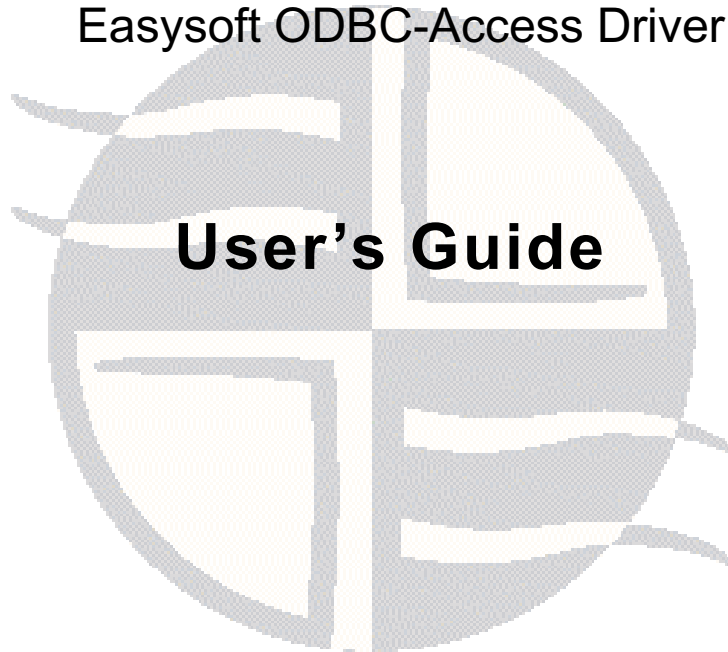# Easysoft Data Access

Easysoft ODBC-Access Driver

User's Guide

This manual documents version 1.0.n of the Easysoft ODBC-Access Driver.

Publisher: Easysoft Limited

Thorp Arch Grange

Thorp Arch

Wetherby

LS23 7BA

United Kingdom

CONTENTS

LIST OF FIGURES

This page left blank intentionally

PREFACE

About this manual

This manual is intended for use by anyone who wants to install the Easysoft ODBC-Access Driver, configure it, and then access Microsoft Access data sources from an ODBC-enabled application.

Chapter Guide

- **Intended Audience**

- **Notational Conventions**

- **Typographical Conventions**

- **Contents**

- **Trademarks**

Intended Audience

The Unix-based sections require experience of using Unix shell commands. You need to be able to do basic tasks such as editing text files. More complex tasks are described in detail, but it helps to understand how your system handles dynamic linking of shared objects.

Displaying the Manual

This manual is available in the following formats:

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

- HTML.

Notational Conventions

A *note box* provides additional information that may further your understanding of a particular topic in this manual:

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

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

Linux	On Linux, you must log on as the `root` user to make many important changes.

A *caution box* provides 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!

Typographical Conventions

This manual uses the following typographical conventions:

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

  Click **Next** to continue.

- Commands to be typed are displayed in a `monotype` font, for example:

  At the command prompt, type `admin`.

- File listings and system names (such as file names, directories and database fields) are displayed in a `monotype` font.

Contents

- **Introduction**

  Introduces the Easysoft ODBC-Access Driver.

- **Installation**

  Explains how to install the Easysoft ODBC-Access Driver.

- **Configuration**

  Explains how to configure the Easysoft ODBC-Access Driver.

- Appendices

  Technical Reference and **Glossary**.

Trademarks

Throughout this manual, *Windows* refers generically to Microsoft Windows 95, 98, 2000, NT, XP, ME, 2003 Server, Vista or 2008 server, 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.

CHAPTER 1 INTRODUCTION

Introducing the Easysoft ODBC-Access Driver

The Easysoft ODBC-Access Driver is an ODBC 3.51 driver for Microsoft Access. It lets ODBC-enabled applications access Microsoft Access databases from Linux and Unix platforms.

Chapter Guide

- **Overview**
- **Product Status**
- **Deployment**

Overview

The Easysoft ODBC-Access Driver connects ODBC-enabled applications on Linux and Unix to Access databases. For example, connect to Access databases from Apache, ApplixWare, Informatica, OpenOffice.org and StarOffice. In addition, the Easysoft ODBC-Access Driver supports the Perl DBI and DBD::ODBC modules, PHP, PEAR DB, the Python pyodbc and mxODBC interfaces, C and any other ODBC-enabled programming language or interface.

Product Status

The Easysoft ODBC-Access Driver is currently available on Unix and Linux platforms. The most up to date list of Easysoft ODBC-Access Driver platforms is available at:

http://www.easysoft.com/products/data_access/odbc-access-driver/index.html

Software problems can be reported to **support@easysoft.com** by users who have either purchased support or registered at the Easysoft web site at **http://www.easysoft.com** and are evaluating Easysoft products.

Deployment

Access has four components: a structure to hold data (tables), a way to manipulate that data (the Microsoft Jet Database Engine engine), an environment to create a front end for the data (Design view of forms and reports) and tools that can run the front end (Data view of forms and reports). Access data is stored in a database file, which has the extension .mdb. The Easysoft ODBC-Access Driver communicates directly with the .mdb file. The Easysoft ODBC-Access Driver does not use Jet Database Engine to access the .mdb file. (However, the driver does use the Jet locking mechanisms to manage concurrent access to the .mdb file.)

If your Access database is split into two files, you can use the Easysoft ODBC-Access Driver to access your data by loading the backend (_be.mdb) file in which the tables are stored.

To access your Access data with the Easysoft ODBC-Access Driver:

• On the Easysoft ODBC-Access Driver machine, mount the remote Windows share share where the .mdb file is stored.

– OR –

• Copy the .mdb file to the Easysoft ODBC-Access Driver machine.

The Easysoft ODBC-Access Driver does not support Access Project files (.adb), which allow you to create an Access client front end to a SQL Server databases. If you need to access your SQL Server data from Linux or Unix, use the Easysoft ODBC-SQL Server Driver, which is available to download at:

http://www.easysoft.com/products/data_access/odbc-sql-server-driver/index.html

Supported Access Database Versions

The Easysoft ODBC-Access Driver supports version 4.0 .mdb files, which is the default format for databases created in Microsoft Access 2000, 2002 and 2003.

Although Microsoft Office Access 2007 introduced a new database file format, .accdb, version 4.0 .mdb files continue to be supported by later versions of Access.

The Easysoft ODBC-Access Driver also supports .accdb databases created in Microsoft Office Access 2007.

CHAPTER 2 INSTALLATION

Installing the Easysoft ODBC-Access Driver

This chapter explains how to install, license and remove the Easysoft ODBC-Access Driver.

The installation instructions assume you are, or are able to consult with, a system administrator.

Chapter Guide

- **Obtaining the Easysoft ODBC-Access Driver**

- **What to Install**

- **Installing the Easysoft ODBC-Access Driver**

- **Uninstalling the Easysoft ODBC-Access Driver**

Obtaining the Easysoft ODBC-Access Driver

There are three ways to obtain the Easysoft ODBC-Access Driver:

- The Easysoft web site is available 24 hours a day at **http://www.easysoft.com** and lets you download product releases and documentation.

  Choose **Download** from the Easysoft ODBC-Access Driver section of the web site 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 site is available 24 hours a day at **ftp://ftp.easysoft.com** and lets you download free patches, upgrades, documentation and beta releases of Easysoft products, as well as definitive releases.

  Change to the `pub/access` subdirectory and then choose the platform release that you require.

- You can order Easysoft software on CD. To do this, **contact us** by email, telephone or post.

What to Install

The name of the Easysoft ODBC-Access Driver distribution file varies from platform to platform. The file name format is:

- `odbc-access-`*`x.y.z-platform`*`.tar.gz`

  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` depends on the operating system distribution you require. File names may have this format:

- `odbc-access-`*`x.y.z-platform-variation`*`.tar`

  where *`platform-variation`* refers to alternative versions available for a single platform.

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

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

Note	If you download a Unix file with a Windows browser, the browser may strip the file name extension. For example, if you download a `.gz` file and the browser strips the file name extension, it may not be obvious that the file is gzipped. Use `file` *filename* to find out the file type of the downloaded file.

Caution! As long as you stop all Easysoft software first (or software that uses the Easysoft drivers), it is safe to reinstall or upgrade the Easysoft ODBC-Access Driver without uninstalling.

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

Installing the Easysoft ODBC-Access Driver

These instructions show how to install the Easysoft ODBC-Access Driver on Unix platforms. Please read this section carefully **before** installing the Easysoft ODBC-Access Driver.

BEFORE YOU INSTALL

Requirements

To install the Easysoft ODBC-Access Driver on Unix you need:

*   The Bourne shell in `/bin/sh`. If your Bourne shell is not located there, you may need to edit the first line of the installation script.

*   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 do not have any of these commands, they can usually be obtained from the **Free Software Foundation**. As the `tee` command does not work correctly on some systems, the distribution includes a `tee` replacement.

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

- For Easysoft Licensing to work, you must do one of the following:

  - Install the Easysoft ODBC-Access Driver in `/usr/local/easysoft`.

  - Install the Easysoft ODBC-Access Driver elsewhere and 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`.

  - Install the Easysoft ODBC-Access Driver elsewhere and set the `EASYSOFT_ROOT` environment variable.

    For more information about setting the `EASYSOFT_ROOT` environment variable, see **"Post installation" on page 39**.

- An ODBC Driver Manager. Easysoft ODBC-Access Driver distributions include the unixODBC Driver Manager.

- You do not have to be the `root` user to install, 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-Access Driver with unixODBC.

  2. Create the example data source in the SYSTEM `odbc.ini` file.

  3. Update the dynamic linker entries (some platforms only).

  If you are not `root`, these tasks will have to be done manually later.

  Easysoft recommend you install all components as the `root` user.

What you can Install

This distribution contains:

- The Easysoft ODBC-Access Driver.

- The unixODBC Driver Manager.

You will need an ODBC Driver Manager to use the Easysoft ODBC-Access Driver from your applications. The distribution therefore contains the **unixODBC Driver Manager**. Most (if not all) Unix applications and interfaces support the unixODBC Driver Manager. For example, Perl DBD::ODBC, PHP, Python and so on.

You do not have to install the unixODBC Driver Manager included with this distribution. You can use an existing copy of unixODBC. For example, a version of unixODBC installed by another Easysoft product, a version obtained from your operating system vendor or one that you built yourself. However, as Easysoft ensure that the unixODBC distributed with the Easysoft ODBC-Access Driver has been tested with that driver, we recommend you use it.

If you choose to use an existing unixODBC Driver Manager, the installation script will attempt to locate it. The installation script looks for the Driver Manager in the standard places. If you have installed it in a non-standard location, the installation script will prompt you for the location. The installation primarily needs unixODBC's `odbcinst` command to install drivers and data sources.

Where to Install

This installation needs a location for the installed files. The default location 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 accept the default location `/usr/local`, the product will be installed in `/usr/local/easysoft` and below.

If you choose a different installation path, the installation script will try to symbolically link `/usr/local/easysoft` to the `easysoft` subdirectory in your chosen location. 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, the installation will be unable to create the symbolic link. For information about how to correct this manually, see **"Post Installation Steps for non-root Installations" on page 39**.

Note that you cannot license Easysoft products until either of the following is true:

- `/usr/local/easysoft` exists either as a symbolic link to your chosen installation path or as the installation path itself.

- You have set `EASYSOFT_ROOT` to *installation_path*`/easysoft`.

Changes Made to Your System

This installation script installs files in subdirectories of the path requested at the start of the installation, Depending on what is installed, a few changes may be made to your system:

1. If you choose to install the Easysoft ODBC-Access Driver into unixODBC, 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` is in *installation_path*`/easysoft/unixODBC/bin`, if you are using the unixODBC included with this distribution.)

   The `odbcinst.ini` entry for the Easysoft ODBC-Access Driver will look similar to this:

```
[Easysoft ODBC-ACCESS]

Description     = Easysoft ODBC-Access Driver

Driver          = /usr/local/easysoft/access/lib/libesmdb.so

Setup           = /usr/local/easysoft/access/lib/libesmdbS.so

Threading       = 0

FileUsage       = 1

DontDLClose     = 1

UsageCount      = 1
```

For information about removing these entries, see **"Uninstalling the Easysoft ODBC-Access Driver" on page 44**.

2. The installation script installs example data sources into unixODBC. The data sources will be added to your SYSTEM `odbc.ini` file. You can locate your SYSTEM `odbc.ini` file by using `odbcinst -j`. The data source for the standard driver will look similar to this:

```
[ACCESS_SAMPLE]

Description      = Easysoft ODBC-Access Driver

Driver          = Easysoft ODBC-ACCESS

mdbfile         = /path/to/data.mdb

smbpath         =

lockfile        =

smblib          =

smbuser         =

smbauth         =

readonly        = yes

exclusive       = no

ignore_rel      = no
```

For information about removing these data sources, see
**"Uninstalling the Easysoft ODBC-Access Driver" on page
44**.Easysoft ODBC-Access Driver

3. Dynamic Linker.

   On operating systems where the dynamic linker has a file listing locations for shared objects, the installation script will attempt to add Easysoft ODBC-Access Driver shared object paths to this list. On Linux, this dynamic linker configuration file is usually `/etc/ld.so.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 `installation_path/easysoft/lib`. When you run the installation script, the dates and versions of these files will be compared with the same files in the distribution. The files are 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. This is because on some platforms, files in use cannot be replaced. If a file cannot be updated, you will see a warning during the installation. All warnings are written to a 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, 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 pieces of information. Before installing, you need to find out whether you have unixODBC already installed and where it is installed. The installation script searches standard places like `/usr` and `/usr/local.` However, if you installed the Driver Manager in a non-standard place and you do not install the included unixODBC, you will need to know the location.

I N S T A L L A T I O N

Unpacking the Distribution

The distribution for Unix platforms is a tar file. There are multiple copies of the same distribution with different levels of compression. You unpack the distribution in one of the following ways.

If the distribution file has been gzipped (`.gz`), use:

```
gunzip odbc-access-x.y.z-platform.tar.gz
```

If the distribution file has been bzipped (`.bz2`), use:

```
bunzip2 odbc-access-x.y.z-platform.tar.bz2
```

If the distribution file has been compressed, (`.Z`), use:

```
uncompress odbc-access-x.y.z-platform.tar.Z
```

You may have a distribution file which is not compressed at all (`.tar`). To extract the installation files from the tar file, use:

```
tar -xvf odbc-access-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 installation script and various other installation files.

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

License to Use

The End-User License Agreement is contained in the file `license.txt`. Be sure to understand the terms of the agreement 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 answer some questions. In each case, the default choice will be displayed in square brackets and you need only press Enter to accept the default. If there are alternative responses, these will be shown in round brackets; to choose one of these, type the response 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 answer when you type nothing and press Enter is `n`.

Running the Installer

Before you run the installer, make sure you have read **"Installation" on page 29**. If you are considering running the installation as a non `root` user, make sure you also read:

http://www.easysoft.com/support/kb/kb01009.html

To start the installation, run:

```
./install
```

You will need to:

* Confirm your acceptance of the license agreement by typing "yes" or "no".

    For more information about the license agreement, see **"License to Use" on page 30**.

* Supply the location where the software is to be installed. Easysoft recommend accepting the default installation path.

    For more information, see **"Where to Install" on page 24**.

Note	If you are upgrading, you will need a new license from Easysoft.

Locating or Installing unixODBC

Easysoft strongly recommend you use the unixODBC Driver Manager because:

• The installation script is designed to work with unixODBC and can automatically add Easysoft ODBC-Access Driver and data sources during the installation.

• Most ODBC-enabled applications and interfaces support unixODBC. The Easysoft ODBC-Access Driver and any data sources that you add during the installation will be automatically available to your applications and interfaces therefore.

• The unixODBC project is currently led by Easysoft developer Nick Gorham. This means that there is a great deal of experience at Easysoft of unixODBC in general and of supporting the Easysoft ODBC-Access Driver 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 installation starts by searching for unixODBC. There are two possible outcomes here:

1. If the installation script finds unixODBC, the following message will be output:

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

2. If the installation script cannot find unixODBC in the standard places, you will be asked whether you have it installed.

If unixODBC is installed, you need to provide the unixODBC installation path. Usually, the path required is the directory above where `odbcinst` is installed. For example, if `odbcinst` is in `/opt/unixODBC/bin/odbcinst`, the required path is `/opt/unixODBC`.

If unixODBC is not installed, you should install the unixODBC included with this distribution.

If you already have unixODBC installed, you do not have to install the unixODBC included with the distribution, but you might consider doing so if your version is older than the one included.

The unixODBC in the Easysoft ODBC-Access Driver distribution is not built with the default options in unixODBC's configure line.

Option	Description
--prefix=/etc	This means the default SYSTEM `odbc.ini` file where SYSTEM data sources 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.

Option	Description
--enable-stats=no	Disables unixODBC statistics, which use system semaphores to keep track of used handles. Many systems do not have sufficient semaphore resources to keep track of used handles. In addition, the statistics are only available in the GUI ODBC Administrator.
--enable-readline=no	This disables readline support in `isql`. We disabled this because it ties `isql` to the version of libreadline on the system we build on. We build on as old a version of the operating system as we can for forward compatibility. Many newer Linux systems no longer include the older readline libraries and so enabling readline support makes `isql` unusable on these systems.
--prefix=/usr/local/easysoft/unixODBC	This installs unixODBC into `/usr/local/easysoft/unixODBC`.

Figure 1: Easysoft unixODBC configure line options.

Installing the Easysoft ODBC-Access Driver

The Easysoft ODBC-Access Driver installation script:

- Installs the driver.

- Registers the driver with the unixODBC Driver Manager.

  If the Easysoft ODBC-Access Driver is already registered with unixODBC, a warning will be displayed that lists the drivers unixODBC knows about. If you are installing the Easysoft ODBC-Access Driver 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. unixODBC's `odbcinst` will not update these paths if a driver is already registered.

- Creates an example Easysoft ODBC-Access Driver data source.

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

  If a data source called "ACCESS_SAMPLE" already exists, the existing data source will be displayed and you have the option to replace it.

Licensing

The *installation_path*`/easysoft/license/licshell` program lets you obtain or list licenses.

Licenses are stored in the *installation_path*`/easysoft/license/licenses` file. After obtaining a license, you should make a backup copy of this file.

The installation script asks you if you want to request an Easysoft ODBC-Access Driver license:

```
Would you like to request a Easysoft ODBC-Access
Driver 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, the installation runs the `licshell` script. The process of obtaining a license is best described in the **Licensing Guide**.

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, the License Client can create a license request file that you can mail or fax to Easysoft. You can also supply the details to us by telephone.

Start the License Client. The following menu is displayed:

```
[0] exit
```

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

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

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 generates a key that is used to identify the product and operating system (we need this key to license you).

After you have chosen the product to license (Easysoft ODBC-Access Driver), you need to supply:

- Your full name.

- Your company name.

- An email contact address. This **must** be the email address that you used when you registered on the Easysoft web site.

- Your telephone number (you need to specify this if you telephone us to request a license).

- Your fax number (you need to specify this if you fax the license request to us).

- A reference number. When applying for a trial license, just press Enter when prompted for a reference number. This field is used to enter a reference number that we will supply you for full (paid) licenses.

You will then be asked to specify how you want to obtain the license. The choices are:

```
[1] Automatically by contacting the Easysoft
License Daemon
```

This requires a connection to the Internet and the ability to support an outgoing TCP/IP connection to `license.easysoft.com` on port 8884.

```
[2] Write information to file so you can fax,
telephone it
```

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

```
[3] Cancel this operation
```

If you choose to obtain the license automatically, the License Client will start a TCP/IP connection to `license.easysoft.com` on port 8884 and send the details you supplied and your machine number. No other data is sent. The data sent is transmitted as plain text, so if you want to avoid the possibility of this information being intercepted by someone else on the Internet, 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`. You should then exit the License Client by choosing option `[0]` and complete the installation. After you have mailed, faxed or telephoned the license request to us, we will return a license key. Add this to the end of the file *installation_path*`/easysoft/license/licenses`.

If any warnings or errors are output during this process, please mail the output to **support@easysoft.com** and we will correct the problem.

P OST INSTALLATION

Supplied Documents and Examples

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

*   The Easysoft ODBC-Access Driver documentation is installed in *installation_path*/easysoft/access/doc:

    *   The Easysoft ODBC-Access Driver manual in PDF format.

    *   The Easysoft ODBC-Access Driver EULA.

installation_path/easysoft/access/doc/CHANGES.txt lists all the changes in each version of the Easysoft ODBC-Access Driver.

There are also many resources at the **Easysoft web site**.

Post Installation Steps for non-root Installations

If you installed the Easysoft ODBC-Access Driver as a non-root user (not recommended), there may be some additional steps you will need to do manually:

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

    You can manually install the driver under unixODBC by adding an entry to the odbcinst.ini file. Run odbcinst -j to find out the location of the DRIVERS file then append the lines from the drv_template file to the odbcinst.ini file. (drv_template is in the directory where the distribution was untarred to).

2. No example data sources can be added into unixODBC if you do not have write permission to the SYSTEM `odbc.ini` file. Run `odbcinst -j` to find out the location of the SYSTEM DATA SOURCES file then add your data sources to this file.

3. On systems where the dynamic linker has a configuration file defining the locations where it looks for shared objects (Linux), you will need to add:

```
installation_path/easysoft/lib
installation_path/easysoft/unixODBC/lib
```

    The latter entry is only required if you installed the unixODBC included with this distribution. Sometimes, after changing the dynamic linker configuration file, you need to run a program to update the dynamic linker cache. (For example, `/sbin/ldconfig` on Linux.)

4. If you did not install the Easysoft ODBC-Access Driver in the default location, you need to do one of the following:

    *   Link `/usr/local/easysoft` to the `easysoft` directory in your chosen installation path.

        For example, if you installed in `/home/user`, the installation will create `/home/user/easysoft` and you need to symbolically link `/usr/local/easysoft` to `/home/user/easysoft`:

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

    *   Set and export the `EASYSOFT_ROOT` environment variable to *installation_path*/easysoft.

5. If your system does not have a dynamic linker configuration file, you need to add the paths listed in step 3 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 such as `/etc/profile`.

    The environment variable depends on the dynamic linker. Refer to your `ld` or `ld.so` man page. It is usually:

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

SETTING DYNAMIC LINKER SEARCH PATHS

Your applications will be linked against an ODBC Driver Manager, which will load the ODBC Driver you require. The dynamic linker needs to know where to find the ODBC Driver Manager shared object. The ODBC Driver Manager will load the Easysoft ODBC-Access Driver, which is dependent on further common Easysoft shared objects; the dynamic linker needs to locate these too.

On operating systems where the dynamic linker has a file specifying locations for shared objects (Linux, for example), the installation will attempt to add paths under the path you provided at the start of the installation to the end of this list; no further action should be required. For more information, see **"Dynamic Linker." on page 28**.

On other Unix platforms, there are two methods of telling the dynamic linker where to look for shared objects:

1. You add the search paths to an environment variable and export it.

    This method always works and overrides the second method, described below.

2. At build time, a run path is inserted into the executable or shared objects. On most System V systems, Easysoft distribute Easysoft ODBC-Access Driver shared objects with an embedded run path. The dynamic linker uses the run path to locate Easysoft ODBC-Access Driver shared object dependencies.

   For the first method, the environment variable you need to set depends on the platform (refer to the platform documentation for `ld(1), dlopen` or `ld.so(8))`.

Environment Variable	Platform
LD_LIBRARY_PATH	System V based operating systems and Solaris.
LIBPATH	AIX
SHLIB_PATH	HP-UX
LD_RUN_PATH	Many platforms use this in addition to those listed above.

   Figure 2: Dynamic linker search path environment variables.

   To use the Easysoft ODBC-Access Driver, you need to add:

   installationdir/easysoft/access:*installationdir*/easysoft/lib

   where *installationdir* is the directory in which you chose to install the Easysoft ODBC-Access Driver. If you accepted the default location, this is `/usr/local`.

An example of setting the environment path in the Bourne shell on Solaris is:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/easyso
ft/access:/usr/local/easysoft/lib
```

```
export LD_LIBRARY_PATH
```

Note	The exact command you need to set and export an environment variable depends on your shell.

If you installed the unixODBC Driver Manager included in the Easysoft ODBC-Access Driver distribution, you also need to add *installationdir*/easysoft/unixODBC/lib to the dynamic linker search path.

Uninstalling the Easysoft ODBC-Access Driver

There is no automated way to remove the Easysoft ODBC-Access Driver in this release. However, removal is quite simple. To do this, follow these instructions.

To uninstall the Easysoft ODBC-Access Driver

1. Change directory to `installation_path`/`easysoft` and delete the `access` directory. `installation_path` is the Easysoft ODBC-Access Driver installation directory, by default `/usr/local`.

2. If you had to add this path to the dynamic linker search paths (for example, `/etc/ld.so.conf` on Linux), remove it. You may have to run a linker command such as `/sbin/ldconfig` to get the dynamic linker to reread its configuration file. Usually, this step can only be done by the `root` user.

3. If you were using unixODBC, the Easysoft ODBC-Access Driver entry needs to be removed from the `odbcinst.ini` file. To check whether the Easysoft ODBC-Access Driver is configured under unixODBC, use `odbcinst -q -d`. If the command output contains `[Easysoft ODBC-ACCESS]`, uninstall the drivers from unixODBC by using:

   `odbcinst -u -d -n "Easysoft ODBC-ACCESS"`

   If a reduced usage count message is displayed, repeat this command until `odbcinst` reports that the drivers have been removed.

4. If you created any Easysoft ODBC-Access Driver data sources under unixODBC, you may want to delete these. To do this, first use `odbcinst -j` to locate USER and SYSTEM `odbc.ini` files. Then check those files for data sources that have the driver attribute set to `Easysoft ODBC-Access.`

5. Remove `access_install.info` from the *installation_path*/`easysoft` directory.

This page left blank intentionally

CHAPTER 3 CONFIGURATION

Configuring the Easysoft ODBC-Access Driver

The Easysoft ODBC-Access Driver is installed on the computer where your applications are running. ODBC applications access ODBC drivers through the ODBC Driver Manager and a data source. The data source tells the Driver Manager which ODBC driver to load and which Access database to connect to. This chapter describes how to create data sources, use DSN-less connections and configure the Easysoft ODBC-Access Driver.

Before setting up a data source, you must have successfully installed the Easysoft ODBC-Access Driver.

For Easysoft ODBC-Access Driver installation instructions, see **"Installation" on page 17**.

Chapter Guide

- **Connecting to your Access Database**

- **Configuring the Easysoft ODBC-Access Driver**

- **Setting Up Data Sources**

- **DSN-less Connections**

- **Troubleshooting Database Connection Problems**

- **Permissions Summary**

Connecting to your Access Database

The Easysoft ODBC-Access Driver is a file-based ODBC driver that communicates directly with the Access database file. The database file must be accessible through the local file system on the machine where the Easysoft ODBC-Access Driver is installed.

DEPLOYMENT OPTIONS

If both Windows and Easysoft ODBC-Access Driver users will need to access the database file you can:

• Copy the database file to a Samba share on the Easysoft ODBC-Access Driver machine.

– OR –

• Mount the Windows share where the database file is located as part of the local file system on the Easysoft ODBC-Access Driver machine.

Alternatively, if only Easysoft ODBC-Access Driver users need to access the database file, you should:

• Copy the database file to a directory on the Easysoft ODBC-Access Driver machine.

Read this section to see which Easysoft ODBC-Access Driver deployment option is appropriate for you.

1. **Where Does Your Database File Need To Be Located?**

   For performance reasons, you should locate the database file nearest to the applications that use it most intensively. If your Windows applications would be most adversely affected by opening a database file located on a remote machine, locate the database file on Windows. Otherwise, copy the database file to the Easysoft ODBC-Access Driver machine, and, if applicable, use Samba to expose the file to Windows users.

2. **What Platform Do You Need To Access The Database File From?**

   LINUX

   - If the database file is located on a Windows share and cannot be moved, use `smbmount` on the Easysoft ODBC-Access Driver machine to mount the remote share.

     Note For read-write access to the database file, you need to use a version of `smbmount` that can disable local file caching (`directio` option). There is a performance penalty associated with this option.

     You also need to install the `libsmbclient` library, which the Easysoft ODBC-Access Driver uses to let Windows applications know when it has a database file open and to prevent other users from opening a database file it has opened for exclusive access. `libsmbclient` is part of the Samba suite.

   - If the database file is located on the Easysoft ODBC-Access Driver machine, copy the file to a Samba share on that machine, if Windows users also need to access the database.

U N I X

- If the database file is located on Windows, mount the remote share as a CIFS filesystem. Open the database for read-only access (`readonly` attribute set to `yes`) if Windows users may also be opening the database for read-write or exclusive access.

- If the database file is located on the Easysoft ODBC-Access Driver machine, copy the file to a Samba share on that machine, if Windows users also need to access the database.

Configuring the Easysoft ODBC-Access Driver

This section describes how to configure the Easysoft ODBC-Access Driver to connect to an Access database by using a data source or a DSN-less connection string. The section assumes you are, or are able to consult with, a database administrator.

Setting Up Data Sources

There are two ways to set up a data source to your Access data:

• Create a SYSTEM data source, which is available to anyone who logs on to this Unix machine.

– OR –

• Create a USER data source, which is only available to the user who is currently logged on to this Unix machine.

By default, the Easysoft ODBC-Access Driver installation creates a SYSTEM data source named `[ACCESS_SAMPLE]`. If you are using the unixODBC included in the Easysoft ODBC-Access Driver distribution, the SYSTEM `odbc.ini` file is in `/etc`.

If you 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`.

If you accepted the default choices when installing the Easysoft ODBC-Access Driver, USER data sources must be created and edited in `$HOME/.odbc.ini`.

Note	To display the directory where unixODBC stores SYSTEM and USER data sources, type `odbcinst -j`. By default, you must be logged in as `root` to edit a SYSTEM data source defined in `/etc/odbc.ini`.

You can either edit the sample data source or create new 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.

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

The `Driver` attribute identifies the ODBC driver in the `odbcinst.ini` file to use for a data source.

When the Easysoft ODBC-Access Driver is installed into unixODBC, an entry for the driver (`Easysoft ODBC-ACCESS`) is placed in `odbcinst.ini`.

For Easysoft ODBC-Access Driver data sources, you need to include a `Driver = Easysoft ODBC-ACCESS` entry.

To configure an Access data source, in your `odbc.ini` file, you need to specify:

- The path to the Access database file (.mdb or .accdb).

For example:

```
[ACCESS_SAMPLE]

Driver    = Easysoft ODBC-ACCESS

mdbfile    = /home/myuser/ms-access/Northwind.mdb
```

– OR –

```
[ACCESS_SAMPLE]

Driver    = Easysoft ODBC-ACCESS

mdbfile    = /home/myuser/ms-access/Northwind 2007.accdb
```

If you are using the Easysoft ODBC-Access Driver to open an database file that Windows users may also be accessing, specify these settings:

- The SMB URL for the database file.

- The path to the SMB client library on the Easysoft ODBC-Access Driver machine.

- The user name and password of a user who can access the Windows or Samba share where the database file is located.

For example:

```
[ACCESS_SAMPLE_WINDOWS_SHARE]

Driver  = Easysoft ODBC-ACCESS

mdbfile = /mnt/accounts/access/Northwind.mdb

smbpath = smb://mswin_machine/accounts/access/Northwind.mdb

smblib = /usr/lib/libsmbclient.so

smbuser = mywindows_user

smbauth = mywindows_password
```

– OR –

```
[ACCESS_SAMPLE_SAMBA_SHARE]

Driver  = Easysoft ODBC-ACCESS

mdbfile = /home/samba/sales/access/Northwind.mdb

smbpath = smb://samba_server/sales/access/Northwind.mdb

smblib = /usr/lib/libsmbclient.so

smbuser = my_samba_share_valid_user

smbauth = my_samba_share_valid_password
```

Attribute Fields

The following attributes may be set in the `odbc.ini` file:

Attribute	Description
Driver = *value*	The name of the ODBC driver to use with this data source. Set this attribute value to `Easysoft ODBC-ACCESS`.
Description = *value*	A single line of descriptive text that may be retrieved by some applications to describe the data source.

Attribute	Description
mdbfile = *value*	The path to the Access database file (.mdb or .accdb) on the machine where the Easysoft ODBC-Access Driver is installed.
	If the database file is stored on the same machine as the Easysoft ODBC-Access Driver, specify the path to the database file. For example:
	`# Path to .mdb file stored on`
	`# Easysoft ODBC-Access Driver machine.`
	`mdbfile = /home/myuser/ms_access/Northwind.mdb`
	– OR –
	`# Path to .accdb file stored on`
	`# Easysoft ODBC-Access Driver machine.`
	`mdbfile = /home/samba/ms_access/Northwind 2007.accdb`
	If the database file is stored on a Windows machine, you need to share the directory where the file is located and then mount the Windows share from the Easysoft ODBC-Access Driver machine. Specify the mounted share's path with `mdbfile`, along with the remote database file name. For example:
	`# Local path to .mdb file stored on`
	`# Windows. /mnt/access_windows is local mount`
	`# point for a remote Windows share.`
	`mdbfile = /mnt/access_windows/Northwind.mdb.`

Attribute	Description
dbq = *value*	The path to the Access database file (.mdb or .accdb) on the machine where the Easysoft ODBC-Access Driver is installed.
	The `dbq` attribute is an alternative to the `mdbfile` attribute, and is provided for symmetry with Microsoft's Access ODBC driver. You do not need to specify both the `dbq` and `mdbfile` attributes in your data source.
	Note The version of the Easysoft ODBC-Access Driver driver that supports the `dbq` attribute is not currently available on the Easysoft web site, but can be obtained by contacting the Easysoft support team (**support@easysoft.com**).

Attribute	Description

smbpath = *value*	The SMB URL for the database file. This attribute (along with `smblib`, `smbuser` and `smbauth`) is required when you open a database file that Windows users may also be accessing.
	
	Important When you specify a SMB URL, the Easysoft ODBC-Access Driver uses the `libsmbclient` library to: let Windows know that it has opened a database file; prevent Windows users from opening a database file it has opened for exclusive access. Without this mechanism, there is the potential for database file corruption when Windows users and Easysoft ODBC-Access Driver users are working simultaneously with same the database.
	
	Use this syntax for the SMB URL:
	
	`smb://host/share/path/filename`
	
	where:
	
	`host` is the name or IP address of the machine on which the share is located.
	
	`share` is the share name.
	
	`path` specifies any subdirectories under *share.*
	
	`filename` is the database file name, which should be the same one as specified by the `mdbfile` attribute.
	
	For example:
	`# SMB URL for a .mdb file shared by a Windows machine.`
	`smbpath = smb://mswin_box/accounts/access/Northwind.mdb`
	
	`# SMB URL for a .accdb file shared from a Unix`
	`# machine through Samba server.`
	`smbpath = smb://unix_box/sales/access/Northwind 2007.accdb`
	
	Note that SMB URLs for Samba shares are case sensitive; SMB URLs for Windows shares are not case sensitive.

Attribute	Description
smblib = *value*	The path to the `libsmbclient` library on the machine where the Easysoft ODBC-Access Driver is installed. For example: `smblib = /usr/lib/libsmbclient.so` `libsmbclient` is part of the Samba suite. The Easysoft ODBC-Access Driver uses `libsmbclient` to: • Allow Linux and Unix users to safely edit a database file that Windows users may also be accessing. • Prevent Windows users from opening a database file it has opened for exclusive access. • Prevent users from locking records in Windows that are already locked in the Easysoft driver and vice versa.
smbuser = *value*	The name of a user who can access the share where the database file is located. If the database file is stored on a Samba share, specify a Samba user. If the database file is stored on a Windows share, specify a Windows user. The user must have either read or read-write permissions for the share. If the user only has read access to the share, set the `readonly` data source attribute to `yes`.
smbauth = *value*	The password for `smbuser`.

Attribute	Description
lockfile = *value*	By default, when you open a database file for shared access, the Easysoft ODBC-Access Driver uses a locking information file named *dbfiledir/dbfilename*.ldb. For example, if you open a database named Nwind.mdb, the Easysoft ODBC-Access Driver expects a locking information file named Nwind.ldb in the same directory as Nwind.mdb.
	Use the lockfile attribute to specify an alternative .ldb file path if directory permissions prevent the Easysoft ODBC-Access Driver from manipulating a .ldb file in the default directory. Specify a directory where all users who connect to this data source have write access to. For example:
	`lockfile = /tmp/Nwind.ldb`
	Do not use the lockfile attribute if users may access the database file from Windows at your site. Windows applications access database files through the Jet/Access database engine, which expects the *dbfiledir/dbfilename*.ldb convention for lock files. There must only be one lock file per database file.

Attribute	Description
readonly = yes | no	To open the database file for read-only access, set `readonly` to `yes`. When opened for read-only access, you can view the Access database, but not change it.
	By default, `readonly` is set to `no`, which means the Easysoft ODBC-Access Driver opens the database for read-write access.
	If you open a database for read-write access without setting the `smb*` data source attributes, the Easysoft ODBC-Access Driver returns the warning:
	`read write access without SMB channel can potentially allow corruption of the MDB file`
	The Easysoft ODBC-Access Driver uses the `smb*` attributes to allow users to safely edit a database file that Windows users may also be accessing. If your database is shared with Windows users, configure the `smb*` attributes before opening the database for read-write access. Otherwise, set `readonly` to `yes`.

Attribute	Description
exclusive = yes | no	To open the database file for exclusive access, set `exclusive` to `yes`. By default, `exclusive` is set to `no`.
	Notes For libsmbclient Users
	When you set `exclusive` to `yes`, the Easysoft ODBC-Access Driver uses `libsmbclient` to the mark the database as `share mode DENY_ALL`. This `share mode` denies all open requests on the database file, which prevents other users from opening the database while you are connected to the Easysoft ODBC-Access Driver data source.
	Note Using `share mode DENY_ALL` is the only way to prevent Windows users from opening a database that you have opened for exclusive access with the Easysoft ODBC-Access Driver.
	To enable the Easysoft ODBC-Access Driver to use `libsmbclient`, you need to set the `smb*` attributes in the data source. If you set `exclusive` to `yes` without setting the `smb*` attributes, the Easysoft ODBC-Access Driver cannot prevent other users from opening the database and so returns the warning:
	``` exclusive access without SMB channel can not exclude other access, and can potentially allow corruption of the MDB file ```
	Notes For Non-libsmbclient Users
	Setting `exclusive` to `yes` prevents other Easysoft ODBC-Access Driver users from opening the database file while you are connected to the Easysoft ODBC-Access Driver data source.

Attribute	Description
ignore_rel = yes | no	Whether the Easysoft ODBC-Access Driver takes into account any relationships defined for a particular table when doing INSERTs, UPDATEs or DELETEs. For example, with ignore_rel set to yes, the Easysoft ODBC-Access Driver would prevent a record from being deleted if one of its columns was a foreign key for another table. By default, ignore_rel is set to no.
work_mem_size = *value*	The number of rows that the Easysoft ODBC-Access Driver caches into memory before storing them on disk. For example: `work_mem_size = 256`
work_dir_path = *value*	The directory where the Easysoft ODBC-Access Driver temporarily stores result set rows when work_mem_size is exceeded. For example: `work_dir_path = /tmp`
htime_pattern = *value*	Use htime_pattern to specify the format used for hash (#) quoted date constants in queries. Available formats are: MDY (month/day/year) DMY (day/month/year) YMD (year/month/day) For example: `htime_pattern = MDY`

Attribute	Description
unicode_map = 0 | 1 | 2	How Easysoft ODBC-Access Driver maps 16-bit characters to 8-bit characters.
	The `unicode_map` attribute affects: `TEXT`, `MEMO` and `HYPERLINK` data in result sets; metadata (column names); `DELETE`, `INSERT INTO`, `UPDATE` statements (`TEXT` columns only); SQL statement parameters.
	Use `unicode_map` if you experience data loss/corruption when working with character data.
	The available values for the `unicode_map` attribute are:
	`0` When retrieving data, the Easysoft ODBC-Access Driver preserves the low 8 bits of the encoded character and discards the high 8 bits. When `unicode_map` is set to `0`, some applications may replace non-ASCII characters retrieved from the Access database with question marks (?).
	`1` When retrieving data, the Easysoft ODBC-Access Driver preserves ASCII characters and replaces non-ASCII characters with question marks. For example, `Forêts d'érables` becomes `For?ts d'?rables`.
	`2` The Easysoft ODBC-Access Driver converts characters to UTF-8 when retrieving data and from UTF-8 when submitting data (for example, inserting data into `TEXT` columns).
	By default, `unicode_map` is set to `0`.
	Note The version of the Easysoft ODBC-Access Driver driver that supports the `unicode_map` attribute is not currently available on the Easysoft web site, but can be obtained by contacting the Easysoft support team (**support@easysoft.com**).

Figure 3: Easysoft ODBC-Access Driver data source settings.

ENVIRONMENT

The Easysoft ODBC-Access Driver must be able to find the following shared objects, which are installed during the Easysoft ODBC-Access Driver installation:

* `libodbcinst.so`

    By default, this is located in `/usr/local/easysoft/unixODBC/lib`.

* `libeslicshr.so`

    By default, this is located in `/usr/local/easysoft/lib`.

* `libessupp.so`

    By default, this is located in `/usr/local/easysoft/lib`.

* `libestdscrypt.so`

    By default, this is located in `/usr/local/easysoft/lib`.

You may need to set and export `LD_LIBRARY_PATH`, `SHLIB_PATH` or `LIBPATH` (depending on your operating system and run-time linker) to include the directories where `libodbcinst.so`, `libeslicshr.so` and `libessupp.so` are located.

Note	The shared object file extension (`.so`) may vary depending on the operating system (`.so`, `.a` or `.sl`).

E STABLISHING A T EST C ONNECTION

The `isql` query tool lets you test your Easysoft ODBC-Access Driver data sources.

To test the Easysoft ODBC-Access Driver connection

1. Change directory into `/usr/local/easysoft/unixODBC/bin`.

2. Type `./isql -v data_source`, where *data_source* is the name of the Easysoft ODBC-Access Driver data source.

   If you are unable to connect, see **"Troubleshooting Database Connection Problems" on page 76** for help on solving some common connection problems.

3. At the prompt, type an SQL query. For example:

   ```
   SQL> select * from mytable;
   ```

   – OR –

   Type `help` to return a list of tables:

   ```
   SQL> help
   ```

Example: CONNECTING TO AN ACCESS DATABASE ON WINDOWS FROM LINUX

1. On the Easysoft ODBC-Access Driver machine, use `smbmount` to mount the Windows share where the database file is located.

   For example:

```
# smbmount //mywindowsmachine/myshare /mylinuxmountpoint -o
username=mywindowsuser,rw,directio
```

> **Important** For read-write access to the database, you must specify the `directio` option when mounting the share. Check the man page for your version of smbmount to see whether it supports the `directio` option. Specifying `directio` turns off inode data caching on files opened on the mount. The Easysoft ODBC-Access Driver relies on this functionality when managing concurrent access to the database file.
>
> When mounting the share, you need to supply the user name and password of a Windows user who has read, write, create and delete privileges for the share. When testing the Easysoft ODBC-Access Driver with a database file located in a Windows Vista share, this requirement equated to these Windows permissions:

Share permissions	Shared folder permissions
Change Read	Modify Read Write

Figure 4: Windows Vista permissions needed by mount user.

These permissions allow the Easysoft ODBC-Access Driver to write to or create a locking information file (.ldb).

2. Configure the Easysoft ODBC-Access Driver data source.

Example data sources:

```
# This Easysoft ODBC-Access Driver data source opens an Access
# database that is stored on a Windows share. The data source
# opens the database for shared read/write access; other users can
# open the database therefore.
[ACCESS_SAMPLE_WINDOWS_SHARE]
Driver = Easysoft ODBC-ACCESS


# The Windows share (accounts) has been attached under /mnt on the
# Easysoft ODBC-Access Driver machine.
mdbfile = /mnt/accounts/access/Northwind.mdb


# The smb* attributes allow the Easysoft ODBC-Access Driver to
# manage concurrent access to the database.


# The SMB URL for the database on the Windows share.
smbpath = smb://mswin_machine/accounts/access/Northwind.mdb
# The path to the SMB client library on the
# Easysoft ODBC-Access Driver machine
smblib = /usr/lib/libsmbclient.so
```

```
# The user name and password of a Windows user who has read and
# write privileges to the share.
smbuser = mywindows_user
smbauth = mywindows_password


# Opens the database for shared read/write access.
readonly = no
exclusive = no
```

– OR –

```
# This data source opens the same Access database for exclusive
# access; the Easysoft ODBC-Access Driver will prevent other
# users from accessing the database, therefore.
[ACCESS_SAMPLE_WINDOWS_SHARE]
Driver = Easysoft ODBC-ACCESS
mdbfile = /mnt/accounts/access/Northwind.mdb
smbpath = smb://mswin_machine/accounts/access/Northwind.mdb
smblib = /usr/lib/libsmbclient.so
# The user name and password of a Windows user who can access the
# share and who has read and write privileges to the .mdb file.
smbuser = mywindows_user
smbauth = mywindows_password

# Opens the database for exclusive read/write access.
readonly = no
exclusive = yes
```

– OR –

```
# This data source opens the same Access database for read-only

# access; the Easysoft ODBC-Access Driver will prevent its users

# from making updates to the database, therefore.

[ACCESS_SAMPLE_WINDOWS_SHARE]

Driver = Easysoft ODBC-ACCESS

mdbfile = /mnt/accounts/access/Northwind.mdb

smbpath = smb://mswin_machine/accounts/access/Northwind.mdb

smblib = /usr/lib/libsmbclient.so


# The user name and password of a Windows user who has read

# privilege to the share.

smbuser = mywindows_user

smbauth = mywindows_password


# Opens the database for read-only access.

readonly = yes

exclusive = no
```

For more information about Easysoft ODBC-Access Driver data source attributes, see **"Attribute Fields" on page 55**.

EXAMPLE: CONNECTING TO AN ACCESS DATABASE LOCATED ON A SAMBA SHARE

1. Copy the Access database file to a Samba share on the Easysoft ODBC-Access Driver machine.

   The Easysoft ODBC-Access Driver requires the following Samba configuration options to be set for the share in which the database file is located.

Option	Value
oplocks	yes This is the default setting.
posix locking	yes This is the default setting.
blocking locks	yes This is the default setting.
veto oplock files	/*.mdb/*.MDB/*.ldb/*.LDB/*.accdb/*.ACCDB

Figure 5: smb.conf settings for the Access database file share.

2. Configure the Easysoft ODBC-Access Driver data source.

Example data sources:

```
# This Easysoft ODBC-Access Driver data source opens an Access
# database that is stored on a Samba share. The data source
# opens the database for shared read/write access; other users can
# open the database therefore.
[ACCESS_SAMPLE_SAMBA_SHARE]
Driver = Easysoft ODBC-ACCESS


mdbfile = /home/samba/sales/access/Northwind.mdb


# The smb* attributes allow the Easysoft ODBC-Access Driver to
# manage concurrent access to the database.


# The SMB URL for the database on the Samba share.
smbpath = smb://samba_server/sales/access/Northwind.mdb
# The path to the SMB client library on the
# Easysoft ODBC-Access Driver machine
smblib = /usr/lib/libsmbclient.so
# The user name and password of a Samba user who has read/write
# access to the share.
smbuser = my_samba_share_valid_user
smbauth = my_samba_share_valid_password
```

```
# Opens the database for shared read/write access.

readonly = no

exclusive = no
```

                              – OR –

```
# This data source opens the same Access database for exclusive

# access; the Easysoft ODBC-Access Driver will prevent other

# users from accessing the database, therefore.

[ACCESS_SAMPLE_SAMBA_SHARE]

Driver = Easysoft ODBC-ACCESS

mdbfile = /home/samba/sales/access/Northwind.mdb

smbpath = smb://samba_server/sales/access/Northwind.mdb

smblib = /usr/lib/libsmbclient.so


# The user name and password of a Samba user who has read/write

# access to the share.

smbuser = my_samba_share_valid_user

smbauth = my_samba_share_valid_password


# Opens the database for exclusive read/write access.

readonly = no

exclusive = yes
```

– OR –

```
# This data source opens the same Access database for read-only

# access; the Easysoft ODBC-Access Driver will prevent its users

# from making updates to the database, therefore.

[ACCESS_SAMPLE_SAMBA_SHARE]

Driver = Easysoft ODBC-ACCESS

mdbfile = /home/samba/sales/access/Northwind.mdb

smbpath = smb://samba_server/sales/access/Northwind.mdb

smblib = /usr/lib/libsmbclient.so


# The user name and password of a Samba user who has read

# access to the share.

smbuser = my_samba_share_valid_user

smbauth = my_samba_share_valid_password


# Opens the database for read-only access.

readonly = yes

exclusive = no
```

Troubleshooting Database Connection Problems

This section lists some common connection problems and their solutions.

- **Failed to open MDB file**
- **Failed to open SMB channel**
- **Could not open/create lock file, check sharing permissions**

Failed to open MDB file

The Easysoft ODBC-Access Driver can open either a local or remote Access database file (.mdb or .accdb). The path to the database file is specified in the Easysoft ODBC-Access Driver data source in `/etc/odbc.ini.`

In the Easysoft ODBC-Access Driver data source, check the `mdbfile` attribute value.

Use `ls` to check that the `mdbfile` attribute in your data source specifies a valid path and that the directory is accessible.

If the `ls` command output contains "permission denied", contact your administrator. The directory containing the database file needs to be accessible (execute permission set) to you.

If the path is not valid, check with your administrator whether the database file is located on another machine. (The Easysoft ODBC-Access Driver can open a database file that is located on a Windows share.) If the database file is remote, use the `mount` command to check whether the share has been attached on the Easysoft ODBC-Access Driver machine. There should be an entry in the `mount` command output that corresponds with the directory specified with `mdbfile`. For example:

```
$ mount

/dev/sda3 on / type ext3 (rw,errors=remount-ro)

proc on /proc type proc (rw,noexec,nosuid,nodev)

/sys on /sys type sysfs (rw,noexec,nosuid,nodev)

.

.

.

//mywindowsmachine/myshare on /mnt/myshare type smbfs (rw)

$ grep mdbfile /etc/odbc.ini

mdbfile = /mnt/myshare/Northwind.mdb
```

If the `mount` command does not show the mount point specified in the `mdbfile` attribute, consult with your administrator. The share on which the database file is located needs to be mounted.

If the database file is on a Windows share, the user specified when mounting the share needs to have read, write, create and delete permissions for the share. When testing the Easysoft ODBC-Access Driver with a database file located in a Windows Vista share, this requirement equated to these Windows permissions:

Share permissions	Shared folder permissions
Change Read	Modify Read Write

If the database file is located on Unix, the Easysoft ODBC-Access Driver will also return a "Failed to open MDB file" if:

1. The user running the application that is connecting to the data source only has read access to a local database file.

   – OR –

   The user specified by `smbuser` only has read access to a database file located on a Samba share.

2. The `readonly` attribute in the data source is set to `yes`. (Change this attribute's value to `no` and try again.)

Failed to open SMB channel

The Easysoft ODBC-Access Driver uses the `libsmbclient` library, which is part of the Samba suite, to:

- Let Windows know that it has opened a database file (.mdb or .accdb).

- Prevent Windows users from opening a database file it has opened for exclusive access.

Without this mechanism, there is the potential for database file corruption when Windows users and Easysoft ODBC-Access Driver users are working with the database specified by `mdbfile`.

To configure the driver for use with `libsmbclient` you use the `smb*` attributes in the driver data source.

If the "Failed to open SMB channel" error contains "file not found", check the `smbpath` attribute in the data source. The `smbpath` attribute needs to specify the SMB URL for the database file specified with `mdbfile`. The SMB URL has the format `smb://host/share/path/filename`.

In the following example, a remote Access database named `Northwind.mdb` is located on the Easysoft ODBC-Access Driver machine in `/mnt/myshare_mountdir`. The `mount` command shows that `/mnt/myshare_mountdir` is the mount point for a share named `myshare` on a machine named `myremotemachine`.

```
$ grep mdbfile /etc/odbc.ini

mdbfile = /mnt/myshare_mountdir/Northwind.mdb

$ mount

//myremotemachine/myshare on /mnt/myshare_mountdir type smbfs (rw)
```

The SMB URL for the database file would therefore be:

```
smb://myremotemachine/myshare/Northwind.mdb
```

If the "Failed to open SMB channel" error contains "errno=13", check the `smbuser` and `smbauth` attributes in the data source.

If the database file is located on a Samba share, the `smbuser` and `smbauth` attributes need to specify the user name and password for a Samba user (created with Samba tools such as `smbpasswd`) who has read access to the share. To open a database file for read-write access (`readonly` data source attribute set to `no`), the Samba user also needs write access to the share).

If the database file is located in a Windows share, the `smbuser` and `smbauth` attributes need to specify the user name and password for a Windows user who can access the share. If `smbuser` only has read privilege to the database file (as opposed to read and write), the `readonly` attribute in the data source needs to be set to `yes`. Otherwise, an "errno=13" error will be returned.

Could not open/create lock file, check sharing permissions

To allow its users to safely edit a database file that Windows users may also be accessing and updating, the Easysoft ODBC-Access Driver uses a locking information file (.ldb).

The driver uses a .ldb file to:

- Determine which records in the database are locked and by whom.

- Let other applications know which records in the database are locked by a Easysoft ODBC-Access Driver user.

The Easysoft ODBC-Access Driver may have to create and delete a .ldb file as well as write to it. If directory permissions prevent the Easysoft ODBC-Access Driver from writing to/creating a .ldb file, the driver may return a "Could not open/create lock file, check sharing permissions" error.

If you get this error when opening a database file that **Windows users do not require access to**, specify an alternative .ldb file directory with the `lockfile` data source attribute. Specify a directory where you and other users who connect to the data source have write access to. For example:

```
lockfile = /tmp/Nwind.ldb
```

Permissions Summary

This table lists the various permissions that may need to be set to allow the Easysoft ODBC-Access Driver to open your Access database file (.mdb or .accdb).

Permission	Applies to	Type	Notes
execute write	database file directory	Local Unix permissions on Easysoft ODBC-Access Driver machine	The local directory from which the database file is opened must be accessible (execute permission set) by the user who is connecting to the data source. The local directory must also be writeable by the user who is connecting to the data source if the database file is opened for shared access (`exclusive` data source attribute set to `no`).
read write	database file	Local Unix permissions on Easysoft ODBC-Access Driver machine	The database file must be readable (read permission set) by the user who is connecting to the data source. If the database file is opened for read/write access (`readonly` data source attribute set to `no`), the database file must also be writeable (write permission set) by this user.

Permission	Applies to	Type	Notes
read write	Samba share containing database file	Samba	These permissions are only applicable if the database file is located on a Samba share on Unix. The Samba user specified by the `smbuser` data source attribute needs to have read access to the share. If the database file is opened for read/write access (`readonly` data source attribute set to `no`), this user must also have write access to the share.
read write create delete	Windows share containing database file	Windows	These permissions are only applicable if the database file is located on a Windows share. The Windows user whose authentication details are supplied when mounting the share needs to have read, write, create and delete privileges on the share. Note that on some versions of Windows, the create and delete privileges may be replaced by the modify privilege.

Permission	Applies to	Type	Notes
read write	Windows share containing database file	Windows	These permissions are only applicable if the database file is located on a Windows share. The Windows user specified by the `smbuser` data source attribute needs to have read privilege on the share. If the database file is opened for read/write access (`readonly` data source attribute set to `no`), this user must also have write privilege.

Figure 6: Access database file and directory permissions.

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-ACCESS};
                MDBFILE=path;"...)
```

where *path* is the path to the database file on the Easysoft ODBC-Access Driver machine.

Other Easysoft ODBC-Access Driver attribute settings, as described in **"Setting Up Data Sources" on page 52**, can be added to the connection string using the same `PARAMETER=value;` format. For example:

```
"DRIVER={Easysoft ODBC-ACCESS};MDBFILE=/home/myuser/ms_access/Northwind.mdb;"
```

APPENDIX A TECHNICAL REFERENCE

Technical Reference for the Easysoft ODBC-Access Driver

This section contains extra information relating to the deployment of the Easysoft ODBC-Access Driver.

Appendix Guide

- **ODBC Conformance**
- **Supported Data Types**
- **Threading**
- **Tracing**

ODBC Conformance

The Easysoft ODBC-Access Driver complies with the ODBC 3.51 specification.

The Easysoft ODBC-Access Driver is Level 1 compliant.

ODBC API SUPPORT

All ODBC 3.51 calls are fully supported apart from.

- `SQLBrowseConnect`
- `SQLBulkOperations`
- `SQLColumnPrivileges`
- `SQLTablePrivileges`

Supported Data Types

The Easysoft ODBC-Access Driver supports the following Access data types:

- `BINARY`

  `VARBINARY`

- `BIT`

- `TINYINT`

  `BYTE`

- `COUNTER`

- `CURRENCY`

- DATETIME

  DATE

  TIME

- REAL

- FLOAT

  DOUBLE

- SMALLINT

- INTEGER

  INT

- DECIMAL

  NUMERIC

  DEC

- TEXT

  LONGCHAR

  MEMO

- IMAGE

- HYPERLINK

Threading

The Easysoft ODBC-Access Driver is thread safe in accordance with the ODBC specification and can safely be used behind threaded applications.

Tracing

The ODBC calls an application makes can be traced:

- Within the Driver Manager by an application.

- From within the Driver Manager.

WITHIN THE DRIVER MANAGER BY AN APPLICATION

An application can turn tracing on in the Driver Manager by using the ODBC API `SQLSetConnectAttr` (...,`SQL_ATTR_TRACE`,...).

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

FROM WITHIN THE DRIVER MANAGER

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

```
Trace = Yes

TraceFile = logfile
```

For example:

```
[ODBC]

Trace = Yes

TraceFile = /tmp/unixodbc.log
```

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

APPENDIX B GLOSSARY

Terms and definitions

Application Programmer Interface (API)

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

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.

Data source

A database or other data repository coupled with an ODBC Driver, which has been given a Data Source Name (see **"DSN" on page 90**) to identify it to the ODBC Driver Manager.

Data type

An attribute that specifies what type of information can be stored in a column, parameter, or variable.

Driver Manager

Software whose main function is to load ODBC drivers. ODBC applications connect to the Driver Manager and request a data source name (DSN). The Driver Manager loads the driver specified in the DSN's configuration file. On 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.

DSN-less connection

A type of data connection that is created based on information in a data source name (DSN), but is stored as part of a project or application. DSN-less connections are especially useful for Web applications because they let you move the application from one server to another without re-creating the DSN on the new server.

License key

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

ODBC

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

ODBC driver

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

INDEX

E

H

I

P

R

S

T