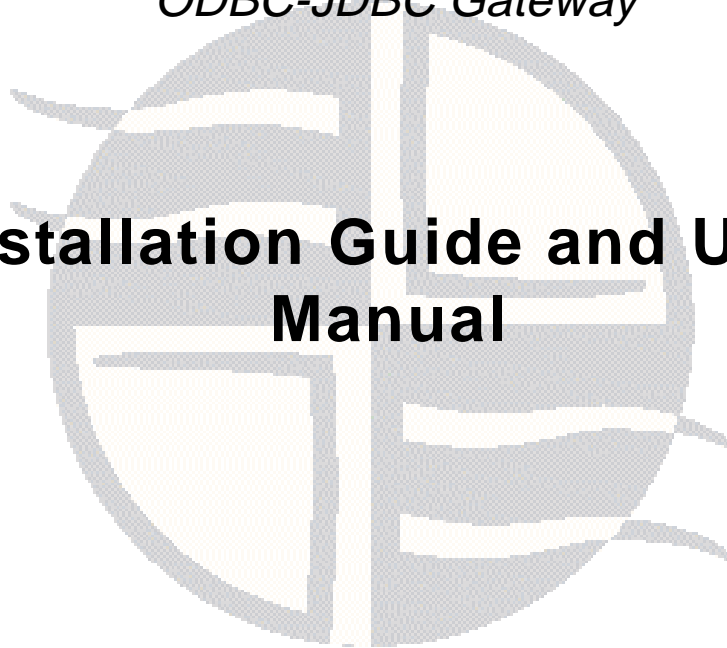


# Easysoft Data Access

*ODBC-JDBC Gateway*

## **Installation Guide and User Manual**





Version 2.x.

This manual documents version 2.x of the Easysoft ODBC-JDBC Gateway.

Publisher: Easysoft Limited

Thorp Arch Grange

Thorp Arch

Wetherby

LS23 7BA

United Kingdom

Copyright © 1993-2008 by Easysoft Limited.

All rights reserved.

You may not reverse engineer, decompile or disassemble this manual. Information in this document is subject to change without notice. Companies, names, and data used in examples are fictitious unless otherwise noted.


The names of companies referred to herein, their corporate logos, the names of their hardware and software may be trade names, trademarks or registered trademarks of their respective owners.

Easysoft and the Easysoft logo are registered trademarks of Easysoft Limited.

The software described in this document is provided under a licence agreement and may be used only in accordance with the terms of that agreement (see the [Easysoft License Agreement](#)).

# CONTENTS

<b>List of Figures</b>	.....	<b>5</b>
<b>Preface</b>	.....	<b>7</b>
	Intended Audience	8
	Displaying the Manual	8
	Notational Conventions	9
	Typographical Conventions	10
	Contents	11
	Trademarks	12
<b>Chapter 1</b>	<b>Introduction</b> .....	<b>13</b>
	Why ODBC?	14
	Driver Managers	16
	Why JDBC?	17
	The Easysoft ODBC-JDBC Gateway	18
<b>Chapter 2</b>	<b>Installation</b> .....	<b>19</b>
	Obtaining the Easysoft ODBC-JDBC Gateway	20
	What to install.	21
	Installing on Windows	23
	Uninstalling on Windows	34
	Installing on Unix	35
	Uninstalling on Unix	59

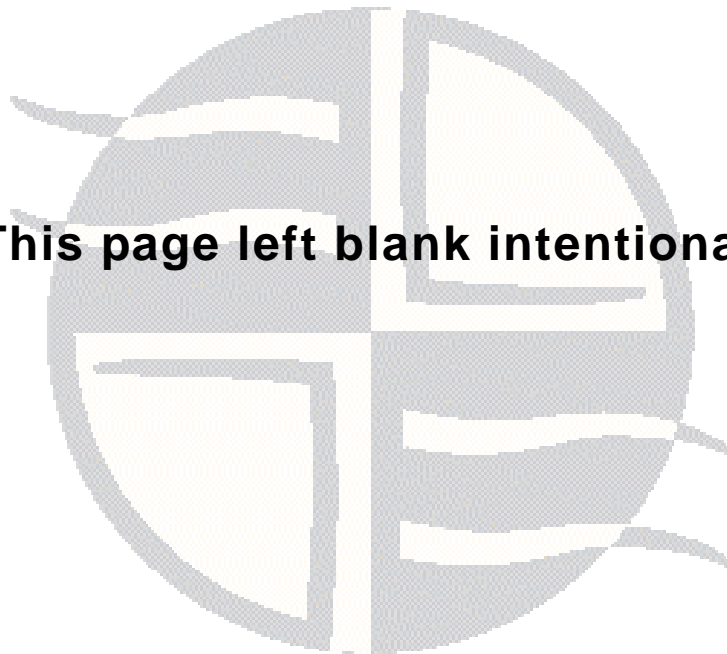


<b>Chapter 3</b>	<b>Configuration</b> .....	<b>61</b>
	DSN-less connections .....	62
	Windows Setup .....	63
	Unix Setup .....	69
	Attribute Fields .....	74
<b>Appendix A</b>	<b>Technical Reference</b> .....	<b>89</b>
	Threading .....	90
	Supported API Calls .....	90
	XA Support .....	92
	Tracing .....	95
<b>Appendix B</b>	<b>Glossary</b> .....	<b>97</b>
<b>Index</b>	.....	<b>103</b>

# LIST OF FIGURES

<b>Figure 1: Before and after ODBC</b> .....	<b>15</b>
<b>Figure 2: The Driver Manager as a dynamic linker</b> .....	<b>16</b>
<b>Figure 3: The Setup Java Interface dialog box</b> .....	<b>24</b>
<b>Figure 4: The Select JVM dialog box</b> .....	<b>25</b>
<b>Figure 5: The License Manager window</b> .....	<b>27</b>
<b>Figure 6: The ODBC Data Source Administrator Drivers tab</b> .....	<b>30</b>
<b>Figure 7: The ODBC Data Source Administrator</b> .....	<b>63</b>
<b>Figure 8: The Create New Data Source dialog box</b> .....	<b>64</b>
<b>Figure 9: The Easysoft ODBC-JDBC Gateway DSN Setup dialog box</b> .....	<b>65</b>

**This page left blank intentionally**



# PREFACE



---

## About this manual

This manual is intended for use by anyone who wants to access JDBC data sources from ODBC applications.

---

## Chapter Guide

- **Intended Audience**
- **Displaying the Manual**
- **Notational Conventions**
- **Typographical Conventions**
- **Contents**
- **Trademarks**

# PREFACE

*About this manual*

---

## 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 experience of using a Unix shell and basic functions like editing a file.

---

## 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 the Easysoft ODBC-JDBC Gateway.
- **Installation**  
Explains how to install the Easysoft ODBC-JDBC Gateway.
- **Configuration**  
Explains how to configure a data source for the Easysoft ODBC-JDBC Gateway.
- **Appendices**  
Technical Reference and Glossary.

## PREFACE

*About this manual*

---

### Trademarks

Throughout this manual, *Windows* refers generically to Microsoft Windows 95, 98, 2000, NT, XP, ME, 2003 Server or Vista, 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

---

## Introducing the Easysoft ODBC-JDBC Gateway

The Easysoft ODBC-JDBC Gateway enables ODBC applications to access JDBC data sources.

The Easysoft ODBC-JDBC Gateway is currently available on Windows, Linux and Unix platforms. The most up to date list of Easysoft ODBC-JDBC Gateway platforms is available at:

[http://www.easysoft.com/products/data\\_access/odbc\\_jdbc\\_gateway/index.html](http://www.easysoft.com/products/data_access/odbc_jdbc_gateway/index.html)

---

### Chapter Guide

- [Why ODBC?](#)
- [Driver Managers](#)
- [Why JDBC?](#)
- [The Easysoft ODBC-JDBC Gateway](#)

## INTRODUCTION

*Introducing the Easysoft ODBC-JDBC Gateway*

---

### Why ODBC?

Historically, corporate data was held on large, centralized computing resources that performed all the processing required on it. Changes to the business practice meant changes had to be made to the corporate mainframe system. Worse still was the problem of integrating two or more of these highly individual systems, for example in the event of a corporate merger.

As the desktop computer improved in power, users began to want to access corporate data in order to process it on their own desktop. The client-server architecture became a popular goal: the central computing resource (server) would produce a subset of its data for a user-friendly tool (client). The client would use desktop computing power to format and present the data.

Database application writers and their customers found themselves with a key problem: it was necessary to produce one version of a piece of software for each DataBase Management System (DBMS) they wished to use it with.

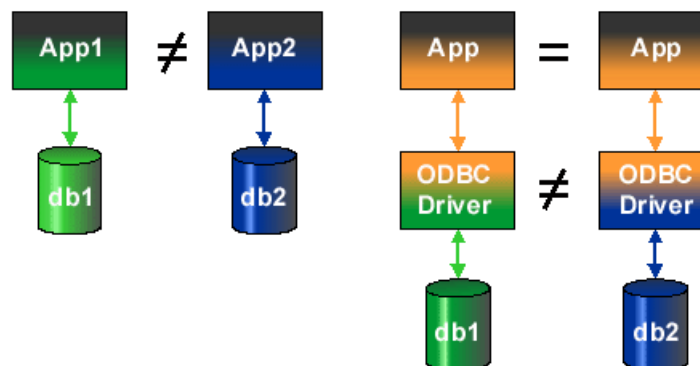
Relational databases and SQL went part way toward alleviating the problem. For the first time there was a defined, open, standard language for querying databases. In theory at least, it was possible to use the same language in dealing with databases from a variety of manufacturers. The X/Open consortium went on to define a Call Level Interface (CLI) so that programmers could effectively use SQL within their own programs.

ODBC (Open Database Connectivity) has for many years been recognised as the de facto data access method for Windows applications and ODBC drivers are available on Windows for all leading database systems.

ODBC is an API definition, compliant with ANSI SQL and the X/Open SQL CLI which allows an application to be written without considering the intricacies of the particular database engine to which it connects.

An ODBC *driver* takes care of all database-specific code, if necessary transforming the structure of the underlying system into a relational framework.

**Figure 1 on page 15** illustrates the principle of separating the driver from the application:



**Figure 1: Before and after ODBC**

Before ODBC, even if *App1* and *App2* were functionally equivalent, there had to be two programs, one for each DataBase Management System (DBMS), whereas ODBC permits the DBMS-specific parts of the program to be separated from the part that fulfils the functional requirement.

The result is that the completed application can be attached to any DBMS that has a corresponding driver. There are currently over fifty DataBase Management Systems supported in this way.

# INTRODUCTION

Introducing the Easysoft ODBC-JDBC Gateway

---

## Driver Managers

The barest ODBC system would include an ODBC-conformant driver accessing some data, and an ODBC-conformant application, linked to the driver library.

If commercial applications were distributed in this way, users would need to re-link their applications to their chosen driver whenever they wanted to access a different data source.

Instead, the application program is linked to a *driver manager*, which loads the required driver at runtime.

The driver manager also initializes the driver with a stored set of attributes.

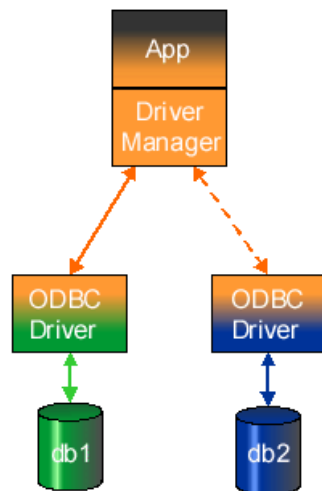


Figure 2: The Driver Manager as a dynamic linker



This approach provides three key results:

- Once developers have written applications to satisfy a business requirement, the application can be 'plugged in' to whatever database management system satisfies the technical demands.
- Administrators can connect a variety of applications (such as generic query tools) to their databases to browse and investigate the data.
- *Data access middleware* can be inserted between the ODBC application and driver to add strategic functionality such as joining heterogeneous databases into one data source or bridging a network.

---

### **Why JDBC?**

JDBC provides a call-level API for SQL-based database access and is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases.

The JDBC API makes it possible to:

- Establish a connection with a database
- Send SQL statements
- Process the results

Support for JDBC technology has been built into many new products, ensuring that customers are not locked into any proprietary architecture and can continue to access information stored on different database management systems.

No configuration is required on client machines, as all connection information is defined by the JDBC URL used to access the server.

## INTRODUCTION

*Introducing the Easysoft ODBC-JDBC Gateway*

---

### **The Easysoft ODBC-JDBC Gateway**

ODBC has for many years been recognised as the de facto data access method for Windows applications. Java has introduced an alternative standard, known as JDBC (Java DataBase Connectivity).

A growing number of database systems targeted at the Java marketplace only provide access via JDBC, thus excluding the use of popular ODBC applications such as Microsoft Excel and Crystal Reports for reporting and analysis.

The Easysoft ODBC-JDBC Gateway is the solution to this problem, a fully functional ODBC 3.5 driver which allows you to access all your existing JDBC data sources from any ODBC 2.0, 2.5, 3.0 or 3.5 compliant application.

The Easysoft ODBC-JDBC Gateway uses the JNI (Java Native Interface) to load JDBC Driver classes and to call JDBC methods in the loaded JDBC drivers.

Version 1.3x, 1.4x, 1.5x or the latest 1.6x build of the Java Virtual Machine (JVM) and the Java2 Runtime Environment (JRE) must be installed on the target machine before attempting to install the Easysoft ODBC-JDBC Gateway.

All details concerning Java downloads and installation can be found on the Sun Java web site at <http://java.sun.com>.

The driver DLL file (o2jg.dll) contains an implementation of all the ODBC API functions and a separate DLL file (o2jgs.dll) contains ODBC Setup API functions.

# INSTALLATION

---

## Installing the Easysoft ODBC-JDBC Gateway

This section explains how to install, license and remove the Easysoft ODBC-JDBC Gateway 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 the Easysoft ODBC-JDBC Gateway**
- **What to install**
- **Installing on Windows**
- **Uninstalling on Windows**
- **Installing on Unix**
- **Uninstalling on Unix**

---

**Obtaining the Easysoft ODBC-JDBC Gateway**

There are three ways to obtain the Easysoft ODBC-JDBC Gateway:

- 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 ODBC-JDBC Gateway section of the website and then choose the platform release that you require.

If you have not already done so, you will need to register at the web site to download Easysoft software.

- 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/odbc-jdbc-gateway` 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 name of the Easysoft ODBC-JDBC Gateway distribution file varies from platform to platform, but is of the form:

- `odbc-jdbc-gateway-x_y_z-platform.msi` (Windows)

– OR –

- `odbc-jdbc-gateway-x_y_z-platform.tar.gz` (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.

**NB**

Select the highest release available for your platform within your licensed major and minor version number (installing software of a different major or minor 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.

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

**Caution!**

If you do uninstall the Easysoft ODBC-JDBC Gateway before upgrading, 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).

## INSTALLATION

*Installing the Easysoft ODBC-JDBC Gateway*

Refer to the section relevant to your platform to continue:

- **"Installing on Windows" on page 23**
- **"Uninstalling on Windows" on page 34**
- **"Installing on Unix" on page 35**
- **"Uninstalling on Unix" on page 59**

---

## Installing on Windows

- Execute the file distribution that you downloaded in "**Obtaining the Easysoft ODBC-JDBC Gateway**" on page 20.

Follow the on screen instructions.

### UPDATING FILES THAT ARE IN USE

To avoid rebooting your computer, the Easysoft ODBC-JDBC Gateway installer prompts you when files that it needs to update are in use by another application or service. This frees the locked files and allows the installation to complete without a system restart.

On Windows Vista, the Easysoft ODBC-JDBC Gateway installer uses the Restart Manager to locate the applications that are using files that need updating. These applications are displayed in the Files in Use dialog box. To avoid a system restart, choose **Automatically close applications and attempt to restart them after setup is complete**. The Easysoft ODBC-JDBC Gateway installer then uses the Restart Manager to try to stop and restart each application or service in the list. If possible, the Restart Manager restores applications to the same state and with the same data that they were in before it shut them down.

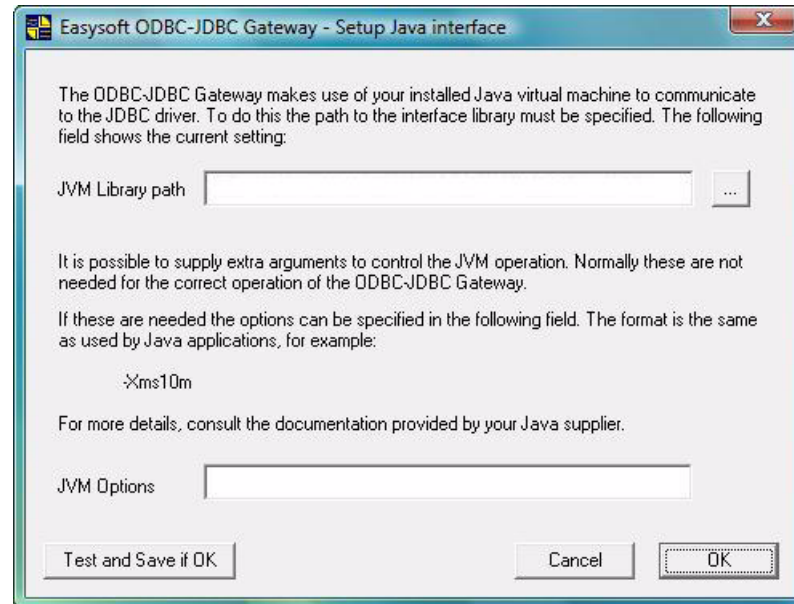
On earlier versions of Windows, when the Files in Use dialog is displayed, manually shut down each application in the list and then click **Retry** to avoid a system restart.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

#### SPECIFYING THE JVM TO USE WITH THE EASYSOFT ODBC-JDBC GATEWAY

If Setup cannot find a JVM on your computer, the **Setup Java Interface** dialog box is displayed:



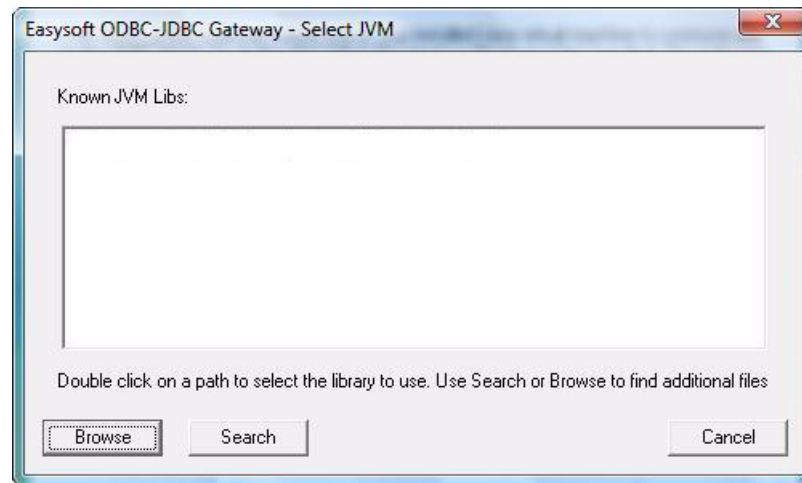
**Figure 3: The Setup Java Interface dialog box**

1. If you know the location of the JVM that you want the Easysoft ODBC-JDBC Gateway to use, in the **JVM Library Path** box, type the path of the JVM. For example:

`C:\Java\jvm.dll`

Alternatively, to browse for the JVM, click the ... button. In the **Select JVM** dialog box, click the **Browse** button. In the **Known JVM Libs** list, double-click the JVM.





**Figure 4: The Select JVM dialog box**

– OR –

If you are unsure where the JVM is installed, click the ... button. In the **Select JVM** dialog box, click the **Search** button and then browse to the directory you want to search. The search results are displayed in the **Known JVM Libs** list. In this list, double-click the JVM that you want to use. If the list is empty, search a different directory for JVMs.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

If **Setup Java Interface** is unable to find a JVM, continue with the Easysoft ODBC-JDBC Gateway Setup. You then need to download a JVM and run **Programs>Easysoft>ODBC-JDBC Gateway>Configure Java Interface** again, so that the Easysoft ODBC-JDBC Gateway knows where to find the JVM. Until you have done this, you will not be able to use the Easysoft ODBC-JDBC Gateway.

#### **NB**

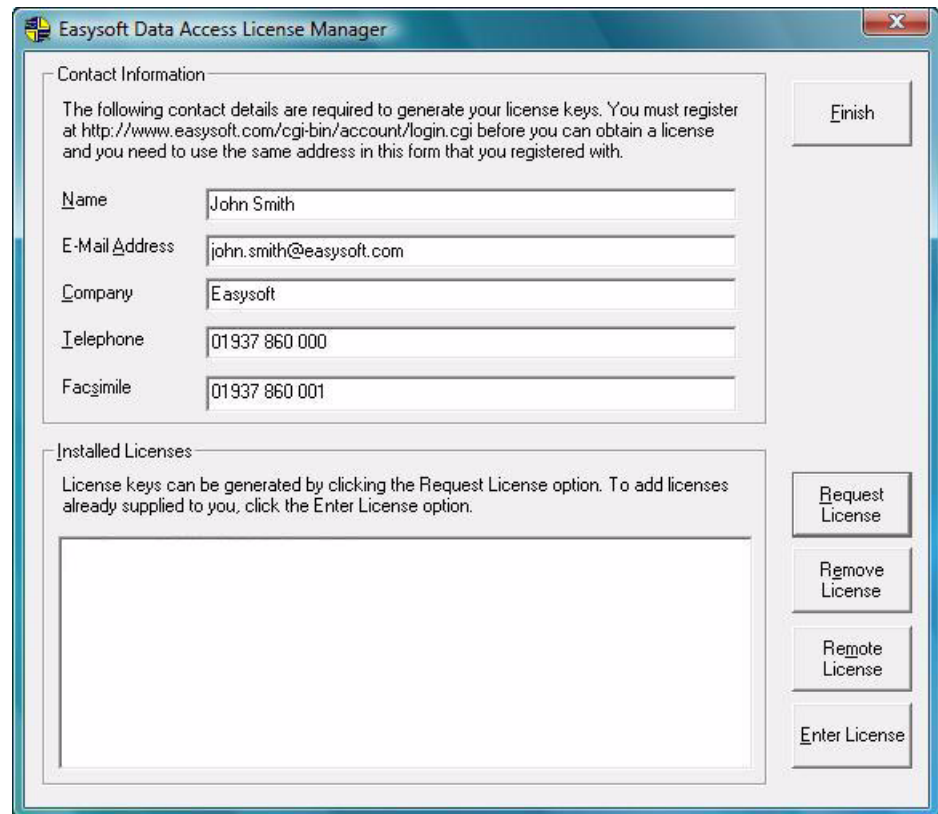
To obtain a JVM, download the JRE from either the Sun web site at <http://java.sun.com/j2se/1.4/download.html> or the IBM web site at <http://www-106.ibm.com/developerworks/java/jdk/>. The JVMs included with these JREs have been tested with the Easysoft ODBC-JDBC Gateway.

## LICENSING ON WINDOWS

The install program starts the Easysoft License Manager (documented in the [Licensing Guide](#)).

The following types of license are available:

- a *free time-limited trial license* which gives you free and unrestricted use of the product for a limited period (usually 14 days).
- a *full license* if you have purchased the product. On purchasing the product you are given an authorization code, which you use to obtain a license.



**Figure 5: The License Manager window**

1. Enter your contact details.

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.

2. Click **Request License**.

You are asked for a license type.

3. For a trial license, click **Time Limited Trial** and then click **Next**.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

The License Manager asks what software you are licensing:

Select **Easysoft ODBC-JDBC Gateway** from the drop-down list and then click **Next**.

– OR –

If you have obtained an authorization code for a purchased license, select **Non-expiring License** and then click **Next**.

The License Manager requests an authorization code.

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

The License Manager displays a summary of your details and allows you to choose the method of applying for your license.

5. Choose **On-line Request** if your machine is connected to the internet and can make outgoing connections to port 8884.

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 6 on page 29**.

<p><b>NB</b> Only your license request identifier and contact details as they are displayed in the main License Manager screen are sent to Easysoft.</p>
--

The remaining three options (**Email 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 at Easysoft's web site and your license key will be emailed to you automatically.

To use this method, click **View Request**, and then visit:

- [http://www.easysoft.com/support/licensing/trial\\_license.html](http://www.easysoft.com/support/licensing/trial_license.html)  
(trial licenses)
- [http://www.easysoft.com/support/licensing/full\\_license.html](http://www.easysoft.com/support/licensing/full_license.html)  
(purchased licenses)

In the Licensing page, enter your machine number (and authorization code for purchased license), click **Submit** and your license key will be emailed to you.

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 is displayed, telling you how many licenses have been added.

### **NB**

If you use the **Email Request** option, the license key is emailed to the email address as displayed on the License Manager main screen, not the `from:` address of your email.

6. Click **Finish** in the License Manager.

The installation is complete.

A new **Programs>Easysoft>ODBC-JDBC Gateway** program group is added with links to this manual in PDF format, the Easysoft web site, a Getting Started Guide, Licensing documentation, and the Easysoft ODBC-JDBC Gateway News Group.

## INSTALLATION

*Installing the Easysoft ODBC-JDBC Gateway*

### VALIDATING THE INSTALLATION

The version number and installation details of the Easysoft ODBC-JDBC Gateway can be verified by running the **ODBC Data Source Administrator**.

1. In **Control Panel**, double-click **Administrative Tools** and then **Data Sources (ODBC)**.

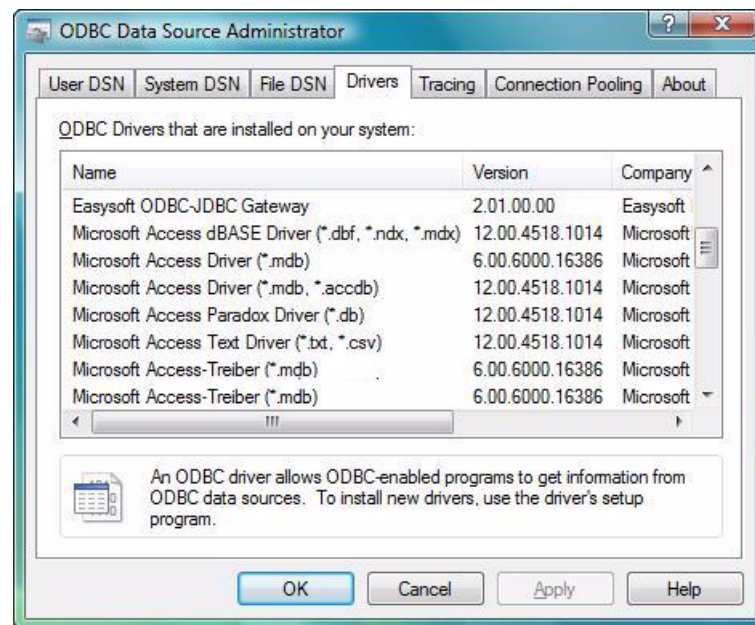
**9x**

In **Control Panel**, double-click **ODBC Data Sources (32bit)**.

**NT**

In **Control Panel**, double-click **Data Sources (ODBC)**.

2. Select the **Drivers** tab and the Easysoft ODBC-JDBC Gateway installation details are displayed:



**Figure 6: The ODBC Data Source Administrator Drivers tab**

## DEFINING WHICH JVM TO USE

The Easysoft ODBC-JDBC Gateway uses a JVM to load the target JDBC driver. When you install the Easysoft ODBC-JDBC Gateway, Setup searches for a compatible JVM. To check which JVM the Easysoft ODBC-JDBC Gateway is using, run

**Programs>Easysoft>ODBC-JDBC Gateway>Configure Java Interface.**

You should also run **Configure Java Interface** if:

- You want to use a different or newer JVM with the Easysoft ODBC-JDBC Gateway. For example, you may want to do this if you have updated your JVM since you ran Setup.
- You ran Easysoft ODBC-JDBC Gateway Setup without a JVM being installed.

1. Start **Programs>Easysoft>ODBC-JDBC Gateway>Configure Java Interface.**

The JVM that the Easysoft ODBC-JDBC Gateway is using is displayed in the **JVM Library Path** box.

If the **JVM Library Path** box is empty or you want to use a different JVM, follow these steps:

2. If you know the location of the JVM that you want the Easysoft ODBC-JDBC Gateway to use, in the **JVM Library Path** box, type the path of the JVM. For example:

```
C:\Java\jvm.dll
```

Alternatively, to browse for the JVM, click the ... button. In the **Select JVM** dialog box, click the **Browse** button. In the **Known JVM Libs** list, double-click the JVM.

– OR –

## INSTALLATION

*Installing the Easysoft ODBC-JDBC Gateway*

If you are unsure where the JVM is installed, click the ... button. In the **Select JVM** dialog box, click the **Search** button and then browse to the directory you want to search. The search results are displayed in the **Known JVM Libs** list. In this list, double-click the JVM that you want to use. If the list is empty, search a different directory for JVMs.

3. To test whether the JVM is compatible with the Easysoft ODBC-JDBC Gateway, click the **Test and Save if OK** button.

If the test results show that the JVM is incompatible and you have more than one JVM installed, click **Cancel** to exit **Configure Java Interface**. Start **Configure Java Interface**, click the ... button and then double-click a different JVM. Click the **Test and Save if OK** button to check the new JVM.

If your JVM is incompatible and you do not have another JVM installed, you need to obtain a different one

### NB

To obtain a JVM, download the JRE from either the Sun web site at <http://java.sun.com/j2se/1.4/download.html> or the IBM web site at <http://www-106.ibm.com/developerworks/java/jdk/>. The JVMs included with these JREs have been tested with the Easysoft ODBC-JDBC Gateway.



## SETTING JVM OPTIONS

Usually, you do not need to change any JVM configuration options to use a JVM with the Easysoft ODBC-JDBC Gateway. If you do need to specify any JVM settings, the Easysoft ODBC-JDBC Gateway provides the following ways to do this:

On Windows:

1. Start **Programs>Easysoft>ODBC-JDBC Gateway>Configure Java Interface**.
2. In the JVM Options box, type the JVM options.

On Unix:

- Add this entry to the [Easysoft ODBC-JDBC Gateway] section in `odbcinst.ini`:

```
JVMOPTIONS = jvmoption
```

Separate multiple JVM options with a space. For example:

```
-Xmx2048M -Xms1024M
```

For more information about the available options, see the documentation for your JVM.

JVM options specified in the **Configure Java Interface** dialog box or `odbcinst.ini` apply to all Easysoft ODBC-JDBC Gateway data sources. On Unix, you can override the global JVM options for a particular data source, by setting the `JVM_OPTIONS` environment variable. For example:

```
JVM_OPTIONS="-Xmx2048M -Xms1024M"
```

```
export JVM_OPTIONS
```

---

## **Uninstalling on Windows**

This section explains how to remove the Easysoft ODBC-JDBC Gateway from your system.

1. Do one of the following:
  - In Windows Vista, in **Control Panel**, open **Programs and Features**.
  - In earlier versions of Windows, in **Control Panel** open **Add or Remove Programs**.
2. Do one of the following:
  - In Windows Vista, double-click **Easysoft ODBC-JDBC Gateway**.
  - In earlier of Windows, select **Easysoft ODBC-JDBC Gateway** and click **Change/Remove**.

Any licenses you obtained for the Easysoft ODBC-JDBC Gateway and other Easysoft products are held in the Windows registry.

Note that your licenses are not removed when you uninstall, so the product does not need to be relicensed if you reinstall or upgrade.

---

## Installing on Unix

These instructions show how to install the Easysoft ODBC-JDBC Gateway on Unix platforms.

### **BEFORE YOU INSTALL**

#### ***Requirements***

The installation script has a minimal set of requirements:

- Bourne shell in `/bin/sh` (if your Bourne shell is not located there you may need to edit the first line of the install file).
- Various commonly used Unix commands such as:

```
grep, awk, test, cut, ps, sed, cat, wc, uname,  
tr, find, echo, sum, head, tee, id
```

If you are missing any of these commands, they can generally be obtained from the Free Software Foundation. As some machines have a broken tee command, the distribution comes with a tee replacement.

- Depending on the platform, you will need up to 10Mb of disk space free for the installed programs and up to 10Mb temporary space of the installation files themselves. If you install the unixODBC Driver Manager as well, these numbers increase by approximately 1.5Mb.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

- For Easysoft Licensing to work, you must either install in `/usr/local/easysoft` or symbolically link `/usr/local/easysoft` to wherever you chose to install the software. The installation will do this automatically for you so long as you run the installation as someone with permission to create `/usr/local/easysoft`.
- An ODBC Driver Manager. Easysoft ODBC-JDBC Gateway distributions contain the unixODBC Driver Manager but you can use an already installed unixODBC if you prefer.
- You do not have to be the `root` user to install Easysoft ODBC-JDBC Gateway but you will need permission to create a directory in the chosen installation path. Also, if you are not the `root` user it may not be possible for the installation to:
  1. Register the Easysoft ODBC-JDBC Gateway Client with unixODBC.
  2. Create the demo data source in the SYSTEM `odbc.ini` file.
  3. Update the dynamic linker entries (only some platforms).

If you are not `root`, these tasks will have to be done manually later. Easysoft recommend you install all Easysoft ODBC-JDBC Gateway components as the `root` user.

#### ***What you can install***

This Easysoft ODBC-JDBC Gateway distribution contains:

- the Easysoft ODBC-JDBC Gateway ODBC Driver
- the unixODBC Driver Manager

You will need an ODBC Driver Manager to use the Easysoft ODBC-JDBC Gateway Client from your applications. Easysoft ODBC-JDBC Gateway distributions contain the unixODBC Driver Manager (see <http://www.unixodbc.org>). Most (if not all) Unix applications and interfaces (for example Perl DBD::ODBC, PHP, Python and so on) support the unixODBC Driver Manager. You do not have to install the unixODBC Driver Manager in this distribution as you can use an already installed unixODBC (whether that was installed with another Easysoft product, from your Operating System Vendor or even if you built it yourself). However, Easysoft ensure the unixODBC distributed with Easysoft ODBC Drivers has been tested with our drivers, so we recommend you use it.

If you choose to use an already installed unixODBC Driver Manager, the installation script will attempt to locate it. The installation looks in the standard places but if you have installed it in a non-standard location you will need to provide that location to the installation script when it prompts you. The installation primarily needs unixODBC's `odbcinst` command to install drivers and any data sources.

### ***Where to install***

This installation needs a location for the installed files. The default is `/usr/local`.

At the start of the installation you will be prompted for an installation path. All files are installed in a subdirectory of your specified path called "easysoft". For example, if you pick the default of `/usr/local`, the Easysoft ODBC-JDBC Gateway will be installed in `/usr/local/easysoft` and below.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

If you choose an install path different from the default then the installation will try to symbolically link `/usr/local/easysoft` to the `easysoft` in your chosen path. This allows us to distribute binaries with built in dynamic linker run paths. If you are not `root` or the path `/usr/local/easysoft` already exists and is not a symbolic link this will fail (see later for how this may be corrected manually). You should note that you cannot license Easysoft products until `/usr/local/easysoft` exists either as a symbolic link to your chosen install path or as the install path itself.

#### ***Changes made to your system***

This installation installs files in subdirectories of the path requested at the start of the installation and depending on what is installed, a few changes may be made to your system as outlined below:

1. If you choose to install the Easysoft ODBC-JDBC Gateway into `unixODBC` then `unixODBC`'s `odbcinst` command will be run to add an entry to your `odbcinst.ini` file. You can locate this file with `"odbcinst -j"` (`odbcinst` will be in `/usr/local/easysoft/unixODBC/bin`) if you are using the `unixODBC` that comes with the Easysoft ODBC-JDBC Gateway.

The entry for Easysoft ODBC-JDBC Gateway will look similar to this:

```
[Easysoft ODBC-JDBC Gateway]
Description      = ODBC-JDBC Gateway
Driver          = /usr/local/easysoft/ojg/libo2jg.so
Setup           = /usr/local/easysoft/ojg/libo2jgS.so
JvmPath         = /usr/jdk1.3.1/jre/lib/i386/client/libjvm.so
FileUsage       = 1
DontDLClose     = 1
UsageCount      = 1
```

Obviously, the paths depend on where you installed Easysoft ODBC-JDBC Gateway and where your JVM is located. See **["Uninstalling on Unix" on page 59](#)** for ways to remove this.

## INSTALLATION

*Installing the Easysoft ODBC-JDBC Gateway*

2. The installation installs an example data source into unixODBC. This entry will be added to your `SYSTEM odbc.ini` file. You can locate your `SYSTEM odbc.ini` file using `odbcinst -j`. The entry will look similar to this:

```
[SAMPLE_OJG]
Driver           = Easysoft ODBC-JDBC Gateway
Description      = sample OJG DSN
DriverClass     = com.borland.datastore.jdbc.DataStoreDriver
User            = uid
Password        = pwd
ClassPath       = /tmp/jdserver6.jar
URL             = jdbc:borland:dsremote://server/C:\JDatastore6\bin\TEST.jds
Strip_Quote     = 1
With_Schema     = 0
Single_Statement = 0
```

See "[Uninstalling on Unix](#)" on page 59 for ways to remove this.

3. Dynamic Linker.

On operating systems where the dynamic linker has a file specifying locations for shared objects (Linux, FreeBSD), the installation will attempt to add paths under the path you provided at the start of the install to the end of this list.

On Linux, this is generally the file `/etc/ld.so.conf`.

On FreeBSD, this is generally the file `/etc/defaults/rc.conf`.



## ***Reinstalling or installing when you already have other Easysoft products installed***

Each Easysoft distribution contains common files shared between Easysoft products. These shared objects are placed in `/usr/local/easysoft/lib`. When you run an installation, the dates and versions of these files will be compared with the same files in the distribution and only updated if the files being installed are newer or have a later version number.

You should ensure that nothing on your system is using Easysoft software before starting an installation because on some platforms, files in use cannot be replaced. If a file cannot be updated, you will see a warning during the installation. You may review all warnings after the installation in the file called "warnings" in the directory you unpacked the distribution into.

If the installer detects you are upgrading a product, the installer will suggest you delete the product directory to avoid having problems with files in use. An alternative is to rename the specified directory.

If you are upgrading the Easysoft ODBC-JDBC Gateway, you will need a new license from Easysoft to use the new driver.

## ***Gathering information required during the installation***

During the installation you will be prompted for various information. Before installing you should determine:

- If you have unixODBC already installed and where it is installed. The install searches standard places like `/usr` and `/usr/local` but if you installed in a non-standard place and you don't install the unixODBC that comes with Easysoft ODBC-JDBC Gateway, you will need to know the location.

## INSTALLATION

*Installing the Easysoft ODBC-JDBC Gateway*

### INSTALLATION

#### ***Unpacking the distribution***

The Easysoft ODBC-JDBC Gateway distribution for Unix platforms is distributed as a tar file. There are multiple copies of the same distribution with different levels of compression. You unpack the distribution as follows.

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

```
gunzip odbc-jdbc-gateway-x.y.z-platform.tar.gz
```

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

```
bunzip2 odbc-jdbc-gateway-x.y.z-platform.tar.bz2
```

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

```
uncompress odbc-jdbc-gateway-x.y.z-platform.tar.Z
```

You may have a distribution file which is not compressed at all (i.e. the filename ends in `.tar`).

To extract the installation files from the tar file use:

```
tar -xvf odbc-jdbc-gateway-x.y.z-platform.tar
```

This will create a directory with the same name as the tar file (without the `.tar` postfix) containing further archives, checksum files, an install script and various other installation files.

Change directory into the directory created by unpacking the tar file.

## ***License to use***

The license text can be found in the file `license.txt` and be sure to understand the terms before continuing, as you will be required to accept the license terms at the start of the installation.

## ***Answering questions during the installation***

Throughout the installation, you will be asked to supply the answer to some questions. In each case, the default will be displayed in square brackets and you need only press `<Enter>` to take the default. If there are alternative responses, these will be shown in round brackets; to pick one of these type them and press `<Enter>`.

For example:

```
Do you want to continue? (y/n) [n]:
```

The possible answers to this question are "y" or "n". The default when you enter nothing and hit `<Enter>` is "n".

## ***Running the installer***

Before you run the installer, make sure you have read the "**Before you install**" section. If you are considering running the installation as a non `root` user, we suggest you review this carefully as you will have to get a `root` user to manually complete some parts of the installation afterwards. Easysoft recommend installing as the `root` user. (If you are concerned about the changes that will be made to your system see "**Changes made to your system**" on page 38.)

To start the installation run:

```
./install
```

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

You will need to:

- Confirm your acceptance of the license agreement with "yes" or "no". See "**License to use**" on page 43.
- Enter a location where the software is to be installed. Easysoft recommend taking the default here. See "**Where to install**" on page 37..

**NB**

If you are upgrading Easysoft ODBC-JDBC Gateway you will need a new license from Easysoft.

The next step is Locating or installing unixODBC.

#### ***Locating or installing unixODBC***

Easysoft strongly recommend you use the unixODBC Driver Manager because:

- The installation is designed to work with unixODBC and can automatically add ODBC drivers and DSNs during the install.
- Most applications and interfaces that can use ODBC know about unixODBC and so any new ODBC drivers or data sources you add with this installation will automatically become available to your applications and interfaces.
- unixODBC is currently maintained by Easysoft developer Nick Gorham. This means there is much greater experience with unixODBC within Easysoft and we will be able to provide better support for Easysoft ODBC-JDBC Gateway running under unixODBC. It also means that if you find a problem in unixODBC it is much easier for us to facilitate a fix.

- The unixODBC package contains much more than a driver manager. The aim of the unixODBC project is to provide all the ODBC functionality available on Windows for Unix operating systems. The unixODBC package may be built with the QT libraries to allow GUI configuration of DSNs and drivers. It also contains the GUI DataManager program which may be used to explore your ODBC data. The Easysoft ODBC-JDBC Gateway contains the code and shared object which is used by unixODBC's GUI ODBCConfig utility to add/delete and configure Easysoft ODBC-JDBC Gateway DSNs.

The installation will start by searching for an installed unixODBC.

There are two possible outcomes here:

1. If unixODBC is located a message will be output saying:

```
Found unixODBC under /path_to_unixODBC and it is  
version n.n.n
```

2. unixODBC is not found.

If unixODBC is not found in the standard places you will be asked whether you have it installed.

If you have it installed, you need to provide the argument given to unixODBC's configure as `--prefix`. i.e. if you built unixODBC with "configure --prefix=/usr/local/unixODBC" you enter "/usr/local/unixODBC". Generally the path required is the directory above where `odbcinst` is installed i.e. if `odbcinst` is in `/opt/unixODBC/bin/odbcinst` the required path is `/opt/unixODBC`.

If you have not got unixODBC installed, you should install the unixODBC included with Easysoft ODBC-JDBC Gateway.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

If you already have unixODBC installed you do not have to install the unixODBC included with Easysoft ODBC-JDBC Gateway but you might consider doing so if your version is older than the one included with Easysoft ODBC-JDBC Gateway.

The unixODBC in the Easysoft ODBC-JDBC Gateway distribution is not built with the default options in unixODBC's configure line:

- `--prefix=/etc`

This means the default `SYSTEM odbc.ini` file where `SYSTEM dsn`s are located will be `/etc/odbc.ini`.

- `--enable-drivers=no`

This means other ODBC drivers that come with unixODBC are not installed.

- `--enable-iconv=no`

This means unixODBC will not look for a `libiconv`. Warnings about not finding an `iconv` library were confusing our customers.

- `--enable-stats=no`

Disables unixODBC statistics which uses system semaphores to keep track of used handles. Many machines do not have sufficient semaphore resources to keep track of statistics and they are only available in the GUI ODBC Administrator anyway.

- `--enable-readline=no`

This disables readline support in `isql`.

We disabled this because it ties `isql` to the version of `libreadline` on the machine we build on. We build on as old a version of the Operating System we can for upwards compatibility. Many newer Linux machines no longer come with the older `readline` libraries and so enabling `readline` support renders `isql` unusable.

- `--prefix=/usr/local/easysoft/unixODBC`

This installs `unixODBC` into

`/usr/local/easysoft/unixODBC`.

### ***Installing the Easysoft ODBC-JDBC Gateway Driver***

The Easysoft ODBC-JDBC Gateway installation comprises of:

- Installing the Easysoft ODBC-JDBC Gateway driver.
- Registering the Easysoft ODBC-JDBC Gateway ODBC Driver with the `unixODBC` Driver Manager

If `unixODBC` is now installed (either installed by this installation or an existing copy was found) the Easysoft ODBC-JDBC Gateway ODBC Driver will be registered as an ODBC driver with the `unixODBC` Driver Manager.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

If you already have the Easysoft ODBC-JDBC Gateway registered with unixODBC, you will see a warning that Easysoft ODBC-JDBC Gateway is already registered and a list of the drivers unixODBC knows about. If you are installing the Easysoft ODBC-JDBC Gateway into a different directory than it was installed before, you will need to edit your `odbcinst.ini` file after the installation and correct the Driver and Setup paths as unixODBC's `odbcinst` will not update them if a driver is already registered.

- Creating an Easysoft ODBC-JDBC Gateway example source in unixODBC

If unixODBC is installed and you registered the Easysoft ODBC-JDBC Gateway with unixODBC, an example data source will be added to your `odbc.ini` file.

To use the example data source, you will need to update it for the JDBC driver you are using (see **"Post Installation" on page 52**).

If a data source called "ODBC\_JDBC\_SAMPLE" already exists the existing data source will be displayed and you have the option to replace it.

### ***Licensing the Easysoft ODBC-JDBC Gateway***

The program `/usr/local/easysoft/license/licshell` is used to obtain or list licenses.

Licenses are stored in the file `/usr/local/easysoft/license/licenses`. After obtaining a license, you should take a copy of this file in case something happens to it.



When you install the Easysoft ODBC-JDBC Gateway, the installation will ask you if you want to request an Easysoft ODBC-JDBC Gateway license:

```
Would you like to request a Easysoft ODBC-JDBC
Gateway license now (y/n) [y]:
```

You do not need to obtain a license during the installation, you can run `licshell` after the installation to obtain or view licenses.

If you answer yes to this the installation will run the `licshell` script. The process of obtaining a license is best described in the **Licensing Guide** and on the Easysoft web site.

To obtain a license automatically you will need to be connected to the Internet and allow outgoing connections to `license.easysoft.com` on port 8884. If you are not connected to the Internet or do not allow outgoing connections on port 8884 then the License Client can create a license request file which you can:

1. Enter in the License Request pages, accessible from the licensing links menu in this web page: <http://www.easysoft.com/support>.
2. Mail, fax or telephone to Easysoft.

Obviously, option 1 is quickest if you have a web browser and access to the Internet.

Once the License Client has started you are presented with a menu of options which allow you to:

```
[0] exit
```

```
[1] view existing license
```

```
[n] obtain a license for the desired product.
```

Obviously, if you have not got any other Easysoft products licensed then option [1] will not show any existing licenses.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

To obtain a license, select one of the options from [2] onwards for the product you are installing. The License Client will then run a program that was installed for that product which generates a key which is used to identify the product and operating system (we need this key to license you).

Once you have picked the product to license (ODBC-JDBC Gateway), you need to supply:

1. Your full name
2. Your company name
3. An email contact address. This (currently) **must** be the email address you registered on the Easysoft web site.
4. Your telephone number (you need to specify this if you telephone the license request to us).
5. Your fax number (you need to specify this if you fax the license request to us).
6. A reference number. When applying for a trial license just hit *<Enter>* on this field as this field is used to enter a reference number we will supply you for full (paid) licenses.

You will then be asked for a method of obtaining the license where the choices are:

[1] Automatically by contacting the Easysoft License Daemon (this requires connection to the Internet and the ability to support an outgoing TCPIP connection to `license.easysoft.com` on port 8884).

[2] Write information to file so you can:

a) obtain your license at

<http://www.easysoft.com/support>

Choose the type of license you require from the licensing links menu in this web page.

b) fax, telephone it.

The license request is output to `license_request.txt`.

[3] Cancel request

If you choose to obtain the license automatically, the License Client will start a TCPIP connection to `license.easysoft.com` on port 8884 and send the details you entered at the prompts above and your machine number. No other data is sent. The data sent is transmitted as plain text so if you do not want this information possibly intercepted by someone else on the net you should choose [2] and telephone or fax the request to us. The License daemon will return the license key, print it to the screen and make it available to the installation script in the file `licenses.out`.

If you choose option [2], the license request is written to the file `license_request.txt` and you should exit the License Client via option [0] and complete the installation. Once you have mailed, faxed or telephoned the license request to us, we will return a license key which should add to the end of the file `install_path/easysoft/license/licenses`.

If during this process any warnings or errors are output, please mail the output to [support@easysoft.com](mailto:support@easysoft.com) and we will rectify the problem.

## INSTALLATION

*Installing the Easysoft ODBC-JDBC Gateway*

### POST INSTALLATION

#### ***Supplied documents and examples***

The last part of the installation runs a post install script which lists resources available to you.

- Included documentation is installed in  
`/usr/local/easysoft/ojg/doc`

The Easysoft ODBC-JDBC Gateway manual (this manual) in PDF format.

`CHANGES.txt` - a list of all the changes in each Easysoft ODBC-JDBC Gateway version.

The Easysoft ODBC-JDBC Gateway license

`dlg_help.html` contains example DSN information for various JDBC drivers.

There are many resources at the Easysoft web site.

#### ***Post installation steps for non-root installations***

If you installed Easysoft ODBC-JDBC Gateway as a non-`root` user (not recommended) then there may be some manual steps you will need to perform:

1. If you attempt to install the Easysoft ODBC-JDBC Gateway ODBC Driver under the unixODBC Driver Manager and you do not have write permission to unixODBC's `odbcinst.ini` file, the Easysoft ODBC-JDBC Gateway driver cannot be added.

You can manually install the Easysoft ODBC-JDBC Gateway driver under unixODBC by adding an entry to the `odbcinst.ini` file. Run `odbcinst -j` to ascertain the `DRIVERS` file then append the lines from the `drv_template` file (in the directory where the Easysoft ODBC-JDBC Gateway distribution was untarred to) to the `odbcinst.ini` file.

2. As step 1, no example dsns can be added into unixODBC if you do not have write permission to the `SYSTEM odbc.ini` file. Run `odbcinst -j` to ascertain the name of the "SYSTEM DATA SOURCES" file then add your DSNs.
3. On machines where the dynamic linker has a configuration file defining the locations where it looks for shared objects (Linux/FreeBSD), you will need to add:

```
install_path/easysoft/lib
```

```
install_path/easysoft/unixODBC/lib
```

The latter one is only required if you installed the unixODBC included with Easysoft ODBC-JDBC Gateway. Sometimes after changing the dynamic linker configuration file you need to run a program to update the dynamic linker cache (e.g. `/sbin/ldconfig` on Linux).

4. If you did not install Easysoft ODBC-JDBC Gateway in the default location then you need to link `/usr/local/easysoft` to the `easysoft` directory in your chosen install path.

## INSTALLATION

*Installing the Easysoft ODBC-JDBC Gateway*

e.g. If you installed in `/home/martin` the installation will create `/home/martin/easysoft` and you need to symbolically link `/usr/local/easysoft` to `/home/martin/easysoft`:

```
ln -s /home/martin/easysoft /usr/local/easysoft
```

5. If your system does not have a dynamic linker configuration file you need to add the paths listed in step **3** above to whatever environment path the dynamic linker uses to locate shared objects. You may want to amend this in a system file run whenever someone logs in like `/etc/profile`.

The environment variable differs per dynamic linker. Consult your `ld` or `ld.so` man page. It is usually:

`LD_LIBRARY_PATH`, `LIBPATH`, `LD_RUN_PATH` or `SHLIB_PATH`.

## ***Defining which JVM to use***

After a typical installation, the Easysoft ODBC-JDBC Gateway will be accessible in `/usr/local/easysoft/ojg`. The installation will also have installed:

- the unixODBC driver manager (or used your existing copy)
- installed Easysoft ODBC-JDBC Gateway as an ODBC driver under unixODBC by adding an entry to your `odbcinst.ini` file.
- installed an example Easysoft ODBC-JDBC Gateway data source in your system `odbc.ini`.

### **NB**

The unixODBC included with Easysoft ODBC-JDBC Gateway has `sysconfdir` set to `/etc`. This means if you use the included unixODBC your `odbcinst.ini` and `odbc.ini` files (defining ODBC drivers and system DSNs) are in `/etc`. If you built unixODBC yourself or are using a copy installed from RPM etc, it is more likely that `sysconfdir` was set to `/usr/local/etc` when unixODBC was built. You can check by running `odbcinst -j`

If you did not obtain a license during the installation you need to do this now (you need to run `/usr/local/easysoft/license/licshell` and see the license instructions in "**Licensing the Easysoft ODBC-JDBC Gateway**" on page 48.

## INSTALLATION

*Installing the Easysoft ODBC-JDBC Gateway*

Currently the installation installs the Easysoft ODBC-JDBC Gateway driver in `odbcinst.ini` with a default `JvmPath`. e.g.

```
[Easysoft ODBC-JDBC Gateway]
Description      = ODBC JDBC Gateway
Driver           = /usr/local/easysoft/ojg/libo2jg.so
Setup            = /usr/local/easysoft/ojg/libo2jgS.so
JvmPath          = /usr/jdk1.3.1/jre/lib/i386/client/libjvm.so
FileUsage        = 4
DontDLClose     = 1
```

You need to change `JvmPath` to point at a valid JVM on your system. You may also need to ensure `LD_LIBRARY_PATH/LIBPATH` etc are pointing to any directories containing the JVM shared objects and JVM required shared objects or the dynamic linker knows where to find dependent shared objects. e.g. With JDK 1.4.1 for Linux, `libjvm.so` loads `libverify` which is found in the `jre/lib/i386` directory.

As an example:

```
LD_LIBRARY_PATH=/usr/local/j2sdk1.4.1_06/jre/lib/i
386/client:/usr/local/j2sdk1.4.1_06/jre/lib/i386
```

- `export LD_LIBRARY_PATH.`

**NB**

Usually, you will not need to change any JVM configuration options to use a JVM with the Easysoft ODBC-JDBC Gateway. If you do need to specify any JVM settings, refer to "[Setting JVM Options](#)" on page 33.



### ***Editing the example DSN for your JDBC Driver***

The sample ODBC data source called ODBC\_JDBC\_SAMPLE is installed into your system `odbc.ini` file. You need to edit this DSN or create a different named one with the correct `DriverClass`, `User`, `Password`, `ClassPath` and `URL` for your JDBC driver. The sample one looks similar to this:

```
[ODBC_JDBC_SAMPLE]
Driver           = Easysoft ODBC-JDBC Gateway
DriverClass     = com.borland.datastore.jdbc.DataStoreDriver
User            = dbo
Password        = password
ClassPath       = /tmp/jdsserver6.jar
URL             = jdbc:borland:dsremote://server/C:\JDatastorein\TEST.jds
Strip_Quote     = 1
With_Schema     = 0
```

"`Driver = Easysoft ODBC-JDBC Gateway`" tells unixODBC which driver to use i.e. which entry in `odbcinst.ini` file defines the ODBC driver.

"`DriverClass`" defines the Java class you need to use for the JDBC driver defined in "`ClassPath`".

"`URL`" is the required URL to connect to your JDBC database.

You can find some examples for different JDBC drivers in the file `dlg_help.html`.

## INSTALLATION

### *Installing the Easysoft ODBC-JDBC Gateway*

#### ***Testing the Easysoft ODBC-JDBC Gateway ODBC Driver***

To test the Easysoft ODBC-JDBC Gateway you need to complete all the processes described in **"Post Installation" on page 52**. In particular, you need to edit the sample DSN or create a new one.

Once your DSN is defined you can use unixODBC's `isql` to try a connection.

e.g.

Suppose the DSN defined in `odbc.ini` is `ODBC_JDBC_SAMPLE` and your JDBC database needs a username and password of `testuser/password` then:

```
cd /usr/local/easysoft/unixODBC/bin (or wherever your  
unixODBC's isql is)
```

```
./isql -v ODBC_JDBC_SAMPLE testuser password
```

If this connects you will get a prompt you can start typing SQL into or just "help" to get a list of tables. Pressing `<enter>` on a new line will exit `isql`.

---

## Uninstalling on Unix

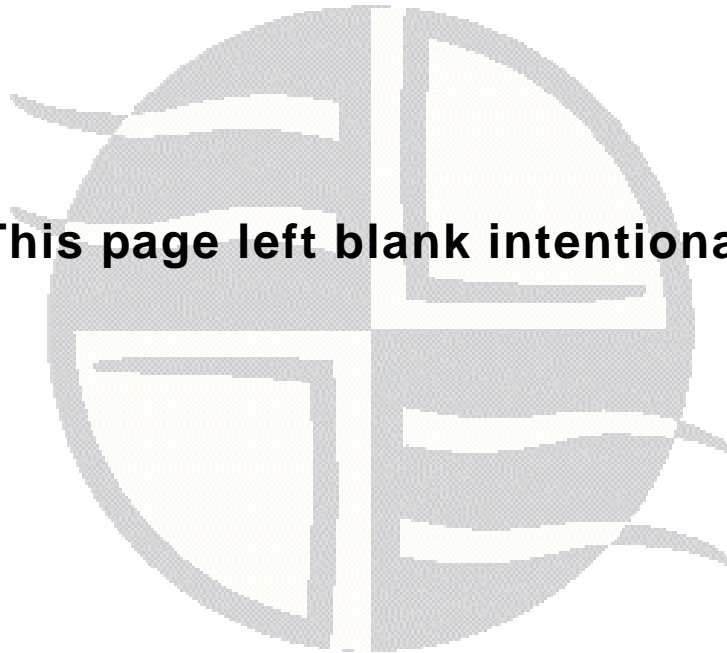
There is no automated method of removal of the Easysoft ODBC-JDBC Gateway in this release. However, removal is quite simple. Follow the instructions below:

1. Change directory to *installpath/easysoft* and delete the directory tree "ojg" (where *installpath* is what was specified at install time).
2. If you had to add the above path to the dynamic linker search paths (e.g. */etc/ld.so.conf* in Linux) remove this entry. You may have to run a linker command such as */sbin/ldconfig* to get the dynamic linker to reread the configuration file. This step can generally only be performed by the `root` user.
3. If you were using unixODBC, the Easysoft ODBC-JDBC Gateway ODBC Driver entry needs to be removed from the `odbcinst.ini` file. To check whether Easysoft ODBC-JDBC Gateway is configured under unixODBC use `odbcinst -q -d`. If this outputs "Easysoft ODBC-JDBC Gateway" you should uninstall the driver using  

```
odbcinst -u -d -n "Easysoft ODBC-JDBC Gateway"
```

If a message is displayed saying reduced usage count, repeat these commands until `odbcinst` states the driver is removed.
4. If you created any Easysoft ODBC-JDBC Gateway data sources under unixODBC, you may want to delete these. You need to use `odbcinst -j` to locate user and system `odbc.ini` files and then check those files for DSNs having the driver attribute set to "Easysoft ODBC-JDBC Gateway".
5. Remove `ojg_install.info` from the `/usr/local/easysoft` directory.

**This page left blank intentionally**



# CONFIGURATION

---

## Configuring the Easysoft ODBC-JDBC Gateway

This section describes how to create an ODBC data source for the Easysoft ODBC-JDBC Gateway.

---

### Chapter Guide

- [DSN-less connections](#)
- [Windows Setup](#)
- [Unix Setup](#)
- [Attribute Fields](#)

---

## **DSN-less connections**

In addition to using a data source, you can also connect to a database by using a DSN-less connection string of the form:

```
SQLDriverConnect(..."DRIVER={Easysoft ODBC-JDBC Gateway};DRIVERCLASS=com.borland.datastore.jdbc.DataStoreDriver;UID=dbuser;PWD=dbpass;CLASSPATH=/tmp/jdsserver6.jar;URL=jdbc:borland:dsremote://server/C:\JDatastore6\bin\TEST.jds"...)
```

In this JDataStore example:

- DRIVERCLASS specifies the JDBC driver class name.
- UID and PWD specify the username and password needed to access the database.
- CLASSPATH specifies the .jar files required to enable Java to load the JDBC driver.
- URL specifies the URL required to connect to the database.

Other Easysoft ODBC-JDBC Gateway attribute settings can be added to this connection string, using the "PARAMETER=value;" format. These attributes are described in "[Attribute Fields](#)" on [page 74](#).

---

## Windows Setup

This section describes how to create ODBC data sources for the Easysoft ODBC-JDBC Gateway on Windows:

1. In **Control Panel**, double-click **Administrative Tools** and then **Data Sources (ODBC)**.

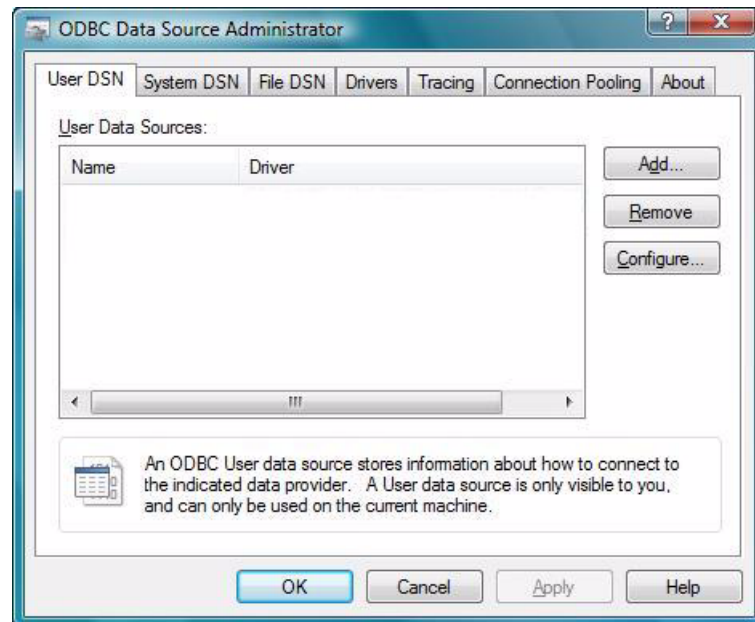
**9x**

In **Control Panel**, double-click **ODBC Data Sources (32bit)**.

**NT**

In **Control Panel**, double-click **Data Sources (ODBC)**.

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



**Figure 7: The ODBC Data Source Administrator**

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

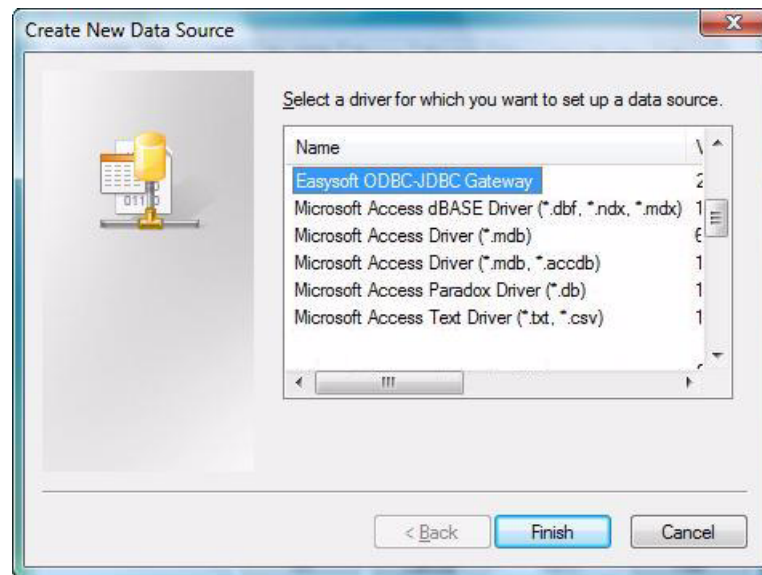
2. Select the **User DSN** tab to set up a data source that only you can access.

– 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:

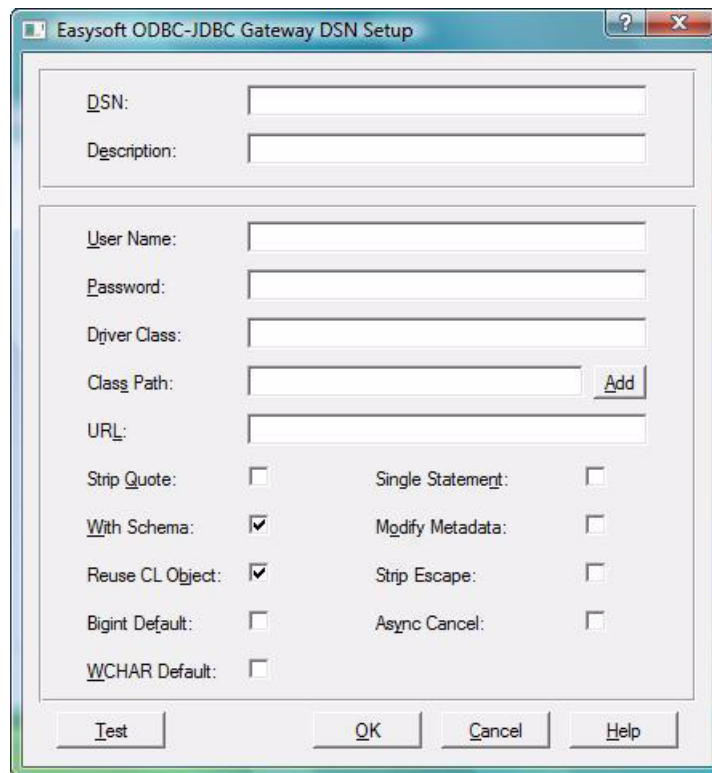


**Figure 8: The Create New Data Source dialog box**

4. Select `Easysoft ODBC-JDBC Gateway` and click **Finish**.



The **Easysoft ODBC-JDBC Gateway DSN Setup** dialog box prompts the user to create an ODBC data source for a specific JDBC driver using the appropriate connection information:



**Figure 9: The Easysoft ODBC-JDBC Gateway DSN Setup dialog box**

The following fields are defined on the **Easysoft ODBC-JDBC Gateway DSN Setup** dialog box:

## DSN

The name of the data source.

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

### DESCRIPTION

A brief description of the data source.

### USER NAME

The name of a user to pass to the JDBC driver in order to access the database (if required).

### PASSWORD

The password for the specified user to pass to the JDBC driver in order to access the database (if required).

### DRIVER CLASS

The JDBC driver class name for this data source that will be passed to the `forName()` function.

#### NB

The syntax and contents of this field are unique to the JDBC driver being used and can be obtained from the individual driver documentation.

An example value (in this case for the **Easysoft JDBC-ODBC Bridge**) would be the string:

```
easysoft.sql.jobDriver
```

### CLASS PATH

A list of any `.jar` files required to enable Java to load the JDBC driver class.

This list is specified in the same way as the Java `CLASSPATH` environment variable, with the separator between entries depending on the operating system used (Windows uses a semi-colon ";" and other platforms use a colon ":").

Click **Add** to launch a file browser to help you to locate the required `.jar` file on your system.

If the target driver requires multiple `.jar` files then click **Add** again to add a further entry.

**Caution!** The order in which the `.jar` files are specified can be significant.

**REF** Please refer to <http://java.sun.com/j2se/1.4/docs/tooldocs/windows/classpath.html> on the Sun Java web site for further information about the setting of class paths under Windows.

### URL

The URL required to connect to your JDBC database.

**NB** The syntax and contents of this field are unique to the JDBC driver being used and can be obtained from the individual driver documentation.

An example value (in this case for the **Easysoft JDBC-ODBC Bridge**) would be the string:

```
jdbc:easysoft://server:8831/dsn_name
```

To display some examples for different JDBC drivers, click **Help**.

For details of the other attributes that can be set in this dialog box, see "**Attribute Fields**" on page 74

### ADD

Displays a dialog box that allows a `.jar` file to be added to the **Class Path** field.

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

### TEST

Takes the values for all Easysoft ODBC-JDBC Gateway Data Source settings and attempts to establish a connection to the specified JDBC data source.

It will report either the first error encountered or a successful completion of the test, and will time out after about 15 seconds.

Click **OK** to exit the Easysoft ODBC-JDBC Gateway **Data Source Setup** program after a successful connection has been made to ensure that those connection settings are saved.

### OK

Creates or modifies an Easysoft ODBC-JDBC Gateway ODBC **Data Source Setup** with the currently displayed settings.

### CANCEL

Exits the Easysoft ODBC-JDBC Gateway ODBC **Data Source Setup** program without affecting any settings.

### HELP

Displays help information.

---

## Unix Setup

### DATA SOURCE ATTRIBUTES

There are two options when setting up a data source to your data:

- 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 "[Windows Setup](#)" on page 63).

**NB**

If you have user and system data sources with the same name, the driver manager will use the user data source.

The default Easysoft ODBC-JDBC Gateway installation will create a system DSN named `ODBC_JDBC_SAMPLE`. If you are using `unixODBC` included with this distribution, the system `odbc.ini` file will be `/etc`.

If you have built `unixODBC` yourself, or installed it from some other source, system data sources are stored in the path specified with the configure option `--sysconfdir=directory`. If `sysconfdir` was not specified when `unixODBC` was configured and built it defaults to `/usr/local/etc`.

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

User data sources are stored in the file `.odbc.ini` in the current user's home directory (e.g. `$HOME/.odbc.ini`).

**NB** By default, you must be logged in as `root` to edit a system data source defined in `/etc/odbc.ini`.

**NB** `odbcinst -j` will show you where the driver manager stores system and user data sources.

You may either change some of the initial configuration parameter values for the sample data source or add extra data sources.

Each section of the `odbc.ini` file starts with a data source name in square brackets `[ ]` followed by a number of *attribute=value* pairs.

**NB** Attribute names in `odbc.ini` are not case sensitive.

For the addition of a data source, all the settings in the following example must be appended to the relevant `odbc.ini` file. The `Driver` attribute names the ODBC driver in the `odbcinst.ini` file to use for this data source. When the Easysoft ODBC-JDBC Gateway is installed into `unixODBC` it places an `Easysoft ODBC-JDBC Gateway` entry into the `odbcinst.ini` file so you should always have `Driver = Easysoft ODBC-JDBC Gateway` in your Easysoft ODBC-JDBC Gateway data sources.

**NB** More details of configuring data sources with `unixODBC` can be found at [www.unixodbc.org](http://www.unixodbc.org).

To configure your Easysoft ODBC-JDBC Gateway DSN in your `odbc.ini` file, you will need to edit:

- The Java class you need to use for the JDBC driver defined in "ClassPath" (`DriverClass`).

- The user name needed to access the database (if required) (User).
- The password for the specified user. (Password).
- A list of any .jar files required to enable Java to load the JDBC driver class (ClassPath).

This list is specified in the same way as the Java CLASSPATH environment variable. If the target driver requires multiple .jar, separate each .jar file in the list with a colon (":").

**Caution!** The order in which the .jar files are specified can be significant.

For more information about setting the class path on Unix, see <http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/classpath.html>.

- The URL required to connect to your JDBC database (URL).

You can find some examples for different JDBC drivers in `/usr/local/easysoft/ojg/doc/dlg_help.html`.

For example, this sample data source has been modified to connect to the hsqldb JDBC driver:

```
[ODBC_JDBC_SAMPLE]
Driver = Easysoft ODBC-JDBC Gateway
DriverClass = org.hsqldb.jdbcDriver
ClassPath = /tmp/hsqldb.jar
URL = jdbc:hsqldb:hsqldb://my_hsql_database_server
With_Schema = 0
```

Other optional attribute values may be set in the `odbc.ini` file, and are described in "**Attribute Fields**" on page 74.

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

### DRIVER ATTRIBUTES

There are a number of driver attributes that are stored in the `odbcinst.ini` file. By default, this file is installed in `/etc`. If you have built unixODBC yourself, then it will be whatever path you specified in the `sysconfdir=directory` configure option (if `sysconfdir` has not been specified then the path will default to `/usr/local/etc`).

**NB** By default, you must be logged in as `root` to edit a system data source defined in `/etc/odbcinst.ini`.

- **JvmPath**

You need to set `JvmPath` to point at a valid JVM on your system. For more information, see ["Defining which JVM to use" on page 55](#).

- **Threaded**

If the applications using the driver does not use threads, you can set `Threads` to be 0. This will provide a performance enhancement, by disabling thread safety in the driver.

### ENVIRONMENT

The Easysoft ODBC-JDBC Gateway must also be able to find the following shared objects that it installs:

- `libodbcinst.so`

By default, this will always be located in `/usr/local/easysoft/unixODBC/lib/`:

- `libeslicshr_r.so` or `libeslicshr.so`

By default, this will always be located in `/usr/local/easysoft/lib/`



- `libessupp_r.so` or `libessupp.so`

By default, this will always be located in  
`/usr/local/easysoft/lib`

It may be necessary to set up and export

`LD_LIBRARY_PATH/LIBPATH` etc.(depending on the operating system and run-time linker) to include the directories where `libodbcinst.so`, `libeslicshr.so` and `libeslicshr.so` are located.

**NB** The shared object file extension (`.so`) varies across operating systems ( `.so`, `.a`, or `.sl`).

You may also need to ensure `LD_LIBRARY_PATH/LIBPATH` etc. are pointing to any directories containing the JVM shared objects and JVM required shared objects or the dynamic linker knows where to find dependent shared objects. For more information, see "[Defining which JVM to use](#)" on page 55.

### ESTABLISHING A TEST CONNECTION

Run the `isql` query tool to prove that the Easysoft ODBC-JDBC Gateway is working:

1. Change directory into

```
/usr/local/easysoft/unixODBC/bin
```

2. Type `./isql -v DSN`

where *DSN* is the name of the target data source

3. Enter SQL commands to query the database, such as:

```
SQL> select table_name from user_tables
```

4. or enter 'help' to return a list of tables

```
SQL>help
```

## CONFIGURATION

Configuring the Easysoft ODBC-JDBC Gateway

---

### Attribute Fields

This section lists the attributes which can be set for the Easysoft ODBC-JDBC Gateway in a table showing:

- The label of the attribute on the Windows dialog box.
- The entry required when editing the Unix `odbc.ini` file.
- The string to be used in a call to `SQLDriverConnect` or in a connect string for ADO type use.

Attributes which are text fields are displayed as “value”.

Attributes which are logical fields can contain either 0 (to set to off) or 1 (to set to on) and are displayed as “0 | 1”.

### DSN

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

Interface	Value
DSN Dialog Box (Windows)	DSN
odbc.ini file (Unix)	[value]
Connect String	DSN=value

**DESCRIPTION**

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

Interface	Value
DSN Dialog Box (Windows)	Description
odbc.ini file (Unix)	Description= <i>value</i>
Connect String	Not Used

**USER NAME**

The name of a user to pass to the JDBC driver in order to access the database (if required).

The attribute value is passed to the JDBC driver as part of a `Properties` object in the connect call.

The equivalent Java code is as follows:

```
Connection con;  
Properties prop;  
if (  
    != null ) {  
        prop.put( "user", uid );  
    }  
con = DriverManager.getConnection( url, p );
```

This value can be overridden at the ODBC level by passing either a non-NULL user name in the call to `SQLConnect()` or a `UID=value` attribute in the connection string passed to `SQLDriverConnect()`.

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

It can also be supplied in the JDBC URL supplied to the JDBC driver, but the syntax of this is JDBC driver dependent..

Interface	Value
DSN Dialog Box (Windows)	User Name
odbc.ini file (Unix)	User= <i>value</i>
Connect String	UID= <i>value</i>

### PASSWORD

The password for the specified user to pass to the JDBC driver in order to access the database (if required).

The attribute value is passed to the JDBC driver as part of a `Properties` object in the connect call.

The equivalent Java code is as follows:

```
Connection con;
Properties prop;
if ( pwd != null ) {
    prop.put( "password", pwd );
}
con = DriverManager.getConnection ( url, p );
```

This value can be overridden at the ODBC level by passing either a non-NULL password in the call to `SQLConnect()` or a `PWD=value` attribute in the connection string passed to `SQLDriverConnect()`.

It can also be supplied in the JDBC URL supplied to the JDBC driver, but the syntax of this is JDBC driver dependent..

Interface	Value
DSN Dialog Box (Windows)	Password
odbc.ini file (Unix)	Password= <i>value</i>
Connect String	PWD= <i>value</i>

## DRIVER CLASS

The JDBC driver class name for this data source that will be passed to the `forName()` function.

The equivalent Java code is as follows:

```
Class.forName( driverClass );
```

The syntax and contents of this field are unique to the JDBC driver being used and can be obtained from the individual driver documentation.

**NB**

An example value (in this case for the **Easysoft JDBC-ODBC Bridge**) would be the string:

```
easysoft.sql.jobDriver
```

Interface	Value
DSN Dialog Box (Windows)	Driver Class
odbc.ini file (Unix)	DriverClass= <i>value</i>
Connect String	DRIVERCLASS= <i>value</i>

## CONFIGURATION

Configuring the Easysoft ODBC-JDBC Gateway

### CLASS PATH

A list of any `.jar` files required to enable Java to load the JDBC driver class.

This list is specified in the same way as the Java `CLASSPATH` environment variable, with the separator between entries depending on the operating system used (Windows uses a semi-colon ";" and other platforms use a colon ":").

#### REF

Please refer to

<http://java.sun.com/j2se/1.4/docs/tooldocs/windows/classpath.html> on the Sun Java web site for further information about the setting of class paths under Windows. For more information about setting the class path on Unix, see <http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/classpath.html>.

Interface	Value
DSN Dialog Box (Windows)	Class Path
odbc.ini file (Unix)	ClassPath= <i>value</i>
Connect String	CLASSPATH= <i>value</i>

### URL

The URL required to connect to your JDBC database (URL).

The URL string will be passed as the first argument to the call:

```
Connection con;
```

```
con = DriverManager.getConnection( url, p );
```

## NB

The syntax and contents of this field are unique to the JDBC driver being used and can be obtained from the individual driver documentation.

An example value (in this case for the **Easysoft JDBC-ODBC Bridge**) would be the string:

```
jdbc:easysoft://server:8831/dsn_name
```

Interface	Value
DSN Dialog Box (Windows)	URL
odbc.ini file (Unix)	URL= <i>value</i>
Connect String	URL= <i>value</i>

## STRIP QUOTE

Controls how any SQL passed through the Easysoft ODBC-JDBC Gateway is scanned and modified.

Some JDBC drivers can not accept double quotes around column and table names, and in some cases the quotes will trigger failure, as it may invoke unwanted case-sensitive behaviour.

As an example, when `Strip Quote` is ON (i.e. set to 1 or checked), the following SQL:

```
SELECT "A" FROM "T"
```

will, when seen by the JDBC driver, be transformed to:

```
SELECT A FROM T
```

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

Interface	Value
DSN Dialog Box (Windows)	Strip Quote
odbc.ini file (Unix)	Strip_Quote=0 1
Connect String	STRIP_QUOTE=0 1

The default for `Strip Quote` is OFF (0).

### SINGLE STATEMENT

Some JDBC drivers are only capable of using a single active result set, but may return a value of either zero or greater than one from `DatabaseMetaData.getMaxStatements()`.

When ON (i.e. set to 1 or checked), the `Single Statement` attribute forces the ODBC driver to return a value of 1 from the `SQLGetInfo` call to find `SQL_MAX_CONCURRENT_ACTIVITIES`.

Interface	Value
DSN Dialog Box (Windows)	Single Statement
odbc.ini file (Unix)	Single_Statement=0 1
Connect String	SINGLESTMT=0 1

The default for `Single Statement` is OFF (0).

### WITH SCHEMA

Controls the return of schema information from ODBC metadata calls.



Some JDBC drivers return schema information from the result set generated by `getTables()` (for example), but are unable to handle SQL that contains the schema information in table and column specifications.

The default for `With Schema` is ON (1). When OFF (i.e. set to 0 or unchecked), schema information is returned to the ODBC application, allowing it to assemble the relevant SQL.

Interface	Value
DSN Dialog Box (Windows)	With Schema
odbc.ini file (Unix)	With_Schema=0 1
Connect String	SCHEMA=0 1

### **MODIFY METADATA**

Some JDBC drivers only return a partial result set from calls such as `DatabaseMetaData.getTypeInfo()`.

When ON (i.e. set to 1 or checked), this attribute causes the ODBC driver to modify the values in the result set returned from metadata calls (such as `SQLGetTypeInfo`) so that they conform to the expected ODBC3 result set specification.

Interface	Value
DSN Dialog Box (Windows)	Modify Metadata
odbc.ini file (Unix)	Clean_Metadata=0 1
Connect String	CLEAN_METADATA=0 1

The default for `Modify Metadata` is OFF (0).

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

### REUSE CL OBJECT

When loading a JDBC driver, the Gateway creates a `java.net.URLClassLoader` and this is then used to load the JDBC driver.

When ON (i.e. set to 1 or checked), the `URLClassLoader` object is retained between calls to `SQLConnect` and `SQLDriverConnect`, thus enabling a faster connection and reducing the system resources required.

This attribute should only be disabled (i.e. set to 0 or unchecked) if problems are encountered connecting to multiple different JDBC drivers.

Interface	Value
DSN Dialog Box (Windows)	Reuse CL Object
odbc.ini file (Unix)	ReuseCL=0 1
Connect String	REUSECL=0 1

The default for `Reuse CL Object` is ON (1).

### STRIP ESCAPE

While some JDBC drivers can accept ODBC-type escape sequences, some JDBC drivers are unable to understand them.

When ON (i.e. set to 1 or checked), this attribute allows the ODBC driver to modify the SQL passed to the JDBC driver, removing the ODBC sequences.

For example, the following SQL when passed to the ODBC driver:

```
SELECT * FROM T WHERE DATE_FIELD = {d '1999-02-01'}
```

will be altered to:

```
SELECT * FROM T WHERE DATE_FIELD = '1999-02-01'
```

when this attribute is enabled.

Interface	Value
DSN Dialog Box (Windows)	Strip Escape
odbc.ini file (Unix)	Strip_Escape=0 1
Connect String	STRIP_ESCAPE=0 1

The default for `Strip Escape` is OFF (0).

### BIGINT DEFAULT

The ODBC specification states that if `SQL_C_DEFAULT` is used in combination with `SQL_BIGINT` fields then the result returned will be in `SQL_C_BIGINT` format.

When ON (i.e. set to 1 or checked), a `CHAR` string is returned under these conditions, as some applications (Microsoft Access in particular) do not know of the `SQL_BIGINT` data type and therefore expect the data to be returned as a string.

Interface	Value
DSN Dialog Box (Windows)	Bigint Default
odbc.ini file (Unix)	Bigint2Char=0 1
Connect String	BIGINT2CHAR=0 1

The default for `Bigint Default` is OFF (0).

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

### ASYNC CANCEL

By default, the Easysoft ODBC-JDBC Gateway will not call `PreparedStatement.cancel()` and `ResultSet.Cancel()` when the ODBC function `SQLCancel` is called. Several JDBC Drivers will fail if this is done in separate threads, which the ODBC specification allows.

When ON (i.e. set to 1 or checked), the Easysoft ODBC-JDBC Gateway will call the `cancel()` methods.

Interface	Value
DSN Dialog Box (Windows)	Async Cancel
odbc.ini file (Unix)	Async_Cancel=0 1
Connect String	ASYNCCANCEL=0 1

The default for `Async Cancel` is OFF (0).

### WCHAR DEFAULT

The ODBC specification allows applications to request the format in which data is returned, and for each SQL datatype there is a default type. The specification says that for a wide character field, the default return type is a `SQL_WCHAR`, a Unicode data type.

By default, the Easysoft ODBC-JDBC Gateway will map a `SQL_WCHAR` column when requested as a `SQL_DEFAULT` to a `SQL_WCHAR`, in accordance with the ODBC specification.

However, for tables with one or more `WCHAR` fields that are part of the primary key of that table, Microsoft Access expects that conversion to result in `SQL_CHAR` data. The symptom for this is that the table opens, but displays `#deleted` in all the fields.

When ON (i.e. set to 1 or checked), the Easysoft ODBC-JDBC Gateway uses the `SQL_CHAR` conversion for `SQL_DEFAULT` requests on a `SQL_WCHAR` type. This is the conversion that Access expects.

This option is a workaround for Access. Enabling `WCHAR Default` causes the Easysoft ODBC-JDBC Gateway to behave in a non-standard way and may cause problems with other applications. If this is the case, create a separate data source for use with Access and only enable `WCHAR Default` in that data source.

Interface	Value
DSN Dialog Box (Windows)	WCHAR Default
odbc.ini file (Unix)	WcharDefaultC=0 1
Connect String	WCHARDEFAULTC=0 1

The default for `WCHAR Default` is OFF (0).

### **XA\_ENLIST**

When ON (i.e. set to 1), the Easysoft ODBC-JDBC Gateway uses the XA interface to access the JDBC driver. This makes the XA connection available for use by the Easysoft ODBC-JDBC Gateway and any work done by the gateway is under the control of the Transaction Manager. (Your ODBC application also needs to turn off the ODBC auto-commit mode by using `SQLSetConnectAttr` with the `SQL_ATTR_AUTOCOMMIT` attribute.)

## CONFIGURATION

### Configuring the Easysoft ODBC-JDBC Gateway

If you want to use the Easysoft ODBC-JDBC Gateway in the context of a distributed XA transaction, enable the `XA_Enlist` option. Otherwise, leave the option set to its default value OFF (0).

Interface	Value
DSN Dialog Box (Windows)	Not available.
odbc.ini file (Unix)	<code>XA_Enlist=0 1</code>
Connect String	<code>XA_ENLIST=0 1</code>

### **XA\_CONNECTION\_STRING**

The DB field value in the `xa_open` string. For example, you specify a database named "payroll" with the following `xa_open` string clause:

```
DB=payroll
```

You also need to specify "payroll" as the value for the `XA_Connection_String` setting:

```
XA_Connection_String=payroll
```

Interface	Value
DSN Dialog Box (Windows)	Not available
odbc.ini file (Unix)	<code>XA_Connection_String=value</code>
Connect String	<code>XA_CONNECTION_STRING=value</code>

## XAClass

The class the JDBC driver implements the `XADataSource` interface with.

Interface	Value
DSN Dialog Box (Windows)	Not available
odbc.ini file (Unix)	<code>XAClass=value</code>
Connect String	<code>XCLASS=value</code>

### Examples

For the Oracle JDBC driver, use:

```
XAClass = oracle.jdbc.xa.client.OracleXADataSource
```

For the Microsoft SQL Server JDBC driver, use:

```
XAClass = com.microsoft.sqlserver.jdbc.SQLServerXADataSource
```

## XIDClass

The method the JDBC driver provides to create XA transaction ids (Xid). The Transaction Manager uses Xids to coordinate the branches of a distributed transaction.

Interface	Value
DSN Dialog Box (Windows)	Not available
odbc.ini file (Unix)	<code>XIDClass=value</code>
Connect String	<code>XIDCLASS=value</code>

### Examples

For the Oracle JDBC driver, use:

```
XIDClass = oracle.jdbc.xa.OracleXid
```

## CONFIGURATION

*Configuring the Easysoft ODBC-JDBC Gateway*

For the SQL Server JDBC driver, use:

```
XIDClass = com.microsoft.sqlserver.jdbc.XidImpl
```



# TECHNICAL REFERENCE



---

## Technical Reference for the Easysoft ODBC-JDBC Gateway

This section contains extra information relating to the deployment of the Easysoft ODBC-JDBC Gateway.

---

### Appendix Guide

- [Threading](#)
- [Supported API Calls](#)
- [XA Support](#)
- [Tracing](#)

---

## **Threading**

The Easysoft ODBC-JDBC Gateway as an ODBC driver is supposed to be thread safe, as it is a requirement of ODBC 3 drivers.

No threading issues were found using the Easysoft ODBC-JDBC Gateway in WinSQL or Microsoft Access, both of which are multithreaded applications.

However, when tested last there appeared to be some threading issues when the Easysoft ODBC-JDBC Gateway runs behind the Easysoft ODBC-ODBC Bridge server.

This requires investigation and eventual resolution.

---

## **Supported API Calls**

All ODBC 3.5 calls are supported **except** for the following:

- `SQLBrowseConnect`
- `SQLBulkOperations`
- `SQLCopyDesc`
- `SQLSetPos`
- `SQLDescribeParam`
- `SQLNumParams`
- `SQLSetScrollOptions` (Deprecated in ODBC 3)

The parameter based calls are not supported because support for them is not provided in JDBC 2. When JDBC 3 drivers become more common targets for the Easysoft ODBC-JDBC Gateway, support for these functions will be added.

**UNICODE ODBC CALLS**

If available, the Easysoft ODBC-JDBC Gateway supports the Unicode version (with suffix "W") of the ODBC calls it implements. The Easysoft ODBC-JDBC Gateway supports:

- SQLColAttributeW
- SQLColAttributesW
- SQLColumnPrivilegesW
- SQLColumnsW
- SQLConnectW
- SQLDescribeColW
- SQLDriverConnectW
- SQLExecDirectW
- SQLForeignKeysW
- SQLGetConnectAttrW
- SQLGetCursorNameW
- SQLGetDescFieldW
- SQLGetDescRecW
- SQLGetDiagFieldW
- SQLGetDiagRecW
- SQLGetInfoW
- SQLGetStmtAttrW
- SQLGetTypeInfoW
- SQLNativeSqlW
- SQLPrepareW

- SQLPrimaryKeysW
- SQLProcedureColumnsW
- SQLProceduresW
- SQLSetConnectAttrW
- SQLSetCursorNameW
- SQLSetDescFieldW
- SQLSetStmtAttrW
- SQLSpecialColumnsW
- SQLStatisticsW
- SQLTablePrivilegesW
- SQLTablesW

---

## **XA Support**

The Easysoft ODBC-JDBC Gateway can provide access to XA resources in the context of a distributed transaction.

To access an XA resource, add a data source that connects to the JDBC driver for the resource by using the XA interface.

<b>NB</b>	To be involved in a distributed transaction, the target JDBC driver must implement the XA features of the JDBC 2.0 Optional Package.
-----------	--

In the data source, set the `XA_Enlist` data source option to 1. Use `XAClass` to specify the JDBC driver's `XADatasource` class. The class that the JDBC driver implements `XADatasource` with is used to produce XA connections. Use `XIDClass` to specify the method the JDBC driver provides to create Xids. The Transaction Manager uses Xids to identify each transaction in a distributed transaction. The `XA_Connection_String` attribute is required if the `DB` field is present in the `xa_open` string. The `XA_Connection_String` attribute value must be the same as that of the `DB` field.

This example data source sets up an XA connection to the Oracle JDBC driver:

```
[XA-OJG]
Driver          = Easysoft ODBC-JDBC Gateway
DriverClass     = oracle.jdbc.OracleDriver
User            = system
Password        = manager
ClassPath       = /usr/local/oracle/jdbc/lib/ojdbc14.jar
URL             = jdbc:oracle:oci8:@//my_host:1521/my_servicename
XA_Enlist       = 1
XA_Connection_String = my_database
XAClass         = oracle.jdbc.xa.client.OracleXADatasource
XIDClass        = oracle.jdbc.xa.OracleXid
```

Note that if the XA attributes are present, the data source can only be used to access the target database as an XA resource under the control of a Transaction Manager. If you need to access the same database with a non-XA connection, configure a separate data source that does not contain the XA attributes.

**THE XA\_OPEN STRING**

The Transaction Manager uses the `xa_open` string to connect to the Resource Manager.

The format for the Easysoft ODBC-JDBC Gateway `xa_open` string is:

```
ODBCJDBC_XA+DSN=data_source+LogFile=log_file+DB=db_name
```

`ODBCJDBC_XA` is the Easysoft ODBC-JDBC Gateway Resource Manager name. Use the `DSN` field to specify the Easysoft ODBC-JDBC Gateway data source that you configured for the resource. To enable logging, include the optional `LogFile` field. You need to use this field rather than the standard tracing mechanisms (see ["Tracing" on page 95](#)) to enable logging for an XA connection. The `DB` field is used to allow the ODBC connection to find the matching XA connection.

This WebSphere MQ example shows the corresponding `xa_open` string to use for the `XA-OJG` data source:

```
XAOpenString=ODBCJDBC_XA+DSN=XA-OJG+LogFile=/tmp/sql.log+DB=my_database
```

**THE XA\_SWITCH\_T STRUCTURE**

The Easysoft ODBC-JDBC Gateway `xa_switch_t` structure name is `ojg_xaosw`. This structure contains the entry points for the Easysoft ODBC-JDBC Gateway Resource Manager. Applications requiring XA support should be built to use this structure. The Easysoft ODBC-JDBC Gateway will translate calls into the entry points into the corresponding Java `javax.sql.XADataSource` methods.

---

## Tracing

The ODBC calls an application makes can be traced:

- within the driver manager by an application
- from within the driver manager
- from within the Easysoft ODBC-JDBC Gateway

### WITHIN THE DRIVER MANAGER BY AN APPLICATION

An application can turn tracing on in the driver manager via the ODBC API `SQLSetConnectAttr (... ,SQL_ATTR_TRACE,...)`.

The trace filename may also be specified with the `SQLSetConnectAttr` attribute `SQL_ATTR_TRACEFILE`.

### FROM WITHIN THE DRIVER MANAGER

Tracing in the driver manager is platform-specific:

- On Windows:

Start the ODBC driver manager administration interface via **Start Menu > Control Panel > ODBC Data Sources**.

Click on **Tracing**, ensure the specified filename is valid and click **Start Tracing Now**.

- On Unix:

For the unixODBC driver manager, add two attributes to the [ODBC] section (create one if none exists) in `odbcinst.ini`.

```
Trace = Yes|No
```

```
TraceFile = /path/filename
```

e.g.

## TECHNICAL REFERENCE

*Technical Reference for the Easysoft ODBC-JDBC Gateway*

```
[ODBC]
Trace = Yes
TraceFile = /tmp/sql.log
```

Ensure that the user who is running the application to be traced has write permission to `TraceFile` (and to the directory containing it), or no tracing information will be produced.

### FROM WITHIN THE EASYSOFT ODBC-JDBC GATEWAY

- On Unix:

Add a `LOG` attribute to the DSN section of the `odbc.ini` file. e.g.

```
[Easysoft ODBC-JDBC Gateway]
.
.
LOG = /tmp/ojg.log
```

The `LOG` value is the path and file name of the log file (e.g. `/tmp/ojg.log`).



# GLOSSARY

# B

---

## Terms and definitions

### **API**

Application Programmer Interface. An API is a published set of function calls and constants allowing different programmers to utilize a ready-written library of subroutines.

### **Applet**

A Java program that has certain restrictions placed upon it, such as restricted network and filesystem access. Applets are able to run in a web browser.

### **Application**

An Application Program ("Application" or "App") is 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.

### **Bitmask**

A value which, when written out in binary, has a meaning assigned to each digit, which can be 0 or 1. This is a very efficient way of storing a number of *flags* in a small amount of memory. When viewed in decimal it is a single number resulting from adding up the values of the individual bits, worth 1, 2, 4, 8, 16, 32 and so on.

**Client/Server**

The name given to 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 requests queries or actions be performed on the data by the server. 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*). The term *column* might refer to only the *definition* of a column (i.e. its name and type), or to all the data in it.

**Connection String**

ODBC *driver managers* accept a connection string when a client connects. Ideally it contains all necessary attribute values to make the connection to a data source, but provision is made for the driver to negotiate with the application or the user for any missing information.

**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 99**) to identify it to the ODBC Driver Manager.

**DLL**

Dynamic Link Library. Windows' mechanism for shared object code. See also **“Shared Object” on page 101**.

**Download**

The transfer of 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 101.**

**Driver Manager**

Software whose main function is to load ODBC drivers. ODBC applications connect to the Driver Manager and request a *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. Called a *cell* in MS Access.

**Flags**

Single-bit values, representing 'Yes' or 'No'. When more than one flag is present, they are normally stored in a *bitmask*.

**Host**

A computer visible on the network.

**HTTP**

HyperText Transfer Protocol. The means of transferring web pages.

**JDBC (Java DataBase Connectivity)**

A Java *API* for database access based on the *ODBC* API, but includes an object-oriented interface to the underlying data source.

**JDK (Java Development Kit)**

A software development environment for writing Java applets and applications.

**JRE (Java RunTime Environment)**

A subset of the Java Development Kit (JDK) which supports the execution (but not the development) of Java applications and consists of the Java Virtual Machine (JVM), the core classes, and supporting files.

**JVM (Java Virtual Machine)**

Software that interprets and executes the byte codes in Java class files like a microprocessor would execute machine code. There are many virtual machines available from different vendors and for different purposes.

**Middleware**

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

**ODBC (Open DataBase Connectivity)**

A standard *API* for connecting application programs to relational database systems through a suitable *driver*. ODBC is available on a wide number of platforms.

### **ODBC driver**

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

### **Operating System**

A collection of software programs, APIs and working practices that control and integrate the execution of system functions on behalf of application programs.

### **Platform**

The term *platform* normally covers the hardware and operating system as a unit. For example; a PC running Microsoft Windows, a PC running BSD Unix, and a Sun running Solaris are three different platforms.

### **Server**

A computer, or *host*, on the network, designed for power and robustness rather than user-friendliness and convenience. Servers typically run round-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. See **“Client/Server” on page 98**.

### **Shared Object**

A piece of object code (i.e. a program fragment) for loading and executing by other programs.

### **SQL (Structured Query Language)**

A standard language for interacting with relational database systems, based on Relational Theory.

### 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 102.**

### Table

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

software	
vendor	name
Easysoft	Easysoft ODBC-JDBC Gateway
MySoft	My ODBC Client Application

This table has two columns; `vendor`, and `name`. It has two rows: one corresponding to the Easysoft ODBC-JDBC Gateway and the other corresponding to MySoft’s ODBC client software. The term *table* can also apply to just the definition of the table, without its data.

### User Data Source

An ODBC Data Source with access limited to a specific user on a given system. See also **“System Data Source” on page 102.**

# INDEX

## Symbols

---

/etc/defaults/rc.conf .....	40
/etc/ld.so.conf .....	40

## A

---

Add .....	67
Add/Remove Programs icon .....	34
ADO .....	74
API .....	97
API conformance .....	90
Applet .....	97
application .....	97
Async Cancel .....	84

## B

---

beta releases .....	20
bitmask .....	97
bunzip2 .....	42
bzip2 .....	21

## C

---

Cancel .....	68
Caution box .....	9
CD .....	20
Class Path .....	66, 67, 78
ClassLoader	
see URLClassLoader .....	82
CLEAN_METADATA .....	81
client-server .....	98
column .....	98

## INDEX

compress .....	21
connection string .....	98
Control Panel	
ODBC .....	30, 63

### D

---

data source .....	98
system .....	102
user .....	102
DBMS .....	14
Description .....	66, 75
distributed transactions .....	92
DLL .....	98
documentation .....	20
download .....	99
driver .....	99
Driver Class .....	66, 77
driver manager .....	99
unixODBC .....	44
DSN .....	65, 74, 99
DSN-less connections .....	62

### E

---

easysoft directory .....	37
Easysoft ODBC-JDBC Gateway	
installing on Unix .....	35
installing on Windows .....	23
uninstalling on Unix .....	59
uninstalling on Windows .....	34



**F**

---

field .....	99
files	
odbcinst.ini .....	95
flags .....	99
FTP .....	20

**G**

---

gunzip .....	42
gzip .....	21

**H**

---

Help .....	68
host .....	99
HTTP .....	100

**I**

---

install directory .....	38
installation	
file name .....	21
requirements on Unix .....	35
installing .....	23
on Unix .....	35
on Windows .....	23

**J**

---

JDBC .....	100
JDK .....	100
JRE .....	100
JVM	
defined .....	100
setting options for .....	33
specifying on Unix .....	55
specifying on Windows .....	31

# INDEX

## L

---

LD_LIBRARY_PATH .....	54, 73
LD_RUN_PATH .....	54
LIBPATH .....	54, 73
license	
authorization code .....	26

## M

---

Microsoft Access .....	90
middleware .....	100
Modify Metadata .....	81

## N

---

Note box .....	9
----------------	---

## O

---

ODBC .....	100
background to .....	14
conformance .....	90
driver .....	101
ODBC Data Source Administrator .....	63
odbc.ini .....	70
odbcinst .....	38
OK .....	68
open database connectivity .....	100
operating system .....	101

## P

---

Password .....	66, 76
patches .....	20
platform .....	101
Platform note .....	9

**R**


---

Reference box .....	9
Reuse CL Object .....	82

**S**


---

SCHEMA .....	81
server .....	101
shared object .....	101
SHLIB_PATH .....	54
Single Statement .....	80
SQL .....	101
SQLBrowseConnect .....	90
SQLBulkOperations .....	90
SQLConnect .....	74
SQLCopyDesc .....	90
SQLDescribeParam .....	90
SQLDriverConnect .....	74
SQLNumParams .....	90
SQLSetPos .....	90
SQLSetScrollOptions .....	90
Strip Escape .....	82, 83
Strip Quote .....	79
structured query language .....	101
symbolic link .....	38
system data source .....	102
System DSN tab .....	64

**T**


---

table .....	102
tar .....	42
Test .....	68
threading .....	90
tracing .....	94, 95

**U**

---

uncompress .....	42
unicode	
Microsoft Access and .....	84
ODBC API Calls .....	91
uninstalling	
on Unix .....	59
on Windows .....	34
unixODBC .....	70
upgrades .....	20
URL .....	67, 78
URLClassLoader .....	82
user data source .....	102
User DSN tab .....	64
User Name .....	66, 75

**W**

---

WCHAR Default .....	84
web site .....	20
wide characters	
Microsoft Access and .....	84
WinSQL .....	90
With Schema .....	80

**X**

---

XA support .....	92
XA_Connection_String .....	86
XA_Enlist .....	85
xa_open string .....	94
xa_switch_t structure .....	94
XAClass .....	87
XIDClass .....	87