download a Unix file using Windows, the browser may corrupt the filename. For example, if you download a `.gz` file and Windows corrupts the filename, it may not be obvious that the file is "`gzipped`". Use "`file` *filename*" to find out the file type of the downloaded file.

CROSS-PLATFORM DATA ACCESS

If you intend to implement cross-platform data access, you also need to install either the Easysoft ODBC-ODBC Bridge (see **http://www.easysoft.com/products/2002/main.phtml**) for remote ODBC access or the Easysoft JDBC-ODBC Bridge (see **http://www.easysoft.com/products/2003/main.phtml**) for remote JDBC access from Java applications.

The Easysoft ODBC-ODBC Bridge consists of separate client and server components and the Easysoft JDBC-ODBC Bridge consists of a single server component.

NB	You cannot 'mix and match' server and client components of the Easysoft ODBC-ODBC Bridge and Easysoft JDBC-ODBC Bridge.

The following components are required for remote ODBC access to ISAM data:

• Easysoft Data Access for ISAM on the server platform

• the Easysoft ODBC-ODBC Bridge server component on the server platform

- the Easysoft ODBC-ODBC Bridge client component on the client platform

NB	The first two digits of the version number must match when the Easysoft ODBC-ODBC Bridge client and server components are installed (see the Easysoft ODBC-ODBC Bridge manual for more information). This does not apply to the Easysoft JDBC-ODBC Bridge, where no specific client installation is required.

The following components are required for remote JDBC access to ISAM data:

- Easysoft Data Access for ISAM on the server platform

- the Easysoft JDBC-ODBC Bridge on the server platform

Win 9x	If you are using Windows 9x with either the Easysoft ODBC-ODBC Bridge or the Easysoft JDBC-ODBC Bridge you will need Winsock2, which can be downloaded from **http://www.microsoft.com/windows95/downloads/contents/wu admintools/s_wunetworkingtools/w95sockets2/**.

Cross-platform data access software can be obtained as follows:

1. From the Easysoft Web site at **http://www.easysoft.com**:

For the Easysoft ODBC-ODBC Bridge:

- Obtain your required platform version for both client and server from the **Multiple Platforms** option on the Easysoft Data Access for ISAM **Download** page.

- Install the **Client Download** onto your client machine (on both Unix and Windows, you need to accept the **Install ODBC-ODBC Bridge Client** option and reject the **Install ODBC-ODBC Bridge Server** option).

  This provides the Easysoft ODBC-ODBC Bridge Client.

- Install the **Server Download** on your server machine:

  On Unix you need to accept both the **remote ODBC access** and **Install ODBC-ODBC Bridge Server** options.

  On Windows you also need to download the Easysoft ODBC-ODBC Bridge via the **Multiple Platforms** option, selecting your required server platform version, and install the **Server Download** on your server machine.

  This provides the Easysoft ODBC-ODBC Bridge Server, the Easysoft ODBC-SQI SQL Engine and the Easysoft SQI-ISAM Driver.

For the Easysoft JDBC-ODBC Bridge:

- Obtain your required platform version from the **Single Platform** option on the Easysoft Data Access for ISAM **Download** page and install it on your server machine (on Unix you need to reject the **Install OOB Server** option).

  This provides the Easysoft ODBC-SQI SQL Engine and the Easysoft SQI-ISAM Driver.

- Obtain your required platform version from the Easysoft JDBC-ODBC Bridge **Download** page and install it on your server machine.

  This provides the Easysoft JDBC-ODBC Bridge.

2. From the Easysoft FTP site:

On Windows:

- for the Easysoft ODBC-ODBC Bridge both client and server components are contained in the same Easysoft ODBC-ODBC Bridge executable file held in the **ftp://ftp.easysoft.com/pub/eda-isam/** directory.

- the Easysoft JDBC-ODBC Bridge server component is contained in the Easysoft JDBC-ODBC Bridge executable file held in the **ftp://ftp.easysoft.com/pub/jdbc-odbc-bridge/** directory. There is no client component to install.

On Unix:

- the Easysoft ODBC-ODBC Bridge server component is bundled within the Easysoft Data Access for ISAM archive file held in the **ftp://ftp.easysoft.com/pub/eda-isam/** directory.

- the Easysoft ODBC-ODBC Bridge client component is contained in the Easysoft ODBC-ODBC Bridge archive file held in the **ftp://ftp.easysoft.com/pub/odbc-odbc-bridge/** directory.

- the Easysoft JDBC-ODBC Bridge server component is contained in the Easysoft JDBC-ODBC Bridge archive file held in the **ftp://ftp.easysoft.com/pub/jdbc-odbc-bridge/** directory.

NB	Both client and server components are held in the same executable installation file for the Easysoft ODBC-ODBC Bridge. There is no specific client installation for the Easysoft JDBC-ODBC Bridge.

You can now download a file and begin the installation process.

Caution!

As long as you stop all running applications (on Windows), or any software either from Easysoft or using Easysoft drivers (on Unix), it is safe to reinstall or upgrade the Easysoft Data Access for ISAM without uninstalling.

If you do uninstall, you should first back up any configuration data that you still need, as uninstalling some Easysoft products will result in this information being deleted (license details remain in place).

Refer to the section relevant to your platform to continue:

Installing on Windows

1. Execute the file distribution that you downloaded in **"Obtaining Easysoft Data Access for ISAM" on page 26**.

Caution! | Please shut down other Windows programs before installing. In particular, Microsoft Outlook can cause the installation routine to pause for several minutes when you start it.

There will be a short delay while setup prepares the wizard to guide you through the rest of the install procedure before the **Welcome** dialog box is displayed:

Figure 5: The Welcome dialog box

2. Click **Next** to continue.

The **Software License Agreement** dialog box then displays
Easysoft End User licensing details:
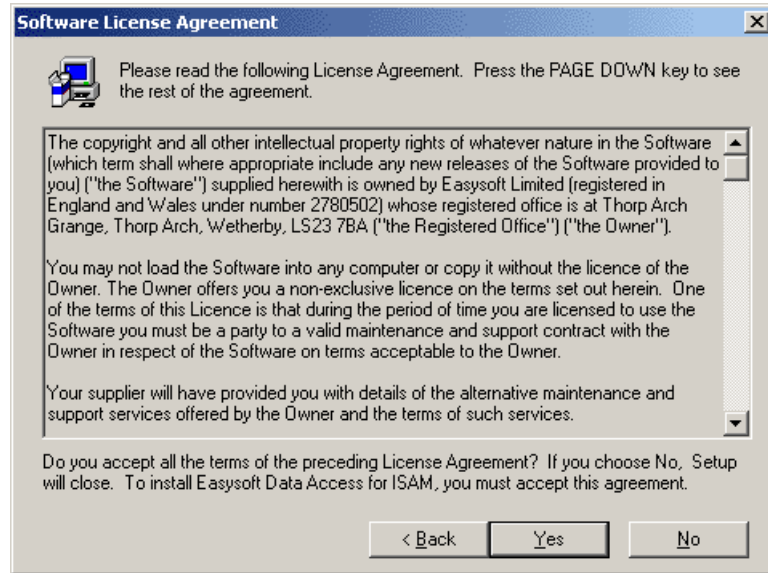
Figure 6: The Software License Agreement dialog box

You are required to accept the terms of the License Agreement
before continuing.

3. If you do not agree to the License Agreement, click **No** to exit the
installation.

– OR –

Click **Yes** to accept the License Agreement and continue with the
installation.

The **Information** dialog box is displayed:
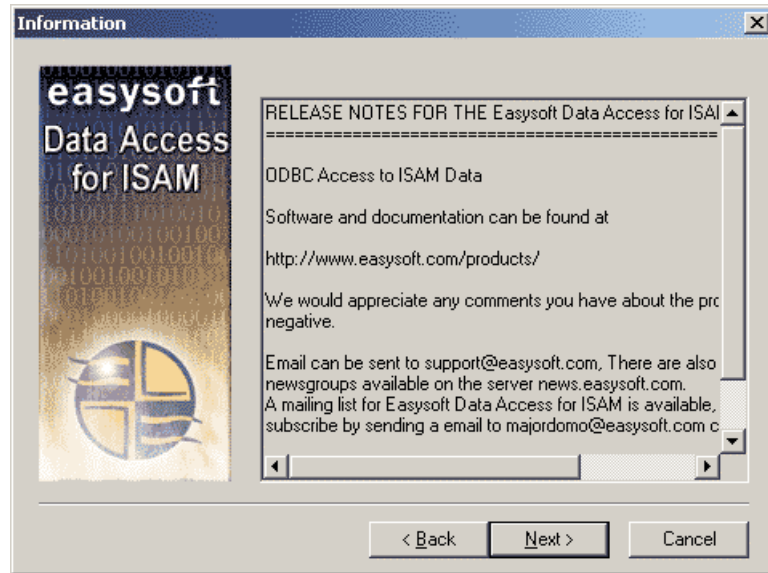
Figure 7: The Information dialog box

4. Click **Next** to continue.

The **User Information** dialog box is displayed:
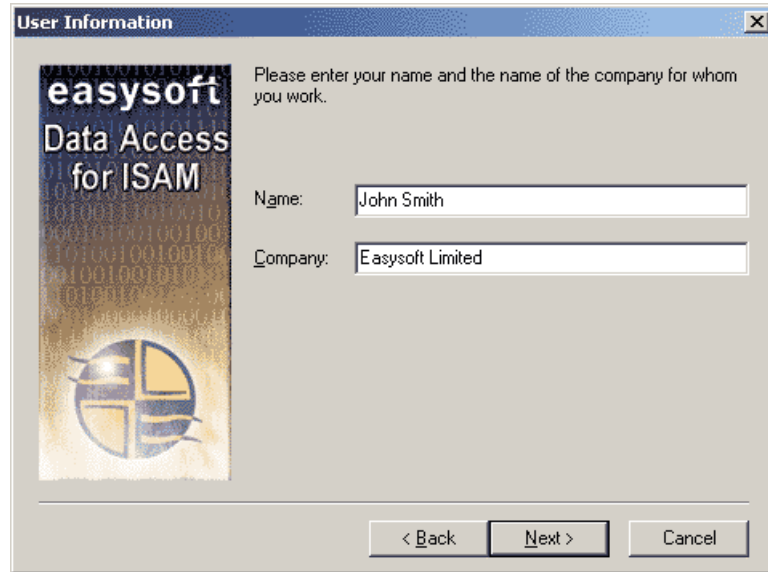
Figure 8: The User Information dialog box

5. Enter your name and company name, then click **Next** to continue.

NB	The name and company that you enter here will become the defaults in the License Manager later on.

The **Choose Destination Location** dialog box is displayed:
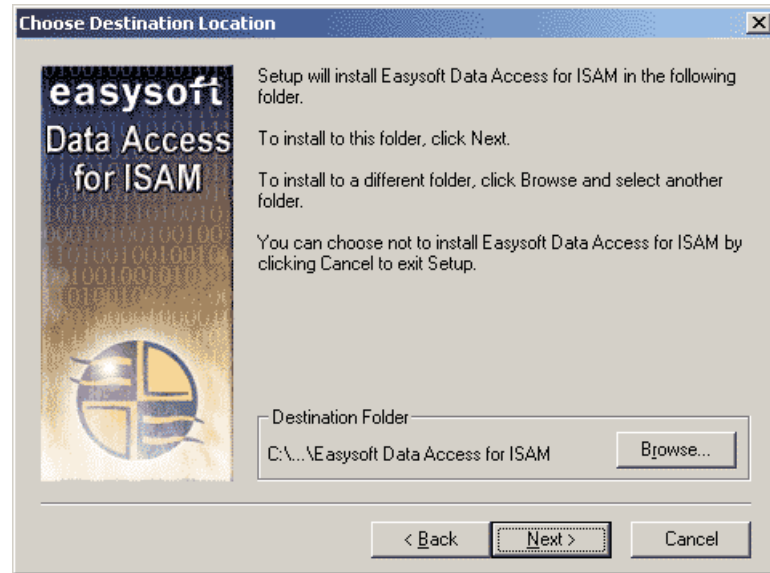
Figure 9: The Choose Destination Location dialog box

Choose the directory in which you want to install Easysoft Data Access for ISAM.

6.   To accept the default, click **Next**.

– OR –

To choose an alternative directory, click **Browse** to select the path you want, then click **Next** to continue.

There is now a short wait while the relevant Easysoft Data Access for ISAM components are copied and configured.

The **Easysoft Data Access for ISAM** dialog box is displayed, pre-configured to set up a data source connecting to a sample database:
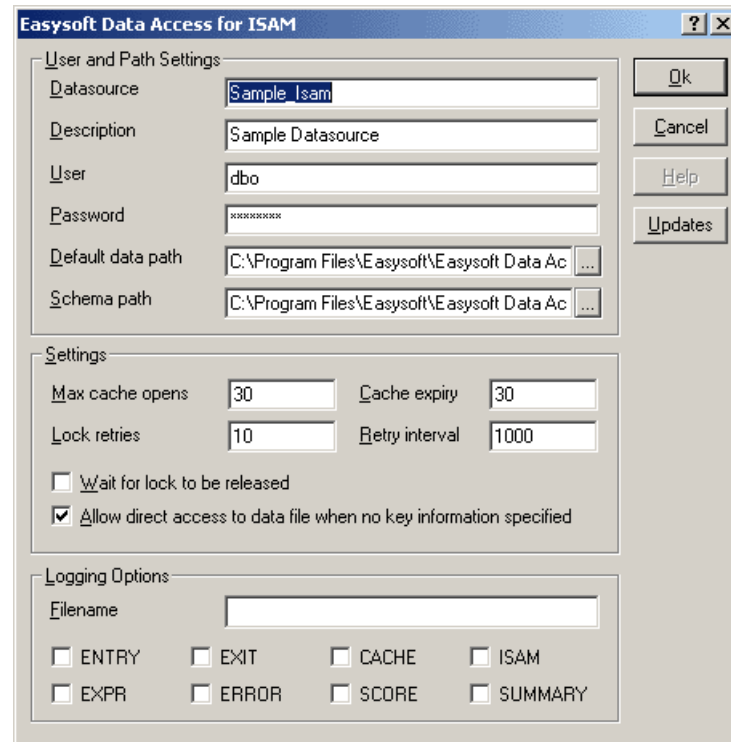
Figure 10: The Easysoft Data Access for ISAM DSN dialog box

"Creating data sources" on page 60 explains how to create a data source to connect to your real ISAM data.

7. Click **OK** to create the sample data source and continue.

LICENSING ON WINDOWS

The install program now starts the Easysoft License Manager (explained fully in the **Licensing Guide**).

The following types of license are available:

• a *trial license* which gives you free and unrestricted use of the product for a limited period (usually 28 days).

• a *purchased license* which gives you unrestricted use of the product. When you purchase the product, an authorization code is emailed to you.

The License Manager dialog box is displayed:

Figure 11: The License Manager window

8. Enter your contact information.

    You MUST enter the **Name**, **E-Mail Address** and **Company** fields.

    The **Telephone** and **Facsimile** fields are important if you require Easysoft to contact you by those methods.

9. Click **Request License**.

    You are asked for a license type:

Figure 12: The License Type dialog box

10. To obtain a trial license, click **Time Limited Trial** and then click **Next**:
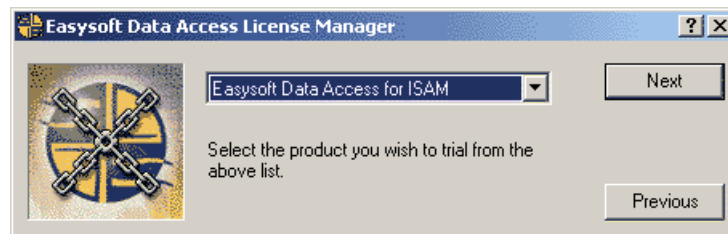
Figure 13: Select the product you are licensing

Select `Easysoft Data Access for ISAM` and click **Next**.

– OR –

If you have purchased the software and received an authorization code, select **Non-expiring License** and click **Next**:
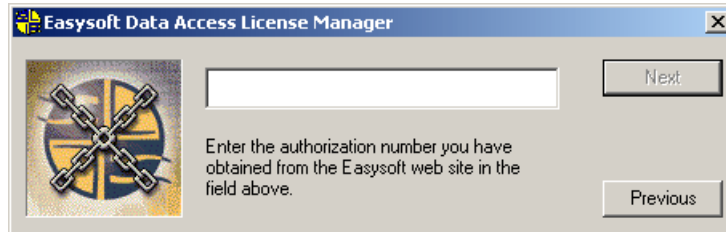
Figure 14: The Authorization Number dialog box

Enter the authorization code and then click **Next**.

11. The License Manager displays a summary of the information entered and a choice of methods with which to apply for a license:

Figure 15: The License Application dialog box

12. Choose **On-line Request** if your machine has a connection to the internet.

   The License Manager then transmits a network packet to the license server at Easysoft. The whole process is automatic and you can proceed to **step 13 on page 43**.

NB	Only your license request identifier and contact details as they are displayed in the License Manager main screen are sent to Easysoft.

   The remaining three options (**E-mail Request**, **Print Request** and **View Request**) are all ways to obtain a license if your machine is offline (i.e. does not have a connection to the internet).

   Each of these methods involves providing Easysoft with information including your machine number (a number unique to your machine) and then waiting to receive your license key.

   Instead of emailing, faxing or telephoning your details to Easysoft, you can enter them directly onto the Easysoft web site and your license key will be emailed to you automatically.

   To use this method, click **View Request** to display your machine number and then run a web browser and log on to the License Generator screen on the Easysoft web site at **http://www.easysoft.com/sales/autolicense.phtml**.

   Choose the type of license you require, enter your machine number and then click **Continue**.

   Your license key will now be emailed to you.

NB	Licensing Easysoft Data Access for ISAM also obtains licenses for the Easysoft ODBC-ODBC Bridge and the Easysoft JDBC-ODBC Bridge.

When you receive the license key, you can activate it either by double-clicking the email attachment or by clicking **Enter License** on the License Manager main screen and pasting the license key into the dialog box.

A message displays the number of licenses that have been added.

NB	The **E-mail Request** option sends the license key to the email address as displayed on the License Manager main screen, not the `from:` address of your email.

For more information about the licensing procedure, refer to the **License Agreement**.

13. Click **Finish** in the License Manager to return to the install program.

The **Setup Complete** dialog box is displayed:
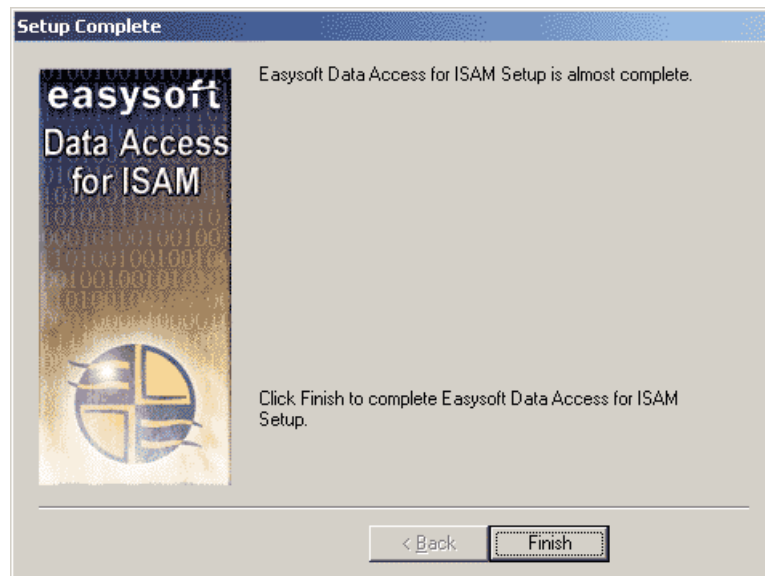
Figure 16: The Setup Complete dialog box

14. Click **Finish** again.

The Windows installation of Easysoft Data Access for ISAM is complete.

The install program will automatically create a SAMPLE_ISAM data source pointing at Easysoft sample ISAM test data (see **step 7 on page 38**).

For details of how to link to this test data to check if the installation of Easysoft Data Access for ISAM has been successful, see **"Querying a data source on Windows" on page 61**.

NB	The sample data source and its accompanying schema data installed into C:\Program Files\Easysoft\Easysoft Data Access for ISAM\Demo. Do not add any important information to the sample database, because it will be overwritten if Easysoft Data Access for ISAM is reinstalled.

The following menu options are installed under **Start > Programs > Easysoft > Easysoft Data Access for ISAM**:

• **Easysoft Web Site**

Links to the Easysoft web site at **http://www.easysoft.com**.

- **Web Administrator**

Runs the Web Administrator (see **"Using the Easysoft Web Administrator" on page 79** for further details).

- **Licensing**

Links to the License Help file and the License Manager.

15. You now need to install the client part of your method of connection if you are linking via the Easysoft ODBC-ODBC Bridge by copying the most recent version available from **ftp://ftp.easysoft.com/pub/eda-isam/** onto each client machine and running the install script.

    You are recommended to choose the **Custom** installation, and select the client module only.

    Please refer to the Easysoft ODBC-ODBC Bridge manual (**ftp://ftp.easysoft.com/pub/docs/oob/edaoob.pdf**) for details.

Uninstalling on Windows

To uninstall Easysoft Data Access for ISAM:

1. Select **Start > Settings > Control Panel** and then double-click the **Add/Remove Programs** icon.

   A list is displayed of applications that can be automatically removed.

2. Select **Easysoft Data Access for ISAM** and click **Add/Remove**.

3. Click **Yes** to confirm that you wish to remove Easysoft Data Access for ISAM and all its components.

   The system begins to remove all the components. If shared components seem not to be required, you will be prompted to decide whether or not to delete them.

> **NB** The Windows install/uninstall procedure incorporates a mechanism in the registry to determine whether or not shared files are still required by other programs. Sometimes this database can become out-of-date, for instance if the user deleted an application directly without using **Add/Remove Programs**, or if the registry was 'repaired' after a system crash.

4. If you feel confident with the registry (i.e. your system has had relatively few programs installed and removed) you should click **Yes** or **Yes to All** to continue.

   – OR –

   If you have any doubts (e.g. uninstall procedures have failed in the past) you should click **No** or **No to All**.

   The uninstall process removes Easysoft Data Access for ISAM components from your system.

> **NB** If files have been created in any of the installation directories then these directories will not be removed. In this case, the uninstall program will issue a warning and you can click **Details** to find out what directories remain.

5. Click **OK** to return to the **Control Panel Install/Uninstall** window.

6. The uninstall process is complete.

> **NB** Any licenses that are obtained for Easysoft Data Access for ISAM or any other Easysoft products are held in the Windows registry and are not removed when a product is uninstalled, so it is not necessary to relicense after a reinstall or upgrade.

Installing on Unix

Although this section covers a range of platforms and the precise output may vary from system to system, the installation process is essentially the same.

INSTALLATION REQUIREMENTS

The Unix installation routine has the following requirements:

- The Bourne shell (or BASH) must either be named (or linked to) `/bin/sh` or the first line of the `install` file updated to the correct location.

- Various common Unix commands such as `grep`, `awk`, `cut`, `ps`, `sed`, `cat`, `wc`, `uname`, `tr` and `find`.

  If any of these commands are missing they can be obtained from the Free Software Foundation (**http://www.fsf.org**).

- Depending on the platform, you will need up to 10Mb of free disk space for the installed programs and up to 10Mb temporary space for the installation files themselves.

PREPARATION

1. Log on to your Unix machine as the `root` user.

2. Download the Easysoft Data Access for ISAM (see **"Obtaining Easysoft Data Access for ISAM" on page 26**).

3. Place the distribution file in a temporary directory on your Unix machine.

EXTRACTING THE INSTALLATION FILES

4. Change to the directory in which the distribution file resides.

5. Extract the installation files from the distribution file:

   If the distribution file has been `gzip`ped (i.e. the filename ends in `.gz`), then use:

   ```
   gunzip eda-isam-x.y.z.platform.tar.gz
   ```

   – OR –

   If the distribution file has been `bzip2`ed (i.e. the filename ends in `.bz2`), then use:

   ```
   bunzip2 eda-isam-x.y.z.platform.tar.bz2
   ```

   – OR –

   If the distribution file has been `compress`ed (i.e. the filename ends in `.Z`), then use:

   ```
   uncompress eda-isam-x.y.z.platform.tar.Z
   ```

6. If the distribution file has not been compacted at all (i.e. the filename ends in `.tar`), then it is ready for extraction:

   ```
   tar -xvf eda-isam-x.y.z.platform.tar
   ```

   The `tar` program creates a directory with the same name as the `tar` file (without the final `.tar`) containing further archives, checksum files, a script called `install` and a text file called `INSTALL`.

It also contains a versioned directory to ensure that any shared components already installed by other Easysoft products are only overwritten if those included in this distribution are newer.

NB	If you do not wish to keep the original downloaded distribution file you can now delete it safely.

7. Change into the newly-created `xml-odbc-x.y.z-platform` directory.

Caution!	Check through the `INSTALL` file before continuing. It gives full installation instructions for the Unix-literate, and if you are confident in the use and administration of your system, you can follow the instructions in the `INSTALL` file instead of working through the remainder of this section.

B EGINNING THE INSTALLATION

8. Type:

```
./install
```

NB	During the installation, you are asked to answer some questions. The default reponse is displayed in square brackets [], which you can press *<Enter>* to accept or you can choose any of the alternative responses shown in round brackets () by typing the required response and then pressing *<Enter>*. Occasionally, the install program pauses to give you time to read the information displayed on screen. Press *<Enter>* to continue when you have read the current screen of information.

9. If you have read and agree to the **Easysoft License Agreement**, type `yes` and then press `<Enter>` to continue.

> **NB** You must type `yes`, not `y`, to continue.

10. Specify the directory into which you want to install Easysoft Data Access for ISAM. If you accept the default base directory, the files are installed into a subdirectory called `/easysoft` within the specified path. For example, if you accept the default path of `/usr/local`, the files are installed into `/usr/local/easysoft`.

    If you specify an alternative directory, the files are installed into that directory but a symbolic link is created from `/usr/local/easysoft` pointing to the install directory. This link is necessary for licensing to work.

11. Depending on the server platform, there may be more than one configuration of Easysoft Data Access for ISAM in the distribution file. If this is the case, you will be asked to choose the configuration that best suits your system.

12. Easysoft Data Access for ISAM requires an ODBC Driver Manager for Unix to be installed and includes one with your software. Details of the ODBC Driver Manager provided by Easysoft are displayed on screen, and if you do not already have one then unixODBC will be installed now. If you already have a version then you will be asked if you wish to overwrite it or to use your existing installed version.

LICENSING ON UNIX

13. You are now asked if you want to obtain a license. Accept the default [y] to start the licensing procedure.

    A menu of options is displayed.

14. Enter the number corresponding to Easysoft Data Access for ISAM.

    By licensing Easysoft Data Access for ISAM, you also obtain a license for the Easysoft ODBC-ODBC Bridge and the Easysoft ODBC-SQI SQL Engine.

    You will then be asked to provide some contact information.

15. Enter your **Name**.

16. Enter your **Company Name**.

17. Enter at least one of **Email**, **Phone** and **Fax** (preferably all three).

18. At the **Ref** prompt, do either of the following:

    • If you want a trial license, leave it blank and press *<Enter>*.

    • If you want a purchased license, enter your authorization code then press *<Enter>*.

19. The License Manager displays a menu of options for acquiring your license.

    If you have an internet connection you should select `1, Automatic`. This is the quickest and easiest method unless your firewall or other network obstacles prevent the message from getting through.

    – OR –

If you do not have an internet connection, or the `Automatic` option fails, select option `2` to write the required information to a text file in the current working directory called `license_request.txt`, which you will then need to view to obtain a license.

20. The License Manager displays the menu again. Select 0 to exit.

> **NB** The View Existing Licenses option will not display any license you obtained during this License Manager session. You must quit the License Manager and allow the install script to finish applying the license.

If you chose the automatic licensing method and licenses are retrieved, the licensing script outputs them to a file called `licenses.out`.

If the install script detects possible new licenses in the file `licenses.out` it asks whether you want to add them to the license database.

21. If you obtained licenses and you want to add them now, enter `y`.

– OR –

If you did not obtain licenses, enter `n`.

– OR –

If you do not see this message, then read on. You will need to complete the offline licensing procedure as explained in **"Completing the offline licensing procedure" on page 55**.

22. After the licensing is complete, the install program will ask you to confirm that you wish to install the empty [EASYSOFT_ISAM] and sample [SAMPLE_ISAM] data sources for Easysoft Data Access for ISAM.

It is recommended that both data sources are installed, especially [SAMPLE_ISAM], which is referred to elsewhere in the Easysoft Data Access for ISAM documentation.

> **NB**
>
> An empty directory is created to hold schema files called `<InstallDir>/easysoft/isam/schema`.
>
> The sample data is installed into `<InstallDir>/easysoft/isam/demo/data`.
>
> Schema files for the sample data source are installed into `<InstallDir>/easysoft/isam/demo/schema`.
>
> Do not add any important information to the sample database, because it will be overwritten if Easysoft Data Access for ISAM is reinstalled.

`odbc.ini` is placed in the `/etc` directory by default and automatically updated with definitions of the [EASYSOFT_ISAM] and [SAMPLE_ISAM] data sources that are added.

23. You will now be asked whether or not to install the Easysoft ODBC-ODBC Bridge.

    Enter `y` if you wish to provide remote ODBC access to your data.

24. Follow the defaults through the rest of the installation.

    At the end of the install, information is displayed which describes the data sources which have been set up and explains how the installation can be verified by using the `demosql` SQL script provided with Easysoft Data Access for ISAM to query the sample data source (see **"Querying a data source on Unix" on page 61**).

    A security notice is then displayed, which the user is recommended to follow by immediately updating the password for the default database owner ("`dbo`") which was created during the install.

If you wish to query the data source now, press *<Enter>* after you have finished to return to the system prompt.

NB	You will not be able to query the sample data source if you have still to complete the offline licensing procedure. Complete the licensing, as explained in **"Completing the offline licensing procedure" on page 55** and then try running demosql.

If you have more than one set of ISAM data, you can create more data sources by manually editing odbc.ini (see **"Creating additional data sources on Unix" on page 70**).

"odbc.ini settings for Unix" on page 72 describes the settings must be specified for a data source before the data can be accessed.

Once you have installed Easysoft Data Access for ISAM you can run the Web Administrator to create users and set up their access rights to restrict access to the ISAM data (see **"Using the Easysoft Web Administrator" on page 79**).

25. You now need to install the client part of your method of connection if you are linking via the Easysoft ODBC-ODBC Bridge by copying the most recent version available from **ftp://ftp.easysoft.com/pub/eda-isam/** onto each client machine and running the install script.

    You are recommended to choose the **Custom** installation and select the client module only.

    Please refer to the Easysoft ODBC-ODBC Bridge manual (**ftp://ftp.easysoft.com/pub/docs/oob/edaoob.pdf**) for further details.

C OMPLETING THE OFFLINE LICENSING PROCEDURE

If at **step 19 on page 51** you chose to write the licensing information to a file, you still need to complete the licensing procedure before Easysoft Data Access for ISAM can be used.

Licensing information is written to the `license_request.txt` file, which contains information including a machine number (a number unique to your machine) which Easysoft require before a license key can be issued.

Licensing information which Easysoft require before a license key can be issued (including a machine number unique to your machine) is written to the `license_request.txt` file.

1. Do any one of the following:

    *   Display the `license_request.txt` file (e.g. using
        `cat license_request.txt`) and note the machine number.
        Now run a web browser and go to
        http://www.easysoft.com/sales/autolicense.phtml. Log in to
        the Easysoft web site. On the License Generator screen, choose
        the type of license you want, then enter your machine number
        and click **Continue**. You can now close the web browser. You
        will shortly receive your license key(s) via email.

    *   Email the file `license_request.txt` to
        autolicense@easysoft.com. Your license key(s) will be emailed
        to you automatically.

    *   Email the file `license_request.txt` to
        license@easysoft.com. A member of the Licensing Department
        will email the license keys(s) to you.

2. When you receive your license key(s), append them to the file
   `/usr/local/easysoft/license/licenses`, removing any `LIC:`
   prefixes.

> **NB** The Easysoft license responder puts an attachment in its outgoing emails that allows Windows users to activate their licenses with a double-click. If you read your email in Windows, this attachment will be visible but it will not work for licensing Unix versions of the software.

The Easysoft Data Access for ISAM is now licensed and you can begin using it.

If you need further information about licensing, please refer to the **Licensing Guide**.

Uninstalling on Unix

To uninstall Easysoft Data Access for ISAM:

• stop the Web Administrator if it is running (see **"Using the Easysoft Web Administrator" on page 79**).

• remove the `<InstallDir>/easysoft/isam` directory.

To stop the Web Administrator:

1. Log onto your Unix machine as `root`.

2. Change into the `<InstallDir>/easysoft/isam` directory.

3. Type:

   `./stop_http.sh`

4. Type:

   `y`

when asked if you want to stop all HTTP administration servers.

Caution! | If you do not stop the Web Administrator and then later perform another installation of Easysoft Data Access for ISAM then the installation may not be successful.

To remove the `<InstallDir>/easysoft/isam` directory:

1. Change into the `<InstallDir>/easysoft/isam` directory.

2. Type

   `rm -rf *`

   to delete all the files and directories underneath the current directory.

3. Type

   `cd ..`

   to move back up one level in the directory structure.

4. Type

   `rmdir isam`

   to remove the `isam` subdirectory.
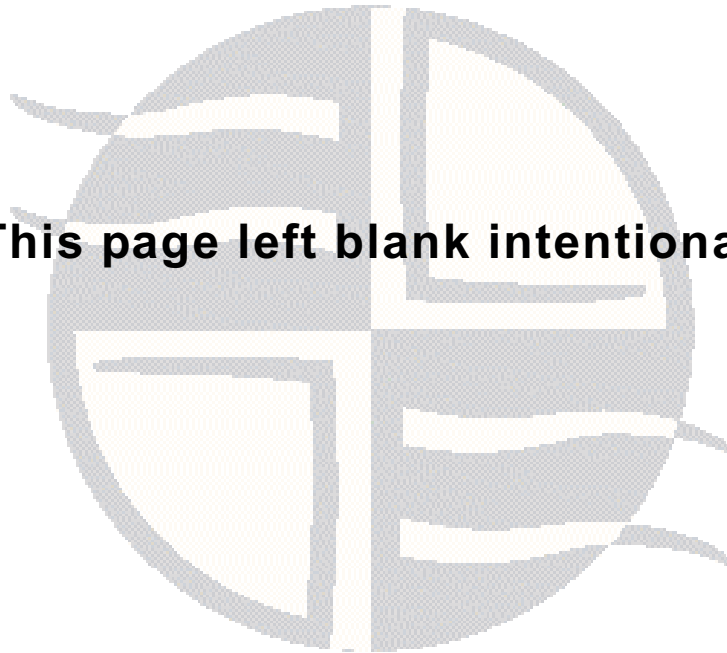
If you have installed the Easysoft ODBC-ODBC Bridge or the Easysoft JDBC-ODBC Bridge and you wish to uninstall one of them, please refer to the manual supplied with that product for details of its uninstall procedure.

Any licenses you obtain for Easysoft Data Access for ISAM and other Easysoft products are stored in the `<InstallDir>/easysoft/license/licenses` file, and unless you deleted this file after uninstalling you will not need to relicense Easysoft Data Access for ISAM when you reinstall or upgrade.

However, for security purposes you may want to make a copy of `<InstallDir>/easysoft/license/licenses` before uninstalling.

This page left blank intentionally

CONFIGURATION

Configuring Easysoft Data Access for ISAM

This section explains how to check that Easysoft Data Access for ISAM is connecting to a sample ISAM database and how to set up additional local data sources on both Windows and Unix.

Chapter Guide

- **Creating data sources**
- **Querying a data source on Windows**
- **Querying a data source on Unix**
- **Creating additional data sources on Windows**
- **Creating additional data sources on Unix**

Creating data sources

A data source called SAMPLE_ISAM is automatically set up to point at an Easysoft sample ISAM database on the server machine when Easysoft Data Access for ISAM is installed.

This is used as an example in **"Querying a data source on Windows" on page 61** and **"Querying a data source on Unix" on page 61** to illustrate how to validate that a connection has been made after the installation process has been completed.

Another data source called EASYSOFT_ISAM is also created, which contains empty data and schema directories which are ready to accept data definitions entered via the Easysoft Administrator.

The EASYSOFT_ISAM data source can be used to point directly at your own ISAM data or new data sources may be created (see **"Creating additional data sources on Windows" on page 64** and **"Creating additional data sources on Unix" on page 70**) to:

- allow access to multiple ISAM files with the same name and layout, but located in different directories

- provide different sets of permissions and privileges for the same database

- allow groups of files from the same database to be configured differently

- allow multiple copies of the same database to be separately configured

- permit an application into which the data source name has been hard-coded to access a database

A correctly configured data source can then be connected to by any ODBC-compliant application (see **"Demonstration" on page 99**.

Querying a data source on Windows

On Windows, a program called `sql.exe` is installed with Easysoft Data Access for ISAM into `<InstallDir>\Easysoft\Easysoft Data Access for ISAM`, which can be run using the required data source as a parameter.

e.g.

```
sql.exe "dsn=sample_isam"
```

which points at the `SAMPLE_ISAM` Easysoft demo database.

When Easysoft Data Access for ISAM is installed, a System Data Source called `SAMPLE_ISAM` is automatically created and the demo database is loaded into `<InstallDir>\Easysoft\Easysoft Data Access for ISAM\demo\data` (for the data) and `<InstallDir>\Easysoft\Easysoft Data Access for ISAM\demo` (for the schema).

Querying a data source on Unix

On Unix you can use one of the two scripts which are installed with Easysoft Data Access for ISAM to query ISAM data sources using SQL statements:

*   `demosql`, for the `SAMPLE_ISAM` Easysoft demo database

– OR –

*   `easysql`, for the `EASYSOFT_ISAM` blank schema database.

Both of these scripts are stored in `<InstallDir>/easysoft/isam/bin` and run an Easysoft utility called `sql`, which in turn executes SQL statements.

These scripts enable the `sql` command to be executed without needing to set `LD_LIBRARY_PATH`, as the script determines the LD_LIBRARY_PATH values from a separate configuration file (`es.ld.so.conf`) and then runs the `sql` command, specifying the data source to query and the user name and password to use to access the data source.

By default, both scripts specify:

- the user name as `dbo` (`uid=dbo`)

- the password as `easysoft` (`pwd=easysoft`)

The `demosql` script specifies the data source name as [`SAMPLE_ISAM`] (`dsn=sample_isam`).

The `easysql` script specifies the data source name as [`EASYSOFT_ISAM`] (`dsn=easysoft_isam`).

User names and passwords can be set up by the database owner (see **"odbc.ini settings for Unix" on page 72**).

If you want to query a data source other than `SAMPLE_ISAM` or `EASYSOFT_ISAM` you need to create a copy of either `demosql` or `easysql` for each new data source, rename it and specify the appropriate data source name in each new script.

You then have a script ready to run for querying each data source against which you can execute a standard SQL command as follows (where the new data source is called "`new_dsn_name`" and the new script is called "`new_sql`"):

```
./sql -d "dsn=new_dsn_name" -f new_sql -o7
```

To query a data source using the `demosql` script:

1. Change into the `<InstallDir>/easysoft/isam/bin` directory.

2. Type:

```
./demosql
```

If a line beginning "`conn =`" is displayed, a connection has been made and the data source can be queried with SQL statements.

For example, to display a list of tables in the data source, type:

```
select * from info_schema.tables;
```

> **NB** SQL lines in `demosql` scripts must end in the ";" character, rather than a line feed.

If an '`invalid user`' or '`invalid password`' error is displayed, check that the script specifies the correct user name and password.

3. Press *<Enter>* to stop typing SQL and return to the system prompt.

Another utility called `isql` is included in the unixODBC open source driver manager, which is installed as part of Easysoft Data Access for ISAM and can be run by going to the `<InstallDir>/easysoft/unixODBC/bin` directory and typing:

```
./isql DSN [UID [PWD]] [options]
```

where

- `DSN` is the data source name

- `UID` is the database user name

- `PWD` is the database password

Type `./isql` only for option details.

> **NB** SQL lines in `isql` scripts do NOT use the ";" character as a line terminator.

Creating additional data sources on Windows

To create a data source connecting to your local System Z data:

1. Select **Start > Settings > Control Panel**, double-click
    Administrative Tools and then **Data Sources (ODBC)**.

9x	Select **Start > Settings > Control Panel** and double-click **ODBC Data Sources (32bit)**.
NT	Select **Start > Settings > Control Panel** and double-click **Data Sources (ODBC)**.

The **ODBC Data Source Administrator** dialog box is displayed:
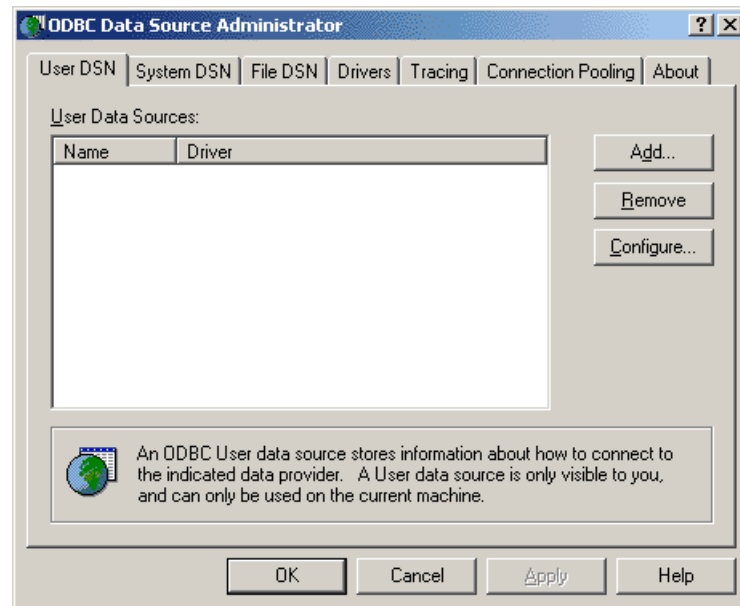
Figure 17: The ODBC Data Source Administrator

2. Select the **User DSN** tab to set up a data source that only you can access.

   – OR –

   Select the **System DSN** tab to create a data source which is available to anyone who logs on to this Windows machine.

3. Click **Add** to add a new DSN.

   The **Create New Data Source** dialog box displays a list of the drivers which are available:
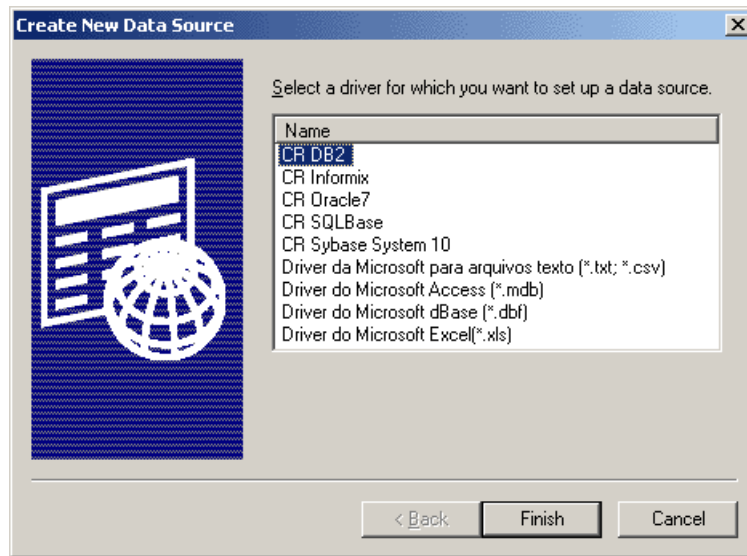
Figure 18: The Create New Data Source dialog box

4. Select `Easysoft Data Access for ISAM` and then click **Finish**.

A dialog box displays the fields required to configure an Easysoft Data Access for ISAM data source:
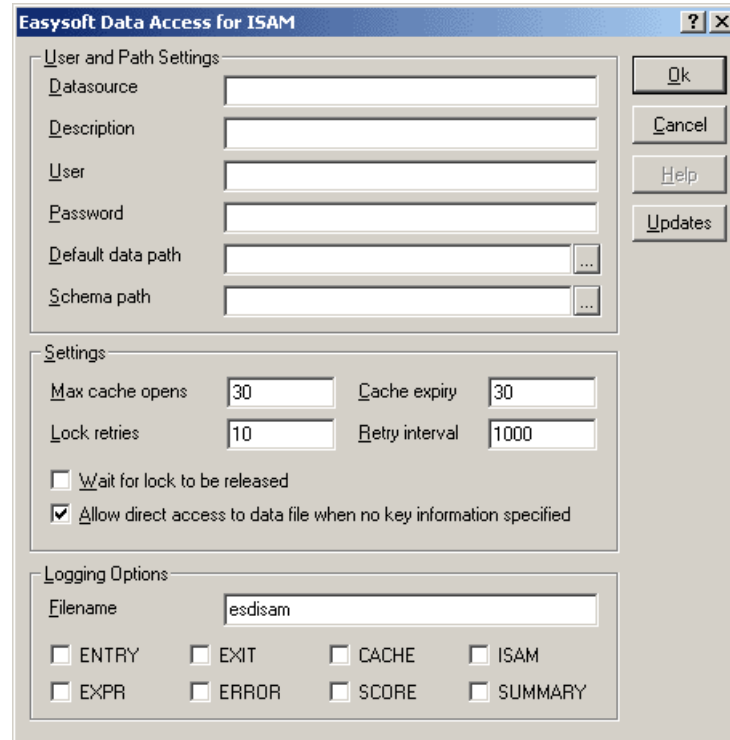
Figure 19: The Easysoft Data Access for ISAM setup dialog box

- **Datasource**

  The name of the User or System data source to be created, as used by the application when calling the `SQLConnect` or `SQLDriverConnect` functions.

  This name cannot be changed once the data source has been created.

- **Description**

  Descriptive text which may be retrieved by certain applications to describe the data source.

- **User**

  The valid ISAM user name as defined in the ISAM schema (if any) with which to connect to the data source.

- **Password**

  The valid ISAM password as defined in the ISAM schema (if any) for the user name entered in the **User** field.

- **Default data path**

  The default path of the data to be accessed with this data source.

- **Schema path**

  The default path of the schema to be accessed with this data source.

- **Max cache opens**

  The maximum number of areas of cache memory that may be opened.

  This field is set to `30` by default.

- **Cache expiry**

  The time (in seconds) after which an area of cache memory will be closed.

  This field is set to `30` by default.

- **Lock retries**

  The number of times that a retry will be attempted if a lock is encountered.

  This field is set to `10` by default.

- **Retry interval**

  The time (in seconds) between each lock retry.

  This field is set to `1000` by default.

- **Wait for lock to be released**

  Select this attribute to wait for a lock from another application to be released before attempting an insert, delete or update.

  If any of these actions is attempted when this option is not selected the user will receive an error message immediately.

- **Allow direct access to data file when no key information specified**

  This attribute is selected by default and optimises the access speed to ISAM data files consisting of fixed length records.

  Note that this option cannot be used if any of the underlying data files is of variable length.

- **Filename**

  The filename prefix to which a log file will be generated.

  This may consist of a directory path only, a directory path and a filename prefix, or be left blank.

The log filename takes this prefix and adds an underscore followed by a unique number (derived from the process ID) and ".`log`".

For example, if you set this field to "`C:\isam`" then the log filename will be of the format `C:\isam_026503.log`.

If no path is specified then the file will be written to the directory where the data source is located.

- **ENTRY**

  Select this attribute to log the entries to all functions and the attributes of all calls in the SQI layer.

- **EXIT**

  Select this attribute to log the exits from all functions.

- **CACHE**

  Select this attribute to log information regarding the file cache.

- **ISAM**

  Select this attribute to log the selection of indexes used when starting a query.

- **EXPR**

  Select this attribute to log any expression checking that the SQI layer performs.

- **ERROR**

  Select this attribute to log any errors reported at the SQI layer.

- **SCORE**

  Select this attribute to log information on the opening of files and how many reads are performed.

- **SUMMARY**

  Select this attribute to log summary information about the SQI function calls.

Caution! Enabling logging will seriously impair performance so remember to disable it once you have finished.

5. Click **OK** to create the data source, or click **Cancel** to abandon it.

   The **Updates** button on the Easysoft Data Access for ISAM DSN dialog box attempts to connect your PC to the Internet and get the latest version number for this software from the Easysoft version server.

   A message will be displayed if a more recent version of the software is available.

   If you have changed your data source configuration, you are advised to click **OK** to save the changes before accepting the new software.

Creating additional data sources on Unix

To configure a data source for your own ISAM data when the data is local (i.e. is on the same Windows machine as the ODBC compliant application that you want to use to connect to the data), you can either:

- Create a system data source (which is available to anyone who logs onto this Unix machine)

– OR –

- Create a user data source (which is only available to the user who is currently logged into this Unix machine)

This is exactly the same mechanism as is used on the ODBC Data Source Administrator on Windows platforms (see **"Creating additional data sources on Windows" on page 64**).

CREATING A SYSTEM DATA SOURCE

1. Copy `<InstallDir>/easysoft/isam/schema/blank.dsn` to either your home directory or a temporary directory.

2. Edit the following settings:

   - replace the data source name (`EASYSOFT_ISAM` by default) with a unique name for your data source.

   - change the `data_path` variable to point to the directory containing your ISAM application data.

3. Log on as the `root` user

4. Type:

   ```
   ODBCSEARCH="ODBC_SYSTEM_DSN"

   export ODBCSEARCH
   ```

5. Run:

   ```
   /usr/local/easysoft/unixODBC/bin/odbcinst -i -s -f filename
   ```

   where `filename` is the new name in **step 2 on page 71**, including the path name.

   This creates a system-wide data source in `/etc/odbc.ini`, where it can be accessed by the unixODBC driver manager.

CREATING A USER DATA SOURCE

1. Copy `<InstallDir>/easysoft/isam/schema/blank.dsn` to either your home directory or a temporary directory.

2. Edit the following settings:

   - replace the data source name (`EASYSOFT_ISAM` by default) with a unique name for your data source.

   - change the `data_path` variable to point to the directory containing your ISAM application data.

3. Run:

   ```
   /usr/local/easysoft/unixODBC/bin/odbcinst -i -s -f filename
   ```

   where `filename` is the new name in **step 2 on page 72**, including the path name.

   This creates a user-specific data source in a file called `.odbc.ini` (residing in the home directory of the user you are currently logged on as), where it can be accessed by the unixODBC driver manager.

NB	Data sources can also be created by entering their attributes directly into either the `odbc.ini` or `.odbc.ini` files (see **"odbc.ini settings for Unix" on page 72** for details of the attributes you need to specify).

ODBC.INI SETTINGS FOR UNIX

If you are running Easysoft Data Access for ISAM on a Unix system, you will probably set up data sources on the Unix machine by editing `blank.dsn` (see **"Creating additional data sources on Unix" on page 70**).

However, the following data source attributes can also be specified directly in `/etc/odbc.ini` (system data sources) or the relevant `.odbc.ini` file (user data sources):

Setting	Description/example
[data source name]	e.g. [ISAM].
driver = EASYSOFT_ISAM	
sort_path = `/tmp`	Directory where temporary sort data is saved if sorting requires more than `sort_mem_size` memory
sort_mem_size = 256	Amount of memory (in Kb) allocated for sorts before the results are saved to disk.
rs_mem_size = 256	Number of rows that will cache into memory before being stored on disk.
rs_path = /tmp	Directory where result set rows are temporarily stored when `rs_mem_size` is exceeded.
blob_path = /tmp	Directory where `blob` data types in the result set are temporarily stored.
sqicount = 1	Number of SQIs used by the data source.
target_string1 = Data Access for ISAM	Connection string for SQI.
target_driver1 = `/usr/local/easysoft/ isam/lib/libesdisam_ sqi.so`	Required SQI driver.
dtcount = 1	Number of data type libraries required for ISAM.
dtlibrary1 = `/usr/local/easysoft/ isam/lib/libesdisam_ dt.so`	Data type library for data type conversions.

Setting	Description/example
data_path = path of the ISAM application data	e.g. `/usr/local/easysoft/isam/` `demo/data/`.
schema_path = path of the ISAM application schema	Directory where the data source schema files (views, users, passwords and privileges) are stored (specify a different schema path for each data source to allow unique users, privileges etc. to be configured e.g. `/usr/local/easysoft/isam/` `demo/schema/`).
isamlibrary = path of the ISAM library	e.g. `/usr/local/easysoft/isam/` `lib/libesisam.so`.
cache_expiry = 2	How long files are cached open for (default is 2).

Setting	Description/example
schema_null = 1	Use NULL rather than an empty string ("") to denote a table that doesn't have a schema.
catalog_null = 1	Use NULL rather than an empty string ("") to denote a table that doesn't have a catalog.
	If you're using an ISAM data source with a linked server in Microsoft SQL Server, add the following lines to the section for the data source in odbc.ini:
	<code>schema_null = 1</code> <code>catalog_null = 1</code>
	Doing this will prevent SQL Server from generating the following error when working with tables that don't have a schema or catalog:
	<code>Invalid schema or catalog specified for provider 'MSDASQL'. OLE DB error trace [Non-interface error: Invalid schema or catalog specified for the provider.].</code>
	Returning an empty string for the schema or catalog name is permitted by the ODBC specification. However, SQL Server treats an empty string as an invalid schema or catalog name.

Setting	Description/example
statistics=*num*	The level of cardinality that the `SQLStatistics` function returns. The default value 31 means that `SQLStatistics` cardinality is handled by the ODBC client application. The values can be added together. For example, to retrieve the number of table rows and to use `SQL_QUICK` for index statistics use the value 25. If you're working with large amounts of ISAM data in SQL Server, using the statistics setting may improve performance. If you experience performance problems, try using `SQL_QUICK` rather than the default `SQL_ENSURE` to retrieve index statistics. Note that if you override SQL Server's default behaviour for `SQLStatistics` in this way, you'll need to check that the results are as expected. This is because SQL Server is unaware that the potentially less accurate `SQL_QUICK` is being used. The application can't, therefore, allow for the possibility that the statistics are incorrect.
statistics=1	Return the number of rows in the table.
statistics=2	Use the `SQL_ENSURE` argument to specify the level of accuracy when retrieving unique index statistics. `SQL_ENSURE` requests that the statistics are retrieved unconditionally.
statistics=4	Use the `SQL_ENSURE` argument to specify the level of accuracy when retrieving nonunique index statistics.

Setting	Description/example
statistics=8	Use the SQL_QUICK argument to specify the level of accuracy when retrieving unique index statistics. SQL_QUICK requests that the statistics are retrieved only if they are readily available. The values may not be current.
statistics=16	Use the SQL_QUICK argument to specify the level of accuracy when retrieving nonunique index statistics.
logging = 3	See **"Logging options" on page 77**.
logfile = /tmp/isam	See **"Logging options" on page 77**.
default_uid =	Specify a default user name for the data source. This overrides any user name specified on remote client machines.
default_pwd =	Specify a default password for the data source. This overrides any password specified on remote client machines.

Figure 20: odbc.ini settings

LOGGING OPTIONS

Various levels of logging may be enabled by insertng the `logfile=` and the `logging=` lines into your `odbc.ini` file:

- `logfile=` specifies the directory path and the filename prefix to which the log file is generated.

  The log filename adds to this prefix an underscore and a unique number derived from the Unix PID (process Identification number).

For example, setting `logfile=/tmp/isam` will result in a log filename of the format `isam_026503.log`.

- `logging=` specifies the level of logging required

Valid logging values are:

Log Number	Log Description
1	Entries to all functions and attributes to all calls in the SQI layer.
2	Exits from all functions.
4	File caching.
8	The opening of files and how many reads are performed.
16	Expression checking performed by the SQI layer on schema calls.
32	Errors reported at the SQI layer.
64	Indexes used when starting a query.
128	Summary information about SQI function calls.
256	Qualifiers (criteria) passed into the SQI start query function.
512	Summary information regarding the number of ISAM calls made.

Figure 21: odbc.ini log level options

Multiple logging options can be enabled by adding values together.

For example, to enable entry, exit and expression logging, use:

```
1+2+16 = 19
```

Caution! The enabling of logging will seriously impair performance and it should be disabled after use.

ADMINISTRATION

Using the Easysoft Web Administrator

This section describes how to create and grant user access permissions to your data using the Easysoft Web Administrator provided with Easysoft Data Access for ISAM.

Chapter Guide

- **Introduction**
- **Starting the Web Administrator**
- **Logging on to the Web Administrator**
- **Adding users**
- **Modifying and removing users**
- **Specifying user access rights**

Introduction

The Web Administrator utility allows you to limit the access that people have to your data by creating users for a data source and specifying their access rights to the data within that data source.

This allows individual users to be prevented from editing, deleting or viewing certain files.

Starting the Web Administrator

To run the Web Administrator it is necessary to start the Web Administrator server and then connect to it from a web browser.

Follow the instructions in the appropriate platform box to start the server:

Win	1. Select **Start > Programs > Easysoft > Easysoft Data Access for ISAM > Web Administration Server**. The Web Administrator starts up in a minimized DOS session. The port at which the server is listening is displayed in this DOS window. The default port is `8454`.

Unix	1. Log onto your machine as `root`. 2. Change into the `<InstallDir>/easysoft/isam/bin` directory. 3. Type: `./start_http.sh` Accept the default port (`8451`) or specify another unused port.

Run a Web browser on the machine from where you want to manage your users, and go to `http://server:8451` (where *server* is the name or IP address of the machine on which Easysoft Data Access for ISAM is installed and `8451` is the port at which the Web Administrator is listening).

For example, if the Web Adminstrator is running on the local machine, type:

```
http://localhost:8451
```

The main screen of the Web Administrator displays a list of the data sources which have been configured for use with Easysoft Data Access for ISAM:
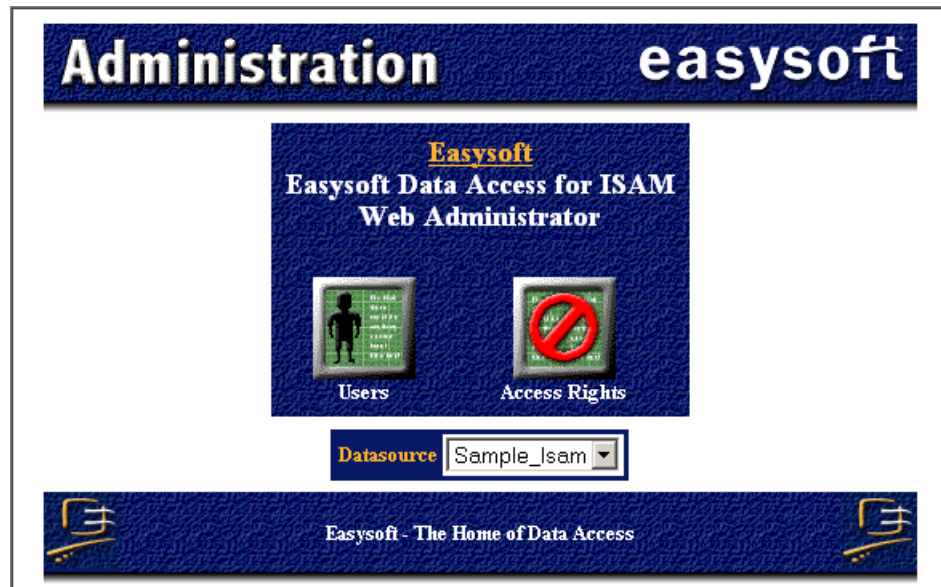
Figure 22: The main Web Administrator screen

Unix	During installation, if you did not specify the file locations needed to link to your ISAM data, and you have not specified them since, any attempt to work with a ISAM data source will produce an error. See **"odbc.ini settings for Unix" on page 72** for details of the attributes you need to set to configure the data source.

If you ever need to stop the Web Administrator:

- Windows users should close the DOS window in which the server is running.

- Unix users should log on as `root`, change into the `<InstallDir>/easysoft/isam/bin` directory and run `./stop_http.sh`.

The Web Administrator should only ever need to be restarted if the server fails or if it is to be run from another port.

NB	Sufficient permissions are required in order to stop the Web Administrator (i.e. users other than `root` can only stop processes they started). Also, in Unix (especially) file permissions dictate who can change the schema files where the users and privileges are stored.

Logging on to the Web Administrator

To create users and specify their rights, you must log on to the Web Administrator as the top level user (known as the database owner) and enter the password for that user.

You are required to log on when you click **Users** or **Access Rights** for the first time in a Web Administrator session, and again if you select a different data source or change the password of the database owner.

When asked to log on for the first time, enter a user name of `dbo` and password of `easysoft`, but once logged on, the password of the database owner must be changed immediately:

1. Click **Users** to display a list of users.

   Only the `dbo` user is set up by default.

2. Enter a new `dbo` password in the **Password** box and the old `dbo` password in the **Old Password** box.

   The old password must be entered for the new password to take effect.

3. Click **Submit** to save the new `dbo` password.

   "Adding users" on page 83 explains how to add other users so that their access to the ISAM data can be restricted.

NB	The database owner sees all tables that contain ISAM data within the Web Administrator.

Adding users

To add users:

1. Log in to the Web Administrator, as described in **"Logging on to the Web Administrator" on page 82**.

   You must log on as the `dbo` user if you have not already done so.

2. Click **Users**.

3. Select the data source for which you want to create users.

   Any users already set up for this data source are listed on the page. The last row in the list of users is blank, which is where the details are entered to add a new user.

> **NB** Click **Refresh** whenever you select a different data source to ensure that you are viewing the correct list of users for that data source.

4. To add a user, type a name in the empty **User** box.

5. Enter a description for the user if required (this is not mandatory).

6. Enter a password for the user if you require additional security for the selected data source (this is not mandatory).

7. Click **Submit** to create the user.

Figure 23: Adding users in the Web Administrator

By default, new users do not have access to any tables in the selected data source.

To find out how to enable privileges for a user, see **"Specifying user access rights" on page 86**.

When a user has been enabled, they must be informed of their user name and password (if any), because these fields must be entered when a user data source is configured to connect to the ISAM data.

NB	Users must be given their user name and password in the correct case, as these details are case sensitive.

Where the user enters these details depends on how they are connecting to the data, as the user name and password may be entered either as part of the data source configuration during set up or via an application dialog box at run time.

To configure the data source:

*   when connecting to local data on a Windows NT machine, enter these details on the **Easysoft Data Access for ISAM** DSN dialog box.

*   when connecting to local data on a Unix machine, enter these details in the appropriate DSN section of `odbc.ini`.

*   when using an Easysoft ODBC-ODBC Bridge client on Windows NT to connect to data on a Unix server, enter these details on the **Easysoft ODBC-ODBC Bridge** DSN dialog box that they complete to connect to the data source.

If log on details are not specified in the data source, then the user name and password should be requested as part of the application, but note that this is not necessarily the case and some applications may not run correctly.

Modifying and removing users

You can change a user's password, and you can delete users to permanently remove their access to the ISAM data.

To modify or remove a user:

1. Log in to the Web Administrator as described in **"Logging on to the Web Administrator" on page 82**.

2. Click **Users**.

3. Select the data source whose users you want to modify or delete and then click **Refresh** to ensure that you are viewing the users for that data source.

4. Edit the details in the **Password** box for the appropriate user and then click **Submit** to confirm your changes.

5. To delete a user, click the **Delete** box for that user (the box contains an X when it is selected) and then click **Submit**.

The user is removed from the list on this page.

Specifying user access rights

Once a user has been created for a data source, you can choose the data source tables to which that user has access and whether they have select, insert, update or delete access to each table.

To grant a user access to specific data source tables:

1. Log in to the Web Administrator as described in **"Logging on to the Web Administrator" on page 82**.

2. Click **Access Rights**.

3. Choose the data source for which you are granting access and then click **Refresh** to ensure that the correct list of users is displayed.

4. Choose the user for whom you want to grant access.

Choose the `public` user to specify the same rights for all users and the click **Refresh** to ensure that the correct permissions are displayed for this user.

All the tables in the selected data source are listed, and the current user's rights to those tables are shown.

5. To grant the user all rights to all tables, click **Grant All**.

6. To remove all the rights of a user to all tables, click **Revoke All**.

7. To grant specific rights to specific tables, select or deselect the appropriate boxes and click **Submit Selection** to apply the changes.

   For example, to give a user read-only access to all tables in a data source:

   • click **Revoke All** to remove all the rights of that user

   • click the **Select** box for each table (these boxes contain an "`x`" when selected)

   • apply the changes by clicking **Submit Selection**

   In **Figure 27 on page 88**, the access rights for user `test` are:

   • SELECT access for all eight tables in the data source

   • INSERT access to the LINEITEM table

   • UPDATE and DELETE access to the PART table

Figure 24: Specifying user access rights in the Web Administrator

Note that the permissions table also shows any **Catalog** or **Schema** details in the two left-hand columns for each table that is displayed.

Additional columns show **Grantor** and **Grantee** information relating to the table permissions that have been allocated.

CLIENT SETUP

5

Setting up clients to connect to ISAM data

This section explains how to set up Windows client machines to connect to ISAM data on the server.

The procedure involves installing the Easysoft ODBC-ODBC Bridge client and setting up a data source.

Chapter Guide

- **Installing the Easysoft ODBC-ODBC Bridge client**
- **Setting up a data source on your Windows client**
- **Worked example for Windows clients**

Installing the Easysoft ODBC-ODBC Bridge client

You can install the Easysoft ODBC-ODBC Bridge client on either Windows or Unix platforms, but, while installing the Easysoft ODBC-ODBC Bridge client on Unix makes it possible to access your ISAM data from Perl, CGI and Apache/PHP, most users of Easysoft Data Access for ISAM will probably be running Windows machines to access their ISAM server data.

To install the Easysoft ODBC-ODBC Bridge client on Windows:

1. Run the Easysoft ODBC-ODBC Bridge installation file.

   This file is supplied as a `.exe` on Windows (see **"Obtaining Easysoft Data Access for ISAM" on page 26**).

2. Follow the instructions on screen to install the Easysoft ODBC-ODBC Bridge client component, which must be installed on each machine that will be used to access the server database.

   You do not need to license the Easysoft ODBC-ODBC Bridge client, because the client license is covered by the Easysoft ODBC-ODBC Bridge server license that you obtained when licensing Easysoft Data Access for ISAM.

   Once you have installed the Easysoft ODBC-ODBC Bridge client, you can set up a data source to connect to the ISAM data on the server (see **"Setting up a data source on your Windows client" on page 91**).

   Please refer to the Easysoft ODBC-ODBC Bridge manual for full details of the client installation on both Windows and Unix.

   For details of connecting from a remote JDBC client, please refer to the Easysoft JDBC-ODBC Bridge manual.

Setting up a data source on your Windows client

To enable cross-platform ODBC access to your data using the Easysoft ODBC-ODBC Bridge you need to create a remote data source on your client to connect to your ISAM data on the server.

1. Select **Start > Settings > Control Panel**, double-click **Administrative Tools** and then **Data Sources (ODBC)**.

9x	Select **Start > Settings > Control Panel** and double-click **ODBC Data Sources (32bit)**.
NT	Select **Start > Settings > Control Panel** and double-click **Data Sources (ODBC)**.

The **ODBC Data Source Administrator** dialog box is displayed:

Figure 25: The ODBC Data Source Administrator

2. Select the **User DSN** tab to set up a data source that only you can access.

    – OR –

    Select the **System DSN** tab to create a data source which is available to anyone who logs on to this Windows machine.

3. Click **Add** to add a new DSN.

    The **Create New Data Source** dialog box displays a list of the drivers which are available:
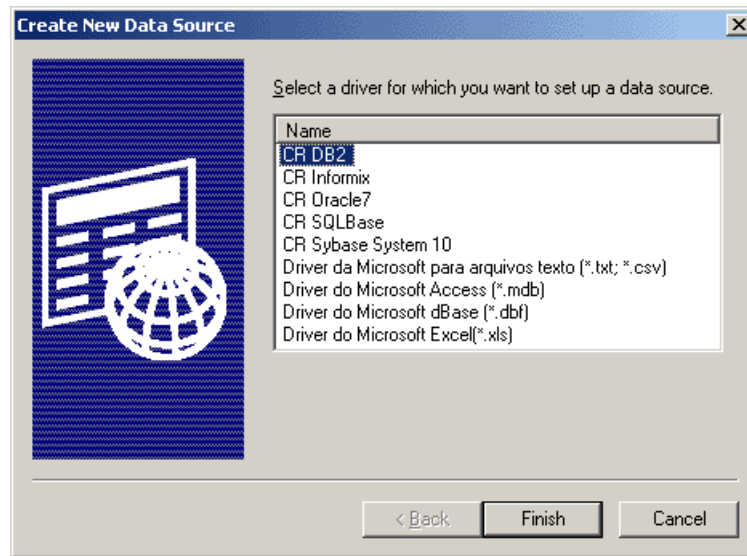
Figure 26: The Create New Data Source dialog box

4. Select `Easysoft ODBC-ODBC Bridge` and then click **Finish**.

The Easysoft ODBC-ODBC Bridge dialog box is displayed:
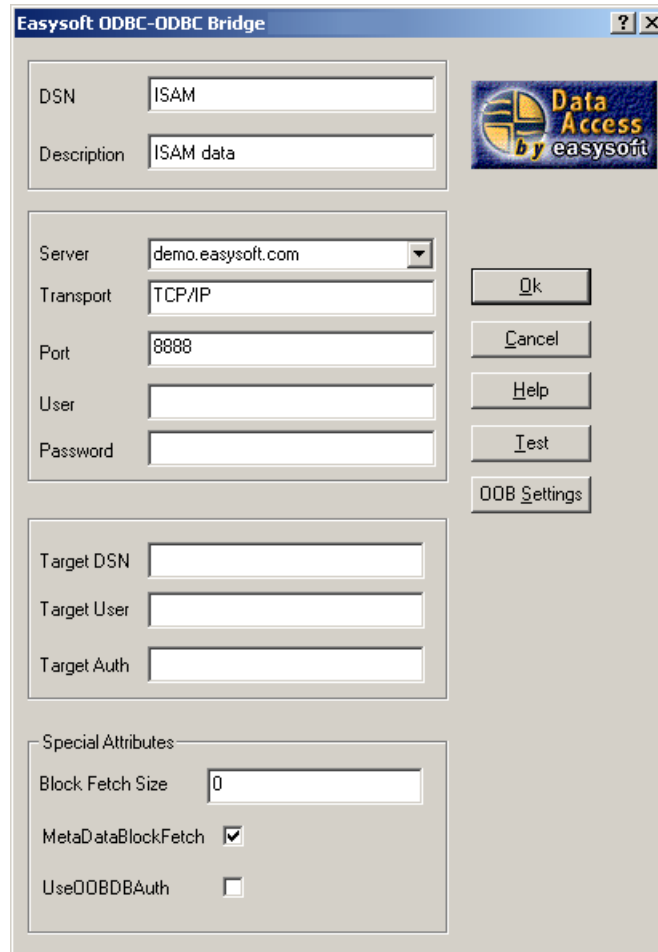
Figure 27: The Easysoft ODBC-ODBC Bridge DSN setup dialog box

5. In the **DSN** field, type a name for the data source, e.g. "ISAM".

   Choose carefully because you cannot change this later.

6. In the **Description** field, type a description for the data in the data source, e.g. "ISAM data".

7. In the **Server** box, type the hostname or IP address of the ISAM server.

8. The **Transport** field shows the network transport protocol that will be used to connect to the data.

    By default this is set to TCP/IP because this is the only protocol currently supported.

9. The **Port** field shows the port number on the server at which the Easysoft ODBC-ODBC Bridge server is listening for connections.

    Accept the default (8888) unless you know that the server is listening at a different port.

10. In the **User** and **Password** fields, type the user name and password that you use to log on to the server machine on which the ISAM data is stored.

11. In the **TargetDSN** field, type the name of the ISAM data source on the server machine.

    Your system administrator will be able to tell you what this name is.

12. In the **Target User** and **Target Auth** fields, type the user name and password that you use to log on to your ISAM server database.

    Your system administrator will be able to tell you what your logon details are.

13. Set **Block Fetch Size** to 10.

    This means that rows of data will be returned in blocks of 10 instead of one row at a time, resulting in faster performance.

    However, you should set this value to 1 if your ODBC application uses cursors or positioned updates/deletes.

14. Click **Test** to see if this data source can connect to the data on the server.

    The results are displayed in a separate window.

15. If the test is successful, click **OK** in the **Easysoft ODBC-ODBC Bridge** dialog box.

    The new data source is listed on the **User DSN** or **System DSN** tab of the **ODBC Data Source Administrator**.

16. Click **OK** to close the **ODBC Data Source Administrator**.

    Possible reasons for an unsuccessful test are:

    - if you see an "*Authentication Failure*" error, double-check the **User** and **Password** settings.

    - if you see a "*Failed to connect to remote driver*" error and additional text in which a login failure is noted, double-check the **Target User** and **Target Auth** settings.

    - if you see an error referring to data sources or DSNs, double-check the **Target DSN** name.

    - if you see a message referring to RPC, the client cannot connect to the Easysoft ODBC-ODBC Bridge server, so double-check the **Server**, **Transport** and **Port** settings.

Worked example for Windows clients

This section provides a worked example of using Microsoft Excel 97 to connect to your ISAM data source.

This worked example assumes that:

- you have a data source connecting to your ISAM data on the server. Consult your system administrator if you need to create a data source and are unsure where the data is stored on the server.

- your system administrator has given you access (via Easysoft Data Access for ISAM) to at least the SALES_CUSTOMER table in the data source.

- you have Microsoft Query installed (so that the **Data > Get External Data** command is available).

- you are familiar with the basics of using Microsoft Excel.

To obtain a list of customers and display the results in Microsoft Excel:

1. Run Microsoft Excel.

2. Select **Data > Get External Data > Create New Query**.

3. On the **Choose Data Source** dialog box, select the data source that connects to your ISAM data.

4. Make sure the **User Query Wizard** to create/edit queries option is selected (the box is ticked when the option is selected).

5. Click **OK** on the dialog box.

6. On the **Query Wizard - Choose Columns** dialog box, scroll down the list of tables to find the SALES_CUSTOMER table.

7. Click the + adjacent to this table to view the columns within it.

8. Select each of the following columns in turn, and click **>** so that they are listed in the **Columns in your query** box:

   CUSTOMER

   ALPHA

   NAME

   ADDRESS1

   ADDRESS2

   ADDRESS3

   ADDRESS4

   ADDRESS5

9. Click **Next** on this and the following dialog boxes until the **Query Wizard - Finish** dialog box is displayed.

10. Ensure that **Return Data to Microsoft Excel** is selected, then click **Finish**.

11. On the **Returning External Data to Microsoft Excel** dialog box, select **Existing worksheet** then click **OK**.

    A list of customers and their addresses is displayed in the worksheet.

    You could try creating another query using columns from both the SOP_ORDER_HEADER table and SOP_ORDER_DETAIL table to display a list of what these customers have purchased.

    When you have finished querying your data, close Microsoft Excel unless you want to continue using it for another task.

This page left blank intentionally

DEMONSTRATION

6

Worked example of connecting to ISAM data

This section demonstrates connecting to local ISAM data from an ODBC-compliant application on your Windows or Unix machine.

If you are implementing cross-platform access to your ISAM data using the Easysoft ODBC-ODBC Bridge or the Easysoft JDBC-ODBC Bridge, please refer to the documentation provided with those products for an example of connecting to remote data.

Chapter Guide

- **Demonstration on Windows**
- **Demonstration on Unix**

Demonstration on Windows

This section provides a worked example of using Microsoft Access to connect to the sample ISAM data provided with Easysoft Data Access for ISAM.

This worked example assumes that:

- you have a data source connecting to the Easysoft sample data which is created automatically during the Easysoft Data Access for ISAM installation. If this is not the case, refer to **"Creating additional data sources on Windows" on page 64** for details of how to set up a data source.

- you are familiar with the basics of using Microsoft Access.

NB	This demonstration uses Microsoft Access 2000. You may find that other versions of this product work slightly differently.

To obtain a list of customer order shipping details and display the results in Microsoft Access:

1. Run Microsoft Access.

2. Create a new blank database.

3. Display the **Tables** tab on the database window.

4. Right-click in the empty window, then select **Link Tables**.

   The **Link** dialog box is displayed.

5. From the **Files of Type** drop-down list box, select **ODBC Databases ()**.

   The **Select Data Source** dialog box is displayed.

6. Click the **Machine Data Source** tab.

7. Select the data source that connects to the sample data.

   The default name for this data source is `Sample_Isam`, but you may have given it an alternative name.

8. Click **OK**.

   The **Link Tables** dialog box is displayed.

9. Click **Select All** to link all the tables into the data base and then click **OK**.

   After a few moments, the tables are listed in the **Tables** tab of the database window in Microsoft Access.

10. Click the **Queries** tab in the database window and then click **New** to create a new query.

11. On the **New Query** wizard, click **Design View** and then click **OK**.

12. On the **Show Table** dialog box, select the CUSTOMER table and then click **Add**.

    The table is added to the **Select Query** window.

13. Select the ORDERS table and then click **Add**.

14. Select the LINEITEM table and then click **Add**.

15. Click **Close** to close the **Select Table** dialog box.

16. Resize the tables in the **Select Query** window until you can see all the column names in full.

The **Select Query** window will look something like:
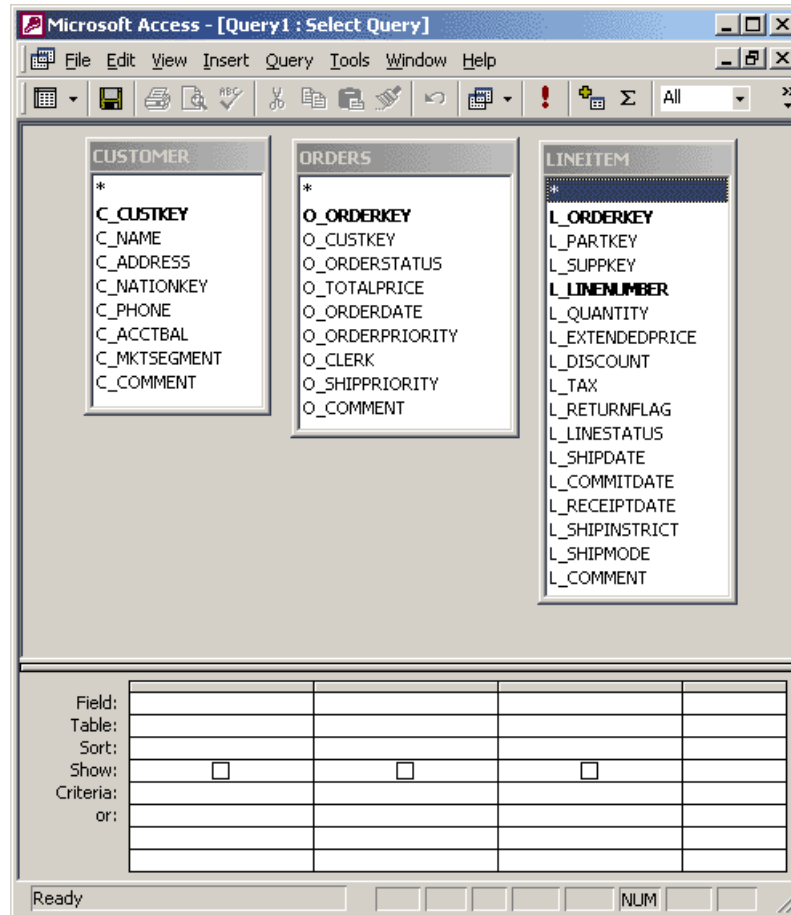
Figure 28: Tables on the Microsoft Access Select Query screen

17. Create the following joins:

- `C_CUSTKEY on CUSTOMER to O_CUSTKEY on ORDERS.`

- `O_ORDERKEY on ORDERS to L_ORDERKEY on LINEITEM`
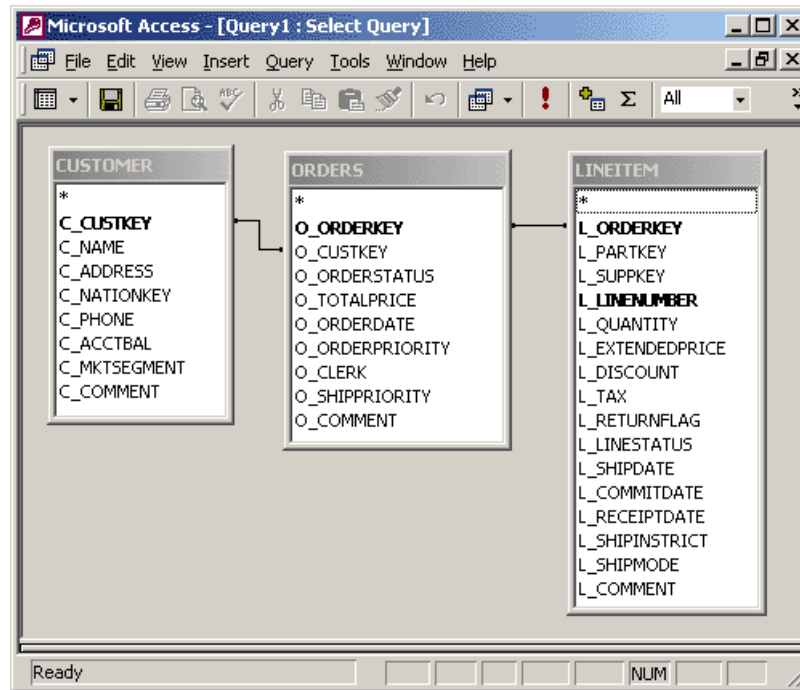
Figure 29: Joins on the Microsoft Access Select Query screen

18. From the CUSTOMER, ORDERS and LINEITEM tables, drag each of the following columns in turn and release them in the columns in the lower half of the **Select Query** window:

    C_NAME from the CUSTOMER table

    O_ORDERDATE from the ORDERS table

    L_QUANTITY from the LINEITEM table

    L_SHIPDATE from the LINEITEMtable

    O_ORDERKEY from the ORDERS table

The **Select Query** window will look something like:
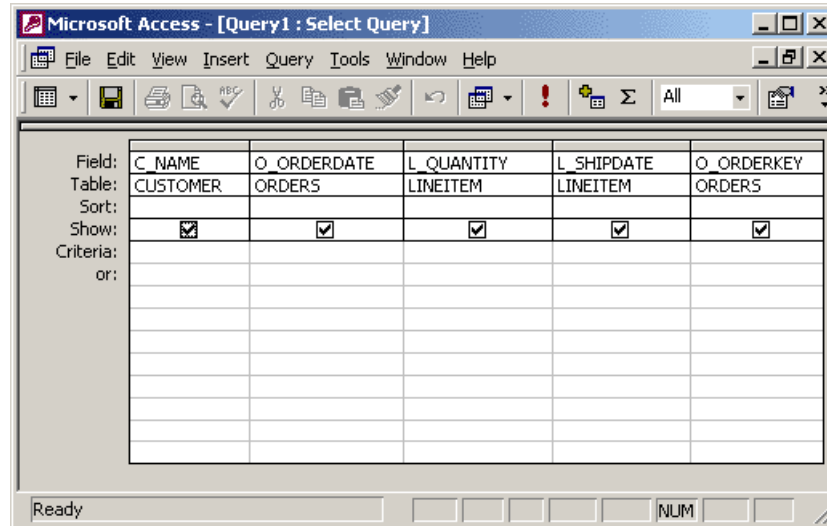
Figure 30: A Query on the Microsoft Access Select Query screen

19. Click the **Close** box to close the **Select Query** window. When asked if you want to save the query, click **Yes**.
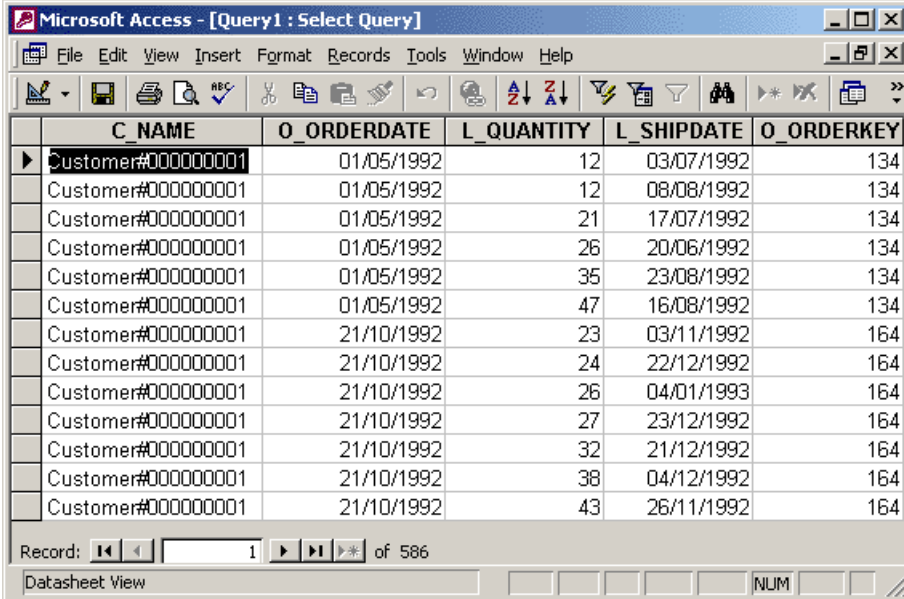
    The **Save As** dialog box is displayed.

20. Enter a name for the query then click **OK**.

    The query is now listed on the **Queries** tab of the database window.

21. Double-click the query to open it.

A list of customer order shipping details is displayed:

Figure 31: Column details on the Microsoft Access Select Query screen

Close Microsoft Access unless you want to continue using it for another task.

Demonstration on Unix

This section provides a worked example of using interactive SQL queries to connect to the sample ISAM data provided with Easysoft Data Access for ISAM.

NB	This demonstration uses a script called "`demosql`" to run an SQL script, as described in **"Querying a data source on Unix" on page 61**. This demonstration may be ignored if the instructions in that section have already been understood.

This worked example assumes that:

- a data source has been configured to connect to the Easysoft sample ISAM data which is created automatically during the Easysoft Data Access for ISAM installation. If this is not the case, refer to **"Creating additional data sources on Unix" on page 70** for details of how to set up a data source.

- the `dsn=` attribute in the `demosql` script is set to the sample data source ("`sample_isam`" by default), and the `uid=` and `pwd=` attributes are set to the user name and password of an existing user who has full access to all the tables in the sample data source.

To query the sample data source using `demosql`:

1. Change into the `usr/local/easysoft/isam/bin` directory.

2. Type:

   `./demosql`

If an "*invalid user*" or "*invalid password*" error is displayed, check that the correct user name and password are specified in the script.

If a line beginning "*conn =*" is displayed, then the connection has been made and you can type an SQL statement to query the data source.

3. To display a list of tables in the data source, type:

```
select * from info_schema.tables;
```

4. To display a list of customer names and their account balances, type:

```
select "C_NAME", "C_ACCTBAL" from CUSTOMER;
```

5. To display a list of order numbers and their total prices, type:

```
select "O_ORDERKEY", "O_TOTALprice" from ORDERS;
```

6. To display a list of customer order shipping details, type:

```
SELECT "C_NAME", "O_ORDERDATE", "L_QUANTITY",
"L_SHIPDATE", "O_ORDERKEY"
from CUSTOMER, ORDERS, LINEITEM
where "C_CUSTKEY"="O_CUSTKEY" and
"O_ORDERKEY"="L_ORDERKEY";
```

If the returned data scrolls out of view, press *<Pause>* and *<Enter>* (if using a Windows terminal) or *<Ctrl-S>* and *<Ctrl-Q>* (if using Unix xterm) to halt and then continue the listing on the screen.

To finish your SQL session, press *<Enter>* to return to the system prompt.

This page left blank intentionally

A
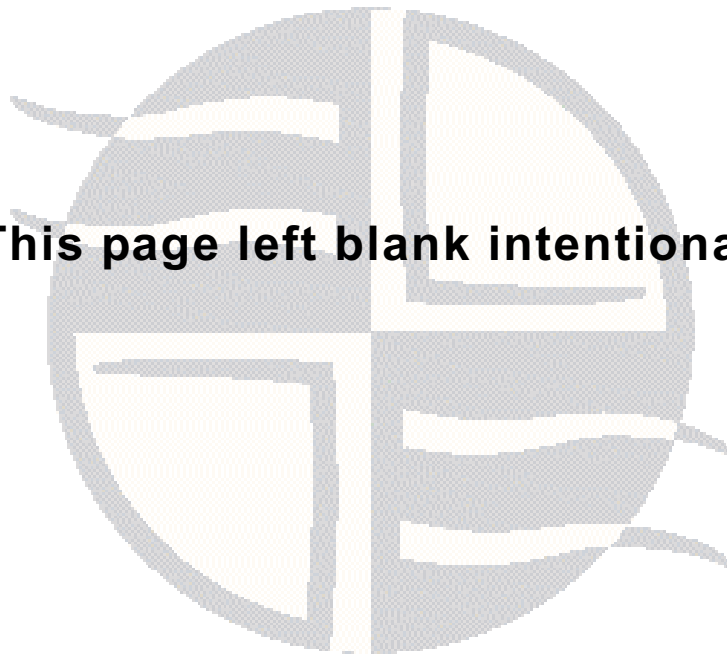
TECHNICAL REFERENCE

Technical Reference for Easysoft Data Access for ISAM

This appendix documents the ODBC and SQL conformance in Easysoft Data Access for ISAM, supported ISAM data types and transactions when using D-ISAM data.

Appendix Guide

- **Conformance**
- **ISAM to SQL data type conversions**
- **Easysoft Data Access for ISAM data types**
- **Transactions with D-ISAM**

Conformance

This section specifies the level of SQL and ODBC conformance in Easysoft Data Access for ISAM.

API SUPPORT

Easysoft Data Access for ISAM provides the following API Entry points:

Function	ODBC Conformance
SQLAllocHandle	Core
SQLBindCol	Core
SQLBindParameter	Core
SQLBrowseConnect	Level 1
SQLCancel	Core
SQLCloseCursor	Core
SQLColAttribute	Core
SQLColumnPrivileges	Level 2
SQLColumns	Core
SQLConnect	Core
SQLDescribeCol	Core
SQLDescribeParam	Level 2
SQLDisconnect	Core
SQLDriverConnect	Core
SQLEndTran	Core
SQLExecDirect	Core
SQLExecute	Core

Function	ODBC Conformance
SQLFetch	Core
SQLFetchScroll	Core
SQLForeignKeys	Level 2
SQLFreeHandle	Core
SQLFreeStmt	Core
SQLGetConnectAttr	Core
SQLGetCursorName	Core
SQLGetData	Core
SQLGetDescField	Core
SQLGetDescRec	Core
SQLGetDiagField	Core
SQLGetDiagRec	Core
SQLGetEnvAttr	Core
SQLGetFunctions	Core
SQLGetInfo	Core
SQLGetStmtAttr	Core
SQLGetTypeInfo	Core
SQLMoreResults	Level 1
SQLNativeSql	Core
SQLNumParams	Core
SQLNumResultCols	Core
SQLParamData	Core
SQLPrepare	Core
SQLPrimaryKeys	Level 1

Function	ODBC Conformance
SQLProcedureColumns	Level 1
SQLProcedures	Level 1
SQLPutData	Core
SQLRowCount	Core
SQLSetConnectAttr	Core
SQLSetCursorName	Core
SQLSetDescField	Core
SQLSetDescRec	Core
SQLSetEnvAttr	Core
SQLSetPos	Level 1
SQLSetStmtAttr	Core
SQLSpecialColumns	Core
SQLStatistics	Core
SQLTablePrivileges	Level 2
SQLTables	Core

Figure 32: Easysoft Data Access for ISAM API Entry Points

The following functions are provided by the ODBC Driver Manager:

Function	ODBC Conformance
SQLDataSources	Core
SQLDrivers	Core
SQLAllocConnect	Core
SQLAllocEnv	Core
SQLError	Core

Function	ODBC Conformance
SQLGetConnectOption	Core
SQLSetParam	Core

Figure 33: Easysoft Data Access for ISAM ODBC Driver Manager functions

The following functions are provided by the Setup DLL:

Function	ODBC Conformance
SQLConfigDriver	Core
SQLConfigDSN	Core

Figure 34: Easysoft Data Access for ISAM Setup DLL functions

The following functions are currently not supported, but are planned for future release:

Function	ODBC Conformance
SQLBulkOperations	Level 1
SQLCopyDesc	Core

Figure 35: Easysoft Data Access for ISAM unsupported functions

STATEMENT TYPES

Easysoft Data Access for ISAM supports the following statements:

- ALTER TABLE
- COMMIT
- CREATE INDEX
- CREATE TABLE
- CREATE VIEW
- DELETE STATEMENT (positioned)
- DELETE STATEMENT (searched)
- DROP INDEX
- DROP TABLE
- DROP VIEW
- GRANT
- INSERT
- REVOKE
- ROLLBACK
- SELECT
- SELECT FOR UPDATE
- UPDATE (positioned)
- UPDATE (searched)

UNIONS

Easysoft Data Access for ISAM supports UNION and UNION ALL.

TABLE REFERENCE

The Table reference list in a select can contain all or any of:

*   Table name

*   Sub Query

*   Join

JOINS

Easysoft Data Access for ISAM supports the following types of joins:

*   INNER

*   LEFT OUTER

*   RIGHT OUTER

*   FULL OUTER

*   NATURAL

The joining condition may be specified with the ON or USING clause. Both the left and right source can be a table name, sub query or another join. Joins can be nested with no restriction on depth.

Joins can be specified in both SQL92 and ODBC format:

SQL

```
SELECT * from x LEFT OUTER JOIN y ON x.a = y.a
```

ODBC

```
SELECT * from {oj x LEFT OUTER y ON x.a = y.a}
```

PREDICATES

Easysoft Data Access for ISAM supports the following predicates:

- Comparison (a = b)

- BETWEEN (a BETWEEN b AND c)

- LIKE (a LIKE '%green%')

- NULL (a IS NOT NULL)

- IN value_list (a IN (1, 2, 3))

- IN sub_query (a IN (SELECT x FROM y)

- Quantified comparison (a = ALL (SELECT x FROM y))

- Exists (EXISTS(SELECT x from y))

The sub queries in predicates can be correlated or non correlated:

Correlated

```
SELECT a FROM b WHERE c = ALL ( SELECT x FROM y
WHERE z = a )
```

Non-correlated

```
SELECT a FROM b WHERE c = ALL ( SELECT x FROM y
WHERE z = 12 )
```

SCALAR FUNCTIONS

Easysoft Data Access for ISAM provides all the functions required by ODBC and also functions from SQL92. Functions can be specified in SQL92 or ODBC format:

SQL

```
SELECT CURRENT_DATE, EXTRACT( YEAR FROM
Employee.data_of_birth ) FROM Employee
```

ODBC

```
SELECT {fn CURRENT_DATE()}, {fn EXTRACT( YEAR FROM
Employee.data_of_birth )} FROM Employee
```

The following is a complete set of supported functions:

Pseudo Variable Functions

- CURRENT_DATE
- CURRENT_TIME
- CURRENT_TIMESTAMP
- CURRENT_USER
- USER

SQL92 Functions

- BIT_LENGTH
- CHAR_LENGTH
- CHARACTER_LENGTH
- OCTET_LENGTH
- POSITION
- SUBSTRING
- TRIM

ODBC Functions

- ASCII
- CHAR
- CONCAT
- DIFFERENCE
- INSERT
- LCASE
- LEFT
- LENGTH
- LOCATE
- LTRIM
- REPEAT
- REPLACE
- RIGHT
- RTRIM
- SOUNDEX
- SPACE
- UCASE
- ABS
- ACOS
- ASIN
- ATAN
- ATAN2

- CEILING
- COS
- COT
- DEGREES
- EXP
- FLOOR
- LOG
- LOG10
- MOD
- PI
- POWER
- RADIANS
- RAND
- ROUND
- SIGN
- SIN
- SQRT
- TAN
- TRUNCATE
- CURDATE
- CURTIME
- DAYNAME
- DAYOFMONTH

- DAYOFWEEK

- DAYOFYEAR

- EXTRACT

- HOUR

- MINUTE

- MONTH

- MONTHNAME

- NOW

- QUARTER

- SECOND

- TIMESTAMPADD

- TIMESTAMPDIFF

- WEEK

- YEAR

- DATABASE

- IFNULL

AGGREGATE FUNCTIONS

Easysoft Data Access for ISAM supports the following aggregate functions:

- COUNT(* | ALL | DISTINCT)

- AVG(ALL | DISTINCT)

- MIN(ALL | DISTINCT)

- MAX(ALL | DISTINCT)

- SUM(ALL | DISTINCT)

CONVERSION FUNCTIONS

Easysoft Data Access for ISAM supports both the SQL92 CAST function and the ODBC CONVERT FUNCTION for conversion between compatible data types.

CONDITIONAL FUNCTIONS

Easysoft Data Access for ISAM supports CASE statements and the shorthand forms NULLIF and COALESCE.

LITERALS

All SQL92 and ODBC32 literals are supported and can be specified in either form:

SQL92

```
DATE '1999-01-02', INTERVAL '10-2' YEAR TO MONTH
```

ODBC

```
{d '1999-01-02'}, {INTERVAL '10-2' YEAR TO MONTH}
```

OPTIMIZATION

Easysoft Data Access for ISAM performs several optimizations to improve performance including the following:

Query optimization

The WHERE clause of a query will be rewritten into a form that allows more efficient processing of data.

For example the query:

```
SELECT * FROM x WHERE (a = 10 or b = 20) and c = 30
```

will be changed into the equivalent:

```
SELECT * FROM x WHERE a = 10 and c = 30

UNION

SELECT * FROM x WHERE b = 20 and c = 30
```

Table Optimization

In cases where indexes are present on tables, Easysoft Data Access for ISAM will, if necessary, rearrange the order in which tables are processed to enable the index to be used, which can lead to considerable increases in performance.

For example, consider the following query where tables `a`, `b` and `c` each have 800 rows and an index on `catalog_number`.

```
SELECT * FROM a, b, c

WHERE a.catalog_number = c.catalog_number

AND b.catalog_number = a.catalog_number

AND a."desc" = b."desc"

AND c.retail = a.retail

AND a.catalog_number = b.catalog_number
```

When run with the table order specified the query takes about 350 seconds on a given test configuration. If table optimization is enabled then, on the same machine, the query takes under 2 seconds.

The effect of this optimization is most noticeable on some of the queries that comprise the TPC-D benchmark set. Without this optimization some of the queries can be considered to never end (still running after a day), with optimization the same query will return in under 10 seconds.

INFORMATIONAL SCHEMA

Easysoft Data Access for ISAM provides an informational schema view of the tables supplied by the target data sources. The following tables are available:

- INFO_SCHEMA.CHARACTER_SETS
- INFO_SCHEMA.COLLATIONS
- INFO_SCHEMA.COLUMN_PRIVILEGES
- INFO_SCHEMA.COLUMNS
- INFO_SCHEMA.INDEXES
- INFO_SCHEMA.SCHEMATA
- INFO_SCHEMA.SERVER_INFO
- INFO_SCHEMA.SQL_LANGUAGES
- INFO_SCHEMA.TABLE_PRIVILEGES
- INFO_SCHEMA.TABLES
- INFO_SCHEMA.USAGE_PRIVILEGES
- INFO_SCHEMA.VIEWS

DATA TYPES

Easysoft Data Access for ISAM supports the following SQL data types:

- SQL_CHAR
- SQL_VARCHAR
- SQL_LONGVARCHAR
- SQL_NUMERIC
- SQL_DECIMAL
- SQL_SMALLINT
- SQL_INTEGER
- SQL_REAL
- SQL_FLOAT
- SQL_DOUBLE
- SQL_BIT
- SQL_TINYINT
- SQL_BIGINT
- SQL_BINARY
- SQL_VARBINARY
- SQL_LONGVARBINARY
- SQL_TYPE_DATE
- SQL_TYPE_TIME
- SQL_TYPE_TIMESTAMP
- SQL_INTERVALS (all types)

ODBC FEATURES

Cursors

Easysoft Data Access for ISAM provides FORWARD ONLY, STATIC and KEYSET CURSORS.

It also provides the following additional ODBC features (reported via the `SQLGetIinfo` API call):

- SQL_ASYNC_MODE

  Asynchronous operation is supported, both at the statement and connection level (SQL_AM_STATEMENT)

- SQL_COLUMN_ALIAS

  Column alias is supported using the optional AS clause

- SQL_CORRELATION_NAME

  Correlation names are supported and can be any valid user-defined name

- SQL_DATETIME_LITERALS

  All SQL92 Datetime literals are supported

- SQL_GETDATA_EXTENSIONS

  `SQLGetData` can be called for any column, bound or unbound

- SQL_GROUPBY

  The columns in the GROUP BY clause and the select list are not related (SQL_GB_NO_RELATION)

- SQL_INDEX_KEYWORDS

  All keywords are supported

ISAM to SQL data type conversions

The following table lists the ISAM data types and edit masks which are supported by Easysoft Data Access for ISAM, and the SQL data types to which the data is converted.

Edit Mask	SQL data type
CHAR	Non-terminated Character string padded with trailing spaces
DEC	Byte designs packed decimal
DOUBLE	Platform specific double
FLOAT	Platform specific float
INT	2 byte big endian integer
LONG	4 byte big endian integer
MINT	2 byte platform specific integer
MLONG	4 byte platform specific integer
STRING	Character string trimmed and padded with 0x00

Figure 36: Easysoft Data Access for ISAM and ISAM data types

Easysoft Data Access for ISAM data types

Group	Name	Description	SQL Group
SQL92	BIGINT	Exact numeric value with precision 19	number
SQL92	BINARY	Binary data of fixed length with a maximum length 255	binary
SQL92	BIT	Single bit binary data	number
SQL92	CHAR	Character string of fixed string length with a maximum length 255	string
SQL92	DATE	Year, month, and day fields, conforming to the rules of the Gregorian calendar	datetime
SQL92	DECIMAL	Signed, exact, numeric value with a precision and scale	number
SQL92	DOUBLE	Signed, approximate, numeric value with a binary precision 53	number

Group	Name	Description	SQL Group
SQL92	FLOAT	Signed, approximate, numeric value	number
SQL92	INTEGER	Exact numeric value with precision 10 and scale 0	number
SQL92	INTERVAL_DAY	Number of days between two dates	interval
SQL92	INTERVAL_DAY_TO _HOUR	Number of days/hours between two date/times	interval
SQL92	INTERVAL_DAY_TO _MINUTE	Number of days/hours/minut es between two date/times	interval
SQL92	INTERVAL_DAY_TO _SECOND	Number of days/hours/minut es/seconds between two date/times	interval
SQL92	INTERVAL_HOUR	Number of hours between two dates/times	interval
SQL92	INTERVAL_HOUR_T O_MINUTE	Number of hours/minutes between two dates/times	interval
SQL92	INTERVAL_HOUR_T O_SECOND	Number of hours/minutes/se conds between two dates/times	interval

Group	Name	Description	SQL Group
SQL92	INTERVAL_MINUTE	Number of minutes between two dates/times	interval
SQL92	INTERVAL_MINUTE _TO_SECOND	Number of minutes/seconds between two dates/times	interval
SQL92	INTERVAL_MONTH	Number of months between two dates	interval
SQL92	INTERVAL_SECON D	Number of seconds between two dates/times	interval
SQL92	INTERVAL_YEAR	Number of years between two dates	interval
SQL92	INTERVAL_YEAR_T O_MONTH	Number of years and months between two dates	interval
SQL92	INTEGER AUTO	Exact numeric value with precision 10 auto incrementing	number
SQL92	LONG VARBINARY	Variable length binary data	binary
SQL92	LONG VARCHAR	Variable length character data	string
SQL92	NUMERIC	Signed, exact, numeric value with a precision and scale	number

Group	Name	Description	SQL Group
SQL92	REAL	Signed, approximate, numeric value with a binary precision 24	number
SQL92	SMALLINT	Exact numeric value with precision 5 and scale 0	number
SQL92	TIME	Hour, minute, and second fields	datetime
SQL92	TIMESTAMP	Year, month, day, hour, minute, and second fields	datetime
SQL92	TINYINT	Exact numeric value with precision 3 and scale 0	number
SQL92	VARBINARY	Variable length binary data with a maximum length 255	binary
SQL92	VARCHAR	Variable length character data with a maximum length 255	string
ASCII	NULL-PAD	CROSS PLATFORM : Null padded (0x00) character string with a maximum length of 32000 with a null values of 0x00.	string

Group	Name	Description	SQL Group
ASCII	SPACE-PAD	CROSS PLATFORM : Space padded (0x20) character string with a maximum length of 32000 with a null value of spaces.	string
ASCII-NUMERIC	NULL-LEFT	CROSS PLATFORM : Null padded left justified unsigned numeric.	number
ASCII-NUMERIC	NULL-LEFT-D	CROSS PLATFORM : Null padded left justified unsigned numeric with a decimal point '.'.	number
ASCII-NUMERIC	NULL-LEFT-LS	CROSS PLATFORM : Null padded left justified numeric with a leading sign (+ and -).	number
ASCII-NUMERIC	NULL-LEFT-LSD	CROSS PLATFORM : Null padded left justified numeric with a leading sign (+ and -) and contains a decimal point '.'.	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	NULL-LEFT-LN	CROSS PLATFORM : Null padded left justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	NULL-LEFT-LND	CROSS PLATFORM : Null padded left justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	NULL-LEFT-TS	CROSS PLATFORM : Null padded left justified numeric with a trailing sign (+ and -).	number
ASCII-NUMERIC	NULL-LEFT-TSD	CROSS PLATFORM : Null padded left justified numeric with a trailing sign (+ and -) and contains a decimal point '.'.	number
ASCII-NUMERIC	NULL-LEFT-TN	CROSS PLATFORM : Null padded left justified numeric with a trailing negative sign (-).	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	NULL-LEFT-TND	CROSS PLATFORM : Null padded left justified numeric with a trailing negative sign (-).	number
ASCII-NUMERIC	NULL-RIGHT	CROSS PLATFORM : Null padded right justified unsigned numeric.	number
ASCII-NUMERIC	NULL-RIGHT-D	CROSS PLATFORM : Null padded right justified unsigned numeric with a decimal point '.'.	number
ASCII-NUMERIC	NULL-RIGHT-LS	CROSS PLATFORM : Null padded right justified numeric with a leading sign (+ and -).	number
ASCII-NUMERIC	NULL-RIGHT-LSD	CROSS PLATFORM : Null padded right justified numeric with a leading sign (+ and -) and contains a decimal point '.'.	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	NULL-RIGHT-LN	CROSS PLATFORM : Null padded right justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	NULL-RIGHT-LND	CROSS PLATFORM : Null padded right justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	NULL-RIGHT-TS	CROSS PLATFORM : Null padded right justified numeric with a trailing sign (+ and -).	number
ASCII-NUMERIC	NULL-RIGHT-TSD	CROSS PLATFORM : Null padded right justified numeric with a trailing sign (+ and -) and contains a decimal point '.'.	number
ASCII-NUMERIC	NULL-RIGHT-TN	CROSS PLATFORM : Null padded right justified numeric with a trailing negative sign (-).	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	NULL-RIGHT-TND	CROSS PLATFORM : Null padded right justified numeric with a trailing negative sign (-).	number
ASCII-NUMERIC	SPACE-LEFT	CROSS PLATFORM : Space padded left justified unsigned numeric.	number
ASCII-NUMERIC	SPACE-LEFT-D	CROSS PLATFORM : Space padded left justified unsigned numeric and contains a decimal point '.'.	number
ASCII-NUMERIC	SPACE-LEFT-LS	CROSS PLATFORM : Space padded left justified numeric with a leading sign (+ and -).	number
ASCII-NUMERIC	SPACE-LEFT-LSD	CROSS PLATFORM : Space piadded left justified numeric with a leading sign (+ and -) and contains a decimal point '.'.	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	SPACE-LEFT-LN	CROSS PLATFORM : Space padded left justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	SPACE-LEFT-LND	CROSS PLATFORM : Space padded left justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	SPACE-LEFT-TS	CROSS PLATFORM : Space padded left justified numeric with a trailing sign (+ and -).	number
ASCII-NUMERIC	SPACE-LEFT-TSD	CROSS PLATFORM : Space padded left justified numeric with a trailing sign (+ and -) and contains a decimal point '.'.	number
ASCII-NUMERIC	SPACE-LEFT-TN	CROSS PLATFORM : Space padded left justified numeric with a trailing negative sign (-).	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	SPACE-LEFT-TND	CROSS PLATFORM : Space padded left justified numeric with a trailing negative sign (-).	number
ASCII-NUMERIC	SPACE-RIGHT	CROSS PLATFORM : Space padded right justified unsigned numeric.	number
ASCII-NUMERIC	SPACE-RIGHT-D	CROSS PLATFORM : Space padded right justified unsigned numeric with a decimal point '.'.	number
ASCII-NUMERIC	SPACE-RIGHT-LS	CROSS PLATFORM : Space padded right justified numeric with a leading sign (+ and -).	number
ASCII-NUMERIC	SPACE-RIGHT-LSD	CROSS PLATFORM : Space padded right justified numeric with a leading sign (+ and -) and contains a decimal point '.'.	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	SPACE-RIGHT-LN	CROSS PLATFORM : Space padded right justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	SPACE-RIGHT-LND	CROSS PLATFORM : Space padded right justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	SPACE-RIGHT-TS	CROSS PLATFORM : Space padded right justified numeric with a trailing sign (+ and -).	number
ASCII-NUMERIC	SPACE-RIGHT-TSD	CROSS PLATFORM : Space padded right justified numeric with a trailing sign (+ and -) and contains a decimal point '.'.	number
ASCII-NUMERIC	SPACE-RIGHT-TN	CROSS PLATFORM : Space padded right justified numeric with a trailing negative sign (-).	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	SPACE-RIGHT-TND	CROSS PLATFORM : Space padded right justified numeric with a trailing negative sign (-).	number
ASCII-NUMERIC	ZERO-RIGHT	CROSS PLATFORM : Zero padded right justified unsigned numeric.	number
ASCII-NUMERIC	ZERO-RIGHT-D	CROSS PLATFORM : Zero padded right justified unsigned numeric with a decimal point '.'.	number
ASCII-NUMERIC	ZERO-RIGHT-LS	CROSS PLATFORM : Zero padded right justified numeric with a leading sign (+ and -).	number
ASCII-NUMERIC	ZERO-RIGHT-LSD	CROSS PLATFORM : Zero padded right justified numeric with a leading sign (+ and -) and contains a decimal point '.'.	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	ZERO-RIGHT-LN	CROSS PLATFORM : Zero padded right justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	ZERO-RIGHT-LND	CROSS PLATFORM : Zero padded right justified numeric with a leading negative sign (-).	number
ASCII-NUMERIC	ZERO-RIGHT-TS	CROSS PLATFORM : Zero padded right justified numeric with a trailing sign (+ and -).	number
ASCII-NUMERIC	ZERO-RIGHT-TSD	CROSS PLATFORM : Zero padded right justified numeric with a trailing sign (+ and -) and contains a decimal point '.'.	number
ASCII-NUMERIC	ZERO-RIGHT-TN	CROSS PLATFORM : Zero padded right justified numeric with a trailing negative sign (-).	number

Group	Name	Description	SQL Group
ASCII-NUMERIC	ZERO-RIGHT-TND	CROSS PLATFORM : Zero padded right justified numeric with a trailing negative sign (-).	number
ASCII-DATE	DAYS	CROSS PLATFORM : Space padded left justified numeric containing a count of days.	datetime
ASCII-DATE	FORMAT	CROSS PLATFORM : Space padded left justified numeric containing a formatted date.	datetime
ASCII-TIME	NANOSECONDS	CROSS PLATFORM : Space padded left justified numeric containing a count of nano seconds.	datetime
ASCII-TIME	SECONDS	CROSS PLATFORM : Space padded left justified numeric containing a count of seconds.	datetime

Group	Name	Description	SQL Group
ASCII-TIME	FORMAT	CROSS PLATFORM : Space padded left justified numeric containing a formatted time.	datetime
ASCII-TIMESTAMP	NANOSECONDS	CROSS PLATFORM : Space padded left justified numeric containing a count of nano seconds.	datetime
ASCII-TIMESTAMP	SECONDS	CROSS PLATFORM : Space padded left justified numeric containing a count of seconds.	datetime
ASCII-TIMESTAMP	FORMAT	CROSS PLATFORM : Space padded left justified numeric containing a formatted timestamp.	datetime
FLOATING-POINT	FLOAT	PLATFORM SPECIFIC : 32 bit single precision floating point number.	number

Group	Name	Description	SQL Group
FLOATING-POINT	DOUBLE	PLATFORM SPECIFIC : 64 bit double precision floating point number.	number
FLOATING-POINT	IEEE-FLOAT-BE	CROSS PLATFORM : 32 bit single precision IEEE big endian float.	number
FLOATING-POINT	IEEE-FLOAT-LE	CROSS PLATFORM : 32 bit single precision IEEE little endian float.	number
FLOATING-POINT	IEEE-DOUBLE-BE	CROSS PLATFORM : 64 bit double precision IEEE big endian float.	number
FLOATING-POINT	IEEE-DOUBLE-LE	CROSS PLATFORM : 64 bit double precision IEEE little endian float.	number
INTEGER	SIGNED	PLATFORM SPECIFIC : signed integer with a maximum length of 8.	number
INTEGER	UNSIGNED	PLATFORM SPECIFIC : unsigned integer with a maximum length of 8.	number

Group	Name	Description	SQL Group
INTEGER	SIGNED-BE	CROSS PLATFORM : signed big endian integer with a maximum length of 8.	number
INTEGER	UNSIGNED-BE	CROSS PLATFORM : unsigned big endian integer with a maximum length of 8.	number
INTEGER	SIGNED-LE	CROSS PLATFORM : signed little endian integer with a maximum length of 8.	number
INTEGER	UNSIGNED-LE	CROSS PLATFORM : unsigned little endian integer with a maximum length of 8.	number
INTEGER	SIGNED-RE	PLATFORM SPECIFIC : signed reversed endian integer with a maximum length of 8.	number

Group	Name	Description	SQL Group
INTEGER	UNSIGNED-RE	PLATFORM SPECIFIC : unsigned reversed endian integer with a maximum length of 8.	number
INTEGER-DATE	DAYS	PLATFORM SPECIFIC : unsigned integer (maximum length 8) containing a count of days.	datetime
INTEGER-DATE	DAYS-BE	CROSS PLATFORM : unsigned big endian integer (maximum length 8) containing a count of days.	datetime
INTEGER-DATE	DAYS-LE	CROSS PLATFORM : unsigned little endian integer (maximum length 8) containing a count of days.	datetime
INTEGER-DATE	DAYS-RE	CROSS PLATFORM : unsigned reverse endian integer (maximum length 8) containing a count of days.	datetime

Group	Name	Description	SQL Group
INTEGER-DATE	FORMAT	PLATFORM SPECIFIC : unsigned integer (maximum length 8) containing a date with a default format of YYYYMMDD.	datetime
INTEGER-DATE	FORMAT-BE	CROSS PLATFORM : unsigned big endian integer (maximum length 8) containing a date with a default format of YYYYMMDD.	datetime
INTEGER-DATE	FORMAT-LE	CROSS PLATFORM : unsigned little endian integer (maximum length 8) containing a date with a default format of YYYYMMDD.	datetime
INTEGER-DATE	FORMAT-RE	CROSS PLATFORM : unsigned reverse endian integer (maximum length 8) containing a date with a default format of YYYYMMDD.	datetime

Group	Name	Description	SQL Group
INTEGER-TIME	FORMAT	PLATFORM SPECIFIC : unsigned integer (maximum length 8) containing a time with a default format of HHMNSS.	datetime
INTEGER-TIME	FORMAT-BE	CROSS PLATFORM : unsigned big endian integer (maximum length 8) containing a time with a default format of HHMNSS.	datetime
INTEGER-TIME	FORMAT-LE	CROSS PLATFORM : unsigned little endian integer (maximum length 8) containing a time with a default format of HHMNSS.	datetime
INTEGER-TIME	FORMAT-RE	CROSS PLATFORM : unsigned reverse endian integer (maximum length 8) containing a time with a default format of HHMNSS.	datetime

Group	Name	Description	SQL Group
INTEGER-TIME	NANOSECONDS	PLATFORM SPECIFIC : unsigned integer (maximum length 8) containing a count of nano seconds.	datetime
INTEGER-TIME	NANOSECONDS-BE	CROSS PLATFORM : unsigned big endian integer (maximum length 8) containing a count of nano seconds.	datetime
INTEGER-TIME	NANOSECONDS-LE	CROSS PLATFORM : unsigned little endian integer (maximum length 8) containing a count of nano seconds.	datetime
INTEGER-TIME	NANOSECONDS-RE	CROSS PLATFORM : unsigned reverse endian integer (maximum length 8) containing a count of nano seconds.	datetime
INTEGER-TIME	SECONDS	PLATFORM SPECIFIC : unsigned integer (maximum length 8) containing a count of seconds.	datetime

Group	Name	Description	SQL Group
INTEGER-TIME	SECONDS-BE	CROSS PLATFORM : unsigned big endian integer (maximum length 8) containing a count of seconds.	datetime
INTEGER-TIME	SECONDS-LE	CROSS PLATFORM : unsigned little endian integer (maximum length 8) containing a count of seconds.	datetime
INTEGER-TIME	SECONDS-RE	CROSS PLATFORM : unsigned reverse endian integer (maximum length 8) containing a count of seconds.	datetime
INTEGER-TIMESTAMP	FORMAT	PLATFORM SPECIFIC : unsigned integer (maximum length 8) containing a formatted timestamp.	datetime
INTEGER-TIMESTAMP	FORMAT-BE	CROSS PLATFORM : unsigned big endian integer (maximum length 8) containing a formatted timestamp.	datetime

Group	Name	Description	SQL Group
INTEGER-TIMESTAMP	FORMAT-LE	CROSS PLATFORM : unsigned little endian integer (maximum length 8) containing a formatted timestamp.	datetime
INTEGER-TIMESTAMP	FORMAT-RE	CROSS PLATFORM : unsigned reverse endian integer (maximum length 8) containing a formatted timestamp.	datetime
INTEGER-TIMESTAMP	NANOSECONDS	PLATFORM SPECIFIC : unsigned integer (maximum length 8) containing a count of nano seconds.	datetime
INTEGER-TIMESTAMP	NANOSECONDS-BE	CROSS PLATFORM : unsigned big endian integer (maximum length 8) containing a count of nano seconds.	datetime

Group	Name	Description	SQL Group
INTEGER-TIMESTAMP	NANOSECONDS-LE	CROSS PLATFORM : unsigned little endian integer (maximum length 8) containing a count of nano seconds.	datetime
INTEGER-TIMESTAMP	NANOSECONDS-RE	CROSS PLATFORM : unsigned reverse endian integer (maximum length 8) containing a count of nano seconds.	datetime
INTEGER-TIMESTAMP	SECONDS	PLATFORM SPECIFIC : unsigned integer (maximum length 8) containing a count of seconds.	datetime
INTEGER-TIMESTAMP	SECONDS-BE	CROSS PLATFORM : unsigned big endian integer (maximum length 8) containing a count of seconds.	datetime
INTEGER-TIMESTAMP	SECONDS-LE	CROSS PLATFORM : unsigned little endian integer (maximum length 8) containing a count of seconds.	datetime

Group	Name	Description	SQL Group
INTEGER-TIMESTAMP	SECONDS-RE	CROSS PLATFORM : unsigned reverse endian integer (maximum length 8) containing a count of seconds.	datetime
PACKED	DECIMAL_STRING	CROSS PLATFORM : Right justified packed decimal with sign stored in low nibble of last byte and data encoded from left to right (high and low nibble).	number
PACKED	DECIMAL_STRING-U	CROSS PLATFORM : Right justified unsigned packed decimal with data encoded from left to right (high and low nibble).	number

Group	Name	Description	SQL Group
PACKED	BCD	CROSS PLATFORM : Binary coded decimal with encoded exponent in first byte and each subsequent byte contains a two digit number, negative number digits are stored subtracted from 100 (to maintain order).	number
PACKED	BCD-ONESCOMP	CROSS PLATFORM : Binary coded decimal with encoded exponent in first byte and each subsequent byte contains a two digit number, negative numbers are stored in ones complement. e.g. 0x2234 (decodes to 3252).	number
ISAM	CHAR	Non-terminated character string padded with trailing spaces	string
ISAM	DECIMAL	Byte designs packed decimal	number

Group	Name	Description	SQL Group
ISAM	DOUBLE	Platform specific double	number
ISAM	FLOAT	Platform specific float	number
ISAM	INT	2 byte big endian integer	number
ISAM	LONG	4 byte big endian integer	number
ISAM	MINT	2 byte platform specific integer	number
ISAM	MLONG	4 byte platform specific integer	number
ISAM	ROWID	4 byte platform specific integer	number
ISAM	STRING	Character string trimmed and padded with 0x00	string
V14	ABC-PACKED	Packed numeric with positive sign (3) and negative_sign (5)	number
V14	ABC-STRING	Ascii string with length encoded into first byte	string
V14	AMOS-F6	AMOS 6 byte float	number
V14	BASIC-DATE	2 byte native endian integer date with YYDDD format	datetime

Group	Name	Description	SQL Group
V14	BASIC-DATE4	4 byte native endian integer date with YYYYDDD format	datetime
V14	BD-DEC	Byte Designs decimal	number
V14	BDC-OVERPUNCHED-NUMERIC	Right justified overpunched trailing overpunched sign numeric	number
V14	BINDATE	4 byte native endian integer date with format YYYYMMDD	datetime
V14	BINDATE-REVERSED	4 byte reversed endian integer date with format YYYYMMDD	datetime
V14	BINTIME	4 byte native endian integer time with format HHMNSSFF	datetime
V14	BINTIME-2-U	2 byte unsigned native endian integer time with format HHMN	datetime
V14	BINTIME-U	4 byte unsigned native endian integer time with format HHMNSSFF	datetime

Group	Name	Description	SQL Group
V14	BIT-1	Datatype to return bit 1 of a single byte field	number
V14	BIT-2	Datatype to return bit 2 of a single byte field	number
V14	BIT-3	Datatype to return bit 3 of a single byte field	number
V14	BIT-4	Datatype to return bit 4 of a single byte field	number
V14	BIT-5	Datatype to return bit 5 of a single byte field	number
V14	BIT-6	Datatype to return bit 6 of a single byte field	number
V14	BIT-7	Datatype to return bit 7 of a single byte field	number
V14	BIT-8	Datatype to return bit 8 of a single byte field	number
V14	BPDATE	Bridgeport datatypes date	datetime
V14	BPDOUBLE	Byte designs packed decimal	number
V14	BYTE	1 byte native endian integer	number

Group	Name	Description	SQL Group
V14	C-TIME	4 byte native endian integer julian date, starting at 01-01-1970 and incrementing in seconds	datetime
V14	C-TIME-REV	4 byte reverse endian integer julian date, starting at 01-01-1970 and incrementing in seconds	datetime
V14	C-TIME2	4 byte native endian integer julian date, starting at 01-01-1900 and incrementing in seconds	datetime
V14	CDS-DATE	2 byte native endian integer calculated date, starting at 01-01-1900 with format YYDDD	datetime
V14	CDS-DATE2	2 byte reverse endian integer julian date, starting at 24-10-1924 and incrementing in days	datetime

Group	Name	Description	SQL Group
V14	CDS-TIME3	3 byte native endian binary date with bitpattern HH[0:0:8] MN[1:0:8] SS[2:0:8]	datetime
V14	CODA-DATE2	4 byte native endian integer julian date, starting at 17-11-1858 and incrementing in minutes	datetime
V14	D-FLOATING	OpenVMS D-FLOATING	number
V14	D-FLOATING-REV	OpenVMS D-FLOATING	number
V14	DATE	Ascii space padded date with format DD/MMM/YYYY	datetime
V14	DATE-LOWER-CASE	Ascii space padded lowercase date with format DD/MMM/YYYY	datetime
V14	DATE-MIXED-CASE	Ascii space padded mixedcase date (e.g. Jan) with format DD/MMM/YYYY	datetime

Group	Name	Description	SQL Group
V14	DATE-PAD-ZERO	Ascii zero padded date with format DDMMYYYY	datetime
V14	DIAL-199-DATE	3 byte native endian binary date with bitpattern YY[0:0:8] MM[1:0:8] DD[2:0:8], all bytes are stored subtracted from 199	datetime
V14	DIAL-99-DATE	3 byte native endian binary date with bitpattern YY[0:0:8] MM[1:0:8] DD[2:0:8], all bytes are stored subtracted from 99	datetime
V14	DIAL-DATE3	3 byte native endian date with bitpattern YY[0:0:8] MM[1:0:8] DD[2:0:8]	datetime
V14	DIAL-DATE6	6 byte ascii zero padded date with format XYMMDD, starting at 01-01-1900	datetime

Group	Name	Description	SQL Group
V14	DIBOL-DATE-5	5 byte ascii julian date starting at 31-12-1900	datetime
V14	DIBOL-TIME-5	5 byte ascii time (count of seconds)	datetime
V14	DIBOL-OVERPUNCHED-NUMERIC	Dibol right justified overpunched trailing sign numeric	number
V14	DOUBLE-SPACES-NULL	Platform specific double with a null value of spaces	number
V14	F-FLOATING	OpenVMS F FLOAT	number
V14	FLOATING-DATE-4	4 byte platform specific floating point julian date starting at 01-01-0001	datetime
V14	FLOATING-DATE-8	8 byte platform specific floating point julian date stating at 01-01-0001	datetime
V14	FLOATING-JULIAN	8 byte platform specific floating point julian date starting at 01-01-1800	datetime

Group	Name	Description	SQL Group
V14	FMBINDATE	4 byte native endian integer date with format YYYYMMDD and starting at 01-01-1900	datetime
V14	FREEFORM	Right justified leading sign ascii numeric	number
V14	FUG-DATE	3 byte native endian date with bitpattern YY[0:0:8] MM[1:0:8] DD[2:0:8] starting at 01-01-1800	datetime
V14	G-FLOATING	OpenVMS G FLOAT	number
V14	G-FLOATING-REV	Byte reversed OpenVMS G FLOAT	number
V14	H-FLOATING	OpenVMS H FLOAT	number
V14	HEXADECIMAL	Represent binary data as hexadecimal strings	string
V14	HEXADECIMAL-REVERSED	Byte reversed binary data represented as hexadecimal strings	string

Group	Name	Description	SQL Group
V14	HEXDATE	4 byte native endian date with bitpattern DD[0:0:8],MM[1:0:8],YYYY[2:0:8], YYYY[3:0:8]	datetime
V14	HEXDATE-REVERSED	4 byte reversed endian date with bitpattern DD[0:0:8],MM[1:0:8],YYYY[2:0:8], YYYY[3:0:8]	datetime
V14	HEXDATE3	3 byte native endian date with bitpatter DD[0:0:8],MM[1:0:8],YY[2:0:8] stating at 01-01-1900	datetime
V14	HEXDATE3-REVERSED	3 byte reversed endian date with bitpatter DD[0:0:8],MM[1:0:8],YY[2:0:8] stating at 01-01-1900	datetime
V14	IBASEDATE	8 byte native endian julian timestamp in nanoseconds starting at 17-11-1858	datetime
V14	ICC-INT1	1 byte big endian ones complemented integer	number

Group	Name	Description	SQL Group
V14	ICC-INT2	2 byte big endian ones complemented integer	number
V14	ICC-INT3	3 byte big endian ones complemented integer	number
V14	ICC-INT4	4 byte big endian ones complemented integer	number
V14	INFO-L-TYPE	Right justified, leading sign, zero padded ascii numeric	number
V14	INFO-N-TYPE	Right justified, leading sign, space padded ascii numeric	number
V14	INFO-T-TYPE	Right justified, trailing sign, zero padded ascii numeric	number
V14	INFO-V-TYPE	Right justified, absolute leading sign, zero padded ascii numeric	number
V14	INFORMIX-STRING	Character string padded with 0x00	string
V14	INTEGER-S	4 byte native endian integer	number

Group	Name	Description	SQL Group
V14	INTEGER1	1 byte native endian integer	number
V14	INTEGER2	2 byte native endian integer	number
V14	INTEGER3	3 byte native endian integer	number
V14	INTEGER4	4 byte native endian integer	number
V14	INTEGER5	5 byte native endian integer	number
V14	INTEGER6	6 byte native endian integer	number
V14	INTEGER7	7 byte native endian integer	number
V14	INTEGER8	8 byte native endian integer	number
V14	JDATE	2 byte native endian date with bitpattern DDD[0:0:8],DDD[1:0:1],YY[1:1:7] starting at 01-01-1900	datetime
V14	KD-DATE	7 byte ascii numeric date with format YYYYDDD	datetime
V14	LDA-NUMERIC	Right justified, trailing sign, zero padded ascii numeric with decimal point	number

Group	Name	Description	SQL Group
V14	LDA-LEADING-SIGN-NUMERIC	Right justified, leading sign, zero padded ascii numeric with decimal point	number
V14	LEADING-OVERPUNCHED-NUMERIC	Left justified, overpunched leading sign, null padded ascii numeric	number
V14	LEADING-SIGN-NUMERIC	Left justified, leading sign, space padded ascii numeric	number
V14	LONG	4 byte native endian integer	number
V14	LONGWORD	4 byte native endian integer	number
V14	LONGWORD-REVERSED	4 byte reverse endian integer	number
V14	MAGIC-DATE	Magic Julian Date	datetime
V14	MAGIC-INTEGER-DATE	4 byte native endian julian integer date starting at 01-01-0001	datetime
V14	MAGIC-INTEGER-DATE-REVERSED	4 byte reversed endian julian integer date starting at 01-01-0001	datetime

Group	Name	Description	SQL Group
V14	MAGIC-INTEGER-TIME	4 byte native endian julian integer time	datetime
V14	MAGIC-INTEGER-TIME-REVERSED	4 byte reversed endian julian integer time	datetime
V14	MAGIC-NUMBER	Magic Number	number
V14	MAGIC-STRING-MEMO	Character string with length encoded into first two bytes	string
V14	MAGIC-STRING-MEMO-4	Character string with length encoded into first four bytes	string
V14	MAGIC-TIME	Magic Time	datetime
V14	MICROFOCUS-COMP-5	Microfocus COBOL COMP-5	number
V14	MOTOROLA-LONGWORD	Motorola 4 byte native endian, word reversed integer	number
V14	MOTOROLA-LONGWORD-REVERSED	Motorola 4 byte reverse endian, word reversed integer	number
V14	NULL-TERMINATED-STRING	Null terminated/padded character string	string
V14	NUMERIC	Right justified, leading_sign, space padded ascii numeric	number

Group	Name	Description	SQL Group
V14	OVERPUNCHED-DATE-1	6 byte right justified, trailing overpunched sign ascii numeric julian date starting at 31-12-1799	datetime
V14	PACKED	Right justified, signed packed numeric	number
V14	PACKED-S	Right justified, signed packed numeric	number
V14	PHDATE	2 byte native endian date with bitpattern DD[0:0:5],MM[0:5:3],MM[1:0:1],YY[1:1:7] starting at 01-01-1900	datetime
V14	PHDATE4	4 byte native endian date with bitpattern DD[0:0:5],MM[0:5:3],MM[1:0:1],YYYY[1:1:7],YYYY[2:0:8],YYYY[3:0:8]	datetime
V14	PHDATETIME	8 byte unsigned native endian integer timestamp with format YYYYMMDDHHMNSSFF	datetime

Group	Name	Description	SQL Group
V14	QUADWORD	8 byte native endian integer	number
V14	QUADWORD-REVERSED	8 byte reversed endian integer	number
V14	RFA	OpenVMS Record File Address	binary
V14	RMC-ASCDATE-DDMMYY	6 byte ascii date with a format DDMMYY starting at 01-01-1951 with the century for years >= 51 being 19 and < 51 20.	datetime
V14	RMC-ASCDATE-YYMMDD	6 byte ascii date with a format YYMMDD starting at 01-01-1951 with the century for years >= 51 being 19 and < 51 20.	datetime
V14	RMC-BINDATE-DDMMYY	4 byte native endian integer date with a format DDMMYY starting at 01-01-1951 with the century for years >= 51 being 19 and < 51 20.	datetime

Group	Name	Description	SQL Group
V14	RMC-BINDATE-YYMMDD	4 byte native endian integer date with a format YYMMDD starting at 01-01-1951 with the century for years >= 51 being 19 and < 51 20.	datetime
V14	SAIC-TIMESTAMP	4 byte SAIC timestamp.	datetime
V14	SEB-DATE	6 byte native endian binary date with bitpattern YYYY[0:0:8],YYYY[1:0:8],MM[2:0:8],MM[3:0:8],DD[4:0:8],DD[5:0:8]).	datetime
V14	SINT3	3 byte native endian signed integer.	number
V14	SINT5	5 byte native endian signed integer.	number
V14	SINT6	6 byte native endian signed integer.	number
V14	SINT7	7 byte native endian signed integer.	number

Group	Name	Description	SQL Group
V14	SIRSI-MONEY	8 byte monetary values made up of two native_endian integers.	number
V14	SMITHSONIAN-DATE	4 byte native endian julian integer date starting at 17-11-1858.	datetime
V14	SOCIALE-DATE	4 byte packed decimal date with format DDMMYYYY.	datetime
V14	SOCIALE-DATE-REV	4 byte packed decimal date with format DDMMYYYY.	datetime
V14	SOCIALE-INT1	1 byte right justified packed numeric.	number
V14	SOCIALE-INT2	2 byte right justified packed numeric.	number
V14	SOCIALE-INT3	3 byte right justified packed numeric.	number
V14	SOCIALE-INT4	4 byte right justified packed numeric.	number
V14	SOCIALE-INT5	5 byte right justified packed numeric.	number

Group	Name	Description	SQL Group
V14	SOCIALE-INT6	6 byte right justified packed numeric.	number
V14	SOCIALE-INT7	7 byte right justified packed numeric.	number
V14	SPECIAL-1P		string
V14	SPECIAL-2P		string
V14	SPECIAL-3P		string
V14	SPECIAL-4P		string
V14	SPECIAL-5P		string
V14	SPECIAL-6P		string
V14	SPECIAL-DATE-1	2 byte native endian julian integer date starting at 05-08-1948.	datetime
V14	SPECIAL-DATE-2	2 byte native endian julian integer date starting at 31-12-1919.	datetime
V14	SPECIAL-DATE-3	5 byte ascii numeric julian date starting at 31-12-1899.	datetime
V14	SPECIAL-DATE-4	5 byte ascii numeric zero padded date with format YYDDD starting at 01-01-1900.	datetime

Group	Name	Description	SQL Group
V14	SPECIAL-DATE-5	4 byte native endian integer timestamp starting at 01-01-1984.	datetime
V14	SPECIAL-DATE-6	5 byte ascii numeric julian date starting at 31-12-1900.	datetime
V14	SPECIAL-DATE-7	2 byte native endian unsigned integer date with a format YYDDD starting at 01-01-1970.	datetime
V14	SPECIAL-DATE-8	2 byte native endian julian integer date starting at 01-01-1970.	datetime
V14	SPECIAL-DATE-9	3 byte zero padded ascii numeric date with format DDD starting at 01-01-1900.	datetime
V14	SPECIAL-DATE-10	5 byte ascii numeric julian date starting at 31-12-1799.	datetime
V14	SPECIAL-DATE-11	2 byte reversed endian integer date with format YYDDD starting at 01-01-1970.	datetime

Group	Name	Description	SQL Group
V14	SPECIAL-DATE-12	5 byte packed numeric date with format DDMMYYYY.	datetime
V14	SPECIAL-DATE-13	6 byte native endian binary date with bitpattern MM[0:0:8],MM[1:0:8],DD[2:0:8],DD[3:0:8],YYYY[4:0:8],YYYY[5:0:8]	datetime
V14	SPECIAL-DATE-14	4 byte reversed endian binary date with bitpattern DDD[0:0:8],DDD[1:0:8],YYYY[2:0:8],YYYY[3:0:8]	datetime
V14	SPECIAL-DATE-15	Time stored in native float.	datetime
V14	SPECIAL-DATE-16	3 byte big endian julian integer date starting at 31-12-1599.	datetime
V14	SPECIAL-DATE-17	6 byte ascii numeric zero padded date with format YY,MM,DD starting at 01-01-1900.	datetime

Group	Name	Description	SQL Group
V14	SPECIAL-DATE-18	2 byte native endian julian integer date starting at 31-12-1969.	datetime
V14	SPECIAL-DATE-19	4 byte packed numeric date with format YYMMDD starting at 01-01-1927 with century for years > 26 being 19 and <= 26 being 20.	datetime
V14	SPECIAL-DATE-20	3 byte packed numeric time with format HHMN.	datetime
V14	SPECIAL-DATE-21	2 byte native endian unsigned integer date with YYDDD format	datetime
V14	STAPLES-DATE	6 byte ascii numeric julian date starting at 31-12-1899.	datetime
V14	STARDATE	8 byte native endian julian timestamp in nanoseconds starting at 17-11-1858.	datetime
V14	STRING	Space padded non-terminated character string.	string

Group	Name	Description	SQL Group
V14	TETRA-DATE	8 byte ascii numeric zero padded date with format DD/MM/YY starting at 01-01-1940 with century for years > 20 being 19 and <= 40 being 20.	datetime
V14	TETRA-JULIAN-DATE	6 byte ascii numeric julian date starting at 31-12-1899.	datetime
V14	TIME	Ascii numeric time with variable format.	datetime
V14	TIMESTAMP	8 byte native endian julian timestamp in nanoseconds starting at 17-11-1858.	datetime
V14	TRAILING-NEGSIGN-DECIMAL	Right justified, space padded ascii numeric with trailing sign.	number
V14	TRAILING-OVERPUNCHED-NUMERIC	Right justified, zero padded ascii numeric with trailing overpunched sign.	number

Group	Name	Description	SQL Group
V14	TRAILING-SIGN-NUMERIC	Left justified, space padded ascii numeric with trailing sign.	number
V14	TRESTLE-DATE	2 byte native_endian binary date with bitpattern DD[0:0:5],MM[0:5:3],MM[1:0:1],YY[1:1:7] starting at 01-01-1980.	datetime
V14	TRESTLE-TIME	3 byte native endian time with bitpattern HH[0:0:8], MN[1:0:8], SS[2:0:8].	datetime
V14	UINT3	3 byte unsigned native endian integer.	number
V14	UINT5	5 byte unsigned native endian integer.	number
V14	UINT6	6 byte unsigned native endian integer.	number
V14	UINT7	7 byte unsigned native endian integer.	number
V14	UINTEGER1	1 byte native endian unsigned integer.	number

Group	Name	Description	SQL Group
V14	UINTEGER2	2 byte native endian unsigned integer.	number
V14	UINTEGER3	3 byte native endian unsigned integer.	number
V14	ULONG	4 byte native endian unsigned integer.	number
V14	ULONGWORD-REVERSED	4 byte reversed endian unsigned integer.	number
V14	UP3-DATE	3 byte packed date with format YYDDD stating at 01-01-1900.	datetime
V14	UQUADWORD	8 byte native endian unsigned integer.	number
V14	UQUADWORD-REVERSED	8 byte reversed endian unsigned integer.	number
V14	UWORD	2 byte unsigned native endian integer.	number
V14	UWORD-REVERSED	2 byte reversed endian unsigned integer.	number

Group	Name	Description	SQL Group
V14	VMSDATE4	4 byte native endian julian integer timestamp starting at 17-11-1858.	datetime
V14	VMSDATE4-REVERSED	4 byte reversed endian julian integer timestamp starting at 17-11-1858.	datetime
V14	VMSDATE8	8 byte native endian julian integer timestamp in nanoseconds starting at 17-11-1858.	datetime
V14	VMSDATE8-REVERSED	8 byte reversed endian integer timestamp in nanoseconds starting at 17-11-1858.	datetime
V14	WORD	2 byte native endian integer.	number
V14	WORD-REVERSED	2 byte reversed endian integer.	number
V14	ZONED	Right justified, zero padded ascii numeric.	number

Group	Name	Description	SQL Group
V14	ZONED-NUM	Right justified, zero padded, overpunched trailing sign ascii numeric.	number
V14	ZONED-S	Right justified, zero padded, overpunched trailing sign ascii numeric.	number
V14	ZONED-U	Right justified, zero padded ascii numeric.	number
V14	PACKED-U	Right justified, unsigned packed numeric.	number
V14	SP3-DATE	5 byte signed packed numeric date with format YYDDD starting at 01-01-1900.	datetime
V14	SPD-DATE	6 byte signed packed numeric date with format YYYYMMDD.	datetime
V14	SPD-DATE-5	5 byte signed packed numeric date with format YYYYMMDD.	datetime
V14	PSI-DATE	4 byte signed packed numeric date with format YYYYDDD.	datetime
V14	DOUBLE	Native double.	number

Group	Name	Description	SQL Group
V14	DOUBLE-REVERSED	Byte reversed double.	number
V14	FLOAT	Native float.	number
V14	FLOAT-REVERSED	Byte reversed double.	number
V14	INFORMIX-DATE-4	4 byte native endian julian integer date starting at 31-12-1899 with a null value of 0x00000080.	datetime
V14	INFORMIX-DATE-4-REV	4 byte reversed endian julian integer date starting at 31-12-1899 with a null value of 0x80000000.	datetime
V14	INFORMIX-INTEGER	4 byte native endian integer with a null value of 0x00000080.	number
V14	INFORMIX-INTEGER-REVERSED	4 byte reversed endian integer with a null value of 0x80000000.	number
V14	INFORMIX-SMALLINT	2 byte native endian integer with a null value of 0x0080.	number

Group	Name	Description	SQL Group
V14	INFORMIX-SMALLINT-REVERSED	2 byte reversed endian integer with a null value of 0x8000.	number
POWERHOUSE	BIG INTEGER SIGNED	PowerHouse INTEGER SIGNED datatype with a length between 4 and 8 bytes	number
POWERHOUSE	BIG INTEGER UNSIGNED	PowerHouse INTEGER UNSIGNED datatype with a length between 4 and 8 bytes	number
POWERHOUSE	CHARACTER	PowerHouse CHARACTER datatype one character per byte, using the native character set of the host.	string
POWERHOUSE	FLOAT	PowerHouse 4 byte FLOAT datatype	number
POWERHOUSE	DOUBLE	PowerHouse 8 byte FLOAT datatype.	number
POWERHOUSE	FREEFORM	PowerHouse FREEFORM datatype.	number
POWERHOUSE	IBASEDATE	PowerHouse IBASEDATE datatype.	datetime

Group	Name	Description	SQL Group
POWERHOUSE	INTEGER SIGNED	PowerHouse INTEGER SIGNED datatype with a maximum length of 4 bytes (see BIG INTEGER SIGNED for sizes > 4 bytes).	number
POWERHOUSE	INTEGER UNSIGNED	PowerHouse INTEGER UNSIGNED datatype with a maximum length of 4 bytes (see BIG INTEGER UNSIGNED for size > 4 bytes).	number
POWERHOUSE	JDATE	PowerHouse JDATE datatype.	datetime
POWERHOUSE	PACKED SIGNED	PowerHouse PACKED SIGNED datatype.	number
POWERHOUSE	PACKED UNSIGNED	PowerHouse PACKED UNSIGNED datatype.	number
POWERHOUSE	PHDATE	PowerHouse PHDATE datatype.	datetime
POWERHOUSE	VARCHAR	PowerHouse VARCHAR datatype.	string

Group	Name	Description	SQL Group
POWERHOUSE	ZONED	PowerHouse ZONED datatype.	number

Figure 37: Easysoft Data Access for ISAM data types

Transactions with D-ISAM

HOW D-ISAM PERFORMS TRANSACTIONS

Transaction processing defines a collection of changes to a group of data files in such a way as to be able to undo these changes at any time during the execution of a given operation.

Changes to records during D-ISAM transactions are performed immediately (other users will see changes made during a D-ISAM transaction before the transaction is committed), but it is possible to undo these changes by rolling back a given transaction.

Note that the transaction processing logic will lock all records that are changed within a transaction for the duration of that transaction.

In some instances it pays to keep the transaction cycle as short as possible to allow other processes access to these records.

ENABLING D-ISAM TRANSACTIONS

Two settings must be made to the server side data source to enable D-ISAM based transactions:

- transactions must be enabled

- a transaction log file must be specified

All users using this facility must have read/write access to the transaction log file

NB	The transaction log file will continually grow in size and can be backed up and purged on a regular basis to reduce the file size.

On Unix

Edit the relevant data source section of `/etc/odbc.ini` as follows:

- to enable transactions, add:

  ```
  transactions = 1
  ```

- to define a transaction log file, add:

  ```
  transaction_log =
  {full transaction log path and file name}
  ```

On Windows

Edit one of the following sections of the Registry:

- `HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBC.INI`

  for a system data source

– OR –

- `HKEY_CURRENT_USER\SOFTWARE\ODBC\ODBC.INI`

  for a user data source

From the list of data sources, click on the required data source and all current data source values will be listed in the right hand panel.

To enable transactions, right click in the right hand panel and select `New -> String Value`. Name the entry "`transactions`" and set the value to "`1`" (by double clicking the newly created entry).

To define a transaction log file, repeat the procedure, creating an entry called "`transaction_log`" and setting the value to the current location of the ISAM transaction log (if no transaction log exists, set the value to "`C:\disam.trn`").

TRANSACTION RESTRICTIONS

ALTER, CREATE, DROP, GRANT and REVOKE statements are performed outside of the transaction processing cycle.

INSERT, UPDATE and DELETE are the only statements on which transaction processing can be performed.

This page left blank intentionally
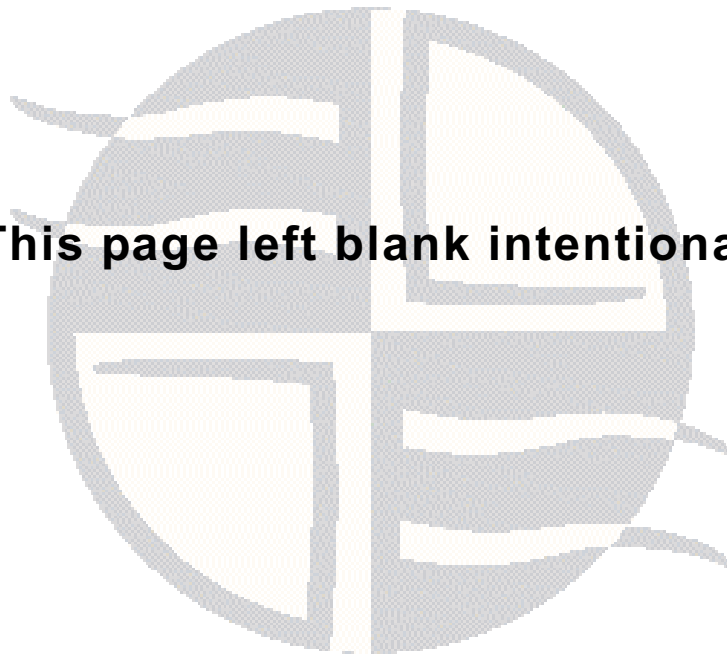
GLOSSARY

Terms and definitions

API (Application Programmer Interface)

A published set of function calls and constants allowing different programmers to utilize a ready-written library of subroutines.

Application

A program that applies the computer to solving some real-world problem. In ODBC terms, it is the program connecting to the data source.

Authorization code

You must have an authorization code for the Easysoft product you wish to license in order to obtain a purchased license. When you purchase a product your authorization code is emailed to you. You do not need an authorization code to obtain a trial license.

Client

A process performing tasks local to the current user, for example, formatting and displaying a report from data retrieved from the server.

Client/Server

The architecture whereby one process (the server) keeps track of global data, and another task (the client) is responsible for formatting and presenting the data. The client connects to the server and requests queries or actions be performed on its behalf. Often these processes run on different hosts across a local-area network.

Column

> The vertical dimension of a table. Columns are named and have a domain (or type).

Database

> A collection of data files.

Data source

> In ODBC terms, a data source is a database or other data repository coupled with an ODBC Driver, which has been given a Data Source Name (see **"DSN" on page 189**) to identify it to the ODBC Driver Manager.

Data type

> The specification of permitted values. A data type limits the values which are allowed to be used.

DBMS

> Database Management System. Software that handles access to a database.

Download

> To retrieve data from a remote machine (or the Internet) to your local machine. Mechanisms for achieving this include FTP and the World Wide Web.

Driver

> See **"ODBC driver" on page 190**.

Driver Manager

Software whose main function is to load ODBC drivers. ODBC applications connect to the Driver Manager and request a data source name (DSN). The Driver Manager loads the driver specified in the DSN's configuration file. In Windows, the ODBC Data Source Administrator is used to set up the Driver Manager.

DSN

Data Source Name. A name associated with an ODBC data source. Driver Managers, such as unixODBC or the Microsoft Windows Driver Manager, use the Data Source Name to cross-reference configuration information and load the required driver.

Field

A placeholder for a single datum in a record. For example, you can have a Surname field in a Contact Details record. Fields are sometimes referred to as cells.

FTP

File Transfer Protocol. A standard method of transferring files between different machines.

Host

A computer visible on the network.

HTTP

HyperText Transfer Protocol. The means of transferring web pages.

Middleware

Software that is placed between the client and the server to improve or expand functionality.

License key

A string which is provided by Easysoft for use in the licensing process.

ODBC

Open Data Base Connectivity. A programming interface that enables applications to access data in database management systems that use Structured Query Language (SQL) as a data access standard.

ODBC driver

Software that accesses a proprietary data source, providing a standardized view of the data to ODBC.

Row

The horizontal dimension of a table. At its most basic, a row equates to a record within a file.

Schema

A specification of the structure of a database, including the tables, their column headings and keys.

Server

A computer, or host, on the network, designed for power and robustness rather than user-friendliness and convenience. Servers typically run around-the-clock and carry central corporate data.

OR

A process performing the centralized component of some task, for example, extracting information from a corporate database.

SQL

Structured Query Language. An international standard text language for querying and manipulating databases.

System data source

In the context of ODBC under Microsoft Windows, a data source which can be accessed by any user on a given system. See also **"User data source" on page 191**.

Table

A data set in a relational database, composed of rows and columns.

TCP/IP

Transmission Control Protocol/Internet Protocol. A standard method of accessing data on different machines.

User data source

In the context of ODBC under Microsoft Windows, a data source which can only be accessed by a specific user on a given system. See also **"System data source" on page 191**.

This page left blank intentionally

INDEX