

# Oracle9i

## Release Notes

Developer Release 2 (9.2.0.1.0) for Mac OS X

August 2002

**Part No. B10046-01**

This document accompanies Oracle9i Developer release 2 (9.2.0.1.0) for Mac OS X. Its contents supplement or supersede information in the Oracle9i documentation library.

This is a developer release. It is not intended to be used in a production environment. You should try this software only in a test environment.

This version supports only the core rdbms database. It does not support database options such as Real Application Clusters, JavaVM, and other advanced database features.

Topics:

- [System Requirements](#)
- [Documentation](#)
- [Installation and Configuration](#)
- [Using the Product Demonstration Application](#)
- [Known Issues](#)

## System Requirements

You must have a Power PC G4 based Apple Mac OS X server for this Oracle software. The following are minimal requirements:

- 512 MB memory
- 2 GB available disk space
- Mac OS X Server "Jaguar" 10.2 version 6c115 (August 2002) or later, which is available from Apple Corporation.

**ORACLE®**

Oracle is a registered trademark, and Oracle9i, SQL\*Net and SQL\*Plus are trademarks or registered trademarks of Oracle Corporation. Other names may be trademarks of their respective owners.

Copyright © 2002, Oracle Corporation.  
All Rights Reserved.

## Documentation

These release notes provide basic installation, configuration and usage instructions for Oracle9i Developer's Release 2 (9.2.0.1.0). For additional documentation, refer to the generic Oracle product documentation, which is available at the following Web site:

<http://technet.oracle.com/>

## Installation and Configuration

Complete the following tasks:

### Install the OS and Oracle9i

1. Install the Mac OS X Server 10.2 version 6c115 (August 2002) or later on the system. Oracle Corporation recommends that you do a clean installation, and wipe out the old operating system.
2. Set up the network to match your preferences. Either use static IP, or use DHCP to obtain an IP address.
3. Create the initial user as `admin`, and grant that user root access. You can use another user name, as long as it can log in as root.
4. Create the user account `oracle`, with the password `oracle`. Set Autologin for user `oracle`. Enable OS X administrative privileges for user `oracle`.

The `oracle` user is the account that owns Oracle9i software after installation. All installation is done by the `oracle` user.

5. Log in as the `oracle` user.
6. Using a text writer such as `vi`, open the configuration file `/etc/inetd.conf`. Activate the FTP and Telnet services by removing the `#` sign in front of the FTP and Telnet lines. Save the file and close it.

You need to have Sudo access to make these changes.

7. Using a text writer, open the file `/etc/hostconfig`. Add the following two lines:

```
PORTMAP=-YES-  
RPCSERVER=-YES-
```

You need to have Sudo access to make these changes.

8. If you have been obtaining a dynamic IP address from DHCP and have had problems with the network recognizing the hostname, then modify

the file `/etc/hosts` to include your computer IP address and hostname, using the following syntax:

```
ipaddress hostname
```

where the variable *ipaddress* is the numeric IP address for your system, and the variable *hostname* is the host name of your system.

For example, if your computer IP address is 130.35.154.53 and your hostname is ab-mac, then you would add the following:

```
130.35.154.53 ab-mac
```

You need to have Sudo access to make these changes.

To ensure that you correctly modified the file, run the following command:

```
> ping ab-mac
```

9. Restart the computer, and log in as the `oracle` user.
10. Create the directory `/Users/oracle/9iR2`
11. From either the CD-ROM or the Web, copy `oradev.tar.gz` to the directory `/Users/oracle/9iR2`.
12. Using the following command, untar `oradev.tar.gz` in the directory `/Users/oracle/9iR2`:

```
cd /Users/oracle/9iR2;  
tar xvzf oradev.tar.gz
```

This command creates the directory `orahome`. The directory `orahome` is the `ORACLE_HOME` directory. `ORACLE_HOME` refers to the directory `/Users/oracle/9iR2/orahome`.

## Creating the Database

Use the following procedure to create the database:

1. If necessary, log in as the `oracle` user.
2. Go to the directory `$ORACLE_HOME/DBCreate`.
3. Run the script `set_grp`. The script `set_grp` creates the OSDBA group (`dba`), and makes the `oracle` administrative account a member of the `dba` group. You must have sudo access to run this script. Use the following command:

```
./set_grp
```

4. Start the bash shell with the following command:

```
bash
```

5. Set system limits with the following command:

```
. set_ulimit
```

You may need to run this script again if you start a new shell session.

6. Using the DBENV script, set required environment variables with the following command:

```
. DBENV
```

7. Using the createDB script, create the database with the following command:

```
./createDB >& dblog
```

During database creation, log files are generated in the directory `$ORACLE_HOME/dbs`. Check the log files for any errors.

For reference, sample log files are available in the directory `samplelogs`.

Please be patient; this operation takes awhile to complete.

8. Log in to the database with the following SQL\*Plus command:

```
sqlplus "/ as sysdba"
```

## Starting Up and Shutting Down the Database

After installation, shut down the database and start it up again. A restart ensures a clean database instance. Use these startup and shutdown SQL commands each time you use the database.

To shut down the database:

```
SQL> shutdown abort
```

To start up the database, use the following command:

```
SQL> startup pfile=/Users/oracle/9iR2/orahome/dbs/initab.ora
```

At this point, the database is started and available for use.

## Configuring SQL\*Net

Complete the following tasks to configure SQL\*Net.

1. Copy the file `$ORACLE_HOME/DBCreate/listener.ora` to the directory `$ORACLE_HOME/network/admin`.

2. Copy the file `$ORACLE_HOME/DBCreate/tnsnames.ora` to the directory `$ORACLE_HOME/network/admin`.
3. Change the hostname in the `listener.ora` and `tnsnames.ora` files to your system's hostname. Additional changes may be required, depending on your environment. Refer to comments in the files for further information.
4. Stop listener processes with the following command:

```
lsnrctl stop
```

5. Start the listener processes with the following command:

```
lsnrctl start
```

6. Shut down and start up the database, as described in the preceding section "[Starting Up and Shutting Down the Database](#)". This is called "bouncing" the database. You must bounce the database to enable the listener and the database to communicate.
7. Use the following command to check the Listener status:

```
lsnrctl status
```

If the listener is configured correctly, then you should see a message similar to the following at the end of the status statement:

```
Instance "ab", status READY, has 1 handler(s) for this service...
```

If you do not see a statement similar to this, then check your `tnsnames.ora` network configuration file in the `listener.ora` and `tnsnames.ora` files.

---

---

**Note:** For information on how to check network configuration files, refer to *Oracle9i Net Services Administrator's Guide Release 2 (9.2)*.

---

---

8. Connect to the database.

If you connect to the database with the TCP/IP protocol, then use the following SQL\*Plus command:

```
sqlplus system/manager@abmactcp
```

where the variable `abmactcp` is an Oracle service defined in the file `tnsnames.ora`.

If you connect to the database with the IPC protocol, then use the following SQL\*Plus command:

```
sqlplus system/manager@abmacipc
```

where the variable *abmacipc* is an Oracle service defined in the file `tnsnames.ora`.

9. Conform that your database connection is valid with the following SQL command:

```
SQL> Select * from v$instance;
```

Review the information returned by this command about the instance to confirm that the instance is active.

## Utility Scripts in DBCreate Directory

The following section describes utility scripts provided with this release to automate database processes. These scripts are located in the directory `DBCreate`.

### **allup.sh**

The script `allup.sh` starts all required processes in the specific order required. Review the comment text in the script for further information on the script's operations.

### **alldown.sh**

The script `alldown.sh` shuts down all processes. Review the comment text in the script for further information on the script's operations.

## Using the Product Demonstration Application

To run the product demonstration (demo) application, you must have the Apple Apache server and Tomcat installed.

Change directories to the `$ORACLE_HOME/DBCreate/oradev` directory, open the file `Readme.txt`, and follow the instructions to install and use the application.

## Known Issues

The following is a list of known issues that affect Oracle9i release 2 (9.2.0.1.0):

## Database Bouncing Required

The database may need to be bounced (shut down and started up again) so that the listener can start listening on the TCP/IP port. Refer to "[Starting Up and Shutting Down the Database](#)" for instructions.

## Datafile Corruption on Improper Database Shutdown

You must stop all running Oracle database instances before shutting down your system. Data files can become corrupted by system crashes, or by shutting down the system without stopping databases.

## JDBC Libraries with Tomcat

The Tomcat server is unable to locate oci8-based JDBC dynamic libraries. To work around this problem, refer to the steps in the script `allup.sh`.

