Oracle Darwin

Darwin Installation and Administration

Release 3.6 for Solaris

January 2000

Part No. A81050-01

ORACLE®

Darwin Installation and Administration, Release 3.6 for Solaris

Part No. A81050-01

Contents

Send Us Your Comments vii

Preface ix

Intended Audience ix
Structure ix
Related Documents x
Conventions xii

1 Overview

1.1 Installation Overview 1-1
1.2 Where to Go for More Information 1-2

2 Requirements

2.1 Hardware and Operating System Requirements 2-1
2.1.1 Solaris Requirements and Recommendations 2-2
2.2 Disk Space Requirements 2-3
2.2.1 Server Disk Space Requirements 2-3
2.2.2 Client Disk Space Requirements 2-3
2.3 Network Connection Requirements 2-3
2.4 Client Requirements 2-3
2.5 Requirements for Code Generation 2-4
2.6 SAS Conversion Utilities Requirements (Optional) 2-5
2.7 Database Connectivity Requirements (Optional) 2-6
2.7.1 Oracle Database 2-6

2.7.2 Database Drivers ... 2-6
2.8 Requirements for Darwin on Multi-Node Servers ... 2-7
2.8.1 Performance on Multi-Node Servers ... 2-7
2.8.2 Model Building on Multi-Node Servers ... 2-7
2.8.3 Server Capacity Planning Guidelines ... 2-7
2.8.4 Distributed Datasets ... 2-8
2.8.5 Moving Serial and Distributed Datasets ... 2-9

3 Installing the Darwin Server

3.1 Upgrading Darwin ... 3-1
3.1.1 Converting Darwin Datasets ... 3-1
3.1.2 Converting Darwin Models ... 3-2
3.1.3 Converting Darwin Server Configurations ... 3-2
3.2 Installing Darwin Server Software on Solaris ... 3-3

4 Configuring Darwin Servers

4.1 Darwin Servers ... 4-2
4.2 Darwin Configuration Files ... 4-3
4.2.1 darwinservers File ... 4-3
4.2.2 DarwinServers.txt File ... 4-4
4.3 Before Using darwinconfig ... 4-4
4.3.1 DARWINHOME Environment Variable ... 4-5
4.3.2 Upgrading Existing Configuration Files ... 4-5
4.4 Configuring and Managing Darwin Servers ... 4-5
4.4.1 Listing Available Servers ... 4-6
4.4.2 Adding and Configuring a Server ... 4-7
4.4.3 Removing a Server ... 4-8
4.4.4 Describing a Server ... 4-8
4.4.5 Verifying the Validity of a Configuration File ... 4-8
4.4.6 Starting a Server ... 4-9
4.4.7 Stopping a Server ... 4-9
4.4.8 Saving a Configuration File ... 4-10
4.4.9 Ending a darwinconfig Session ... 4-10
4.4.10 Getting Help ... 4-10
4.4.11 Modifying Configuration Files ... 4-10

4.4.12 Darwin Environment Variables 4-10
4.4.13 Troubleshooting Configurations 4-11

5 Database Connectivity

5.1 Supported Database Software 5-2
5.2 Obtain and Install Database Driver 5-2
5.2.1 Connecting to Oracle Using DataDirect Connect ODBC 5-3
5.3 Permissions and Accessibilty for Oracle 8*i* 5-4
5.4 Darwin Server Architecture for Database Connectvity 5-4
5.5 Darwin Environment Variables for Oracle 8*i* 5-4
5.5.1 ORACLE_HOME Environment Variable 5-5
5.5.2 LD_LIBRARY_PATH Environment Variable 5-5
5.5.3 ORACLE_SID Environment Variable 5-5
5.6 Required Files 5-5
5.6.1 /etc/services File 5-6
5.6.2 .odbc.ini File 5-6
5.6.3 tsnames.ora File 5-6
5.7 Installation Check 5-7
5.8 Using Database Connectivity 5-7
5.8.1 Connecting to a Database Server 5-7
5.8.2 Disconnecting from a Database 5-8
5.8.3 Database Import Wizard 5-8
5.8.4 Database Export Wizard 5-9

6 Installing the Darwin Client

6.1 Collect Server Information 6-1
6.1.1 Create a Customization Disk or File (Optional) 6-2
6.2 Installing Client Software 6-3
6.2.1 DARWINHOME Windows Environment Variable 6-3
6.3 First-Time Installation 6-3

7 Starting and Stopping Darwin

7.1 Starting Darwin 7-1
7.1.1 Starting Darwin Servers 7-1

7.1.2 Starting a Darwin Client ... 7-1
7.2 Stopping Darwin ... 7-2
7.2.1 Stopping Darwin Clients ... 7-2
7.2.2 Stopping Darwin Servers ... 7-3

8 Administering Darwin Projects

8.1 Darwin Projects ... 8-1
8.2 Creating Projects ... 8-1
8.2.1 Recording Project Information ... 8-2
8.3 Deleting Projects ... 8-3
8.4 Administrator Decisions about Projects ... 8-3
8.5 The .darwinprojects File ... 8-4
8.5.1 A Sample .darwinprojects File ... 8-4
8.5.2 Modifying the .darwinprojects File ... 8-5

A Installing the SAS Conversion Utilities

A.1 Installing the Software ... A-1

B Server Configuration Files

B.1 Darwin Configuration Files ... B-1
B.1.1 Server Configuration Files ... B-1
B.2 What Configuration Files Mean ... B-5

Index

Send Us Your Comments

Darwin Installation and Administration, Release 3.6 for Solaris

Part No. A81050-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the chapter, section, and page number (if available). You can send comments to us in the following ways:

- DARWINDOC@us.oracle.com
- FAX: 781-684-7738. Attn: Oracle Darwin
- Postal service:
  Oracle Corporation
  Oracle Darwin Documentation
  200 Fifth Avenue
  Waltham, Massachusetts 02451
  U.S.A.

If you would like a reply, please give your name, address, and telephone number below.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Darwin is a data mining application designed specifically to handle multiple gigabytes of data, and to provide answers to complex problems of data classification, prediction, and forecasting.

This manual describes how to install and administer Darwin 3.6 server software on selected Sun platforms running Solaris 2.6 or 2.7 and client software on PCs running Windows NT 4.00, Windows 95, or Windows 98. There is a separate manual that describes installing and administering Darwin 3.6 server software on HP-UX.

Intended Audience

This manual is intended for experienced Solaris system administrators.

Structure

This manual contains eight chapters as follows:

Chapter 1	**Overview** Provides an overview of the installation procedure.
Chapter 2	**Requirements** Lists the hardware and software requirements.
Chapter 3	**Installing the Darwin Server** Describes how to install the Darwin server software.
Chapter 4	**Configuring and Starting a Darwin Server** Describes how to configure and start one or more Darwin servers.

Chapter 5	**Database Connectivity** (optional) Describes how to set up the environment for database connectivity.
Chapter 6	**Installing the Darwin Client** Describes how to install Darwin client software.
Chapter 7	**Starting and Stopping Darwin** Describes how to start and stop Darwin servers.
Chapter 8	**Administering Darwin Projects** Describes some administrative functions associated with Darwin projects, and describes how to create a project.

In addition, there are two appendixes that cover supplementary topics:

Appendix A	**Installing the SAS Conversion Utilities** (optional) Explains how to install `sas2darwin` and `darwin2sas`, utilities used for converting between Darwin datasets and SAS files.
Appendix B	**Server Configuration Files** Explains the meaning of entries in a Darwin configuration file.

Related Documents

Darwin documentation is distributed on the documentation CD in PDF and HTML formats.

The complete Darwin documentation set includes

- *Darwin New Features, Release 3.6*. Describes the features introduced at Release 3.5 and Release 3.6.

  This manual is a revision of *Darwin 3.5 New Features*, which described the functionality introduced at Release 3.5. *Darwin New Features, Release 3.6* contains updated information about the functionality introduced at Release 3.5 plus new material describing functionality introduced at Release 3.6.

  If you are upgrading from 3.5 to 3.6, you can discard the manual *Darwin 3.5 New Feature*s; the present manual supersedes it.

- *Darwin 3.6 Release Notes for Solaris.* Describes the release, documents any problems or bugs in the software, and describes any changes that occurred in the software after the manuals were finalized. There are separate release notes for Solaris and for HP-UX.
- For Solaris system administrators: *Darwin Installation and Administration, Release 3.6 for Solaris* (this manual). Describes how to install server and client software. Changes that developed after this manual was finalized are described in the release notes. There are separate installation / administration guides for Solaris and for HP-UX.
- *Using Darwin, Release 3.0.1.* A how-to manual; describes the user interface and provides detailed instructions for using it. (*Using Darwin* describes all the features available at Release 3.0.1; together with *Darwin New Features*, you have a complete description of the user interface at Release 3.6.)
- *Darwin Reference, Release 3.0.1* (companion volume to *Using Darwin*). Introduces data mining and Darwin; provides background and conceptual material on datasets, Darwin tools, and analyses.

Documentation CD

All the Darwin documentation is available in HTML and PDF format on the documentation CD; you can read or print documentation directly from the CD.

To view the PDF files, you will need

- Adobe Acrobat Reader 3.0 or later, which you can download from `www.adobe.com`.

To view the HTML files, you will need

- Netscape 2.x or later, or
- Internet Explorer 4.0 or later

Darwin Online Help

Darwin includes extensive online help that can be summoned from a list of contents, from the **Help** tab, and from Help buttons or the F1 key on dialog windows. For correct display of Darwin's online help, you need Internet Explorer 4.x. If you do not have it, you can download it from `www.microsoft.com`.

Conventions

The following conventions are used in this manual:

Convention	Meaning
boldface	Darwin commands, menu names, menu items, names of dialogs and screens.
Project > New File	Indicates the path for a command. The example shown means on the **Project** menu, click the **New File** command.
`code`	Data fields and values, special characters, etc., examples of files, data, filenames, and pathnames.
`italics`	Argument names and placeholders in command formats.
`% user input` `system output`	In interactive examples, user input is shown in bold typewriter, and system output is shown in regular typewriter.

1

Overview

This manual describes installation and administrative procedures for Darwin Release 3.6 on selected Sun platforms running Solaris 2.6 or 2.7 with client software running on Microsoft Windows NT 4.00, Windows 95, or Windows 98.

1.1 Installation Overview

Darwin is distributed on three CDs:

- the Darwin server CD, formatted to be read by Solaris
- the Darwin client CD, formatted to be read by Windows NT and Windows 95/98
- the Darwin documentation CD, formatted to be read by Windows NT and Windows 95/98

Each CD contains a file `README.txt` that describes the contents of the CD and contains other useful information.

Note: You *must* install the same version of the Darwin client and server.

To install Darwin, follow these steps:

- Confirm that all requirements are met (Chapter 2)
- Install the Darwin server on Solaris (Chapter 3)
- Configure and start at least one Darwin server (Chapters 4 and 8)

Note: The supported server architectures have changed; existing configurations may no longer work properly; see Chapters 2 and 4.

- Set up for database connectivity (optional) (Chapter 5)
- Install Darwin clients on Windows NT/95/98 (Chapter 6)
- Start and stop Darwin servers (Chapter 7)
- Create a Darwin project (Chapter 8)

The chapters in this manual cover these steps in order, beginning with Chapter 2.

1.2 Where to Go for More Information

See the preface for a complete list of related Darwin documentation.

Darwin also provides extensive online help that can be summoned from a list of contents and from Help buttons or the F1 key on dialog windows.

2

Requirements

This chapter describes the hardware, software, disk space, network, and other requirements for the Release 3.6 client and server.

2.1 Hardware and Operating System Requirements

Darwin 3.6 server software runs on UNIX (Solaris 2.6 and 2.7 or HP-UX 11.0).

The Darwin 3.6 server software for Solaris runs on the following hardware platforms:

- Sun Microsystems UltraSPARC workstations running Solaris 2.6 or 2.7
- Sun Microsystems Ultra Enterprise Servers running Solaris 2.6 or 2.7

If you will run Darwin on a multinode Solaris server, see Section 2.8 for information about hardware requirements.

The Darwin 3.6 client (user interface) software runs on the following platforms:

- personal computers running Microsoft Windows NT 4.00, Microsoft Windows 95, or Microsoft Windows 98

On any of the PC platforms, Microsoft Internet Explorer 4.x is required to display online help. Without Internet Explorer, Darwin functions properly but online help is not available.

We recommend that the PCs that Darwin clients run on should have a CPU that is at least Pentium 100 MHz or equivalent and should have at least 16 Mbytes of RAM for Windows 95 and Windows 98 systems and 32 Mbytes of RAM for Windows NT systems.

2.1.1 Solaris Requirements and Recommendations

Solaris 2.6 Recommendations

If you plan to run Darwin servers on Solaris 2.6, you should upgrade to the latest recommended patches (including Y2K patches) from Sun.The kernel patch level should be version 15 or patch number 105181-15. (Use the command **uname -v** to check the kernel patch level.) This patch, along with others, is included in the Sun recommended patch clusters available via **ftp** at:

`ftp://sunsolve.sun.com/pub/patches`

You should download and install the following patches:

- 2.6_Recommended.tar.Z
- `2.6_y2000_ALL.tar.Z`

Solaris Requirement for Multinode Systems (Optional)

If you plan to run Darwin on a multinode system (Ultra Enterprise server or SMP), you must edit the `/etc/system` file on the system before you install Darwin server. These changes are required to enable shared libraries. If you are not going to install Darwin on a multinode system, you do not have to make these changes.

Log in to the multinode Solaris system as root. Add the following entries to the end of the file `/etc/system`:

```
set pt_cnt=1024
set npty=1024
set sadcnt=2048
set nautopush=1024
set dosynctodr=0
set shmsys:shminfo_shmmax=2120000000
set shmsys:shminfo_shmmni=200
set shmsys:shminfo_shmseg=200
```

After you've made the changes, reboot your system before installing Darwin server software. The changes to `/etc/system` don't take effect until you reboot.

2.2 Disk Space Requirements

2.2.1 Server Disk Space Requirements

The Darwin server requires approximately 20 Mbytes of disk space. This is the amount of space the Darwin software itself requires, and does not include space required for any database files or swap space.

2.2.2 Client Disk Space Requirements

The Darwin client requires approximately 16 Mbytes of disk space on Windows NT, Windows 95, or Windows 98.

2.3 Network Connection Requirements

Darwin requires a direct network connection (via TCP/IP) from any personal computer running Darwin client software to Solaris.

2.4 Client Requirements

The Darwin client requires the following:

- A direct network connection from any personal computer running Darwin client software to Solaris.
- Microsoft Internet Explorer 4.x, required for the operation of Darwin online help. Internet Explorer 4.x must be installed on the PC *before* you install the Darwin client.

Note: If Internet Explorer is not installed on the PC where Darwin client is installed, no online help will be displayed. (**Help** buttons will not work; the **F1** key will not display help; the **Help** tab links will not work; the **Help** button on the tool bar will not work; all items in the **Help** menu, except for **Help > Version** will not work.)

Except for the lack of online help, Darwin installed on a PC without Internet Explorer will work properly.

You can download Internet Explorer 4.x from the following site on the World Wide Web:

`www.microsoft.com/windows/ie/download/ie5all.htm`

Click “To the Internet Explorer download index” at the bottom of the page. This takes you to a complete list of the downloads, arranged by platform. One of the options under "Windows 95 and Windows NT 4.0" and under "Windows 98" is "Internet Explorer 4.01 with Service Pack 2".

Internet Explorer must be installed on your system; it does not have to be your default browser.

Note: Darwin has not been tested with Internet Explorer 5.0.

- Microsoft Excel (used to create plots and graphs); one of the following is required:
  - Microsoft Excel 95 for Windows 95, Version 7.0
  - Microsoft Excel 97 (Excel version 8)

Warning: If Excel 95 or Excel 97 is not installed on your PC, you will not be able to create any plots or graphs using Darwin.

Note: Darwin has not been tested with Office 2000.

2.5 Requirements for Code Generation

You can export Darwin models as C, C++, or Java code using the **Code Generation** command of the **Options** menu. You can generate model code for tree or net models, but not for match models.

Code generation is enabled by default; in previous releases of Darwin, a license was required. If you already have a license for code generation, you do not have to remove it; it is ignored.

2.6 SAS Conversion Utilities Requirements (Optional)

If you do not plan to convert SAS files, you may skip this section.

The SAS conversion utilities run on UNIX (Solaris and HP-UX). They require the following software to be installed on the UNIX system on which you plan to do conversions:

- Conceptual Software, Inc., DBMS/COPY, version 7 or higher
- Perl, version 5.001 or higher

DBMS/COPY

You must obtain DBMS/COPY from

Conceptual Software, Inc.
9660 Hillcroft #510
Houston, TX 77096
USA
Telephone: 1-800-328-2686 or 1-713-721-4200

For information about ordering DBMS/COPY, see Conceptual Software's page on the World Wide Web at `www.conceptual.com`. The price (when this manuals was published) is approximately $500 (US) for an individual license.

Perl

If Perl is not already installed on your Solaris system, you can obtain it free of charge from several locations. For details, see `www.perl.org` or `www.perl.com` on the World Wide Web. If you do not wish to compile and build Perl for your system, you can download a Solaris version of Perl directly from `www.sunfreeware.com`. The web page includes directions for downloading.

Disk Space

DBMS/COPY and the Perl scripts require approximately 6.7 Mbytes of disk space in addition to the space required by Darwin.

When the SAS conversion utilities convert a file, they create temporary files; these temporary files require at least as much space as the file being converted.

For information about installing the conversion utilities, see Appendix A of this manual; for information about using the utility, see *Using Darwin*.

2.7 Database Connectivity Requirements (Optional)

If you are not planning to connect to Oracle databases, you may skip this section.

Darwin supports connectivity to Oracle databases on UNIX only. You cannot connect to databases on Windows 95, Windows 98, or Windows NT.

Note: If you are planning to install Darwin servers that connect to Oracle, you must install Oracle database software and DataDirect Connect ODBC before you install Darwin server software.

2.7.1 Oracle Database

You will need one of the following Oracle products installed at your site:

- Oracle Version 7.3 and SQL*NET Version 2.3
- Oracle Version 8*i* Enterprise Edition

For information about obtaining and installing these products, contact Oracle:

- Telephone: 1-800-ORACLE1
- Oracle's web site: `www.oracle.com/database/oracle8/`

2.7.2 Database Drivers

Darwin uses the MERANT (Micro Focus) DataDirect Connect ODBC version 3.5 drivers for database connectivity. You must purchase them directly from MERANT. For information about DataDirect Connect ODBC drivers, contact DataDirect:

- Telephone: 1-800-876-3101 or 1-919-461-4200
- MERANT's web site: `www.merant.com/datadirect/index.asp`

You can purchase DataDirect Connect ODBC online at the web site or by telephone. You can also download a 30-day evaluation copy of DataDirect Connect ODBC from the web site.

MERANT documentation describes how to install, configure, and test DataDirect.

For information about setting up for database connectivity, see Chapter 5 of this manual; for information about using database connectivity, see *Using Darwin* and *Darwin 3.6 New Features*.

2.8 Requirements for Darwin on Multi-Node Servers

This section describes things to keep in mind when running Darwin on multi-node servers.

If you do not plan to run Darwin on multinode servers, you can ignore this section.

For information about settings that must be added to `/etc/system` before installing Darwin on a multi-node system, see Section 2.1.1.

2.8.1 Performance on Multi-Node Servers

Optimal performance of Darwin running on a multi-node server (SMP) is achieved when Darwin is the only program running on that SMP. If other programs are running on the SMP at the same time that Darwin is running, Darwin performance will be poor.

Also, performance on a multi-node server is best when only one user at a time runs Darwin on any given set of CPUs. This restriction does not apply to multiple Darwin users on a single CPU system.

2.8.2 Model Building on Multi-Node Servers

When you build a model on a multi-node server, model building is always distributed, even if the datasets are serial. A serial dataset will be distributed as the operation begins, and re-serialized at its conclusion. The most efficient way to run a sequence of operations is to explicitly distribute the data, as described in Section 2.8.4, before launching the first operation.

2.8.3 Server Capacity Planning Guidelines

Below are some guidelines to be taken as "rules of thumb" in planning server capacity.

Sizing the Number of Server CPUs

Model building requires the most CPU resources to run the hundreds and thousands of calculations the algorithms perform against the training datasets. The testing and evaluating phases take less CPU resources, and scoring the full database with the deployed model takes the least CPU resources. Below is a Darwin performance table that you can use to derive the number of CPUs necessary to achieve a desired performance level; note the numbers shown are valid for a particular dataset and not necessarily accurate for different datasets.

Model type	Number of CPUs			
	4	6	8	16
C&RT	23 min	15 min	12 min	6 min
Neural Net	93 min	55 min	47 min	24 min

Sizing Server RAM Memory

Model development is most efficient when the full training, testing, and evaluation datasets are in RAM memory at least individually.

Minimum memory size:

max number of records for modeling **x** max record size = ____MG / GB

Sizing Server Disk Storage Capacities

Allow 100 percent additional disk storage space over the raw database total storage requirements for database administration, transient model storage, and scored dataset storage.

2.8.4 Distributed Datasets

All Darwin datasets are created, by default, as serial datasets. If you wish to create a distributed version of a dataset, you must explicitly create it. In particular, if you create a dataset using either **Dataset > Create** or the **Text Import** Wizard, the resulting dataset is always serial.

Operations such as frequency counts or model building, performed on a distributed version of a dataset, are often faster than the same operations performed on the undistributed (serial) dataset. If you want to perform an operation on a distributed dataset, you must distribute the dataset before you perform the operation. Once you distribute the dataset, you can save the distributed version for future use.

Here are two ways to distribute a dataset; in both ways, Darwin creates a new dataset that is the distributed version of the original dataset:

- Use the **Randomize** transform to randomize the dataset. If you are connected to a distributed server, the transformed dataset is distributed in memory; you can save the transformed distributed dataset if you wish. The default name of the

randomized dataset is *dataset-name*[randomize]. (The original dataset that you applied the transformation to is still serial.)

- Select the dataset. Go to the **Datasets** tab of **Options > Advanced**; click the **Create Distributed** button. Darwin creates a distributed version of the dataset named *dataset-name*[distributed]; the dataset is automatically saved. The underlying file for the new dataset has the extension .dst.

For large datasets, either of these operations may take several minutes or more. See also Section 2.8.5, below.

For more information about either of these operations, see *Using Darwin*.

2.8.5 Moving Serial and Distributed Datasets

If you move your dataset between servers with different numbers of nodes, serialize the dataset first. Here's how:

- Click the dataset's name to select it.
- Click **Options > Advanced**.
- Click the **Datasets** tab.
- In the **Serial or Distributed** section, click **Create Serial**.

Darwin creates a serial version of the dataset named *<name>*[serial] (e.g., the serial version of Ceil is Ceil[serial]).

3

Installing the Darwin Server

The first step in the Darwin installation process is installing the Darwin server software on Solaris, as described in this chapter.

3.1 Upgrading Darwin

If Darwin is already on your system, you must stop (kill) any daemons associated with Darwin (via `darwinconfig`) and remove the existing version of Darwin (via `pkgrm`) before you install the new version. If you wish, you can move the current version of Darwin from `/opt/TMCdarwin` before using `pkgrm` and before installing the new one.

If you have a license for Darwin 3.x, you do not have to delete the license. Darwin no longer requires license files; existing license files are ignored.

Before you deinstall pre-3.0 versions of Darwin, you should convert datasets and models, as described below. Darwin 3.5 models and datasets do not need any conversion.

The list of supported architectures has changed. Server configurations that worked with previous versions of Darwin, including Darwin 3.6 Beta, may no longer work. See Section 3.1.3 for information.

3.1.1 Converting Darwin Datasets

Datasets created by Darwin 2.1, 2.0.1, or 2.0 are not recognized properly by Darwin 3.6. If you plan to use Darwin 2.x datasets with Darwin 3.6, you must convert them.

All versions of Darwin recognize *text data files.*You convert Darwin 2.x datasets to Darwin 3.6 as follows:

- Before deinstalling Darwin 2.x, export all 2.x datasets that you intend to use with Darwin 3.6 as data files and associated descriptors.
- After you install Darwin 3.6, use the exported data files and descriptors to create datasets in the Darwin 3.6 environment.

Darwin 3.0, 3.0.1, 3.0.2, or 3.5 datasets do not have to be converted.

3.1.2 Converting Darwin Models

You cannot use models created with pre-3.0 versions of Darwin (such as Darwin 2.0.1) with Darwin 3.6; you must recreate the models using Darwin 3.6.

Models created with Darwin 3.0.1, 3.0.2, or 3.5 do not have to be recreated.

3.1.3 Converting Darwin Server Configurations

The list of supported Darwin server architectures has changed. The following architectures are no longer supported in Darwin 3.6:

- **`sparcsmp`**
- **`hpuxpar`**

The architectures **`sparc`** and **`hpux`** do not have the same meaning that they did in releases of Darwin earlier than the 3.6 final release.

The supported 3.6 server architectures are

- **`sparc`** (Solaris with no database connectivity)
- **`sparcdb`** (Solaris with database connectivity)
- **`hpux`** (HP-UX with no database connectivity)
- **`hpuxdb`** (HP-UX with database connectivity)

Existing server configurations must be recreated to reflect the supported architectures.

3.2 Installing Darwin Server Software on Solaris

Warning: If you are planning to configure Darwin servers that connect to Oracle databases, you must install Oracle database software and DataDirect Connect before you install Darwin server software.

For information about settings that must be added to `/etc/system` before installing Darwin on a multi-node system, see Section 2.1.1.

The Darwin server software is distributed as a tarfile on a CD. To install the Darwin server, follow these steps:

1. Install database software and Data Direct Connect if you plan to configure servers that support database connectivity.
2. Log in as root to the Solaris system where the Darwin server software will reside.
3. If any Darwin servers are already installed on the system and are running, stop each one using the following command, as described in Section 4.4.7

   ```
   /opt/TMCdarwin/etc/darwinconfig stop server-name
   ```

   where *`server-name`* is the name of the server.
4. Remove Darwin server software using the command

   ```
   pkgrm TMCdarwin
   ```

5. Make sure that there is no copy of the Darwin server software in the directory `/tmp` and that `/tmp` is large enough to hold Darwin server software. If the server software will not fit into `/tmp`, use some other directory.
6. Insert the Darwin Server CD in a drive attached to the Solaris system where the servers will reside; copy the files from the CD to `/tmp` and extract them from the tarfile:

   ```
   cd /tmp
   tar xvf /cdrom/cdrom0/Darwin_36_FCS_Server.tar
   ```

   where *`cdrom`* is the name of the CD drive.
7. To install Darwin, use the command `pkgadd`:

   ```
   pkgadd -d /tmp TMCdarwin
   ```

Answer any questions that the command asks. This command creates the directory `/opt/TMCdarwin` (if it doesn't already exist).

Note: If `/opt` is not a distinct partition and is not large enough to contain Darwin, the Darwin software will be dumped into the root partition and may fill the root partition. A simple solution is, before running `pkgadd`, to create a symbolic link to `/opt/TMCdarwin` from a directory in a partition that has enough space for the Darwin software.

---

8. Configure at least one Darwin server, as described in Chapter 4. Start any servers that you configure. Do not restart any existing server configurations until you have corrected the architecture.

4

Configuring Darwin Servers

After you have installed the Darwin server software, you must configure and start one or more Darwin servers, as described in this chapter. If you plan to connect to a database, see Chapter 5.

> **Note:** Darwin server configurations created using versions of Darwin earlier than Darwin 3.6 (including Darwin 3.6 Beta) will not work with Darwin 3.6. See Section 4.3.2 for more information.

Basically, configuring a Darwin server means using the `darwinconfig` command and its subcommands to add, remove, describe, start, stop, etc., Darwin servers. In addition, the `darwinconfig` command creates and maintains several configuration files for both client and server.

Section 4.1 provides a brief overview of Darwin's client-server design, and shows how the various Darwin components interact with each other.

This rest of this chapter explains

- Darwin configuration files (Section 4.2)
- what you need to do before you configure a server (Section 4.3)
- how to add, remove, and configure a server, using `darwinconfig` and its subcommands (Section 4.4)

For examples of configuration files and for an explanation of the entries in a server configuration file, see Appendix B.

4.1 Darwin Servers

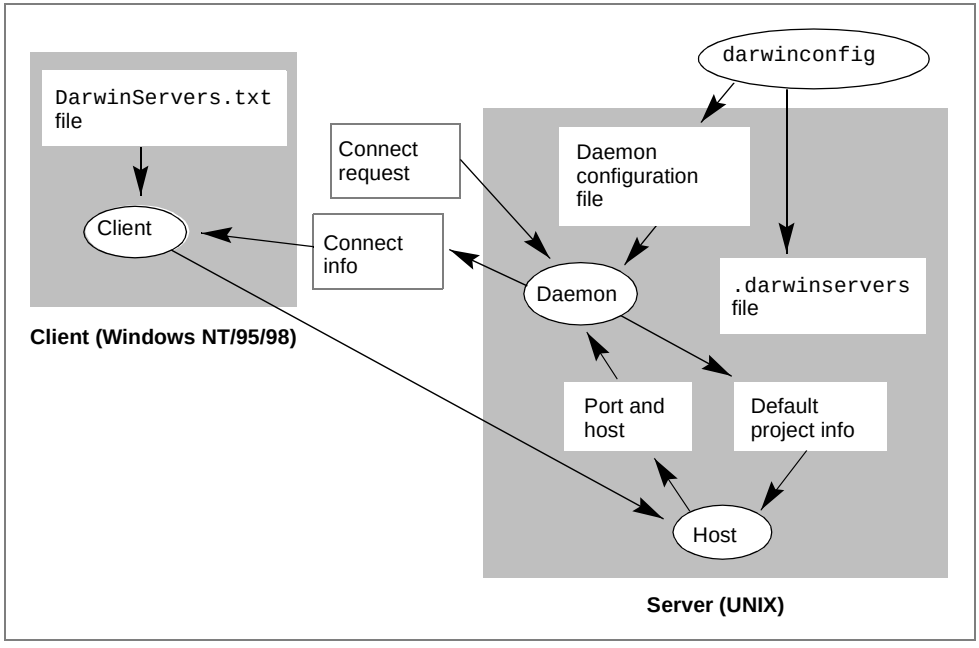
Darwin uses a client-server design, a design that separates client and server functions. Typically, a client workstation provides the user interface, does some of the processing, and communicates with a server. The server does most of the processing and handles requests from the client.

Darwin's client-server design distributes the computing as follows: the Darwin client presents and updates the user interface and the graphics, and also handles some local computation. The Darwin server runs the Darwin data mining algorithms.

The Darwin server is composed of the necessary Darwin executable files, a daemon, and a configuration file created using the `darwinconfig` command. Each daemon runs a particular Darwin executable in a particular way.

Figure 4–1 shows how Darwin components interact.

Figure 4–1 Interaction of Darwin components.



4.2 Darwin Configuration Files

The server's configuration file contains all the information necessary to run the server and set up projects. Appendix B contains sample configuration files for the Solaris platforms that Darwin supports at Release 3.6.

Note: All file locations assume a standard installation in which the software is installed in `/opt/TMCdarwin`, the default location. If the software is installed in some other directory, correct the pathnames accordingly.

The command `darwinconfig`, described in Section 4.4, creates and maintains the following files:

- a configuration file named `.darwinservers`, which covers all configured Darwin servers at your site. The `.darwinservers` file is created in `/opt/TMCdarwin/etc`. Section 4.2.1 describes the `.darwinservers` file.
- a configuration file for each server on the system. Each server's configuration file has the same name as its server. These configuration files are saved on the UNIX server in `/opt/TMCdarwin/etc`. Appendix B shows sample configuration files for single and multinode servers with and without database connectivity.

4.2.1 darwinservers File

The `.darwinservers` file lists all configured Darwin servers. In a standard configuration created by the `darwinconfig` command, the `.darwinservers` file is automatically created and updated in `/opt/TMCdarwin/etc` on UNIX.

Each line in the `.darwinservers` file describes a single server. Each line has five parts:

- the keyword `server`
- the Darwin server's name (the name specified in the `name` line of the server's configuration file)
- the hostname of the system on which the server runs (the name specified in the `server` line of the server's configuration file)
- the port to be used by the server (matching the `port` entry in the configuration file for the server)

- a description of the Darwin server, within quotation marks

Notice that the server's name, the name of the system on which the server runs, and the port number must match the values in the server's configuration file (see examples in Appendix B). Here is a sample `.darwinservers` file for the sample server configuration files for servers `eval-1` and `Ring`:

```
server eval-1 test.mysite.com 606 "Test System 1"
server Ring mysmpserver 1111 "SMP test system"
```

4.2.2 DarwinServers.txt File

A related file, `DarwinServers.txt`, which you create (see Section 6.1.1), resides on each user's C drive on Windows NT or Windows 95/98:

```
C:\Program Files\Tmc\Darwin Windows Client\DarwinServers.txt
```

This file contains a list of servers to which the client can connect. You may wish to copy `.darwinservers` to this file. Darwin does not update `DarwinServers.txt`; you must update it by hand.

When you install a Darwin client, you specify the file `DarwinServers.txt` as the source of the list of servers to which users can connect.

When a user opens a Darwin session, the Darwin client displays, on the login screen, a list of server names and descriptions, taken from the `DarwinServers.txt` file. (Using the sample file shown above, the login screen would offer the user the choice of servers eval-1 and Ring.) The user selects a server from this list; Darwin then attempts to connect the user to that server.

4.3 Before Using darwinconfig

Before you can use the `darwinconfig` command, you must

- be logged in as root to the UNIX system on which the server will run
- confirm that `DARWINHOME` has the appropriate value (see Section 4.3.1, below)

Note: If you will be connecting to a database, see Chapter 5 before starting to configure a Darwin server.

4.3.1 DARWINHOME Environment Variable

If you have a nonstandard installation, you must set the `DARWINHOME` environment variable to the location of the installation in order to run `darwinconfig`. The `DARWINHOME` environment variable does not need to be set if the software is installed in `/opt/TMCdarwin`.

Note: There are also several Darwin environment variables that may need to be set, depending on circumstances. Note that Darwin environment variables must be added to the configuration file using the `darwinconfig add` command. See Section 4.4.12 for details.

4.3.2 Upgrading Existing Configuration Files

Darwin 3.6 supports architectures different from those supported by earlier versions of Darwin, including Darwin 3.6 Beta. Two of the supported architectures, **`sparc`** and **`hpux`**, do not have the same meaning as they did in earleir versions of Darwin. (See the description of **`architecture`** in Section B.2 for details.) Server configurations created using earlier versions of Darwin will not work properly with Darwin 3.6.

You must create a new configuration for each Darwin server.

Note that you *cannot* modify existing configurations files; instead you must create new configurations. See Section 4.4.11 for details.

4.4 Configuring and Managing Darwin Servers

This section describes the `darwinconfig` subcommands and related topics. The subcommands are described in Sections 4.4.1 through 4.4.10. Additional topics not covered by a specific subcommand are

- modifying configuration files (Section 4.4.11)
- Darwin environment variables (Section 4.4.12)

The command `darwinconfig` runs on UNIX (Solaris or HP-UX); you must be logged in as root to perform tasks such as creating configuration files or starting and stopping servers.

Note: If you are not logged in as root when you create configuration files and start or stop servers, the operations will not be performed correctly.

To use the `darwinconfig` command, cd (on UNIX) to the appropriate directory and issue the `darwinconfig` command:

```
# cd /opt/TMCdarwin/etc
# ./darwinconfig
Darwin SCU >
```

The `Darwin SCU >` prompt indicates that you are running `darwinconfig`. The command is interactive; you are prompted for information required for the particular subcommand you are using.

The `darwinconfig` command provides these subcommands:

- **`list`** lists the available Darwin servers (Section 4.4.1).
- **`add`** adds a new Darwin server (Section 4.4.2).
- **`remove`** removes an existing Darwin server (Section 4.4.3).
- **`describe`** displays a description of a Darwin server (Section 4.4.4).
- **`verify`** verifies the configuration of a Darwin server (Section 4.4.5).
- **`start`** starts an existing Darwin server (Section 4.4.6).
- **`stop`** stops a running Darwin server (Section 4.4.7).
- **`save`** saves a new configuration file (Section 4.4.8).
- **`quit`** quits `darwinconfig` without saving changes (Section 4.4.9).
- **`exit`** exits `darwinconfig` after saving changes (Section 4.4.9).
- **`help`** or **`?`** prints a list of the subcommands and their definitions (Section 4.4.10).

4.4.1 Listing Available Servers

Use `list` to display a list of all of the available servers (i.e., all servers in the `.darwinservers` file described in Section 4.2.1). For each server, `list` displays the name, the architecture, the hostname, whether auto restart is specified, and whether the server is running.

An example of output to `list`:

```
Darwin SCU > list
Available servers:
Name         Arch    Hostname              Auto Restart Running
DarthSerial  sparc   darth-smp.think.com   Yes          Yes
DarthSmp-1   sparc   darth-smp.think.com   No           Yes
DarthSmp-2   sparc   darth-smp.think.com   Yes          Yes
DarthSmp-4   sparc   darth-smp.think.com   Yes          Yes
DarthDB      sparcdb darth-smp.think.com   Yes          No
```

4.4.2 Adding and Configuring a Server

Use `add` to create and configure a new Darwin server. The subcommand `add` prompts you for information about server characteristics and uses your answers to create a configuration file for the server. If `add` finds problems with your answers, it generates messages about the problems. Remember that no two servers can have the same name.

> **Note:** The server's configuration file is not automatically saved; to save it, use `save` (Section 4.4.8) or `exit` (Section 4.4.9). You must save a server's configuration file before you can start that server.

When you create and configure your first Darwin server, a `.darwinservers` file is created. Thereafter, the `.darwinservers` file is updated whenever a new server is created or a server is removed. See Section 4.2.1 for more information about the `.darwinservers` file.

On Windows NT or Windows 95/98, the file `DarwinServers.txt` in `C:\Program Files\Tmc\Darwin Windows Client` contains a list of servers to which the client can connect. When you are finished configuring servers, you can use the information in `.darwinservers` to create this file. Since Darwin does not update `DarwinServers.txt`, the user or system administrator must update it.

For examples of complete configuration files, see Appendix B; Appendix B also contains detailed descriptions of the meaning of the entries in the configuration file. You will find it useful to have Appendix B available as you answer the questions generated by `add`.

The `add` subcommand is also used to add any Darwin environment variables to a server's configuration file. See Section 4.4.12.

4.4.3 Removing a Server

To remove a Darwin server, first stop the server using the `stop` subcommand, described in Section 4.4.7. Use `remove` to remove an existing configuration file from `/opt/TMCdarwin/etc`. The server is then no longer available to Darwin clients, and the `.darwinservers` file is automatically updated.

The `DarwinServers.txt` file is not automatically updated; you must update that file yourself from the Windows NT or Windows 95/98 file:

```
C:\Program Files\Tmc\Darwin Windows Client\DarwinServers.txt
```

4.4.4 Describing a Server

Use `describe` to display the contents of the server's configuration file. The meaning of each entry in the configuration file is described in Appendix B.

An example of output to `describe`:

```
Darwin SCU > describe DarthSmp-1
Architecture            : sparc
Server name             : DarthSmp-1
Server description      : 1 CPU of a Sun ES4000 Server (8 CPUs)
Server host name        : darth-smp.think.com
Listen port             : 901
Executable              : /opt/TMCdarwin/bin/darwinhost-sparcsmp
Auto restart on reboot  : No
Number of nodes         : 1
Serial root             : $USERHOME/darwin
Distributed root        : Default
Shared project file     : No shared project file specified
Shared serial root      : No shared serial root specified
Shared distributed root : No shared distributed root specified
Environment entry       : LD_LIBRARY_PATH=/opt/TMCdarwin//lib/darwin/
```

4.4.5 Verifying the Validity of a Configuration File

Use `verify` to confirm that the configuration file for the server is valid. The `verify` subcommand ensures that you can actually connect to specified ports and that any files mentioned in the configuration file actually exist. It also verifies that the configuration is appropriate for the specified architecture. You can use `verify` to ensure the validity of a configuration file that you have created with the subcommand `add`.

4.4.6 Starting a Server

Use `start` *`server-name`* to start the Darwin server with name *`server-name`*. You can only start configured servers; a server is not configured until you save the configuration file (using the `save` subcommand). `darwinconfig` verifies the server's configuration file and tries to start the server. If `darwinconfig` finds problems with the server's configuration file, it indicates the problem(s) and asks whether you want to start the server anyway. If `darwinconfig` cannot start the server, it tells you why.

> **Note:** Be sure you are logged in as root when starting and stopping Darwin servers. If you are not root, you will be able to start and stop the server, but when another user tries to enter Darwin, the servers are automatically stopped.

The `darwinconfig` command is interactive; however, you can use the following one-line version of `darwinconfig` to start a configured Darwin server with name *`server-name`*:

```
# darwinconfig start server-name
```

If you use this one-line version of the command, `darwinconfig` always tries to start the server. You must be root when you execute this command.

4.4.7 Stopping a Server

Use `stop` *`server-name`* to stop the Darwin server with name *`server-name`*. You can only stop servers that are running.

> **Note:** Be sure you are logged in as root when starting and stopping Darwin servers. If you are not root, you will be able to start and stop the server, but when another user tries to enter Darwin, the servers are automatically stopped.

The `darwinconfig` command is interactive; however, you can use the following one-line version of darwinconfig to stop a configured Darwin server with name *`server-name`*:

```
# darwinconfig stop server-name
```

4.4.8 Saving a Configuration File

Use the `save` subcommand to write the information you have entered to Darwin's internal configuration files.

The subcommand add does not automatically save the configuration; you must use `save` to save the configuration file before you can start the server. You can also end the session using `exit`, which saves all changes before ending the session.

4.4.9 Ending a darwinconfig Session

There are two ways to end a `darwinconfig` session:

- `exit` saves changes, then ends the session.
- `quit` ends the session without saving changes.

4.4.10 Getting Help

Enter `help` or `?` at the `Darwin SCU >` prompt to get a list of the subcommands and a brief description of each.

4.4.11 Modifying Configuration Files

You cannot modify configurations. What you must do instead is remove the server and configure it again, as described in Section 4.4.2. If the removed server has ever been used, you must give the server a new name when you add it. The reason it must have a new name is that names of servers that have been used are embedded in the `.darwinprojects` and `.darwinhostrc` files in users' home directories.

4.4.12 Darwin Environment Variables

There are several Darwin environment variables that may need to be added, depending on circumstances. Any Darwin environment variables that are needed must be added and defined using `darwinconfig add` (see Section 4.4.2). An example of adding an environment variable required for a particular database package is shown below.

License File Environment Variable

Darwin no longer requires a license file. The `TMC_LICENSE_FILE` environment variable is ignored.

Environment Variables Required for Database Connectivity

Oracle databases require that you set certain Darwin environment variables. See the documentation for your database software for details, and see Chapter 5 of this installation guide. You must define such variables using `darwinconfig add` for Darwin to be aware of them.

Below is an example showing `darwinconfig add`'s prompts and user responses (shown in bold) for setting a Darwin environment variable required by a database package. This example also shows the system's response when you press the RETURN key for help as prompted.

```
Would you like to add any environment variables to the server environment?
(y/n) : y

Environment variable("exit" to exit)(return for help) : <return>

The name of a environment variable to add to the Darwin server process
environment. This is necessary for most database connectivity setups. See your
database documentation for more information.

Environment variable ("exit" to exit)(return for help): ORACLE_HOME
Value of "ORACLE_HOME" (return for help) : <return>

The value of a environment variable to add to the Darwin server
process environment. This is necessary for most database connectivity setups.

Value of "ORACLE_HOME" (return for help) :
/product/oracleSW/product/8.0.5

Environment variable ("exit" to exit) (return for help) : exit

Done.
```

4.4.13 Troubleshooting Configurations

Here are solutions to common problems associated with server configurations:

1. Use `darwinconfig` to create configuration files; if you make changes using a text editor the changes may not work. In particular, configuration files are not parsed by any standard UNIX shell; among other things, dependent variables will not work.
2. If you change a configuration, make sure that you stop the server and restart it. Changes will not take effect until you restart the server.

3. If you change configuration parameters, you may have to reboot the server.
4. The Darwin executable must be running on the server.

5

Database Connectivity

Note: If you are planning to connect to Oracle databases from Darwin, you must install Oracle database software and Data Direct Connect before you install Darwin server software.

You can skip this chapter if you will not be connecting to a database. This chapter describes setting up the environment for database connectivity. Follow these steps:

- Obtain, install, and connect to the database software (Section 5.1).
- Obtain and install the database driver (Section 5.2).
- Confirm / set permissions and file / directory accessibility (Section 5.3).
- When you configure servers, specify the correct architecture for database connectivity (Section 5.4).
- Set Darwin environment variables for database connectivity (Section 5.5).
- Create required files (Section 5.6).
- Check that the installation is correct and that the environment variables have been set correctly (Section 5.7).
- After you have installed the Darwin client software (Chapter 6), confirm that users can connect to the database software (Section 5.8).

5.1 Supported Database Software

Darwin 3.6 supports connectivity to Oracle databases only.

Darwin 3.6 connects to:

- Oracle Version 7.3 and SQL*NET Version 2.3
- Oracle Version 8i Enterprise Edition

This rest of this chapter describes connecting to an Oracle 8*i* database. For information about connecting to Oracle 7.3, consult your local Oracle Support Services. There is a sample server configuration for Oracle 7.3 connectivity in Section B.1.1.

If you do not already have Oracle on your system, you must contact Oracle, as described in Section 2.7.1, to learn how to obtain and install Oracle software.

Oracle and DataDirect Connect ODBC must be installed and configured before you configure Darwin database connectivity.

5.2 Obtain and Install Database Driver

Darwin does not support the ODBC drivers supplied by Oracle.

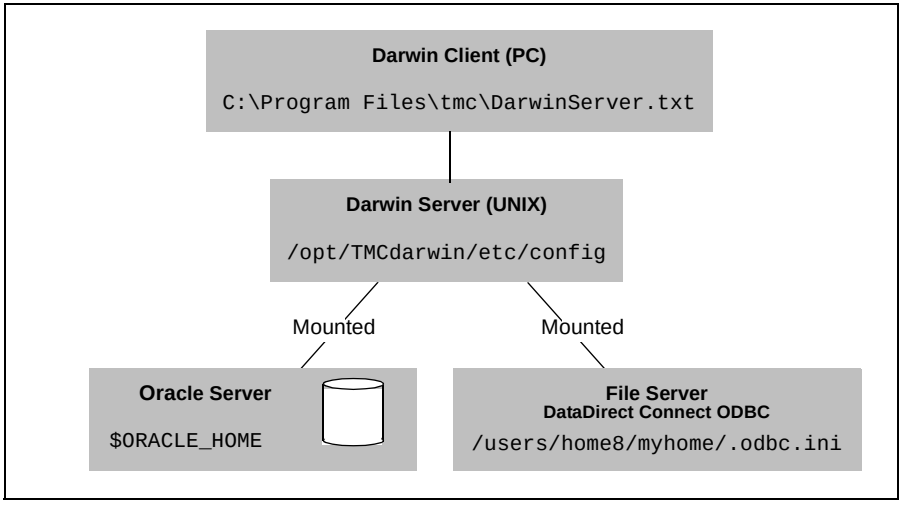
Darwin uses MERANT (Micro Focus) DataDirect Connect ODBC 3.5 drivers to connect to a databases on a UNIX server.

You can connect to and disconnect from databases and create Darwin datasets from the result of an SQL script. Alternatively, you can use the **Database Import** Wizard to create a Darwin dataset from a database table and the **Database Export** Wizard to store datasets and results tables in a database. You cannot perform any other ODBC operations, such as storing models in a database.

You install the DataDirect drivers and Oracle on the UNIX network where the Darwin servers reside. DataDirect drivers and Oracle must be accessible from the Darwin server; they can be NFS mounted. Darwin supports connectivity to UNIX databases only. You cannot connect to databases on Windows 95 or Windows NT.

Figure 5–1 illustrates database connectivity between Darwin and an Oracle database.

Figure 5–1 Connectivity between Darwin and Oracle.



5.2.1 Connecting to Oracle Using DataDirect Connect ODBC

If you plan to use DataDirect Connect ODBC, you must

- Purchase the DataDirect Connect ODBC software from MERANT, as described in Section 2.7.2.
- Install and configure the DataDirect Connect ODBC driver, following instructions provided with the driver.
- Verify that you both DataDirect Connect ODBC drivers and the Oracle server are accessible from the machine on which the Darwin server will be installed.
- Ensure that you have installed Darwin server software that supports database connectivity, as described in Chapter 3.
- Set up the environment, which includes specifying the appropriate environment variables for all the software involved (Darwin, DataDirect, and the database). You must also set certain Darwin environment variables (Section 5.5) when you configure the Darwin server.

5.3 Permissions and Accessibilty for Oracle 8*i*

The following permissions are required to connect to Oracle8*i* from Darwin:

- Users must be recognized by the database server and must have database connect permission.
- Users must have read permission on the database tables.
- Oracle directories must be locally accessible; they can be NFS mounts.
- DataDirect 3.5 directories must be locally accessible; they can be NFS mounts.
- Darwin requires the Oracle network libraries; these libraries must be locally accessible. Darwin also requires the Oracle configuration file `tsnames.ora` and `listener.ora` to be locally accessible if the Oracle server is not on the same node as Darwin.

5.4 Darwin Server Architecture for Database Connectvity

When you configure a Darwin server, you must specify the architecture that supports database connectivity: **`sparcdb`** (for Solaris Systems) or **`hpuxdb`** (for HP-UX systems), as described in Chapter 4 and Appendix B. If you do not specify the architecture correctly, you will not be able to connect to a database.

When you configure the server, you must also set environment variables, as described below.

5.5 Darwin Environment Variables for Oracle 8*i*

This section describes the Darwin environment variables that must be set for database connectivity. The variables described here are specific to Oracle.

For an example of how to set environment variables in Darwin, see Section 4.4.12.

For an example of a configuration files for a Darwin servers that support Oracle connectivity, see Appendix B.

The following environment variables are required:

- `ORACLE_HOME`
- `LD_LIBRARY_PATH`
- `ORACLE_SID`

5.5.1 ORACLE_HOME Environment Variable

The environment variable ORACLE_HOME must be set appropriately in the Darwin server's configuration. ORACLE_HOME is the directory where Oracle is locally accessible. If Oracle is installed in the directory /product/oracleSW/product/8.0.5, then ORACLE_HOME should be defined as follows in the server's configuration file:

```
ORACLE_HOME=/product/oracleSW/product/8.0.5/
```

You must also add the directory where Oracle libraries are install to the LD_LIBRARY_PATH environment variable.

5.5.2 LD_LIBRARY_PATH Environment Variable

The environment variable LD_LIBRARY_PATH must be set appropriately in the Darwin server's configuration. LD_LIBRARY_PATH must include the Oracle libraries, the DataDirect libraries, and the Darwin libraries. For example, if Oracle is installed at /product/oracleSW/product/8.0.5/, and if DataDirect is installed at /opt/odbc35/lib, and if Darwin is installed in the default location, then LD_LIBRARY_PATH should be defined as follows in the server's configuration file:

```
LD_LIBRARY_PATH=/opt/TMCdarwin/lib/darwin:/opt/odbc35/lib:
              /product/oracleSW/product/8.0.5/lib/
```

5.5.3 ORACLE_SID Environment Variable

The environment variable ORACLE_SID must be defined in the Darwin server's configuration; for example,

```
ORACLE_SID=ORCL
```

5.6 Required Files

The following files are required:

- /etc/services (or the NIS or NIS+ map)
- .odbc.ini

If the Oracle database and the Darwin server are on different UNIX systems, the following file is also required:

- tsnames.ora

5.6.1 /etc/services File

The `/etc/services` file (or the NIS or NIS+ map) must have a TCP port matching the name and number of the TCP port for the Oracle listener on the database server. This should already be set by your database administrator. For example, to specify port number 1521, include the following:

```
listento 1521/tcp
```

5.6.2 .odbc.ini File

Each user's home directory on UNIX must contain an appropriate `.odbc.ini` file.

Note: Darwin does not support the DataDirect Connect ODBC environment variable `ODBCINI`.

Here is a sample `.odbc.ini` file for Oracle connectivity on Solaris (with Oracle installed at `/db0/oracle`). The DataDirect Connect Oracle driver library is installed at `/opt/odbc/lib/ivor814.so`. `ServerName` is your Oracle `tnsnames.ora` name.

```
[ODBC Data Sources]
Oracle8=Oracle8 ODBC Driver on Solaris
[Oracle8]
Driver=/opt/odbc/lib/ivor814.so
Description=MERANT Oracle8 ODBC Driver
ServerName=ora_db0_net
[ODBC]
Trace=1
TraceFile=odbctrace.out
TRACEDLL=/opt/odbc/lib/odbctrace.so
InstallDir=/opt/odbc
```

5.6.3 tsnames.ora File

If the Oracle server is not on the same node as Darwin, the file `network/admin/tnsnames.ora` in the directory where Oracle software is installed must point to the listener (i.e., the TCP/IP port) on the database server. (`/db0/oracle` is the directory where Oracle is installed.) For example,

```
ora_db0_net=
  (DESCRIPTION=
```

```
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=test-console.think.com)
      (PORT=1521)
    )
    (CONNECT_DATA=
       (SID=ORCL)
    )
)
```

Note: Do not use tabs in the file `tnsnames.ora`.

5.7 Installation Check

Before you connect to an Oracle database using the Darwin client, check that your installation of Oracle and the DataDirect software is correct. To do so, log in to a machine where Darwin is installed and follow these steps:

1. Connect to the Oracle database using the Oracle utility `sqlplus`.
2. Connect to Oracle using the DataDirect demo utilities; for example, try the following sequence of commands:

   ```
   cd $ODBC/bin
   ivtestlib ../lib/ivor814.sl
   cd $ODBC/demo
   demoodbc -UID userid -PWD password DSN
   ```

   where `$ODBC` is the location where DataDirect is installed and `DSN` is the ODBC Data Source Name.

If you can complete these steps successfully, you will be able to connect to Oracle databases from the Darwin client and create Darwin datasets from the databases.

5.8 Using Database Connectivity

After you have installed the Darwin client software (Chapter 6), you will be able to use the Darwin client to connect to a database. This section describes the different ways a user can connect to a database from the Darwin client.

5.8.1 Connecting to a Database Server

The following commands on Darwin's **Project** menu permit users to connect to a database server and to terminate an existing connection to a database server:

- **Database Connect**
- **Database Disconnect**

The **Database** tab of the **Dataset** menu's **Create** command lets users create a Darwin dataset from a connected database using an SQL script or query. For information about these operations, see *Using Darwin* and *Darwin Reference* or the Darwin online help. See Section 5.8.3 for another way to import data from a database.

SQL Scripts

Darwin allows you to run an SQL script after you have connected to a database. For more information and examples, see *Using Darwin* and the *Darwin Reference.*

SQL Limitations

When you use a script to create a new Darwin dataset, do not use the SQL UNION, INTERSECT, EXCEPT, and JOIN operators; using these operators in a script results in an error.

5.8.2 Disconnecting from a Database

You must close all datasets created using a database product before you try to disconnect from the database. If you forget to do this, you get the error message `Database in Use`. The error message reflects the way a database supporting ODBC creates datasets.

When the database creates a new dataset, it does so by creating a pointer; it does not retrieve data at the same time. If you try to close a connection to the database when there are any active pointers, you get the `Database in Use` message.

Therefore, before disconnecting, you must save any datasets created from a database and then close the original dataset that was created directly from the database.

5.8.3 Database Import Wizard

The **Database Import** Wizard simplifies the process of converting a database table to a Darwin dataset. The Wizard performs the following functions:

- connects you to the database
- permits you to log in to the database
- displays available database tables

- converts the selected table to a Darwin dataset
- closes the connection to the database

To use the Wizard to import data, there must be a table that defines the data that you wish to import, and you must provide the name of that table.

The Wizard manages the connection to and the disconnection from the database automatically.

This Wizard does not work, of course, unless the Darwin server is configured for database connectivity.

For more information about this Wizard, see *Darwin 3.6 New Features.*

5.8.4 Database Export Wizard

The **Database Export** Wizard permits you to save a Darwin dataset or results table as a database table. You can write the dataset or result table to a new table in the database, or you can update or overwrite an existing database table. The Wizard

- connects you to the database
- lets you select a Darwin dataset
- exports the dataset to a database table
- closes the connection to the database

For more information about this Wizard, see *Darwin 3.6 New Features.*

6

Installing the Darwin Client

After you've installed the Darwin server on Solaris and configured and started one or more Darwin servers, you are ready to install Darwin client software on PCs running Windows NT, Windows 95, or Windows 98.

Follow these steps to install Darwin client software:

1. Verify that Microsoft Internet Explorer 4.x is installed on every PC on which you are planning to install Darwin clients. See Section 2.4 for information about obtaining Internet Explorer 4.x. If Internet Explorer 4.x is not installed on a PC, Darwin online help will not be available on that PC.
2. Gather information about the Darwin servers that you will connect to (Section 6.1).
3. Create a Darwin Windows Client customization disk or file (optional) (Section 6.1.1).
4. Install the client software (Section 6.2).

6.1 Collect Server Information

Before you install Darwin client software, you must have the following information for each server that the client will connect to:

- the name of the server
- the name of the host machine on which the server resides
- the port number for the server
- a description of the server (optional)

During the installation process, you will be prompted for this information in a screen titled "Create the Configuration File".

If you are connecting to several servers or are installing several clients, you may want to create a Darwin Windows Client customization disk or file (Section 6.1.1).

6.1.1 Create a Customization Disk or File (Optional)

Follow these steps to create a Darwin Windows Client customization disk or file:

1. Using **Notepad** or a similar editor, create a file named `DarwinServers.txt`, containing one-line descriptions of the Darwin servers that you plan to connect to. For each server, create a line in the following format:

   server *name host port description*

   where *name* is the name of the server, *host* is the system on which the server resides, *port* is the port number for the server, and *description* is the optional description of the server, enclosed in quotation marks.

   Separate each item from the next with a space (not a TAB).

   For example, if you plan to connect to the server `eval-1` on the machine `test.mysite.com` at `port 606` with description `"Test System 1"`, include the following line in `DarwinServers.txt`:

   ```
   server eval-1 test.mysite.com 606 "Test System 1"
   ```

   This line is the entry for the server from the `.darwinservers` file; you can create the customization file or disk by copying the server entries in the `.darwinservers` file.

2. To create a Darwin Windows Client customization disk, copy `DarwinServers.txt` to a diskette. To create a customization file, copy `DarwinServers.txt` to a folder that can be seen by all the personal computers that will run Darwin client software. The `DarwinServers.txt` file also belongs in each user's `C:\Program Files\Tmc\Darwin Windows Client` folder.

If you are installing clients on Windows NT, Windows 95 and Windows 98, you can use the same Darwin Windows Client customization disk or file for all PC platforms.

6.2 Installing Client Software

The client software is installed using an InstallShield wizard. Follow these steps to install the client software:

1. Exit all programs that are running on the PC where you are installing a Darwin client.
2. Insert the Darwin Client CD in the CD drive of the PC where you are installing the Darwin client.
3. If you are using a Darwin Windows Client customization diskette, insert it into the diskette drive of the same PC.
4. The **Darwin Windows Client Setup Program** will start up automatically. The installation wizard will guide you through the installation process.
5. If the setup program does not start automatically, navigate to the CD ROM drive and double-click `Setup.exe`.
6. To complete the process, the PC must be restarted. You can either let the wizard restart the PC or you can restart the PC at your convenience. If you are using a customization disk, remove it before you restart the PC.

6.2.1 DARWINHOME Windows Environment Variable

During installation of the Darwin client, Darwin adds a Windows environment variable named `DARWINHOME`, and sets it to the directory in which you installed the Darwin client, which is by default `C:\Program Files\Tmc\Darwin Windows Client`.

This variable appears on the **Control Panel > System > Environment** tab under **User Variables**. It is possible to modify its setting there, but *do not*. If the setting is modified, Darwin is not likely to work at all.

6.3 First-Time Installation

Before you can use Darwin for the first time, you must create a least one dataset in one of your projects. Until you create a project and copy a text file or dataset into it, everything in the Darwin graphical user interface is greyed out and no commands are available. There are two ways to do this:

- Copy a dataset or text file from the datasets for practice as follows:
  1. Set up the datasets for practice, as described in *Darwin 3.6 New Features*.

2. Create a project as described in Section 8.2.
3. Drag a dataset from the datasets for practice into your project.
4. Commands will now be available.

- Create a project and copy a text file into itas follows:
  1. After both Darwin client and server software are installed, start up the Darwin client.
  2. Create a project (described in Section 8.2).
  3. Exit Darwin.
  4. On UNIX, go to the `darwin` directory, and, in the project directory for the project you just created, create a `.txt` file (or copy a `.txt` file into the directory), and, optionally, a `.des` file.
  5. When you attach to the Darwin client, commands will be available.

7

Starting and Stopping Darwin

After the Darwin server and client software is installed, and the server(s) are configured and running, you and other users can start using Darwin. You, as system administrator, will want to start up Darwin to confirm that Darwin is installed correctly and is ready for users.

7.1 Starting Darwin

Starting Darwin requires the following steps, which must be performed in the order indicated:

- First, the system administrator starts all necessary Darwin servers on UNIX (Section 7.1.1).
- Next, ordinary users start the client software on Windows NT or Windows 95/98 (Section 7.1.2).

7.1.1 Starting Darwin Servers

A system administrator uses the `darwinconfig start` *`name`* command described in Section 4.4.6 to start each server that a user will connect to.

Note: Be sure you are logged in as root when starting and stopping Darwin servers. If you are not root, you will be able to start and stop the server, but when another user tries to enter Darwin, the servers are automatically stopped.

7.1.2 Starting a Darwin Client

To start Darwin,

- Log in as an ordinary user on Windows NT, Windows 95, or Windows 98.

- Click the **Start** button; on the **Start** menu, click **Programs**; on the **Programs** menu, click **Darwin Windows Client** (identified by a protohominid icon); if there is a Darwin icon on your desktop, you can double-click it to start the client.

The Darwin splash screen appears. After the splash screen disappears, the Darwin login window appears (titled **Darwin Login**).

Now log in to a Darwin server. At the top of the login screen are two boxes for you to type in:

- **User Name:** Enter the user name for your account on the server.
- **Password:** Enter the password for your account on the server. This password may or may not be the same password that you use to log on to your PC.
- **Darwin Server:** This text box displays a list of servers for you to choose from. The list comes from the `DarwinServers.txt` file that was created when the client software was installed.) Click the name of the Darwin server you want to connect to. To the right of the server name is a brief description of it.
- Click **OK** or press ENTER.

Darwin then attempts to connect you to the server you selected, and the main Darwin window is displayed.

If the attempt to connect fails, try connecting to different server or contact your system administrator. The administrator can check to see if servers are running and start them if necessary.

7.2 Stopping Darwin

Stop Darwin as follows:

- First, ordinary users exit (stop) all clients on Windows NT, Windows 95, or Windows 98 (Section 7.2.1).
- Then, the administrator stops all servers on UNIX (Section 7.2.2).

7.2.1 Stopping Darwin Clients

The best way to exit Darwin is to click **Project > Exit** (on the **Project** menu, click **Exit**). When you exit by this route, Darwin gives you a last opportunity to save any files or objects that you created in this session but did not save.

7.2.2 Stopping Darwin Servers

The system administrator stops each server using the `darwinconfig stop` *name* command described in Section 4.4.7.

Note: Be sure you are logged in as root when starting and stopping Darwin servers. If you are not root, you will be able to start and stop the server, but when another user tries to enter Darwin, the servers are automatically stopped.

8

Administering Darwin Projects

This chapter provides general information about Darwin projects; these are basic things system administrators should know to support Darwin users.

Note: If you install Darwin on a system for the first time, you must take certain steps to make the GUI usable. See Section 6.3 for details.

8.1 Darwin Projects

All work in Darwin is done within the context of a project, which is a UNIX directory that resides on the server. Ordinary users create and delete projects. At the beginning of your first Darwin session, the only commands available are those that create new projects (**New Project**), select existing projects (**Open Project)**, or end the session (**Exit**).

The first time that you run Darwin, you must create a project. After you have created at least one project, Darwin automatically selects the project you last used whenever you start a new Darwin session. You can work in that project, select another existing and available project, or create a new project.

For more information about projects, see Darwin's online help, *Using Darwin*, and *Darwin Reference*.

8.2 Creating Projects

Ordinary users create projects from the Darwin client, using the **New Project** command of the **Project** menu.

Follow these steps to create a project:

- On the **Project** menu, click **New Project**.

A dialog box appears, and prompts you for the following information:

- In the **Project** box:
  - **Name:** Specify a name for the project.
  - **Description:** (Optional.) Record any information you may want about the project.
  - **Shared:** If you want this project to be shared with other users, click the **Shared** box.

    Note: This option is available only if the Darwin server is configured to permit shared projects.

  - **Note:** In the **In Server** box are two read-only boxes:
  - **Name:** Name of the current server.
  - **Description:** Description of the current server.
  - **Distributed:** If you want the project to be distributed, click the **Distributed** box. **Note:** This option is available only if the Darwin server is a distributed server.
- If you wish, click **Advanced**, which takes you to **Advanced Options**, **Project** tab, where you can record information about the project (see Section 8.2.1, below).
- Click **OK** or press ENTER.

Darwin then creates the project, using the pathname or pathnames specified in the server configuration file.

8.2.1 Recording Project Information

If you click **Advanced Options** on the **New Project** dialog window, you can specify the following optional information about the current project (the project you are creating):

- **Leader:** The name of the project leader.
- **Phone** and **Ext:** The telephone number of the project leader.
- **Business Problem:** The problem that the project addresses.
- **Objectives:** The objectives of the project.

All this information is optional. When you are finished, click **OK** or press ENTER to save the information.

8.3 Deleting Projects

Ordinary users can delete their own projects by right-clicking the project's name in the **Workspace** listing; then, on the context menu that appears, click **Delete**.

You must delete all files associated with a project before you delete the project. If you try to delete a project with files left in it, you get one of these error messages:

```
Serial directory is not empty
```

or

```
Distributed project directory is not empty
```

8.4 Administrator Decisions about Projects

All project directories reside on the UNIX server.

The system administrator defines the location of Darwin project directories within the server configuration files (described in Chapter 4). The system administrator can also set up projects for users. Usually, however, users define projects for themselves.

Darwin supports two types of projects: user projects and shared projects:

- *User projects* are designed for use by a single user. Their UNIX umasks are set to 0022.

  If the administrator uses the USERHOME keyword in the serial path defined in a server's configuration file, then project directories are subdirectories of the home directory of the user who creates them. If the administrator does not use the USERHOME keyword during server configuration, project directories are grouped as subdirectories of a single specified directory.

- *Shared projects* are designed for use by groups of users. Their UNIX umasks are set to 0000, allowing all users to create, open, use, and delete files (if the base directories have write permission set for all users). As part of configuring a server, the system administrator specifies whether users can create shared projects. The USERHOME and USERNAME keywords are not allowed in shared pathnames.

Administrators can choose whether to allow shared projects on any given Darwin server. If they define shared serial root, shared project file, and shared distributed root (for distributed servers only) in a server's configuration file, then shared projects are allowed. If the shared files are not defined, then shared projects cannot be created on that server. See the discussion of `shared_serial_root` in Section B.2 for more information.

8.5 The .darwinprojects File

When Darwin creates a nonshared project for a user, it puts the project name and information in a `.darwinprojects` file, which is stored in the user's home directory on UNIX. The file contains one entry for each project the user creates.

Darwin creates and maintains the `.darwinprojects` file. Under ordinary circumstances, neither the user nor the administrator needs to edit the file.

When the user gives the **Project** menu's **Open Project** command, Darwin displays the project names from the user's `.darwinprojects` file. If the server has a shared projects file, those project names are also displayed.

The user selects a project from the list. Darwin then uses the `.darwinprojects` file to locate the project directories for this project, when running on this server. (If the project has not been run on this particular server before, Darwin goes to the server configuration file for default directory information.)

When you restart Darwin, it automatically selects the project you used last.

8.5.1 A Sample .darwinprojects File

As an example, the project `ad_campaign_3` might have the following entry in a `.darwinprojects` file:

```
project_name ad_campaign_3
directory /users2/csmith/darwin/ad_campaign_3
server eval-1 T /users2/csmith/darwin/ad_campaign_3
server RingD F /export/darwin/csmith/ad_campaign_3
end_project
```

The information for each project is as follows:

- the keyword `project_name`, followed by the project's name
- the keyword `directory`, followed by the pathname of the serial project directory

- the keyword `server`, followed by three items:
  - the name of a server on which the project has run
  - the keyword T or F, indicating that serial and distributed directories are in the same directory (T) or are in different directories (F)
  - the pathname for the distributed directory

  If the project has been run on more than one server, the file contains one line for each server that has been used.
- the phrase `end_project`

8.5.2 Modifying the .darwinprojects File

There are a few circumstances under which the user or administrator might want to alter the `.darwinprojects` file. For example,

- Whenever any directory containing Darwin projects is moved, the user or administrator should check the pathnames in the user's `.darwinprojects` file and correct any pathname affected by the move.
- The system administrator can change the location of the serial or distributed directories for a server by manually editing that server's configuration file. These changes do not propagate into the `.darwinprojects` files. This can be advantageous: For instance, it allows existing projects to remain on one set of disks, while new projects are placed by the server on new disks. On the other hand, if you want a user's projects to migrate to a new disk, then either the user or the administrator must edit the `.darwinprojects` file for that user.
- If two users want to share access to a user project, they can do so by copying the project's entry from the `.darwinprojects` file into the second user's `.darwinprojects` file. Permissions on the project directory may have to be modified; other modifications may be necessary.

A

Installing the SAS Conversion Utilities

Darwin supports importing and exporting datasets in proprietary formats using a combination of a third-party product (DBMS/COPY) and two Perl scripts distributed with Darwin (`sas2darwin` and `darwin2sas`). The Perl scripts automate the use of DBMS/COPY to convert files to and from the format supported by the SAS Institute, Inc.

DBMS/COPY, Perl, and the Perl scripts all run on UNIX; you cannot invoke the SAS conversion utilities from the Darwin client.

This appendix describes how to install the conversion utilities on UNIX. For information about the hardware and software requirements of the SAS conversion utilities, see Section 2.6 of this manual. For information about using the utilities, see *Using Darwin*.

A.1 Installing the Software

If you do not plan to convert to and from SAS files, you should not install the SAS conversion utilities.

Install DBMS/COPY and Perl as directed by the suppliers of the software.

The Perl scripts `sas2darwin` and `darwin2sas` are part of Darwin 3.6 and are installed automatically in the UNIX directory `/opt/TMCdarwin/bin` (`DARWINHOME/bin`) when Darwin is installed.

`sas2darwin` and `darwin2sas` assume that Perl is installed on UNIX in `/usr/local/bin/perl`. If Perl is not installed in this location on your system, you must edit the first line of the script `sas2darwin`, changing the line

```
#!/usr/local/bin/perl
```

to specify the location of Perl on your system. (`darwin2sas` is a link to `sas2darwin`, so you don't have to edit both scripts.)

B

Server Configuration Files

This appendix includes sample Darwin server configuration files and explains what the entries mean.

You create configuration files using the UNIX command `darwinconfig`, as described in Chapter 4.

Note: All file locations assume a standard installation in which the software is installed in `/opt/TMCdarwin`, the default location. If the software is installed somewhere else, replace `/opt/TMCdarwin` with the appropriate pathname.

B.1 Darwin Configuration Files

`darwinconfig` creates and maintains the following UNIX files:

- the file `.darwinservers`, which covers all configured Darwin servers at your site. `.darwinservers` is created in `/opt/TMCdarwin/etc`. Section 4.2.1 describes the `.darwinservers` file.
- a configuration file for each server on the system. Each server's configuration file has the same name as its server. These configuration files are found on UNIX in `/opt/TMCdarwin/etc`. Section B.1.1 shows a sample configuration file for each supported architecture running Solaris 2.7 and a sample configuration file for a server supporting database connectivity.

For more information about configuring and maintaining Darwin servers, see Chapter 4.

B.1.1 Server Configuration Files

The server's configuration file contains all the information necessary to run the server and set up projects. Configuration files are slightly different for each different

platform. The following sections contain sample configuration files for the Solaris platforms that Darwin supports at Release 3.6. Configuration files are created using the UNIX command `darwinconfig`, as described in Chapter 4. In the default installation, configuration files reside in the directory `/opt/TMCdarwin/etc`.

Sample Configuration File for a Single Node Server

Here is the configuration file for a server named `eval-1` that runs on an Ultra-SPARC workstation; this server does not support database connectivity. For a detailed explanation of the various terms, see Section B.2.

```
DARWIN(tm) DAEMON CONFIG
architecture sparc
name eval-1
description Test System 1
server test.mysite.com
port 606
executable /opt/TMCdarwin/bin/darwinhost-sparcsmp
auto_restart true
nnodes 1
serial_root $USERHOME/darwin/
distributed_root Default
shared_project_file /<another>/<disk>/<projects>
shared_serial_root /<another>/<disk>/<darwin>
shared_distributed_root Default
env LD_LIBRARY_PATH=/opt/TMCdarwin/lib/darwin/
env DARWINHOME=/opt/TMCdarwin/
```

The following entries are required for the server to work properly:

```
architecture
name
description
server
port
executable
auto_restart
serial_root
distributed_root
```

The following are optional:

```
shared_project_file
shared_serial_root
shared_distributed_root
```

any environment variables that do not override defaults

Sample Configuration File for an Ultra Enterprise Server

Here is the configuration file for a four-node Darwin server that runs on an Ultra Enterprise (SMP) system; this server does not support database connectivity. For a detailed explanation of the various terms, see Section B.2.

```
DARWIN(tm) DAEMON CONFIG
architecture sparc
name Ring
description SMP test system
server mysmpserver
port 1111
executable /opt/TMCdarwin/bin/darwinhost-sparcsmp
auto_restart false
nnodes 4
serial_root $USERHOME/darwin/
distributed_root Default
shared_project_file /<another>/<disk>/<projects>
shared_serial_root /<another>/<disk>/<darwin/>
shared_distributed_root Default
env LD_LIBRARY_PATH=/opt/TMCdarwin/lib/darwin/
env DARWINHOME=/opt/TMCdarwin/
```

The following entries are required for the server to work properly: architecture

```
name
description
server
port
executable
auto_restart
nnodes
serial_root
distributed_root
```

The following are optional:

```
shared_project_file
shared_serial_root
shared_distributed_root
```

any environment variables that do not override defaults

Sample Configuration File for Oracle 7.3 Connectivity

Here is a configuration file for a single-node SPARC server that supports connection to an Oracle 7.3 database using DataDirect 3.5.

```
DARWIN(tm) DAEMON CONFIG
architecture sparcdb
name Oracle7.3
description sqa-u2-1 Server with Microfocus3.5 to Oracle 7.3
server MyServer
port 873
executable /opt/TMCdarwin/bin/darwinhost-sparcsmpdb
auto_restart true
serial_root $USERHOME/darwin
distributed_root Default
env DARWINHOME=/opt/TMCdarwin
env LD_LIBRARY_PATH=/opt/TMCdarwin/lib/darwin:/db0/oracle/lib:/
proj/project1/odbcbc35/lib
env ORACLE_HOME=/db0/oracle
env ORACLE_SID=sid1
```

All entries, except for the `env DARWINHOME=...` entry, are required for the server to work properly.

Sample Configuration File for Oracle 8i Connectivity

Here is a configuration file for a two-node Darwin server on an Ultra Enterprise (SMP) system that supports connection to an Oracle 8i database. Chapter 5 contains detailed information about the environment variables in this sample.

```
DARWIN(tm) DAEMON CONFIG
architecture sparcdb
name TwoNodes-oracle8i
description MySMP, 2 nodes, Oracle8i, ODBC35
server MySMP
port 890
executable /opt/TMCdarwin//bin/darwinhost-sparcsmpdb
auto_restart true
nnodes 2
serial_root $USERHOME/darwin
distributed_root Default
env DARWINHOME=/opt/TMCdarwin/
env ORACLE_HOME=/opt/oraclient
env ORACLE_SID=tmc0
env LD_LIBRARY_PATH=/opt/TMCdarwin/lib/darwin:/opt/odbc/lib:
                                /opt/oraclient/lib
```

All entries, except for the `env DARWINHOME=...` entry, are required for the server to work properly.

B.2 What Configuration Files Mean

This section lists all the entries that can appear in a configuration file for a Darwin server and explains what they mean.

- **`architecture`** is one of the architectures supported by Darwin. `darwinconfig` prints a list of the currently supported architectures, which are
  - **`sparc`**: Sun Microsystems UltraSPARC workstations or Ultra Enterprise Servers running Solaris 2.7 (single-node or multinode servers)
  - **`sparcdb`**: Sun Microsystems UltraSPARC workstation Ultra Enterprise servers running Solaris 2.7 with database connectivity (single-node or multinode servers)
  - **`hpux`**: Hewlett-Packard 9000 Series 700 or Series 800 single-node or multinode system running HP-UX 11.0
  - **`hpuxdb`**: Hewlett-Packard 9000 Series 700 or Series 800 single-node or multinode system running HP-UX 11.0 with database connectivity

    Note: If you specify either of the architectures **`sparcdb`** or **`hpuxdb`** you must have Merant DataDirect Connect ODBC and Oracle software installed. A server configuration supporting database connectivity must also specify the correct environment variables. (See Chapter 5 for details.)

  You specify a multinode or single node server using the **`nnodes`** entry.

  This item is required for all types of servers.

`name` is the name of the server (that is, configuration) to be displayed in the list of choices presented to the user upon entering Darwin. The name is also used in the `.darwinservers` file, which `darwinconfig` creates and updates automatically. `name` is also the name of the configuration file that describes the server. This item is required for all types of servers. Two different servers cannot have the same name. If a server has been used, you cannot create a new server with the same name, even if you remove the old server, since the name is embedded in `.darwinprojects` and `.darwinhostrc` in users' home directories.

`description` is the text describing the server that is displayed to users on the **Darwin Login** screen. This item is required for all types of servers.

`server` is the hostname of the workstation or server where the Darwin server will run. This address plus the port address are used to connect users to the Darwin

server. You can also specify the IP address of the workstation or server for this item. This item is required for all types of servers.

port is the TCP port number where the daemon listens for connect requests. For security reasons, the port number should be less than 1024 if the daemon is running as root and greater than 1024 if the daemon is running under a specific username. If several users will share the server, the daemon must run as root. The port number specified here must not be used by any other installed applications as described in `/etc/services` or the services map if using NIS or NIS+. This item is required for all types of servers.

executable is the path to the Darwin host executable the daemon will invoke when a user selects this server at the start of a Darwin session. This item is required for all types of servers.

auto_restart specifies that this Darwin server is automatically restarted when the system reboots. All daemons started at reboot will run as root. You must be running `darwinconfig` while logged on as root to specify `auto_restart`. You can use the `list` subcommand of `darwinconfig` to see whether `auto_restart` is specified for a particular server. (Be sure to save the configuration before you list it or reboot the server host.) This item is required for all types of servers.

nnodes is the number of nodes on the server that the Darwin server will use. This number must be greater than or equal to 1 and less than or equal to the total number of CPUs that Darwin will use. This item is required for all servers.

The following two keywords can be used when you specify `distributed_root` and `serial_root`:

- **USERHOME** — Used to specify that project directories are created in a subdirectory of the user's home directory on UNIX. For example, if serial root is specified as `$USERHOME/darwin` and if the user with home directory `/users5/csmith` has a subdirectory `/users5/csmith/darwin`, and created the project `ad_campaign_3`, the serial directory for that project would be `/users5/csmith/darwin/ad_campaign_3`.
- **USERNAME** — Used to specify that a user's projects are stored in UNIX directories that contain the name under which a user is logged in. Darwin creates a subdirectory for each user and places the user's projects in that subdirectory. For example, if the distributed root is `/export/darwin/$USERNAME`, and the user `jjones` creates the project `proj`, the distributed directory for that project would be `/export/darwin/jjones/proj`.

Note that you *cannot* use these keywords to specify shared roots.

serial_root is the base directory for users' nonshared serial project directories. (These are directories in the UNIX file system, usually under the users' home directories.) This item is automatically set to the directory that you speicfy in response to the "Root directory for projects:" prompt in darwinconfig add.This item is required for all types of servers.

> **Note:** When you configure a Darwin server, make sure that the partition where the serial_root directory resides is large enough to hold the datasets that you intend to use.

distributed_root is the base directory for all distributed directories created for projects run by this server. On any server running on a Solaris or HP-UX system, the entry for distributed_root is automatically set to Default, signifying that the distributed root is the same as the serial root.

shared_serial_root, **shared_distributed_root**, and **shared_project_file** are necessary only if you want to allow shared projects on a server. (Shared projects are visible to all users of Darwin on the server. Anyone using a shared project has full access to all project files.) You cannot use the keywords USERHOME and USERNAME when you specify these items.

- **shared_serial_root** is the base directory for shared projects' serial directories. The directory must be writable by all Darwin users. If the shared_serial_root is /users5/darwin_shared, Darwin places the serial part of all shared projects in /users5/darwin_shared. For example, if jjones created the shared project demos, its serial directory would be /users5/darwin_shared/demos.
- **shared_distributed_root** is the base directory for all distributed directories created for shared projects run by this server.
- For servers running on a Solaris system, the entry for shared_distributed_root is Default, signifying that the shared distributed root is the same as the shared serial root.
- **shared_project_file** is the pathname of a file that lists all the shared projects that are available to users of Darwin on this server. It contains information in the same format as the .darwinprojects file. This file must be writable by all users creating shared projects.

env entries will be added to the environment of the executable. Entries can be added as required by the database product that you are using or for other reasons. These variables are used by the Darwin server, not the Darwin daemon. See Section

5.4 for information about the environment variables required for database connectivity. See Sections 4.3.1 and 4.4.12 and the discussions of USERHOME and USERNAME in this section for information about Darwin environment variables.

Index

Symbols

? subcommand, 4-6, 4-10

A

add subcommand, 4-6, 4-7
adding a server, 4-7
adding Darwin environment variables, 4-10
architecture, B-5
auto restart, B-6

C

client
  installation
    server information, 6-1
client requirements, 2-3
  Excel, 2-4
  Internet Explorer, 2-3
client/server design, 4-2
code generation, 2-4
configuration
  subcommands, 4-5
configuration file, 4-1
  adding environment variables, 4-10
  architecture, B-5
  auto_restart, B-6
  description, B-5
  distributed_root, B-6
  env, B-7
  executable, B-6
  keywords, B-6
  meaning of, B-5
  modifying, 4-10
  name, B-5
  nnodes, B-6
  overview, B-1
  port, B-6
  sample file
    multinode system, B-3
    single-node system, B-2
  serial_root, B-6
  server, 4-3, B-1, B-5
  shared_distributed_root, B-7
  shared_project_file, B-7
  shared_serial_root, B-7
  upgrading, 4-5
customization disk, 6-2
customization file, 6-2

D

Darwin
  starting, 7-1
  stopping, 7-2
Darwin environment variables
  adding, 4-10
darwinconfig, 4-1
  before using, 4-4
darwinconfig command
  ? subcommand, 4-10
  add subcommand, 4-7
  describe subcommand, 4-8
  exit subcommand, 4-10
  help subcommand, 4-10
  list subcommand, 4-6
  quit subcommand, 4-10

remove subcommand, 4-8
save subcommand, 4-10
start subcommand, 4-9
stop subcommand, 4-9
terminating, 4-10
verify subcommand, 4-8
darwinconfig subcommands, 4-5
DARWINHOME, 4-5
.darwinhostrc file, 4-10, B-5
.darwinprojects file, 4-10, 8-4, 8-5, B-5
.darwinservers file, 4-3, B-1
DarwinServers.txt file, 6-2
database connectivity
considerations, 4-11, 5-1, B-7
disconnecting, 5-8
installation check, 5-7
ODBC drivers, 2-6, 5-3
permissions, 5-4
required files, 5-5
requirements, 2-6
using, 5-7
database connectivity requirements
accessibility, 5-4
environment variables, 5-4
Database Export Wizard, 5-9
Database Import Wizard, 5-8
DataDirect Connect ODBC drivers, 2-6, 5-1
datasets
size of, B-7
DBMS/COPY, A-1
default
shared distributed root, B-7
deinstallation, 3-2
describe subcommand, 4-6, 4-8
description, B-5
disk space requirements, 2-3
distributed dataset, 2-8, 2-9
distributed root, B-6
distributed_root, B-7
documentation, x
documentation CD, xi
documentation conventions, xii

E

env, B-7
environment variables
adding, 4-10
Darwin variables, 4-10
DARWINHOME, 4-5
database connectivity, 5-4
database packages, 4-11, B-7
USERHOME, 8-3, B-6
USERNAME, 8-3, B-6
/etc/services file, 5-6
executable, B-6
exit subcommand, 4-6, 4-7, 4-10
exporting models, 2-4

F

F1 (context-sensitive) help, xi

H

hardware requirements, 2-1
help
online, xi
help subcommand, 4-6, 4-10

I

installation
client software, 6-3
clients
customization disk, 6-2
customization file, 6-2
Darwin software, 1-1, 2-1
server software, 3-3

K

keywords, B-6

L

list subcommand, 4-6
listing servers, 4-6

M

Merant drivers, 2-6
multi-node servers
  performance, 2-7
  requirements, 2-7
multi-node severs
  capacity planning, 2-7

N

nnodes, B-6

O

ODBC drivers, 2-6
.odbc.ini file, 5-6
online help, xi

P

partition, size of, B-7
Perl, A-1
pkgadd, 3-3
pkgrm, 3-1
port, B-6
projects
  administering, 8-1
  deleting, 8-3
  optional information, 8-2
  shared, 8-3
  user, 8-3

Q

quit subcommand, 4-6, 4-10

R

release media, 1-1, 2-1
remove subcommand, 4-6, 4-8
removing Darwin, 3-1
requirements
  disk space, 2-3
  disk space on client, 2-3
  disk space on server, 2-3
  for exporting models, 2-4
  for SMPs, 2-7
  for Ultra Enterprise Servers, 2-7
  network connection, 2-3
  Solaris version, 2-1
  Windows versions, 2-1
root
  distributed, B-6
  serial, B-6
  shared distributed, B-7
  shared serial, B-7

S

SAS conversion utilities
  DBMS/COPY, 2-5
  perl, 2-5
SAS conversion utility
  installation, A-1
  requirements, 2-5
save subcommand, 4-6, 4-7, 4-10
serial datasets, 2-8, 2-9
serial root, B-6
serial_root, B-7
server, B-5
  configuration file, B-1
  installing, 3-3
  name, B-5
  overview, 4-2
  upgrading, 4-5
shared distributed root, B-7
  default, B-7
shared project file, B-7
shared projects, 8-3
shared serial root, B-7
SMP
  performance, 2-7
  requirements, 2-7
SQL limitations, 5-8
start subcommand, 4-6, 4-9
starting Darwin, 7-1
  UNIX, 7-1
  Windows 95, 7-1
  Windows NT, 7-1
stop subcommand, 4-6, 4-9

stopping Darwin, 7-2
UNIX, 7-3
Windows 95, 7-2
Windows NT, 7-2

T

tsnames.ora file, 5-6

U

Ultra Enterprise Server, 2-1
performance, 2-7
requirements for, 2-7
UltraSPARC, 2-1
user projects, 8-3
USERHOME, 8-3, B-6
USERNAME, 8-3, B-6

V

verify subcommand, 4-6, 4-8