

## **Oracle9i Client**

Installation and Administration Guide Release 2 (9.2.0.4.0) for  
Stratus Contium 600 and 1200 Series

December 2003

Copyright © 2001, 2003, Oracle. All rights reserved.

Contributing Authors: Caroline Johnston, Sharad Lal, Tom Mallory, Binoy Sukumaran

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

---

---

# Contents

<b>Send Us Your Comments .....</b>	<b>vii</b>
<b>Preface .....</b>	<b>ix</b>
<b>1 Pre-Installation Requirements</b>	
<b>Hardware Requirements .....</b>	<b>1-1</b>
<b>Disk Space Requirements.....</b>	<b>1-2</b>
<b>Operating System Requirements .....</b>	<b>1-2</b>
<b>Product-Specific Requirements .....</b>	<b>1-2</b>
Precompilers and Tools .....	1-2
Network and System Management Products .....	1-3
<b>Character Sets .....</b>	<b>1-3</b>
Oracle9i NCHAR Datatypes .....	1-3
AL24UTFFSS Character Set .....	1-3
<b>2 Installation</b>	
<b>Installation Instructions.....</b>	<b>2-1</b>
<b>Additional Information.....</b>	<b>2-2</b>
Relinking Information .....	2-2
<b>3 Post-Installation</b>	
<b>Configuration Tasks to Perform as the root User .....</b>	<b>3-1</b>
<b>Configuration Tasks to Perform as the oracle User .....</b>	<b>3-2</b>
<b>Additional Oracle Product Installation and Configuration.....</b>	<b>3-3</b>
Configuring Oracle Precompilers .....	3-3
<b>4 Administering SQL*Plus</b>	
<b>Administering Command-Line SQL*Plus .....</b>	<b>4-1</b>
Using Setup Files .....	4-1
Using the Site Profile File.....	4-1
Using the User Profile File.....	4-1
Using the PRODUCT_USER_PROFILE Table .....	4-2
Using EMP and DEPT Tables.....	4-2
Creating Demonstration Tables Manually .....	4-2

Deleting Demonstration Tables .....	4-2
SQL*Plus Command-Line Help .....	4-3
Installing the SQL*Plus Command-Line Help.....	4-3
Removing the SQL*Plus Command-Line Help .....	4-3
<b>Using Command-Line SQL*Plus .....</b>	<b>4-4</b>
Using a System Editor from SQL*Plus.....	4-4
Running Operating System Commands from SQL*Plus .....	4-4
Interrupting SQL*Plus.....	4-5
Using the SPOOL Command.....	4-5
<b>SQL*Plus Restrictions .....</b>	<b>4-5</b>
Resizing Windows .....	4-5
Return Codes .....	4-5
Hiding Your Password.....	4-5

## 5 Using Oracle Precompilers and the Oracle Call Interface

<b>Overview of Oracle Precompilers .....</b>	<b>5-1</b>
Precompiler Files.....	5-1
Precompiler README Files .....	5-2
Issues Common to All Precompilers.....	5-2
Uppercase to Lowercase Conversion.....	5-2
Vendor Debugger Programs .....	5-2
Value of IRECLLEN and ORECLLEN.....	5-2
Static Linking .....	5-2
<b>Pro*C Precompiler.....</b>	<b>5-2</b>
Known Issue with Pro*C.....	5-2
Pro*C Demonstration Programs .....	5-3
Pro*C User Program .....	5-3
<b>Pro*COBOL Precompilers .....</b>	<b>5-4</b>
Known Issue with Pro*COBOL.....	5-4
<b>Oracle Call Interface .....</b>	<b>5-5</b>
OCI Demonstration Programs.....	5-5
OCI User Programs.....	5-5
<b>Custom Make Files.....</b>	<b>5-5</b>
<b>Using Signal Handlers .....</b>	<b>5-6</b>

## 6 Configuring Oracle Net Services

Location of Oracle Net Services Configuration Files.....	6-1
<b>Oracle Protocol Support.....</b>	<b>6-2</b>
TCP/IP Protocol Support.....	6-2

## A Oracle9i Components

<b>Oracle9i Client Components.....</b>	<b>A-1</b>
Runtime Edition .....	A-1
Administrator Edition .....	A-1

**Product Descriptions** ..... A-1

**Index**



---

---

# Send Us Your Comments

## **Oracle9i Client Installation and Configuration Guide, Release 2 (9.2.0.4.0) for Stratus Contium 600 and 1200 Series**

Oracle 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 title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Postal service:

Oracle Corporation  
Oracle9i Documentation  
500 Oracle Parkway  
Redwood Shores, CA 94065  
USA

If you would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

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



---

---

# Preface

This guide and the *Oracle9i Administrator's Reference Release 2 (9.2.0.1.0) for UNIX Systems* provide instructions for installing and configuring Oracle9i Client release 2 (9.2.0.4.0) on Stratus Continuum systems.

## Audience

This document is intended for anyone responsible for installing Oracle9i Client release 2 (9.2.0.4.0) on UNIX systems.

## Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

**Accessibility of Code Examples in Documentation** JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

**Accessibility of Links to External Web Sites in Documentation** This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

## Typographic Conventions

The following typographic conventions are used in this guide:

Convention	Description
<code>monospace</code>	Monospace type indicates UNIX commands, directory names, usernames, pathnames, and filenames.
<i>italics</i>	Italic type indicates a variable, including variable portions of filenames. It is also used for emphasis and for book titles.
UPPERCASE	Uppercase letters indicate Structured Query Language (SQL) reserved words, initialization parameters, and environment variables.
<cr>	This string indicates a newline character.

## Command Syntax

UNIX command syntax appears in `monospace` font and assumes the use of the Bourne shell. The "\$" character at the beginning of UNIX command examples is the default UNIX command prompt. Do not enter it as part of the command.

Convention	Description
backslash \	A backslash indicates a command that is too long to fit on a single line. Enter the line as displayed (with a backslash) or enter it on a single line without a backslash:  <pre>dd if=/dev/rdsk/c0t1d0s6 of=/dev/rst0 bs=10b \ count=10000</pre>
braces {}	Braces indicate required items: <code>.DEFINE {macro1}</code>
brackets []	Brackets indicate optional items: <code>cvtcrt termname [outfile]</code>
ellipses ...	Ellipses indicate an arbitrary number of similar items: <code>CHKVAL fieldname value1 value2 ... valueN</code>
<i>italics</i>	Italic type indicates a variable. Substitute a value for the variable: <code>library_name</code>
vertical line	A vertical line indicates a choice within braces or brackets: <code>SIZE filesize [K M]</code>

## Related Documentation

If you are unfamiliar with the concepts or terminology associated with relational database management systems, then refer to *Oracle9i Database Concepts* before beginning your installation. Use the Installation Checklist in Appendix C to ensure that you have the required information and that you have completed the necessary pre-installation steps for a successful installation.

Information on system administration and tuning for a production database system is provided in these documents:

- *Oracle9i Database Administrator's Guide*
- *Oracle9i Net Services Administrator's Guide*
- *Oracle9i Database Performance Guide and Reference*

Information on upgrading from a previous version of the Oracle Server is provided in *Oracle9i Database Migration*.

Information on installing Oracle Workflow is provided in the *Oracle Workflow Server Installation Notes* and *Oracle Workflow Client Installation Notes (Release 2.6.2)*.

# Oracle Services and Support

The Oracle corporate Web page is located at:

<http://www.oracle.com>

Oracle offers a wide range of services from this Web site to help facilitate corporate system solutions, including Oracle Education courses, Oracle Consulting services, and Oracle Support Services. In addition, Oracle provides free trial software, updates on Oracle products and services, and technical brochures and data sheets.

## Oracle Support Services

Technical Support registration and worldwide contact information is available at:

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

At the Oracle support site, templates are available to help you prepare information about your problem before you call so that you may be helped more quickly. You must also provide your CSI (Customer Support Identifier) or SAC (Support Access Code) number if applicable or complete contact details, including any special project information.

## OracleMetaLink

OracleMetaLink is an Oracle Web service that provides technical information. Members of OracleMetaLink can search for updates, alerts, patches, and other information about products, releases, and operating systems, or set preferences to be notified automatically of new information. OracleMetaLink offers a variety of services to assist in setting up and administrating Oracle products, including procedures, scripts, commentary, and tuning and configuration best-practices bulletins. In addition, it offers forums for information sharing among Oracle customers, and direct communication with Oracle Support Services. OracleMetaLink is available to Product Support Customers at no extra cost. Sign up for free membership for this service at:

<http://www.oracle.com/support/metalink>

Use your CSI or SAC number to register.

## Products and Documentation

For U.S.A. customers, Oracle Store is at:

<http://oraclestore.oracle.com>

Links to Stores in other countries are provided at this site.

Product documentation is available at:

<http://docs.oracle.com>

## Customer Service

Global Customer Service contacts are listed at:

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

## Education and Training

Training information and worldwide schedules are available at:

<http://www.oracle.com/education/>

## **Oracle Technology Network**

Register with the Oracle Technology Network (OTN) at:

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

OTN delivers technical papers, code samples, product documentation, an Oracle on-line community and self-service developer support. It also offers the Oracle key developer products to enable rapid development and deployment of applications built on Oracle technology.

---



---

# Pre-Installation Requirements

This chapter describes pre-installation steps required for an Oracle9i Client release 2 (9.2.0.4.0) software installation. Verify that your system meets the hardware, disk space, operating system, and other requirements for installing the Oracle9i Client release 2 (9.2.0.4.0).

This chapter includes the following sections:

- [Hardware Requirements](#)
- [Disk Space Requirements](#)
- [Operating System Requirements](#)
- [Product-Specific Requirements](#)
- [Character Sets](#)

## Hardware Requirements

To install the Oracle9i Client release 2 (9.2.0.4.0) included with this release, your system must meet the minimum hardware requirements listed in the following table:

Hardware	Requirements
Memory	A minimum of 256 MB is required to install Oracle9i Client
Swap space	Disk space equal to the system's physical memory, or 1GB, whichever is greater
Tape drive	A DDS or DLT tape drive
CPU	Stratus Continuum Series module with HP-PA8x00 class CPU
Network adapter	Ethernet adapter
Free disk blocks	At least 100,000 free disk blocks on the installation disk for the base installation, and 350,000 free disk blocks if all the demonstration and sample programs are built.

### Random Access Memory

To determine the amount of random access memory installed on your system, enter the following command:

```
analyze_system
dump_mt
quit
```

**Swap Space**

To determine the amount of swap space currently configured in your system, enter the command:

```
display_paging_usage
```

## Disk Space Requirements

The disk space requirements for the *Oracle9i* client software is 2.0 GB.

---

---

**Note:** This is an approximate value that might vary slightly at installation time.

---

---

## Operating System Requirements

This section lists the *Oracle9i* operating system requirements.

- Operating system version:
  - Stratus VOS 14.6.1 ag or later.
- S115 Streams TCP.
- S579 Open software.
- S877 and S878 GNU tools and library. This software is required to be installed for Oracle support purposes. The user is not required to use this software to build their applications, but it is required to use the Oracle make files.
- S456 ANSI C compiler and S031 C runtime library, if the C precompiler or OCI interface will be used.
- S020 COBOL compiler and S021 COBOL runtime library, if the COBOL precompiler will be used.
- S279 POSIX library, if the C or COBOL precompilers, or the OCI interface will be used.
- Operating system patches:
  - None.

Use the following command to determine the current operating system version:

```
uname -a
```

## Product-Specific Requirements

This section provides product-specific system configuration requirements. Make these additional system configuration changes to use the optional Oracle products.

## Precompilers and Tools

[Table 1-1](#) lists the restrictions and requirements for precompilers and tools.

**Table 1–1 Precompilers and Tools Restrictions and Requirements**

Product Name	Platform	Restrictions and Requirements
Pro*C/C++, 9.2.0.4.0	Continuum 600 and 1200 Series	<ul style="list-style-type: none"> <li>■ S456 VOS standard C compiler</li> <li>■ S031 C Runtime</li> </ul>
Pro*COBOL, 1.8.77	Continuum 600 and 1200 Series	<ul style="list-style-type: none"> <li>■ S020 COBOL compiler</li> <li>■ S021 COBOL Runtime</li> </ul>
Pro*COBOL, 9.2.0.4.0	Continuum 600 and 1200 Series	<ul style="list-style-type: none"> <li>■ S020 COBOL compiler</li> <li>■ S021 COBOL Runtime</li> </ul>

## Network and System Management Products

All network products require the underlying software and operating system libraries for the supported network. The network software must be installed and running prior to installing the Oracle Net products.

**See Also:** Your operating system and third-party vendor networking product documentation for more information on network and system management products.

## Character Sets

This section provides information on restrictions and updates to character sets.

### Oracle9i NCHAR Datatypes

Oracle9i Client release 2 (9.2.0.4.0) limits the SQL NCHAR datatypes to the Unicode character set encoding (UTF8 and AL16UTF16). Alternative this release no longer supports character sets such as the fixed-width Asian character set JA16SJISFIXED in Oracle8i.

To migrate existing NCHAR, NVARCHAR, and NCLOB columns, export and import NCHAR columns, complete the following steps:

1. Export all SQL NCHAR columns from Oracle8i.
2. Drop the SQL NCHAR columns.
3. Migrate the database to Oracle9i.
4. Import the SQL NCHAR columns in to Oracle9i.

### AL24UTFFSS Character Set

Oracle9i Client release 2 (9.2.0.4.0) does not support the Unicode character set AL24UTFFSS introduced in Oracle7. This character set is based on the Unicode standard 1.1, which is now obsolete.

Oracle9i Client release 2 (9.2.0.4.0) supports the Unicode database character sets AL32UTF8 and UTF8. These database character sets include the Unicode enhancements based on the Unicode standard 3.0.

To migrate the existing AL24UTFFSS database, upgrade your database character set to UTF8 before upgrading to Oracle9i. Oracle recommends that you use the Character Set Scanner for data analysis before attempting to migrate your existing database character set.



---

---

# Installation

This chapter describes how to install Oracle9i Client release 2 (9.2.0.4.0) products on your system. Review and complete the tasks listed in [Chapter 1, "Pre-Installation Requirements"](#) before beginning the installation. This chapter contains the following sections:

- [Installation Instructions](#)
- [Additional Information](#)

## Installation Instructions

The following instructions describe how to install Oracle9i Client release 2 (9.2.0.4.0) on the system.

1. Insert the Oracle9i Client release 2 (9.2.0.4.0) distribution tape into a tape drive on the module.
2. Log in to the module using an account with the appropriate access to create the Oracle9i directory structure in the appropriate directory. For example, to install Oracle9i on the master disk in the system directory, log in using an account in the SysAdmin group. You must log in with the `-privileged` option no matter what account is used to log in.
3. Create the Oracle home directory. Oracle9i can be installed on any disk or directory. For example, to install Oracle9i on the master disk in a system directory named Oracle9i, use the following command:

```
create_dir (master_disk) >system>Oracle9i
```

4. Change to the directory that was created in step 3. For example, if using the Oracle9i directory, then enter the following command:

```
change_current_dir (master_disk)>system>Oracle9i
```

5. Attach a port to the tape drive using the `attach_port` command. For example, if the port is `t`, and drive name is `#mt1.0`, then enter the following command:

```
attach_port t #mt1.0
```

6. Mount the tape for read-only access using the `mount_tape` command. For example, if the port is `t`, then enter the following command:

```
mount_tape t -access_rights readonly
```

A system prompt may appear asking if the tape is correct.

7. Extract the Oracle9i distribution into the current directory using the restore command. For example, if the port is `t`, then enter the following command:

```
restore t -restore_acls
```

If the system prompts "What tape is to be used?", then enter the volume id shown in the output from the `mount_tape` command in step 6.

This step may take 45 minutes or more to complete. No error messages should be generated during the process.

8. Detach the tape port to unload the tape. For example, if the port is `t`, then enter the following command:

```
detach_port t
```

9. Create a link as follows, if it does not already exist:

```
link (master_disk)>system>gnu_library>bin (master_disk)>bin
```

This link allows the Oracle make files to invoke the bash shell as `/bin/sh`.

10. Add the Oracle execution (binary) directory to the command library paths. For example, the following command adds the Oracle execution directory to the library path:

```
add_library_path command (master_disk)>system>Oracle9i>bin
```

You may want to make this a default library path for the module.

## Additional Information

The following information is specific for Stratus Contium 600 and 1200 Series installations of Oracle9i:

- Any references in the Oracle documentation to the `make` command must be substituted with the `gmake` command.
- Some Oracle header files contain names that are longer than the 32-character VOS limit. This can cause the binder to report missing symbols due to truncated names. The header file `$ORACLE_HOME/vos/long_name_xlate_table.h` has been supplied to correct the problem. Include the file in the ANSI C compilation with the `-include` compiler directive.

The source file and ANSI options may not occur after the `-include header file` specification.

## Relinking Information

The following relinking information is specific for Stratus Contium 600 and 1200 Series installations of Oracle9i:

---

---

**Note:** You must use the Oracle make files to re-link any of the Oracle9i Client binaries and demonstrations. This requires you be in `bash`.

---

---

- Before re-linking any Oracle utilities, the VOS object library paths must include at least the following object libraries in the order shown:

```
(current_dir)
```

```
(master_disk)>system>stcp>object_library
(master_disk)>system>stcp>object_library>common
(master_disk)>system>stcp>object_library>sbsd
(master_disk)>system>stcp>object_library>socket
(master_disk)>system>posix_object_library>sysv
(master_disk)>system>c_object_library>
(master_disk)>system>object_library>
```

- **To relink SQL\*Plus, use the following command:**

```
bash
cd $ORACLE_HOME/sqlplus/lib
gmake -f ins_sqlplus.mk install
```

- **To relink tnspring, use the following commands:**

```
bash
cd $ORACLE_HOME/network/lib
gmake -f ins_net_client.mk install
```

- **To relink the precompilers, use the following commands:**

```
bash
cd $ORACLE_HOME/precomp/lib
```

- **For Pro\*C:**

```
gmake -f ins_precomp.mk relink EXECNAME=proc
```

- **For Pro\*COBOL:**

```
gmake -f ins_precomp.mk relink EXENAME=procob
```

- **For Pro\*COBOL 18:**

```
gmake -f ins_precomp.mk relink EXENAME=procob18
```



---



---

## Post-Installation

After completing installation, you must perform certain post-installation steps and configure the Oracle9i database. This chapter describes the required steps as well as some optional ones.

- [Configuration Tasks to Perform as the root User](#)
- [Configuration Tasks to Perform as the oracle User](#)
- [Additional Oracle Product Installation and Configuration](#)

---



---

**Note:** This chapter describes basic configuration only. See the *Oracle9i Administrator's Reference Release 2 (9.2.0.4.0) for UNIX Systems* and the product administration and tuning guides for more sophisticated configuration and tuning information.

---



---

### Configuration Tasks to Perform as the root User

Log in as the `root` user and verify database file security. To prevent unauthorized access to secure data, you must protect your files. The file privileges and recommended ownership are as follows:

- The `oracle` account should have read, write, and execute privileges for all files and directories in the Oracle9i installation.
- No user other than the `oracle` user should have write access on any files or directories in the Oracle9i installation.

[Table 3-1](#) summarizes the directory and file permissions for different types of files.

---



---

**Note:** These permissions are the default values and should not be changed.

---



---

**Table 3-1** Access Permissions on Oracle Directories and Files

Directories/Files	Permissions	Comments
<code>\$ORACLE_HOME/bin/</code>	755 rwxr-xr-x	The <code>oracle</code> user should have read, write and execute privileges and all users should have read and execute privileges to this directory.
All other executables	755 rwxr-xr-x	The <code>oracle</code> user should have read, write and execute privileges and all users should have read and execute privileges to this directory.

**Table 3–1 (Cont.) Access Permissions on Oracle Directories and Files**

Directories/Files	Permissions	Comments
\$ORACLE_HOME/lib/	755 rwxr-xr-x	The oracle user should have read, write and execute privileges and all other users should have read and execute privileges to this directory.
All files under \$ORACLE_HOME/lib/	644 rw-r--r--	The oracle user should have read, write and execute privileges and the other users should have read-only privileges to these files.
Product subdirectories such as \$ORACLE_HOME/sqlplus or \$ORACLE_HOME/rdbms	751 rwxr-x--x	The oracle user and ORAINVENTORY group have restricted access to log files. The other users have execute privileges.
Files in \$ORACLE_HOME/sqlplus or \$ORACLE_HOME/rdbms	644 rw-r--r--	The oracle user should have read and write privileges and the other users should have read-only privileges to these files.
\$ORACLE_HOME/network/trace	777 rwxrwxrwx or 730 rwx-wx---	The oracle user and members of the ORAINVENTORY group have access to trace files when 730 is used in the production environment. 777 allows broad access to view and create trace files during development.
All files under product admin directories, like \$ORACLE_HOME/rdbms/admin and \$ORACLE_HOME/sqlplus/admin	644 -rw-r--r--	SQL scripts should typically be run as the SYS user.

## Configuration Tasks to Perform as the oracle User

Perform the following task as the oracle user:

Set the following environment variables in the .profile or .login file of the oracle user before using Oracle9i products. The settings that you use here should correspond to the settings used during installation as described in "Set Environment Variables" on page 2-36. The syntax for setting environment variables is as follows:

For the Bourne or Korn shell:

```
variable_name=value; export variable_name
```

For the VOS command line:

```
set variable_name=value
```

Table 3–2 describes the Environment Variable Settings.

---



---

**Caution:** Do not define environment variables with names that are identical to those used for Oracle processes such as CKPT, PMON, and DBWR.

---



---

**Table 3–2 Environment Variable Settings**

Environment Variable	Recommended Setting
ORACLE_HOME	<p>Set the ORACLE_HOME variable. The ORACLE_HOME variable must be set before an Oracle utility or user application can be used. The directory referenced by the variable must be POSIX format. For example, if Oracle9i is installed in (master_disk)&gt;system&gt;Oracle9i, then enter the following command:</p> <pre>ORACLE_HOME=/system/Oracle9i</pre> <p>The following is also an acceptable command:</p> <pre>ORACLE_HOME=/%my_system#m1_d01/system/Oracle9i</pre> <p>However, the following is not acceptable:</p> <pre>ORACLE_HOME/%system#disk&gt;system&gt;Oracle9i</pre>
PATH	<p>Verify (master_disk)&gt;system&gt;gnu_library&gt;bin is in the default command library path list. If it is not, then it should be added to the list. This is required for Oracle support.</p> <p><b>See Also:</b> <a href="#">Chapter 2, "Pre-Installation Requirements"</a> for more information on other PATH requirements.</p>
TNS_ADMIN	<p>TNS_ADMIN specifies the directory containing the Oracle Net Services configuration files.</p> <p>Oracle Net Services configuration files are typically, but not always located in the \$ORACLE_HOME/admin/network directory. Depending on the file, Oracle Net uses a different search order to locate the file.</p> <p>The search order for the sqlnet.ora and ldap.ora files is as follows:</p> <ol style="list-style-type: none"> <li>1. The directory specified by the TNS_ADMIN environment variable, if set</li> <li>2. The \$ORACLE_HOME/admin/network directory</li> </ol>
TWO_TASK	<p>Set the TWO_TASK variable to the default connect identifier to use in the connect string. The default connect identifier does not need to be specified in the connect string. For example, if the TWO_TASK environment variable is set to sales, you can connect to a database using the CONNECT <i>username/password</i> command rather than the CONNECT <i>username/password@sales</i> command.</p> <p><b>See Also:</b> <i>Oracle9i Net Services Administrator's Reference and Oracle9i Administrator's Reference Release 2 (9.2.0.4.0) for UNIX Systems</i> for more information on connect identifiers.</p>

## Additional Oracle Product Installation and Configuration

This section provides additional information on configuring Oracle Precompilers.

### Configuring Oracle Precompilers

All precompiler configuration files are located in the \$ORACLE\_HOME/precomp/admin directory. [Table 3–3](#) describes the Oracle Precompilers.

---

---

**Note:** You cannot use Oracle Precompilers independently of Oracle9i to convert embedded PL/SQL.

---

---

**Table 3–3 Oracle Precompilers**

<b>Precompilers</b>	<b>Configuration Files</b>	<b>See Also</b>
Pro*C/C++	pcscfg.cfg This file installs without content and can be configured with any text editor according to the site-specific requirements.	<i>Programmer's Guide to the Pro*C/C++ Precompiler</i> for more information on configuring the pcscfg.cfg file.
Pro*COBOL	pcbcfg.cfg This file installs without content and can be configured with any text editor according to the site-specific requirements.	<i>Pro*COBOL Programmer's Guide</i> for more information on configuring the pcbcfg.cfg file.
Pro*COBOL18	pcccob.cfg This file installs without content and can be configured with any text editor according to the site-specific requirements.	<i>Pro*COBOL Programmer's Guide</i> for more information on configuring the pcccob.cfg file.

---

---

## Administering SQL\*Plus

This chapter describes how to use and administer SQL\*Plus on Oracle9i Client. It contains the following sections:

- [Administering Command-Line SQL\\*Plus](#)
- [Using Command-Line SQL\\*Plus](#)
- [SQL\\*Plus Restrictions](#)

**See Also:** *SQL\*Plus Users Guide and Reference* for more information on and SQL\*Plus.

### Administering Command-Line SQL\*Plus

This section describes how to administer command-line SQL\*Plus. In the examples in this section, SQL\*Plus uses the value of the ORACLE\_HOME environment variable wherever a question mark (?) appears.

This section includes the following topics:

- [Using Setup Files](#)
- [Using the PRODUCT\\_USER\\_PROFILE Table](#)
- [Using EMP and DEPT Tables](#)
- [SQL\\*Plus Command-Line Help](#)
- [Installing the SQL\\*Plus Command-Line Help](#)
- [Removing the SQL\\*Plus Command-Line Help](#)

### Using Setup Files

When you start SQL\*Plus, it executes the `glogin.sql` site profile set-up file and then executes the `login.sql` user profile set-up file.

#### Using the Site Profile File

The global site profile file is `$ORACLE_HOME/sqlplus/admin/glogin.sql`. If a site profile already exists at this location, it is overwritten when you install SQL\*Plus. If SQL\*Plus is removed, the site profile file is deleted.

#### Using the User Profile File

The user profile file is `login.sql`. SQL\*Plus looks for this file in the current directory, and then in the directories you specify using the `SQLPATH` environment variable. Set

this environment variable to a colon-separated list of directories. SQL\*Plus searches these directories for the `login.sql` file in the order they are listed.

The options set in the `login.sql` file override those set in the `glogin.sql` file.

**See Also:** *SQL\*Plus User's Guide and Reference* for more information on profile files.

## Using the PRODUCT\_USER\_PROFILE Table

Oracle9i provides the `PRODUCT_USER_PROFILE` table that you can use to disable the SQL and SQL\*Plus commands that you specify.

To create the `PRODUCT_USER_PROFILE` table, run the `$ORACLE_HOME/sqlplus/admin/pupbld.sql` script in the `SYSTEM` schema:

```
$ORACLE_HOME/sqlplus/admin/pupbld.sql
```

For example, enter the following commands, where `SYSTEM_PASSWORD` is the password of the `SYSTEM` user:

```
$ sqlplus SYSTEM/SYSTEM_PASSWORD
SQL> @?/sqlplus/admin/pupbld.sql
```

You can also recreate the `PRODUCT_USER_PROFILE` table manually in the `SYSTEM` schema using the `$ORACLE_HOME/bin/pupbld` shell script. This script prompts for the `SYSTEM` password. If you need to run the `pupbld` script without interaction, set the `SYSTEM_PASS` environment variable to the `SYSTEM` username and password. For example, enter the following command where `SYSTEM_PASSWORD` is the password of the `SYSTEM` user:

```
$ SYSTEM_PASS=SYSTEM/SYSTEM_PASSWORD; export SYSTEM_PASS
```

## Using EMP and DEPT Tables

This section describes how to manually create and delete the `EMP` and `DEPT` demonstration tables.

### Creating Demonstration Tables Manually

Use the `$ORACLE_HOME/sqlplus/demo/demobld.sql` SQL script to create the `EMP` and `DEPT` demonstration tables. In SQL\*Plus, you can use any username to run the `demobld.sql` script to create the demonstration tables in a schema. For example, enter:

```
$ sqlplus SCOTT/TIGER
SQL> @?/sqlplus/demo/demobld.sql
```

You can also use the `$ORACLE_HOME/bin/demobld` shell script to run the `demobld.sql` script, as follows:

```
$ demobld SCOTT TIGER
```

### Deleting Demonstration Tables

Use the `$ORACLE_HOME/sqlplus/demo/demodrop.sql` script to drop the `EMP` and `DEPT` demonstration tables. In SQL\*Plus, you can use any username to drop the demonstration tables from the user's schema. For example, enter:

```
$ sqlplus SCOTT/TIGER
```

```
SQL> @?/sqlplus/demo/demodrop.sql
```

You can also use the `$ORACLE_HOME/bin/demodrop` shell script to run the `demodrop.sql` script, as follows:

```
$ demodrop SCOTT TIGER
```

## SQL\*Plus Command-Line Help

This section describes how to install and remove the SQL\*Plus command-line help.

**See Also:** *SQL\*Plus User's Guide and Reference* for more information on the SQL\*Plus command-line help.

## Installing the SQL\*Plus Command-Line Help

There are three ways to install the SQL\*Plus command-line help:

- Perform an installation that installs a starter database.

When you copy a starter database with pre-built datafiles as part of an installation, SQL\*Plus automatically installs the SQL\*Plus command-line help in the SYSTEM schema.

- Install the command-line help manually in the SYSTEM schema using the `$ORACLE_HOME/bin/helpins` shell script.

The `helpins` script prompts for the SYSTEM password. If you need to run this script without interaction, set the `SYSTEM_PASS` environment variable to the SYSTEM username and password. For example, enter the following command where `SYSTEM_PASSWORD` is the password of the SYSTEM user:

```
$ SYSTEM_PASS=SYSTEM/SYSTEM_PASSWORD; export SYSTEM_PASS
```

- Install the command-line help manually in the SYSTEM schema using the `$ORACLE_HOME/sqlplus/admin/help/helpbld.sql` script.

For example, enter the following commands, where `SYSTEM_PASSWORD` is the password of the SYSTEM user:

```
$ sqlplus SYSTEM/SYSTEM_PASSWORD
SQL> @?/sqlplus/admin/help/helpbld.sql ?/sqlplus/admin/help helpus.sql
```

---



---

**Note:** Both the `helpins` shell script and the `helpbld.sql` script drop existing command-line help tables before creating new tables.

---



---

## Removing the SQL\*Plus Command-Line Help

To manually drop the SQL\*Plus command-line help tables from the SYSTEM schema, run the `$ORACLE_HOME/sqlplus/admin/help/helpdrop.sql` script. For example, enter the following commands, where `SYSTEM_PASSWORD` is the password of the SYSTEM user:

```
$ sqlplus SYSTEM/SYSTEM_PASSWORD
SQL> @?/sqlplus/admin/help/helpdrop.sql
```

## Using Command-Line SQL\*Plus

This section describes how to use command-line SQL\*Plus. It includes the following topics:

- [Using a System Editor from SQL\\*Plus](#)
- [Running Operating System Commands from SQL\\*Plus](#)
- [Interrupting SQL\\*Plus](#)
- [Using the SPOOL Command](#)

### Using a System Editor from SQL\*Plus

If you enter an ED or EDIT command at the SQL\*Plus prompt, the system starts an operating system editor, such as `ed`, `emacs`, `ned`, or `vi`. The `PATH` variable must include the directory where the editor executable is located.

When you start the editor, the current SQL buffer is placed in the editor. When you exit the editor, the changed SQL buffer is returned to SQL\*Plus.

You can specify which editor starts by defining the SQL\*Plus `_EDITOR` variable. You can define this variable in the `glogin.sql` site profile, the `login.sql` user profile, or define it during the SQL\*Plus session. For example, to set the default editor to `vi`, enter:

```
SQL> DEFINE _EDITOR=vi
```

If you do not set the `_EDITOR` variable, the value of either the `EDITOR` or the `VISUAL` environment variable is used. If both environment variables are set, the value of the `EDITOR` variable is used. When `_EDITOR`, `EDITOR`, and `VISUAL` are not specified, the default editor is `ed`.

If you start the editor, SQL\*Plus uses the `afiedt.buf` temporary file to pass text to the editor. You can use the `SET EDITFILE` command to specify a different filename. For example, enter:

```
SQL> SET EDITFILE /tmp/myfile.sql
```

SQL\*Plus does not delete the temporary file.

### Running Operating System Commands from SQL\*Plus

Using the `HOST` command or an exclamation mark (!) as the first character after the SQL\*Plus prompt causes subsequent characters to be passed to a sub-shell. The `SHELL` environment variable sets the shell used to execute operating system commands. The default shell is the Bourne shell (`/bin/sh`). If the shell cannot be executed, an error message is displayed.

To return to SQL\*Plus, enter the `exit` command.

For example, to execute one command, enter:

```
SQL>! command
```

In the preceding example, *command* represents the operating system command that you want to execute.

To execute multiple operating system commands from SQL\*Plus, enter the `HOST` or ! command then press Return. SQL\*Plus returns you to the operating system prompt.

## Interrupting SQL\*Plus

While running SQL\*Plus, you can stop the scrolling record display and terminate a SQL statement by pressing Ctrl+c.

## Using the SPOOL Command

The default file extension of files generated by the SPOOL command is .lst. To change this extension, specify a spool file containing a period (.). For example, enter:

```
SQL> SPOOL query.txt
```

## SQL\*Plus Restrictions

This section describes SQL\*Plus restrictions. It includes the following topics:

- [Resizing Windows](#)
- [Return Codes](#)
- [Hiding Your Password](#)

### Resizing Windows

The default values for the SQL\*Plus LINESIZE and PAGESIZE system variables do not automatically adjust for window size.

### Return Codes

UNIX return codes use only one byte, which is not enough space to return an Oracle error code. The range for a return code is 0 to 255.

### Hiding Your Password

If you set the SYSTEM\_PASS environment variable to the username and password of the SYSTEM user, the output from the ps command might display this information. To prevent unauthorized access, enter the SYSTEM password only when prompted by SQL\*Plus.

If you want to automatically run a script, consider using an authentication method that does not require you to store a password, for example, externally authenticated logins to Oracle9i. If you have a low security environment, you might consider using UNIX pipes in script files to pass a password to SQL\*Plus, for example:

```
$ echo SYSTEM_PASSWORD | sqlplus SYSTEM @MYSCRIPT
```

Alternatively, enter the following lines at the command prompt:

```
$ sqlplus <<EOF
SYSTEM/SYSTEM_PASSWORD
SELECT ...
EXIT
EOF
```

In the preceding examples, SYSTEM\_PASSWORD is the password of the SYSTEM user.



---

---

# Using Oracle Precompilers and the Oracle Call Interface

This chapter describes Oracle Precompilers and the Oracle Call Interface. It contains the following sections:

- [Overview of Oracle Precompilers](#)
- [Pro\\*C Precompiler](#)
- [Pro\\*COBOL Precompilers](#)
- [Oracle Call Interface](#)
- [Custom Make Files](#)
- [Using Signal Handlers](#)

## Overview of Oracle Precompilers

Oracle precompilers are application-development tools used to combine SQL statements for an Oracle database with programs written in a high-level language. Oracle precompilers are compatible with ANSI SQL and are used to develop open, customized applications that run with Oracle9i or any other ANSI SQL database management system.

To build any of the precompiler samples (C or COBOL) using the Oracle make files, do the following:

1. Verify TWO\_TASK is set to the server name.
2. Verify the account scott/tiger exists and the demo tables are installed on the server to which you will connect.
3. Run the following commands to setup the database:

```
cd $ORACLE_HOME/precomp/demo/sql
sqlplus scott/tiger @calldemo.sql
sqlplus scott/tiger @sample11.sql
sqlplus scott/tiger @cv_demo.sql
```

## Precompiler Files

Configuration files for the Oracle precompilers are located in the `$ORACLE_HOME/precomp/admin` directory. The name of the configuration file for the Pro\*C/C++ precompiler is `pcscfg.cfg`.

The executable file for the precompiler is `proc`.

## Precompiler README Files

[Table 5–1](#) lists the location of the precompiler README files. The README files describe changes made to the precompiler since the last release.

**Table 5–1 Location of Precompiler README Files**

Precompiler	README File
Pro*C/C++ release 9.2.0.4.0	\$ORACLE_HOME/precomp/doc/proc2/readme.doc
Pro*COBOL release 9.2.0.4.0	\$ORACLE_HOME/precomp/doc/procob2/readme.doc
Pro*COBOL release 1.8.77.0.0	\$ORACLE_HOME/precomp/doc/pro1x/readme.txt

## Issues Common to All Precompilers

The following issues are common to all precompilers.

### Uppercase to Lowercase Conversion

In languages other than C, the compiler converts an uppercase function or subprogram name to lowercase. This can cause a “No such user exists” error message. If you receive this error message, verify that the function or subprogram name in your option file matches the case used in the IAPXTB table.

### Vendor Debugger Programs

Precompilers and vendor-supplied debuggers can be incompatible. Oracle does not guarantee that a program run using a debugger will perform the same way when it is run without the debugger.

### Value of IRECLLEN and ORECLLEN

The IRECLLEN and ORECLLEN parameters do not have maximum values.

## Static Linking

You can statically link Oracle libraries with precompiler and OCI applications. With static linking, the libraries and objects of the whole application are linked together into a single executable program.

## Pro\*C Precompiler

Before you use the Pro\*C/C++ precompiler, verify that the correct version of the operating system compiler is properly installed.

## Known Issue with Pro\*C

The following is a known issue and its workaround:

### Issue:

char\_map gives extra control character on Stratus.

### Workaround:

Set char\_map to CHARZ or STRING. CHARZ is the default in Oracle9i.

## Pro\*C Demonstration Programs

Demonstration programs are provided to show the features of the Pro\*C/C++ precompiler. There are three types of demonstration programs: C, C++, and Object programs. Object programs demonstrate the new Oracle9i Object features. All of the demonstration programs are located in the `$ORACLE_HOME/precomp/demo/proc` directory.

The programs assume that the demonstration tables created by the `$ORACLE_HOME/sqlplus/demo/demobld.sql` script exist in the SCOTT schema with the password TIGER.

Use the `demo_proc.mk` make file, located in the `$ORACLE_HOME/precomp/demo/proc/` directory, to create the demonstration programs. For example, to precompile, compile, and link the `sample1` demonstration program, enter the following command:

```
bash
cd $ORACLE_HOME/precomp/demo/proc
gmake -f demo_proc.mk samples
```

To create all of the C demonstration programs for Pro\*C, enter:

```
$ gmake -f demo_proc.mk samples
```

To create all of the Object demonstration programs for Pro\*C, enter:

```
$ gmake -f demo_proc.mk object_samples
```

Some demonstration programs require you to run a SQL script, located in the `$ORACLE_HOME/precomp/demo/sql` directory. If you do not run the script, a message displays requesting you to run it. To build a demonstration program and run the corresponding SQL script, include the make macro argument `RUNSQL=run` on the command line. For example, to create the `calldemo` demonstration program and run the required `$ORACLE_HOME/precomp/demo/sql/calldemo.sql` script, enter:

```
$ gmake -f demo_proc.mk calldemo RUNSQL=run
```

To create all of the Object demonstration programs and run all corresponding required SQL scripts, enter:

```
$ gmake -f demo_proc.mk object_samples RUNSQL=run
```

## Pro\*C User Program

You can use the `$ORACLE_HOME/precomp/demo/proc/demo_proc.mk` make file to create user programs. The syntax for creating a program with the `demo_proc.mk` make file is:

```
$ gmake -f demo_proc.mk target OBJS="objfile1 objfile2 ..." EXE=exename
```

In the preceding example:

- *target* is the make file target that you want to use
- *objfile*n** is the object file to link the program
- *exename* is the executable program

For example, to create the program `myprog` from the Pro\*C source file `myprog.pc`, enter one of the following commands, depending on the source and the type of executable you want to create.

- For C source, dynamically linked with the client shared library, enter:  

```
$ gmake -f demo_proc.mk build OBJS=myprog.o EXE=myprog
```
- For C source, statically linked, enter:  

```
$ gmake -f demo_proc.mk build_static OBJS=myprog.o EXE=myprog
```
- To build the Pro\*C examples in the native VOS environment, do the following:
  1. Copy `$ORACLE_HOME/vos/build_proc_demos.cm` to `$ORACLE_HOME/precomp/demo/proc`.
  2. Copy the file `$ORACLE_HOME/vos/c_precomp_template.bind` to `$ORACLE_HOME/vos/c_precomp.bind`.
  3. Edit `$ORACLE_HOME/vos/c_precomp.bind` replacing the occurrences of `MASTER_DISK` and `ORACLE_HOME` according to the instructions in the file.
  4. Change the current directory to `ORACLE_HOME/precomp/demo/proc`.
  5. Ensure `ORACLE_HOME` and `TWO_TASK` are set correctly.
  6. Ensure `ORACLE_HOME/bin` is in the command library paths.
  7. Ensure the account `scott/tiger` exists and the demo tables are installed on the server.
  8. Run `build_proc_demos.cm`. This will install the SQL\*Plus scripts, pre-compile, compile, and bind all the available Pro\*C samples.

## Pro\*COBOL Precompilers

To build the Pro\*COBOL examples using the Oracle make files, do the following:

1. Copy the file `$ORACLE_HOME/vos/cobol_precomp_template.bind` to `$ORACLE_HOME/vos/cobol_precomp.bind`.
2. Edit `$ORACLE_HOME/vos/cobol_precomp.bind` replacing the occurrences of `MASTER_DISK` and `ORACLE_HOME` according to the instructions in the file.
  - For Pro\*COBOL 18:

```
bash
cd $ORACLE_HOME/precomp/demo/procob
gmake -f demon_procob18.mk samples
```
  - For Pro\*COBOL (Pro\*COBOL 2):

```
bash
cd $ORACLE_HOME/procomp/demo/procob2
gmake -f demo_procob.mk samples.
```

## Known Issue with Pro\*COBOL

The following is a known issue and its workaround:

### Issue

Datatype S9(N)binary does not work for small integers.

### Workaround:

Use either `COMP` or `integer`.

## Oracle Call Interface

Before you use the Oracle Call Interface (OCI), verify that the correct version of Pro\*C is installed.

To rebuild the OCI examples using the Oracle make files, use the following commands:

```
bash
cd $ORACLE_HOME/rdbms/demo
gmake -f demo_rdbms.mk demos
```

## OCI Demonstration Programs

Demonstration programs are provided that show the features of the OCI. These demonstration programs are based on C. All of the demonstration programs are located in the `$ORACLE_HOME/rdbms/demo` directory. All programs should be linked statically.

Many of the demonstration programs assume that the demonstration tables created by the `$ORACLE_HOME/sqlplus/demo/demobld.sql` script exist in the SCOTT schema with the password TIGER.

Use the `demo_rdbms.mk` make file, located in the `$ORACLE_HOME/rdbms/demo` directory, to create the demonstration programs. For example, to compile and link the `cdemo1` demonstration program, enter the following command:

```
$ gmake -f demo_rdbms.mk cdemo1
```

To create all of the C demonstration programs for OCI, enter:

```
$ gmake -f demo_rdbms.mk demos
```

## OCI User Programs

You can use the `$ORACLE_HOME/rdbms/demo/demo_rdbms.mk` make file to create user programs. The syntax for creating a user program with the `demo_rdbms.mk` make file is:

```
$ gmake -f demo_rdbms.mk target OBJS="objfile1 objfile2 ..." EXE=exename
```

In the preceding example:

- *target* is the make file target that you want to use
- *objfilen* is the object file to link the program
- *exename* is the executable program

For example, to create the `myprog` program from the C/C++ source `myprog.c`, enter the following command:

- For C source, statically linked, enter:
 

```
$ gmake -f demo_rdbms.mk build_static OBJS=myprog.o EXE=myprog
```

## Custom Make Files

Oracle recommends that you use the provided `demo_product.mk` make files to create user programs as described in the specific product sections of this chapter. If

you modify the provided make file, or if you choose to use a custom-written make file, the following restrictions apply:

- Do not modify the order of the Oracle libraries. Oracle libraries are included on the link line more than once so that all of the symbols are resolved during linking.

The order of the Oracle libraries is essential for the following reasons:

- Oracle libraries are mutually referential. Functions in library A call functions in library B, and functions in library B call functions in library A.
  - The Stratus VOS linkers are two-pass linkers.
- If you add your own library to the link line, add it to the beginning or to the end of the link line. Do not place user libraries between the Oracle libraries.
  - If you choose to use a make utility such as `nmake` or GNU `make`, be aware of how macro and suffix processing differs from the make utility provided with the platform. Oracle make files are tested and are supported with the `make` utility for your platform.
  - Oracle library names and the contents of Oracle libraries are subject to change between releases. Always use the `demo_product.mk` make file that ships with the current release as a guide to determine the required libraries.

## Using Signal Handlers

Oracle9i uses signals for two-task communication. Signals are installed in a user process when you connect to the database and are removed when you disconnect.

[Table 5–2](#) describes the signals that Oracle9i uses for two-task communication.

**Table 5–2 Signals for Two-Task Communication**

Signal	Description
SIGINT	Two-task drivers use SIGINT to detect user interrupt requests. The Oracle process does not catch SIGINT; the user process catches it.
SIGIO	Oracle Net protocols use SIGIO to indicate incoming networking events.
SIGURG	Oracle Net TCP/IP drivers use SIGURG to send out-of-band breaks from the user process to the Oracle process.

The listed signals affect all precompiler applications. You can have more than one signal handle for SIGINT. For SIGINT, use `osnsui()` and `osncui()` to register and delete signal-catching routines.

You can also install as many signal handlers as you want for other signals. If you are not connected to the Oracle process, you can have multiple signal handlers.

[Example 5–1](#) shows how to set up a signal routine and a catching routine.

### Example 5–1 Signal Routine and Catching Routine

```
/* user side interrupt set */
word osnsui( /*_ word *handlp, void (*astp), char * ctx, _*/)
/*
** osnsui: Operating System dependent Network Set User-side Interrupt. Add an
** interrupt handling procedure astp. Whenever a user interrupt (such as a ^C)
** occurs, call astp with argument ctx. Put in *handlp handle for this
** handler so that it may be cleared with osncui. Note that there may be many
** handlers; each should be cleared using osncui. An error code is returned if
```

```

** an error occurs.
*/

/* user side interrupt clear */
word osncui( /*_ word handle _*/ );
/*
** osncui: Operating System dependent Clear User-side Interrupt. Clear the
** specified handler. The argument is the handle obtained from osnsui. An error
** code is returned if an error occurs.
*/

```

**Example 5-2** shows how to use the `osnsui()` and the `osncui()` routines in an application program.

**Example 5-2 *osnsui() and osncui() Routine Template***

```

/*
** User interrupt handler template.
*/
void sig_handler()
{
...
}

main(argc, argv)
int argc;
char **argv;
{

    int handle, err;
    ...

    /* set up my user interrupt handler */

    if (err = osnsui(&handle, sig_handler, (char *) 0))
    {
        /* if the return value is non-zero, an error has occurred
        Take appropriate action for the error. */
        ...
    }

    ...

    /* clear my interrupt handler */

    if (err = osncui(handle))
    {
        /* if the return value is non-zero, an error has occurred
        Take appropriate action for the error. */
        ...
    }

    ...
}

```



---



---

## Configuring Oracle Net Services

This chapter describes how to configure Oracle Net Services on Stratus systems. It contains the following sections:

- [Location of Oracle Net Services Configuration Files](#)
- [Oracle Protocol Support](#)

**See Also:** *Oracle9i Net Services Administrator's Guide* for more information on Oracle networking.

### Location of Oracle Net Services Configuration Files

Oracle Net Services configuration files are typically, but not always, located in the `$ORACLE_HOME/network/admin` directory. Depending on the file, Oracle Net uses a different search order to locate the file.

The search order for the `sqlnet.ora` and `ldap.ora` files is as follows:

1. The directory specified by the `TNS_ADMIN` environment variable, if set
2. The `$ORACLE_HOME/network/admin` directory

The search order for the `cman.ora`, `listener.ora`, and `tnsnames.ora` files is as follows:

1. The directory specified by the `TNS_ADMIN` environment variable, if set
2. The `/etc` directory.
3. The `$ORACLE_HOME/network/admin` directory

For some system-level configuration files, users may have a corresponding user-level configuration file (stored in the user's home directory). The settings in the user-level file override the settings in the system-level file. The following table lists the system-level configuration files and the corresponding user-level configuration files:

System-Level Configuration File	User-Level Configuration File
<code>sqlnet.ora</code>	<code>\$HOME/.sqlnet.ora</code>
<code>tnsnames.ora</code>	<code>\$HOME/.tnsnames.ora</code>

### Sample Configuration Files

The `$ORACLE_HOME/network/admin/samples` directory contains samples of the `cman.ora`, `listener.ora`, `names.ora`, `sqlnet.ora`, and `tnsnames.ora` configuration files.

## Oracle Protocol Support

Oracle protocol support is a component of Oracle Net. It includes TCP/IP protocol support.

The TCP/IP protocol support has an address specification that is used in Oracle Net Services configuration files and in the DISPATCHER initialization parameter in the `initsid.ora` file. The following section describes the address specifications for each of the protocol supports.

**See Also:** *Oracle9i Net Services Administrator's Guide* for more information on Oracle protocol support.

### TCP/IP Protocol Support

TCP/IP is the standard communication protocol used for client/server communication over a network. The TCP/IP protocol support enables communication between client programs and the Oracle9i server, whether they are installed on the same or different systems. If the TCP/IP protocol is installed on your system, the TCP/IP protocol support is installed and linked to all client tools and to Oracle9i.

The TCP/IP protocol support requires an address specification in the following format:

```
(ADDRESS =  
  (PROTOCOL=TCP)  
  (HOST=hostname)  
  (PORT=port)  
)
```

The following table describes the parameters used in this address specification:

Parameter	Description
PROTOCOL	The protocol support to be used. The value is TCP. It is not case sensitive.
HOST	The host name or the host IP address.
PORT	The TCP/IP port. Specify the port as either a number or the alias name mapped to the port in the <code>/STCPservices</code> file. Oracle recommends a value of 1521.

The following shows a sample TCP/IP protocol address:

```
(ADDRESS=  
  (PROTOCOL=TCP)  
  (HOST=MADRID)  
  (PORT=1521)  
)
```

---

---

# Oracle9i Components

This appendix describes the products included with Oracle9i:

- [Oracle9i Client Components](#)
- [Product Descriptions](#)

## Oracle9i Client Components

The following sections provide information on the Oracle9i Client components.

### Runtime Edition

The following alphabetically lists the components available with each installation type of the Oracle9i Client Runtime top-level component. Refer to the release notes for your platform for updates on component availability.

- SQL\*Plus
- Oracle Net

### Administrator Edition

Oracle9i Client Administrator includes the components in the Oracle9i Client Runtime, and the following components. Refer to the release notes for your platform for updates on component availability.

- Oracle Call Interface
- Oracle Net, including Oracle Protocol Support
- Pro\*C/C++ 9.2.0.4.0
- Pro\*COBOL 1.8.77
- Pro\*COBOL 9.2.0.4.0

## Product Descriptions

The following table, [Table A-1](#), provides descriptions and release numbers for products available for installation. Some products described are automatically installed with other products. Refer to the release notes for your platform for updates on component availability.

**Table A-1 Product Descriptions**

<b>Product</b>	<b>Release</b>	<b>Description</b>	<b>For more information, see...</b>
<b>Oracle Call Interface</b>	9.2.0.4.0	Oracle Call Interface is an Application Programming Interface (API) for accessing an Oracle database from a C or C++ program.	<i>Oracle Call Interface Programmer's Guide</i>
<b>Oracle Net Services</b>	9.2.0.4.0	Oracle Net Services provides products that enable client connections to databases across a network. A client-side application sends a request to Oracle Net Services to be transported across the network to the server.  Oracle Net Services installs TCP/IP and Named Pipes.	<i>Oracle Net Administrator's Guide</i>
<b>Pro*C/C++</b>	9.2.0.4.0	Pro*C/C++ takes SQL statements embedded in C and C++ programs and converts them to standard C code. When you precompile this code, the result is a C or C++ program that you can compile and use to build applications that access an Oracle database.  <b>Note:</b> This product requires a separate license as a part of Oracle Programmer.	<i>Pro*C/C++ Precompiler Programmer's Guide</i>
<b>Pro*COBOL</b>	1.8.77 9.2.0.4.0	Pro*COBOL takes SQL statements embedded in a COBOL program and converts them to standard COBOL code. When you precompile this code, the result is a COBOL program you can compile and use to build applications that access an Oracle database.  <b>Note:</b> This product requires a separate license as a part of Oracle Programmer.	<i>Pro*COBOL Precompiler Programmer's Guide</i>
<b>SQL*Plus</b>	9.2.0.4.0	SQL*Plus is a command line interface that enables you to use SQL and PL/SQL database languages with an Oracle database.	<i>SQL*Plus User's Guide and Reference</i>
<b>TCP/IP Protocol Support</b>	9.2.0.4.0	TCP/IP Protocol Support enables client/server conversation over a network using TCP/IP and Oracle Net. This combination of Oracle products enables an Oracle application on a client to communicate with remote Oracle databases through TCP/IP (if the Oracle database is running on a host system that supports network communication using TCP/IP).	<i>Oracle Net Administrator's Guide</i>

---

---

# Index

## A

---

administering command line SQL, 4-1

## C

---

C  
Pro\*C/C++, 5-2  
catching routine, 5-6  
  example, 5-6  
configuration files  
  Oracle Net, 6-1  
  Oracle Net Services, 6-1  
  precompiler, 5-1

## D

---

debugger programs, 5-2  
demonstration programs  
  oracle call interface, 5-5  
  Pro\*C/C++, 5-3  
demonstration tables  
  creating manually, 4-2  
  deleting, 4-2  
directories  
  permissions, 3-1  
documentation  
  on administration and tuning, 0-x  
  on migrating and upgrading from previous  
  release, 0-x  
  related, 0-x

## E

---

environment variables  
  TNS\_ADMIN, 6-1

## F

---

files  
  login files, 3-2  
  permissions, 3-1  
  profile, 3-2

## G

---

glogin.sql file, 4-1

## I

---

initialization files, 3-2  
installation  
  Oracle Internet Directory, 0-x  
  Oracle Workflow, 0-x  
  post-installation tasks, 3-1  
  pre-installation tasks, 1-1  
installation types  
  with Oracle9i Client, A-1  
interrupting SQL\*Plus, 4-5  
ireclen, 5-2

## L

---

login files, 3-2  
login.sql file, 4-1

## M

---

multiple signal handlers, 5-6

## O

---

OCI. *See* Oracle Call Interface  
Oracle Call Interface  
  defined, A-2  
  installation types available with  
  oracle call interface, 5-5  
  demonstration programs, 5-5  
Oracle Net Client  
  installation types available with, A-1  
Oracle Net configuration files, 6-1  
Oracle Net Services  
  defined, A-2  
  protocol support, 6-2  
  protocols, 6-2  
Oracle Net Services configuration files, 6-1  
oracle precompiler and OCI linking and makefiles  
  custom makefiles, 5-5  
Oracle Precompilers, post-installation tasks, 3-3  
Oracle Protocol Support, A-1  
oracle user  
  post-installation tasks, 3-2  
Oracle9i Client top-level component, installation  
  types available with, A-1  
oreclen, 5-2

## P

---

- post-installation tasks
  - Oracle Precompilers, 3-3
- precompiler configuration files, 5-1
- precompilers
  - overview, 5-1
  - requirements, 1-2
  - signals, 5-6
  - uppercase to lowercase conversion, 5-2
  - value of ireclen and oreclen, 5-2
  - vendor debugger programs, 5-2
- Pro\*C/C++
  - demonstration programs, 5-3
  - described, A-2
  - installation types available with, A-1
  - make files, 5-3
  - signals, 5-6
  - using, 5-2
- Pro\*COBOL
  - described, A-2
  - installation types available with, A-1
  - requirements for, 1-3
- PRODUCT\_USER\_PROFILE Table, 4-2
- profile files, 3-2
- protocol support
  - TCP/IP, defined, A-2
- protocols, 6-2

## R

---

- related documentation, 0-x
- requirements
  - precompilers, 1-2
  - Pro\*COBOL, 1-3
  - tools, 1-2
- restrictions (SQL\*Plus), 4-5
  - resizing windows, 4-5
  - return codes, 4-5
- root user
  - post-installation tasks, 3-1
- running operating system commands, 4-4

## S

---

- setup files
  - SQL\*Plus, 4-1
- SIGINT two-task signal, 5-6
- SIGIO two-task signal, 5-6
- signal handlers
  - using, 5-6
- signal routine, 5-6
  - example, 5-6
- SIGURG two-task signal, 5-6
- SPOOL command
  - SQL\*Plus, 4-5
  - using, 4-5
- SQL\*Plus
  - command line help, 4-3
  - default editor, 4-4
  - described, A-2

- editor, 4-4
- installation types available with, A-1
- interrupting, 4-5
- PRODUCT\_USER\_PROFILE Table, 4-2
- restrictions, 4-5
- running operating system commands, 4-4
- setup files, 4-1
- site profile, 4-1
- SPOOL command, 4-5
- system editor, 4-4
- user profile, 4-1
- using command line SQL\*Plus, 4-4

- system editor
  - SQL\*Plus, 4-4

## T

---

- TCP/IP protocol support
  - defined, A-2
- tools
  - requirements, 1-2

## U

---

- user interrupt handler, 5-6
- user profile
  - SQL\*Plus, 4-1
- using command line SQL\*Plus, 4-4