

Oracle Procedural Gateway® for APPC

Installation and Configuration Guide

10g Release 2 (10.2) for UNIX

B16209-01

June 2005

Oracle Procedural Gateway for APPC Installation and Configuration Guide, 10g Release 2 (10.2) for UNIX
B16209-01

Copyright © 1994, 2005, Oracle. All rights reserved.

Primary Author: Maitreyee Chaliha

Contributing Author: Shahrokh Talmoud

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, JD Edwards, PeopleSoft, and Retek are registered trademarks 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

Send Us Your Comments	xi
Preface	xiii
Intended Audience.....	xiii
Documentation Accessibility	xiii
Typographic Conventions.....	xiv
Command Syntax	xiv
Related Documents	xv
Accessing Installed Documentation	xv
Oracle Services and Support.....	xvi
1 Introduction to Oracle Procedural Gateway for APPC	
1.1 Overview of the Gateway	1-1
1.2 Features of the Gateway.....	1-2
1.3 Terms	1-3
1.4 Architecture of the Gateway	1-5
1.5 Implementation of the Gateway	1-6
1.6 Communication With the Gateway.....	1-7
1.7 RPC Functions	1-7
1.7.1 Description of RPC Functions.....	1-8
1.7.1.1 Remote Transaction Initiation	1-8
1.7.1.2 Data Exchange	1-8
1.7.1.3 Remote Transaction Termination.....	1-8
1.8 Transaction Types for Gateway Using SNA.....	1-8
1.9 Transaction Types for Gateway Using TCP/IP.....	1-9
2 Release Information	
2.1 Product Set	2-1
2.2 Changes and Enhancements, Release 10.2.0.1.0	2-1
2.2.1 Enhancements	2-1
2.2.2 Changes.....	2-1
2.2.2.1 Bug 3557168: Init.ora Not Found.....	2-1
2.2.2.2 PG4PWD Replacement Tool Place Holder	2-1
2.2.2.3 Bugs Fixed in Release 10.2.0.....	2-1
2.3 Known Restrictions.....	2-2

2.3.1	Known Restrictions for the Gateway	2-2
2.3.2	Known Restrictions for PGAU	2-2

3 System Requirements

3.1	Hardware Requirements.....	3-1
3.1.1	Processor Requirements.....	3-1
3.1.2	Memory Requirements	3-2
3.1.3	Network Attachment Requirements.....	3-2
3.1.4	CD-ROM Drive	3-2
3.1.4.1	HP-UX Only	3-2
3.1.5	Disk Space Requirements	3-2
3.2	Software Requirements	3-2
3.2.1	Operating System Requirements.....	3-3
3.2.2	Communication Protocol Requirements	3-3
3.2.3	Oracle Server Requirements.....	3-3
3.2.4	Oracle Networking Product Requirements	3-4
3.2.5	IBM Mainframe Requirements	3-4
3.3	Documentation Requirements	3-4

4 Installing the Gateway

4.1	Before You Begin.....	4-1
4.2	Planning to Upgrade or Migrate the Gateway	4-2
4.2.1	Preupgrade Procedures	4-2
4.2.2	Upgrade and Migration Considerations	4-2
4.2.3	Restoration.....	4-3
4.3	Preinstallation Steps for Upgrade.....	4-3
4.3.1	Gateway Installation Methods.....	4-4
4.4	Installing the Gateway Software	4-4
4.5	Installation Steps	4-4
4.5.1	Step 1: Login as DBA and Create Login User ID	4-5
4.5.2	Step 2: Create the Product Installation Directory.....	4-5
4.5.3	Step 3: Setting the Environment Variables.....	4-5
4.5.4	Step 5: Mount the CD-ROM.....	4-6
4.5.4.1	Mounting the CD-ROM for Linux	4-6
4.5.4.2	Mounting the CD-ROM for AIX-Based Systems	4-6
4.5.4.3	Mounting the CD-ROM for HP-UX.....	4-6
4.5.4.4	Mounting the CD-ROM for Solaris Operating System (SPARC 64-bit)	4-7
4.5.5	Step 6: Start the Oracle Universal Installer	4-7
4.5.5.1	For Linux.....	4-7
4.5.5.2	For AIX Systems	4-8
4.5.5.3	For HP-UX	4-8
4.5.5.4	For Solaris	4-8
4.5.6	Step 7: Step through the Oracle Universal Installer.....	4-8
4.5.6.1	Oracle Universal Installer on UNIX platforms.....	4-9
4.6	Deinstalling Oracle Procedural Gateway for APPC	4-10

5	Configuring Your Oracle Network	
5.1	Configuring the Oracle Network for the Gateway	5-1
6	Configuring the SNA Communication Package on Linux	
6.1	Using SNA Security Validation	6-1
6.2	Processing Inbound Connections	6-2
6.3	Independent Versus Dependent LUs.....	6-2
6.4	Definition Types.....	6-2
6.5	Creating IBM Communications Server Definitions for the Gateway.....	6-3
6.5.1	Creating the Configuration	6-3
6.5.2	Creating the Node	6-3
6.5.3	Creating Devices	6-4
6.5.4	Choosing the Device Type.....	6-4
6.5.5	Configuring a LAN Device	6-4
6.5.6	Creating Peer Connections	6-4
6.5.7	Defining the Link Station.....	6-4
6.5.8	Defining the Adjacent Node	6-4
6.5.9	Creating Local LUs	6-5
6.5.10	Defining Local LUs.....	6-5
6.5.11	Creating Partner LUs.....	6-5
6.5.12	Defining Partner LUs	6-5
6.5.13	Creating the CPI-C Side Information Profile	6-5
6.6	Testing the Connection	6-5
6.7	Resume Configuration of the Gateway	6-6
7	Configuring the SNA Communication Package on AIX-Based Systems	
7.1	Processing Inbound Connections	7-1
7.2	Independent Versus Dependent LUs.....	7-2
7.3	Creating SNA Profiles for the Gateway.....	7-2
7.4	Profile Types	7-3
7.5	SNA Server Profiles	7-3
7.5.1	SNA Node Profile	7-3
7.5.2	Link Station Profile	7-3
7.5.3	Mode Profile	7-4
7.5.4	Local LU Profile	7-4
7.5.5	Partner LU Profile.....	7-4
7.5.6	Partner LU Location Profile	7-5
7.5.7	Side Information Profile.....	7-5
7.6	Activating Profiles	7-6
7.7	Resume Configuration of the Gateway	7-6
8	Configuring the SNA Communication Package on HP-UX	
8.1	Processing Inbound Connections	8-1
8.1.1	Steps for Configuring the Communications Interfaces.....	8-2
8.2	SNAPLUS2 Configuration Tool	8-2

8.3	Creating SNAPlus2 Profiles for the Gateway	8-2
8.4	Independent Versus Dependent LUs.....	8-2
8.4.1	Sym Dest Name.....	8-3
8.5	Creating SNA Definitions for the Gateway	8-3
8.6	Sample SNAPlus2 Definitions	8-3
8.7	Configuring SNAPlus2	8-3
8.8	Testing the Connection	8-6
8.9	Resuming Gateway Configuration.....	8-6

9 Configuring the SNA Communication Package on Solaris

9.1	Processing Inbound Connections	9-1
9.2	Configuring SNAP-IX Version 6.....	9-1
9.2.1	Before You Begin.....	9-2
9.2.2	SNAP-IX Configuration Tool.....	9-2
9.2.3	Creating SNAP-IX Profiles for the Gateway.....	9-2
9.2.4	Independent Versus Dependent LUs.....	9-2
9.2.5	Creating SNA Definitions for the Gateway	9-3
9.2.6	Sample SNAP-IX Definitions	9-3
9.2.7	Configuring SNAP-IX	9-3
9.2.8	Starting xsnaadmin.....	9-3
9.2.8.1	Configuring the SNA node	9-3
9.2.8.2	Adding a Port.....	9-3
9.2.8.3	Create a Link Station.....	9-4
9.2.8.4	Creating Local LUs.....	9-4
9.2.8.5	Creating Partner LUs	9-4
9.2.8.6	Creating Mode and CPI-C Profiles	9-5
9.2.9	Testing the Connection	9-5
9.3	Resuming Gateway Configuration.....	9-6

10 Configuring the TCP/IP Communications Package for IMS Connect

10.1	Configuring TCP/IP for IMS Connect on UNIX Platforms.....	10-1
------	---	------

11 Configuring the OLTP

11.1	Configuring the OLTP for an SNA Environment	11-1
11.1.1	Configuring CICS Transaction Server for z/OS	11-1
11.1.2	Configuring IDMS-DC.....	11-2
11.1.3	Configuring IMS/TM.....	11-3
11.1.4	Configuring APPC/MVS.....	11-4
11.2	Configuring the OLTP for a TCP/IP Environment	11-5

12 Gateway Configuration Using SNA Communication Protocol

12.1	Before You Begin.....	12-1
12.2	Preparing to Configure a Gateway Installation/Upgrade	12-2
12.3	Integrating Server Configuration: First-Time Gateway Installations	12-3
12.4	Upgrading or Migrating the Oracle Server from Previous Gateways	12-6
12.4.1	If You Must Reinstall Package Specifications.....	12-6

12.4.2	Upgrading PGAU From Previous Gateway Releases	12-7
12.5	Configuring the Oracle Integrating Server for Gateways to Coexist	12-7
12.6	Optional Configuration Steps to Permit Multiple Users.....	12-7
12.7	Configuring the Gateway	12-10
12.8	Configuring Commit-Confirm.....	12-11
12.8.1	Configuring the Oracle Integrating Server for Commit-Confirm	12-11
12.8.2	Configuring Gateway Initialization Parameters for Commit-Confirm	12-12
12.8.3	Configuring the OLTP for Commit-Confirm.....	12-12
12.9	Verifying the Gateway Installation and OLTP Configuration	12-12
12.9.1	Verifying the Gateway Installation	12-13
12.9.2	Verifying the OLTP Configuration	12-13
12.9.2.1	CICS Verification	12-14
12.9.2.2	IDMS-DC Verification.....	12-14
12.9.2.3	IMS/TM Verification	12-15
12.9.2.4	APPC/MVS Verification	12-16
12.9.3	Verifying OLTP Configuration for Commit-Confirm	12-16
12.10	Performing Postinstallation Procedures	12-17
12.10.1	Installing Sample Applications.....	12-17

13 Gateway Configuration Using TCP/IP Communication Protocol

13.1	Before You Begin.....	13-1
13.2	Preparing to Configure a Gateway Installation/Upgrade	13-2
13.3	Configuring Oracle Integrating Server: First-Time Installation.....	13-3
13.4	Upgrading or Migrating the Oracle Server from Previous Gateways	13-7
13.4.1	If You Must Reinstall Package Specifications.....	13-7
13.4.2	Upgrading PGAU from Previous Gateway Releases	13-8
13.5	Optional Configuration Steps to Permit Multiple Users.....	13-8
13.6	Configuring TCP/IP for the Gateway	13-11
13.7	Configuring the Gateway	13-11
13.8	Loading the PGA_TCP_IMSC Table	13-12
13.9	Verifying the Gateway Installation and OLTP Configuration	13-12
13.9.1	Verifying the Gateway Installation	13-12
13.9.2	Verifying the OLTP Configuration	13-13
13.9.2.1	IMS/TM Verification	13-13
13.10	Performing Postinstallation Procedures	13-14
13.10.1	Installing Sample Applications.....	13-14

14 Security Requirements

14.1	Overview of Security Requirements	14-1
14.2	Authenticating Application Logons.....	14-2
14.3	Defining and Controlling Database Links.....	14-2
14.3.1	Link Accessibility	14-2
14.3.2	Links and CONNECT Clauses.....	14-3
14.4	Using SNA Security Validation	14-3
14.4.1	Specifying SNA Conversation Security	14-3
14.4.1.1	SNA Security Option SECURITY=NONE	14-3

14.4.1.2	SNA Security Option SECURITY=PROGRAM.....	14-4
14.4.1.3	SNA Security Option SECURITY=SAME.....	14-4
14.5	TCP/IP Security	14-4
14.5.1	Specifying TCP/IP Conversation Security	14-5
14.5.1.1	TCP/IP Security Option SECURITY=NONE.....	14-5
14.5.1.2	TPC/IP Security Option SECURITY=PROGRAM	14-5
14.6	Passwords in the Gateway Initialization File	14-6

15 Migration From Existing Gateways

15.1	Migrating An Existing Gateway Instance to a New Release Using SNA Protocol	15-1
15.1.1	Step 1: Install the New Release	15-1
15.1.2	Step 2: Transferring the initsid.ora Gateway Initialization File Parameters.....	15-1
15.1.3	Backout Considerations When Migrating to New Releases.....	15-2
15.1.4	Oracle Net Considerations	15-2
15.1.5	Parameter Changes: Version 4 to Release 10.2.0 of the Gateway	15-2
15.1.6	Parameter Changes: Version 8 or Earlier to Gateway Release 10.2.0	15-4
15.1.7	Migrating from Gateway Release 9.0.1 or 9.2.0 or 10.1.0 to Gateway Release 10.2.0.....	15-4
15.2	Migrating from an Existing Gateway to TCP/IP Using SNA.....	15-4
15.2.1	Using Existing TIPs with Existing Side Profile Definitions.....	15-4
15.2.2	Using Existing TIPs with LU/TP/LOGMODE Names	15-5

A Gateway Initialization Parameters for SNA Protocol

A.1	Gateway Initialization Parameter File Using SNA Protocol	A-1
A.1.1	PGA Parameters.....	A-1
A.1.2	PGA_CAPABILITY Parameter Considerations	A-4
A.1.3	PGA_CONFIRM Parameter Considerations	A-5
A.1.4	Sample listener.ora file for a Gateway Using SNA.....	A-6
A.1.5	Sample tnsnames.ora file for a Gateway Using SNA	A-6

B Gateway Initialization Parameters for TCP/IP Communication Protocol

B.1	Gateway Initialization Parameter File Using TCP/IP.....	B-1
B.1.1	PGA Parameters.....	B-1
B.2	Output for the pg4tcpmap Tool.....	B-3
B.2.1	Sample listener.ora File for a Gateway Using TCP/IP.....	B-4
B.2.2	Sample tnsnames.ora File for a Gateway Using TCP/IP.....	B-5

C Gateway Terminology

C.1	Gateway Term Definitions	C-1
-----	--------------------------------	-----

D Summary of Changes in Previous Versions

D.1	Changes and Enhancements in Previous Versions.....	D-1
D.1.1	Release 10.2.0.1.0.....	D-1
D.1.1.1	pg4hoa1.sh Bootstrap Shell Script No Longer Used When Migrating	D-1
D.1.1.2	SunLink Peer-to-Peer No Longer Supported on Solaris.....	D-1

D.1.1.3	AIX: IBM Comm. Server on AIX 5L Does Not Support SNA 32-bit Libraries ...	D-1
D.1.1.4	TCP/IP Support for Connecting PG4APPC to IMS Connect Release 1.2	D-2
D.1.1.5	Improved Gateway Log File	D-2
D.1.2	Release 9.2.0.1.0	D-2
D.1.3	Release 9.0.1.0.1	D-2
D.1.4	Release 8.0.6.1.0 and 8.0.4.1.0	D-3
D.1.5	Release 4.0.1.1.0	D-3
D.2	Corrected Problems in Previous Versions.....	D-4
D.2.1	Bugs Fixed in Release 10.2.0.1.0.....	D-4
D.2.2	Release 9.2.0.1.0	D-5
D.2.3	Release 9.0.1.0.1	D-6
D.2.4	Release 8.0.6.1.0 and 8.0.4.1.0	D-6
D.2.5	Release 4.0.1.1.0	D-7
D.3	Known Restrictions in Previous Releases	D-7
D.3.1	Release 9.2.0.1.0	D-8
D.3.2	Release 8.0.6.1.0 and 4.0.1.1.0	D-8

E Configuration Worksheet

Index

Send Us Your Comments

Oracle Procedural Gateway for APPC Installation and Configuration Guide, 10g Release 2 (10.2) for UNIX

B16209-01

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:

- Electronic mail: infodev_us@oracle.com
- FAX:(650) 506-7227. Attn: Server Technologies Documentation Manager
- Postal service:

Oracle Corporation
Server Technologies Documentation Manager
500 Oracle Parkway, Mailstop 4op11
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

The Oracle Procedural Gateway for APPC provides Oracle applications with seamless access to IBM mainframe data and services through Remote Procedure Call (RPC) processing.

The UNIX platforms supported by this gateway release are:

- Linux 32-bit Intel Pentium-based processors
- Solaris Operating System (SPARC 64-bit)
- AIX pSeries (64-bit), and
- HP-UX PA-RISC (64-bit)

Refer to the *Oracle Database Installation Guide 10g Release 2 (10.2) for UNIX Systems* and to the certification matrix on the *OracleMetaLink* Web site for the most up-to-date list of certified hardware platforms and operating system versions. The *OracleMetaLink* Web site is available at

<http://metalink.oracle.com>

Intended Audience

Read this guide if you are responsible for tasks such as:

- Determining hardware and software requirements.
- Installing, configuring, or administering an Oracle Procedural Gateway for APPC.
- Developing applications that access remote host databases through the gateway using either the SNA communication protocol or the TCP/IP for IMS Connect communication protocol.
- Determining security requirements.
- Determining and resolving problems.

Before using this guide to administer the gateway, you should understand the fundamentals of the operating system for your platform and Procedural Gateways.

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. Accessibility 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 more information, visit the Oracle Accessibility Program Web site at

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

Accessibility of Code Examples in Documentation

Screen readers 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, some screen readers 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.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Typographic Conventions

The following typographic conventions are used in this guide:

Convention	Description
<code>monospace</code>	Monospace type indicates UNIX commands, directory names, user names, path names, and file names.
<i>italics</i>	Italic type indicates variables, including variable portions of file names. 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.
Bold	Bold type indicates screen names and fields.
SQL*Plus prompts	The SQL*Plus prompt, <code>SQL></code> , appears in SQL statement and SQL*Plus command examples. Enter your response at the prompt. Do not enter the text of the prompt, "SQL>", in your response.

Command Syntax

UNIX command syntax appears in `monospace` font. The dollar character (\$), number sign (#), or percent character (%) are UNIX command prompts. Do not enter them as part of the command. The following command syntax conventions are used in this guide:

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

Related Documents

The *Oracle Procedural Gateway for APPC Installation and Configuration Guide for UNIX* is included as part of your product shipment. Also included is:

- *Oracle Procedural Gateway for APPC Messages Guide for UNIX and Microsoft Windows*
- *Oracle Procedural Gateway for APPC User's Guide for UNIX*

You might also need Oracle server and Oracle Net documentation. The following is a useful list of the Oracle publications that might be referenced in this guide:

- *Oracle Database Installation Guide 10g Release 2 (10.2) for UNIX Systems*
- *Oracle Database Administrator's Guide 10g Release 2 (10.2) for UNIX Systems*
- *Oracle Database Application Developer's Guide*
- *Oracle Database Concepts*
- *Oracle Database Server Distributed Systems*
- *Oracle Database Error Messages*
- *Oracle Net Services Administrator's Guide*

Other related publications are listed in "[Documentation Requirements](#)" on page 3-4. Refer to the *Oracle Technical Publications Catalog and Price Guide* for a complete list of documentation provided for Oracle products.

Accessing Installed Documentation

The following sections provide information about accessing installed documentation.

Documentation for Oracle Procedural Gateway for APPC for UNIX Platforms

Documentation for this product includes this guide and the *Oracle Procedural Gateway for APPC User's Guide 10g Release 2 (10.2)* for UNIX and the *Oracle Procedural Gateway for APPC Messages Guide 10g Release 2 (10.2)*.

To access the documentation in HTML and PDF formats, use a browser to open the top level of the Gateway Documentation CD-ROM. This level contains links to product and UNIX-specific documentation.

Oracle Product Documentation

Oracle Database product documentation is on the Oracle Database Generic Documentation CD-ROM. Instructions for accessing and installing the documents on the CD-ROM are found in the README file on the top level directory of the CD-ROM.

Oracle Services and Support

The Oracle corporate web page is available at

<http://www.oracle.com>

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

Oracle Support Services

Technical Support registration and contact information worldwide is available at the following address

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

At the Oracle support site, you will find templates to help you prepare information about your problem before you call so that you can be helped more quickly. You also need your CSI number (if applicable) or complete contact details, including any special project information.

Oracle Technology Network

Printed documentation is available for sale in the Oracle Store at

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

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at

<http://www.oracle.com/technology/membership/index.html>

If you already have a username and password for OTN, then you can go directly to the documentation section of the OTN Web site at

<http://www.oracle.com/technology/documentation/index.html>

OracleMetaLink

OracleMetaLink is the Oracle Web service for 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.

Log on to *OracleMetaLink* before installing or administrating your product to search for up-to-date information about Oracle Database for UNIX.

In addition, *OracleMetaLink* offers forums for information sharing between 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 the following site

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

Use your Support Access Code (SAC) number to register.

Oracle Products and Other Documentation

For customers in the US, the Oracle Store is at

<http://store.oracle.com>

Links to Stores in other countries are provided from this Web site.

Customer Service

Oracle Support Services contacts are listed at

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

Support for Hearing and Speech Impaired Customers

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week.

- For technical questions, call:
1.800.446.2398
- For non-technical questions, call:
1.800.464.2330

Education and Training

Training information and worldwide schedules are available from

<http://education.oracle.com>

Introduction to Oracle Procedural Gateway for APPC

The Oracle Procedural Gateway for APPC (the gateway) enables users to initiate transaction program execution on remote online transaction processors (OLTPs). The Oracle Procedural Gateway for APPC can establish a connection with OLTP using the SNA communication protocol. The gateway can also use TCP/IP for IMS Connect to establish communication with the OLTP through TCP/IP. The gateway provides Oracle applications with seamless access to IBM mainframe data and services through remote Procedural Call (RPC) processing. The gateway can access any application capable of using the CPI-C API either directly or through a TP monitor such as CICS.

This chapter discusses the architecture, uses and features of the gateway. It contains the following sections:

- [Section 1.1, "Overview of the Gateway"](#)
- [Section 1.2, "Features of the Gateway"](#)
- [Section 1.3, "Terms"](#)
- [Section 1.4, "Architecture of the Gateway"](#)
- [Section 1.5, "Implementation of the Gateway"](#)
- [Section 1.6, "Communication With the Gateway"](#)
- [Section 1.7, "RPC Functions"](#)
- [Section 1.8, "Transaction Types for Gateway Using SNA"](#)
- [Section 1.9, "Transaction Types for Gateway Using TCP/IP"](#)

1.1 Overview of the Gateway

The Oracle Procedural Gateway for APPC extends the RPC facilities available with the Oracle server. The gateway enables any client application to use PL/SQL to request execution of a remote transaction program (RTP) residing on a host. The gateway provides RPC processing to systems using the SNA APPC (Advanced Program-to-Program Communication) protocol and to systems using TCP/IP for IMS Connect protocol. This architecture allows efficient access to data and transactions available on the IBM mainframe and IMS, respectively.

The gateway requires no Oracle software on the remote host system. As a result, the gateway uses existing transactions with little or no programming effort on the remote host.

Note: For gateways using SNA only the use of a generic and standard protocol, APPC, enables the gateway to access a multitude of systems. The gateway can communicate with virtually any APPC-enabled system, including IBM Corporation CICS on any platform, IBM Corporation IMS and APPC/MVS, and Computer Associates IDMS. These transaction monitors provide access to a broad range of systems, allowing the gateway to access many datastores, including VSAM, DB2 (static SQL), IMS, IDMS, ADABAS and others.

The gateway can access any application capable of using the CPI-C API either directly or through a TP monitor such as CICS.

1.2 Features of the Gateway

The Oracle Procedural Gateway for APPC provides the following benefits:

- TCP/IP support for IMS Connect

This release of the gateway includes TCP/IP support for IMS Connect, providing users a choice between the SNA or TCP/IP communication protocol. IMS Connect is an IBM product which enables TCP/IP clients to trigger execution of IMS transactions. The gateway can use a TCP/IP communication protocol to access IMS Connect, which triggers execution of IMS transactions. If you choose to use TCP/IP, then there is no SNA involvement with this configuration.

Related to this new feature of the gateway is the `pg4tcpmap` tool. This release of the gateway includes a tool whose purpose is to map the information from your Side Profile Name to TCP/IP and IMS Connect. For more information about the gateway mapping tool, refer to Chapter 6 of the *Oracle Procedural Gateway for APPC User's Guide*, and to [Chapter 13, "Gateway Configuration Using TCP/IP Communication Protocol"](#) in this guide.

- Fast interface

The gateway is optimized for minimum network traffic when you execute programs remotely. The interface to the gateway is an optimized PL/SQL stored procedure specification called the transaction interface package (TIP). This specification is precompiled in the Oracle Integrating Server. Because there are no additional software layers on the remote host system, overhead occurs only when your program executes.

- Platform independence

Client applications need not be operating system-specific. For example, your application can call a program on a CICS Transaction Server for z/OS. If you move the program to a CICS region on pSeries, then you need not change the application.

- Application transparency

Users calling applications that execute a remote transaction program are unaware that a request is sent to a host.

- Flexible interface

You can use the gateway as an interface with an existing procedural logic or to integrate new procedural logic into an Oracle Integrating Server environment.

- Oracle server integration

Integration of Oracle server with the gateway enables you to benefit from existing and future Oracle features. For example, the gateway can be called from an Oracle stored procedure or database trigger.
- Transactional support

The gateway and the Oracle Integrating Server allow remote transaction updates and Oracle server updates to be performed in a coordinated fashion.
- Wide selection of tools

The gateway supports any tool or application that supports PL/SQL.
- PL/SQL code generator

The Oracle Procedural Gateway for APPC provides a powerful development environment, including:

 - A data dictionary to store information relevant to the remote transaction
 - A tool to generate the PL/SQL Transaction Interface Package, or TIP
 - A report utility to view the information stored in the gateway dictionary
 - A complete set of tracing and debugging facilities
 - A wide set of samples to demonstrate the use of the product against datastores such as DB2, IMS, IDMS, CICS, and ADABAS
- Site autonomy and security

The gateway provides site autonomy, allowing you to perform tasks such as authenticate users. It also provides role-based security compatible with any security package running on your mainframe computer.
- Automatic conversion

Through TIP, the following conversions are performed:

 - ASCII to and from EBCDIC
 - Remote transaction program data types to and from PL/SQL data types
 - National language support for many languages

1.3 Terms

The following terms and definitions are used throughout this guide. Refer to [Appendix C, "Gateway Terminology"](#) for the complete list of terms and definitions pertaining to the gateway, its components, and functions.

Oracle Integrating Server

This is any Oracle server instance that communicates with the gateway for purposes of performing remote procedural calls to execute RTP. The Oracle Integrating Server can be on the same system as the gateway or on a different system. If it is on a different system, then Oracle Net is required on both systems. Refer to [Figure 1–2, "Gateway Architecture"](#) for a view of the gateway architecture.

OLTP (Online Transaction Processor)

OLTP is an online transaction processor available from other vendors, including CICS Transaction Server for z/OS, IMS/TM, z/OS and IDMS-DC.

PGAU (Procedural Gateway Administration Utility)

PGAU is the tool that is used to define and generate PL/SQL transaction interface packages (TIPs). Refer to Chapter 2, Procedural Gateway Administration Utility in the *Oracle Procedural Gateway for APPC User's Guide* for more information about PGAU.

PG DD (Data Dictionary)

This gateway component is a repository of remote host transaction definitions and data definitions. PGAU accesses definitions in the PG DD when generating TIPs. The PG DD has data type dependencies because it supports the PGAU and is not intended to be directly accessed by the customer. Refer to Appendix A, Procedural Gateway for APPC Data Dictionary in the *Oracle Procedural Gateway for APPC User's Guide* for a list of PG DD tables.

RPC

RPC is a programming call that executes program logic on one system in response to a request from another system. Refer to "[Gateway Term Definitions](#)" on page C-1 for more information, and refer to Appendix C, Gateway RPC Interface in the *Oracle Procedural Gateway for APPC User's Guide* as well.

RTP

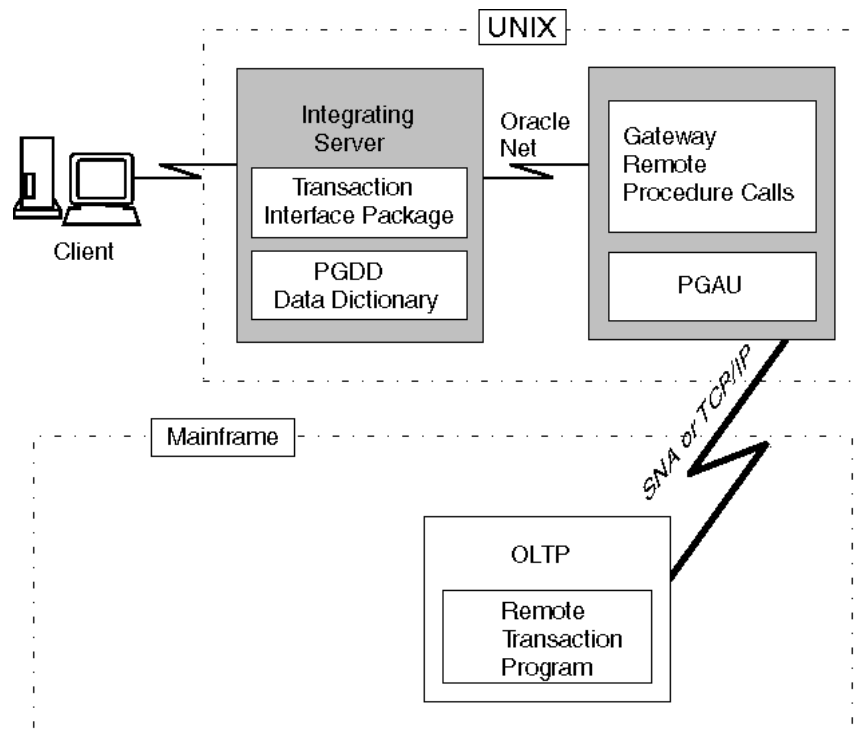
A remote transaction program is a customer-written transaction, running under the control of an OLTP, which the user invokes remotely using a PL/SQL procedure. To execute a remote transaction program through the gateway, you must use RPC to execute a PL/SQL program to call the gateway functions.

TIP (Transaction Interface Package)

A TIP is an Oracle PL/SQL package that exists between your application and the remote transaction program. The transaction interface package, or TIP, is a set of PL/SQL stored procedures that invoke the remote transaction program through the gateway. TIPs perform the conversion and reformatting of remote host data using PL/SQL and UTL_RAW/UTL_PG functions.

[Figure 1-1](#) illustrates the Relationship of Gateway and Oracle Integrating server . The terminology discussed in the preceding sections has been used in the architecture of the gateway.

Figure 1-1 Relationship of Gateway and Oracle Integrating Server on UNIX Host



1.4 Architecture of the Gateway

The architecture of Oracle Procedural Gateway for APPC consists of several components:

- Oracle Integrating Server

The Oracle Integrating Server is usually installed on the same system as Oracle Procedural Gateway for APPC.

If you install the Oracle Integrating Server on a system other than the system on which the gateway is installed, then you must install Oracle Net with the Oracle Integrating Server and with the gateway. The Oracle Integrating Server must be capable of connecting to the gateway through any supported Oracle Net protocol.

Refer to "[Product Set](#)" for a list of Oracle Net protocols currently supported by the gateway and tools.

The Oracle Integrating Server can also be used for non-gateway applications.

- The gateway

Oracle Procedural Gateway for APPC must be installed on a server that can run the required version of the operating system.

- An OLTP (online transaction processor)

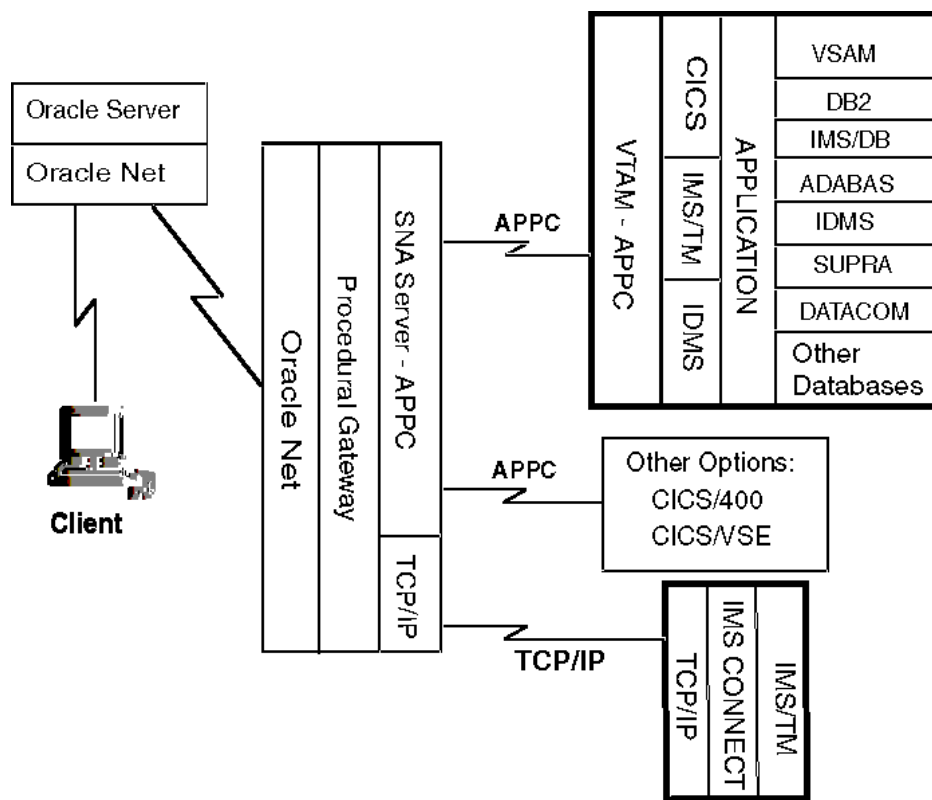
The OLTP must be accessible from the gateway using SNA or TCP/IP communication protocol. Multiple Oracle Integrating Servers can access the same gateway. A single system gateway installation can be configured to access more than one OLTP.

Note: For a gateway using TCP/IP support for IMS Connect the only OLTP that is supported through TCP/IP is IMS through IMS Connect

The OLTP must be accessible to the system using the TCP/IP protocol. Multiple Oracle Integrating Servers can access the same gateway. A single system gateway installation can be configured to access more than one OLTP. Multiple IMS can be accessed from an IMS Connect. If you have a number of IMS Connect systems available, any of these may be connected to one or more IMS systems.

Figure 1-2 illustrates the architecture of Oracle Procedural Gateway for APPC using either SNA or TCP/IP, as described in the preceding section.

Figure 1-2 Gateway Architecture



1.5 Implementation of the Gateway

The basic structure of the gateway is the same whether your communications protocol is SNA or TCP/IP support for IMS Connect. The gateway has some of the same components as an Oracle database instance on UNIX. It has the following components:

- A home directory, similar to the one associated with an Oracle instance ORACLE_HOME environment variable
- A system identifier, identified as *sid* or ORACLE_SID
- An initialization parameter file, similar to the Oracle Integrating Server *initsid.ora* file.

The gateway does not have:

- Control, redo log, or database files
- The full set of subdirectories and ancillary files associated with an installed Oracle server

Because the gateway has no background processes and does not need a management utility such as Oracle Enterprise Manager, you do not need to start the gateway. Each Oracle Integrating Server user session that accesses a particular gateway creates an independent process on UNIX which in turn runs the gateway server and executes either the SNA or TCP/IP functions to communicate with an OLTP.

1.6 Communication With the Gateway

All communication between the user or client program and the gateway is handled through a transaction interface package (TIP), which executes on an Oracle Integrating Server. The TIP is a standard PL/SQL package that provides the following functions:

- Declares the PL/SQL variables that can be exchanged with a remote transaction program
- Calls the gateway packages that handle the communications for starting the conversation, exchanging data and terminating the conversation
- Handles all data type conversions between PL/SQL data types and the target program data types

The Procedural Gateway Administration Utility (PGAU), provided with the gateway automatically generates the TIP specification.

The gateway is identified to the Oracle Integrating Server using a database link. The database link is the same construct used to identify other Oracle server databases. The functions in the gateway are referenced in PL/SQL as:

```
function_name@dblink_name
```

1.7 RPC Functions

The Oracle Procedural Gateway for APPC provides a set of functions that are invoked by the client through RPC. These functions direct the gateway to initiate, transfer data, and terminate remote transaction programs running under an OLTP on another system.

[Table 1–1](#) lists the remote procedural call functions and the correlating commands that are invoked in the gateway and remote host.

Table 1–1 RPC Functions and Commands in the Gateway and Remote Host

Applications	Oracle TIP	Gateway	Remote Host
call tip_init	tip_init call pgainit@gateway	PGAINIT	Initiate program
call tip_main	tip_main call pgaxfer@gateway	PGAXFER	Exchange data
call tip_term	tip_term call pgaterm@gateway	PGATERM	Terminate program

1.7.1 Description of RPC Functions

The following sections describe how the RPC functions perform on gateways using SNA or TCP/IP communication protocols.

1.7.1.1 Remote Transaction Initiation

The TIP initiates a connection to the remote host system, using one of the gateway functions, `PGAINIT`.

When the communication protocol is SNA: `PGAINIT` provides, as input, the required SNA parameters to start a conversation with the target transaction program. These parameters are sent across the SNA network, which returns a conversation identifier to `PGAINIT`. Any future calls to the target program use the conversation identifier as an `INPUT` parameter.

When the communication protocol is TCP/IP: `PGAINIT` provides, as input, the required TCP/IP parameters. Use the `pg4tcpmap` tool to map the parameters. These parameters are sent across the TCP/IP network to start the conversation with the target transaction program. The TCP/IP network returns a socket file descriptor to `PGAINIT`. Any future calls to the target program made by `PGAXFER` and `PGATERM` use the socket file descriptor as an input parameter.

Refer to [Appendix B, "Gateway Initialization Parameters for TCP/IP Communication Protocol"](#), and Chapter 6 in the *Oracle Procedural Gateway for APPC User's Guide*, for more information about the function and use of the `pg4tcpmap` tool.

1.7.1.2 Data Exchange

After the conversation is established, a procedural gateway function called `PGAXFER` can exchange data in the form of input and output variables. `PGAXFER` sends and receives buffers to and from the target transaction program. The gateway sees a buffer as only a `RAW` stream of bytes. The TIP that is residing in the Oracle Integrating Server is responsible for converting the application PL/SQL data types to `RAW` before sending the buffer to the gateway. It is also responsible for converting `RAW` to the PL/SQL data types before returning the results to the application.

1.7.1.3 Remote Transaction Termination

When communication with the remote program is complete, the gateway function `PGATERM` terminates the conversation between the gateway and the remote host.

When the communication protocol is SNA, `PGATERM` uses the conversation identifier as an `INPUT` parameter to request conversation termination.

When the communication protocol is TCP/IP, `PGATERM` uses the socket file descriptor for TCP/IP as an `INPUT` parameter to request conversation termination.

1.8 Transaction Types for Gateway Using SNA

The Oracle Procedural Gateway for APPC supports three types of transactions that read data from and write data to remote host systems:

- One-shot
 - In a one-shot transaction, the application initializes the connection, exchanges data and terminates the connection, all in a single call.
- Persistent

In a persistent transaction, multiple calls to exchange data with the remote transaction can be made before terminating the conversation.

- Multiconversational

In a multiconversation transaction, the procedural gateway server can be used to exchange multiple records in one call to the remote transaction program.

Refer to Remote Host Transaction Types in Chapter 4, Client Application Development of the *Oracle Procedural Gateway for APPC User's Guide* for more information on transaction types.

The following list demonstrates the power of the Oracle Procedural Gateway for APPC:

- You can initiate a CICS transaction on the mainframe to retrieve data from a VSAM file for a PC application.
- You can modify and monitor the operation of a remote process control system.
- You can initiate an IMS/TM transaction that executes static SQL in DB2.
- You can initiate a CICS transaction that returns a large number of records in a single call.

1.9 Transaction Types for Gateway Using TCP/IP

The Oracle Procedural Gateway for APPC using TCP/IP for IMS Connect supports three types of transaction socket connections:

- Transaction socket

The socket connection lasts across a single transaction.

- Persistent socket

The socket connection lasts across multiple transactions.

- Nonpersistent socket

The socket connection lasts across a single exchange consisting of one input and one output.

Note: Do not use the nonpersistent socket type if you plan on implementing conversational transactions because multiple connects and disconnects will occur.

Refer to the section about `pg4tcpmap` commands in Chapter 6 of the *Oracle Procedural Gateway for APPC User's Guide* and to [Chapter 13, "Gateway Configuration Using TCP/IP Communication Protocol"](#) in this guide for more information about how to enter these parameters.

You can initiate an IMS/TM transaction that executes static SQL in DB2; this illustrates the power of the Oracle Procedural Gateway for APPC feature supporting TCP/IP for IMS Connect.

Release Information

This chapter provides information about this release of the Oracle Procedural Gateway for APPC. It contains the following sections:

- [Section 2.1, "Product Set"](#)
- [Section 2.2, "Changes and Enhancements, Release 10.2.0.1.0"](#)
- [Section 2.3.1, "Known Restrictions for the Gateway"](#)

2.1 Product Set

The distribution CD for this release of the Oracle Procedural Gateway for APPC includes the following production components:

- Oracle Procedural Gateway for APPC Release 10.2.0
- Oracle Net Release 10.2.0

2.2 Changes and Enhancements, Release 10.2.0.1.0

This sections provide the details of the changes and enhancements in this release of the Oracle Procedural Gateway for APPC. It contains the following topics:

2.2.1 Enhancements

This release of the gateway adds support for Oracle Procedural Gateway for APPC running on Linux 32-bit: Intel Pentium-based on Intel hardware.

2.2.2 Changes

The following sections provide information on changes to this release of the Oracle Procedural Gateway for APPC.

2.2.2.1 Bug 3557168: Init.ora Not Found

If the gateway cannot find the `initsid.ora` file, an error is returned.

2.2.2.2 PG4PWD Replacement Tool Place Holder

For this release, a tool called `tg4pwd` has replaced `pg4pwd`. Refer to [Section 14.6, "Passwords in the Gateway Initialization File"](#) for more information.

2.2.2.3 Bugs Fixed in Release 10.2.0

No bugs have been reported in Release 10.2.0 of the gateway.

2.3 Known Restrictions

This section lists the known restrictions for the Oracle Procedural Gateway for APPC and PGAU.

2.3.1 Known Restrictions for the Gateway

The following restrictions are known to exist in this release of the Oracle Procedural Gateway for APPC.

Multibyte Character Sets are Not Supported for Numeric Data and Clauses

The Oracle Procedural Gateway for APPC has supported multibyte character set data for COBOL PIC G data types from version 3.4 onwards. However, the non-numeric character data (such as \$, (, +, -,.) that is allowed in `DISPLAY` data types and PIC 9 edit masks must still be specified in EBCDIC. The non-numeric character data is not subject to MBCS translation.

CICS Transactions Do Not Allow PF Key Emulation

When performing a CICS transaction using the Oracle Procedural Gateway for APPC, you cannot emulate CICS PF keys.

APPC PIP Data is Not Supported

You cannot define and transmit APPC PIP data in this release of the Oracle Procedural Gateway for APPC.

Floating Point Datatype Conversion is Not Supported

Conversion of the `floating` point data type is not supported by the Oracle Procedural Gateway for APPC.

Transaction Programs are Responsible for All Data Compression and Decompression

The Oracle Procedural Gateway for APPC does not provide exits for compression and decompression facilities. All data exchanged between the gateway and the transaction must be in uncompressed format.

2.3.2 Known Restrictions for PGAU

When COBOL input to the PGAU `DEFINE DATA` statement contains a `COPY REPLACE` clause, only the first replacement is made.

System Requirements

This chapter describes the system requirements of the gateway. It contains the following sections:

- [Section 3.1, "Hardware Requirements"](#)
- [Section 3.2, "Software Requirements"](#)
- [Section 3.3, "Documentation Requirements"](#)

3.1 Hardware Requirements

The hardware requirements for this release of the gateway on your platform are described in the following sections:

- [Section 3.1.1, "Processor Requirements"](#)
- [Section 3.1.2, "Memory Requirements"](#)
- [Section 3.1.3, "Network Attachment Requirements"](#)
- [Section 3.1.4, "CD-ROM Drive"](#)
- [Section 3.1.5, "Disk Space Requirements"](#)

3.1.1 Processor Requirements

Refer to the *Oracle Database Installation Guide 10g Release 2 (10.2) for UNIX Systems* and to the certification matrix on the *OracleMetaLink* for the most up-to-date list of certified hardware platforms and operating system version requirements to operate the gateway for your Linux, AIX-Based, HP-UX, or Solaris system. The *OracleMetaLink* Web site is available at:

<http://metalink.oracle.com>

The gateway processor requirements for your platform are as follows:

- For Linux 32-bit: Intel Pentium-based processors
- For AIX-Based Systems: IBM pSeries
- For HP-UX: HP 9000 Series HP-UX that can run the required version of HP-UX
- For Solaris: A Solaris Operating System (SPARC 64-bit) that can run the required version of Solaris with 64-bit architecture

3.1.2 Memory Requirements

For most installations, Oracle recommends a minimum of 256 MB of real memory for the system running the Oracle Procedural Gateway for APPC.

The following factors affect the virtual memory requirements of the gateway server process:

- Number of concurrent gateway connections open by each user
- Number of data items being transferred between the gateway and the remote transaction program
- The Oracle Net protocol adapters that were included during the gateway installation
- Additional factors, such as configured network buffer size

3.1.3 Network Attachment Requirements

The gateway requires any network attachment supported by either the SNA communication package for your platform or the TCP/IP Networking Facility for TCP/IP communication.

However, if you are using only the new TCP/IP support for IMS Connect feature, you will not need an SNA package. Your operating system comes with TCP/IP installed.

3.1.4 CD-ROM Drive

A CD-ROM drive is required for installation of the software.

3.1.4.1 HP-UX Only

Oracle uses ISO 9660 format CD-ROM disks with RockRidge extensions.

3.1.5 Disk Space Requirements

[Table 3–1](#) lists the disk space requirements for each platform:

Table 3–1 Disk Space Requirements

Platform	Requirements
Linux (Intel 32-bit)	500 MB
AIX-based Systems (64-bit)	1.75 GB
HP-UX (64-bit)	1.25 GB
Solaris (64-bit)	1.5 GB

3.2 Software Requirements

The system software configuration described in this section is supported by Oracle, provided that the underlying system software products are supported by their respective vendors. Verify the latest support status with your system software vendors.

This section contains the following topics:

- [Section 3.2.1, "Operating System Requirements"](#)
- [Section 3.2.2, "Communication Protocol Requirements"](#)

- [Section 3.2.3, "Oracle Server Requirements"](#)
- [Section 3.2.4, "Oracle Networking Product Requirements"](#)
- [Section 3.2.5, "IBM Mainframe Requirements"](#)

3.2.1 Operating System Requirements

Refer to the *Oracle Database Installation Guide 10g Release 2 (10.2) for UNIX Systems* and to the certification matrix on the *OracleMetaLink* for the most up-to-date list of certified operating system version requirements to operate the gateway for your Linux for Intel Pentium-based 32-bit systems, AIX-Based, HP-UX or Solaris system. The *OracleMetaLink* Web site is available at

<http://metalink.oracle.com>

3.2.2 Communication Protocol Requirements

Each operating system uses specific communications servers, as described in this section.

Note: If you choose to use TCP/IP support for IMS Connect as communication protocol, then you do not need to use an SNA communication package from the following list. Each operating system comes with a TCP/IP protocol automatically installed. If you choose to use the TCP/IP protocol, then you need to configure it to work properly with the gateway. Refer to [Chapter 13, "Gateway Configuration Using TCP/IP Communication Protocol"](#).

Communications Protocols for Linux for Intel Pentium-based 32-bit Systems

Communications protocols for Linux for Intel Pentium-based 32-bit System IBM Communications Server V6.0.1.1 for Linux or higher, or the TCP/IP communication software that comes with your operating system.

Communications Protocols for AIX-based Systems

SNA Server for AIX version 6.01 or higher, or the TCP/IP communications software that comes with your operating system.

Communications Protocols for HP-UX

HP-UX SNAPplus2 Link Release 11.x and HP-UX SNAPplus2 API Release 11.x or higher or the TCP/IP communications software that comes with your operating system.

Communications Protocols for Solaris

SNAP-IX version 6 or higher, or the TCP/IP communications software that comes with your operating system.

3.2.3 Oracle Server Requirements

All UNIX platforms require that the Oracle server, that is to act as the Oracle Integrating Server be up to date with the latest patchset for supported Oracle database releases.

3.2.4 Oracle Networking Product Requirements

Oracle Net is automatically installed on the system where the Oracle Database 10g Server is installed and on the system where the gateway is installed. Refer to [Chapter 5, "Configuring Your Oracle Network"](#) in this guide for detailed configuration information. Additionally, you might refer to the *Oracle Net Services Administrator's Guide*.

3.2.5 IBM Mainframe Requirements

In addition to the other software requirements of the gateway and the platform being used, the following list outlines other requirements necessary on the IBM mainframe:

- OLTP for SNA

The OLTP must support mapped APPC conversations. If the OLTP transaction programs to be executed through the gateway perform database updates, then the APPC verbs CONFIRM, CONFIRMED, and SEND_ERR must be supported by the OLTP. These verbs implement APPC sync level 1.

All resources controlled by an OLTP that can be updated by transaction programs invoked through the gateway must be defined as recoverable resources to the OLTP and host system if COMMIT/ROLLBACK capability is required for those resources. For example, a VSAM file updated by a CICS transaction must be defined to CICS as a recoverable file for COMMIT/ROLLBACK to control the updates.

The gateway is compatible with all supported releases of SNA-enabled products such as CICS, IMS/TM, IDMS and z/OS.

Note: For a list of known restrictions, read the "[Known Restrictions](#)" section on page 2-2 before proceeding with the installation of the gateway.

- OLTP for TCP/IP

IMS/TM: Release 7.1 or later is required, as well as any APARs (patches) listed in the *IBM IMS Connect Guide and Reference*.

IMS Connect: Release 1.2 or higher is required.

3.3 Documentation Requirements

Oracle recommends that you become familiar with the concepts and components of all products related to your gateway and that you read the following documentation on Oracle products other than your gateway:

- *PL/SQL User's Guide and Reference*
- *Oracle Database Installation Guide for UNIX*
- *Oracle Database Administrator's Guide*
- *Oracle Database Concepts*
- *Oracle Database Error Messages*
- *Oracle Net Services Administrator's Guide*
- *Oracle Connectivity Guide*

- *Oracle Call Interface Programmer's Guide*

In addition to your Oracle documentation, ensure that you have appropriate documentation for your platform, for your operating system and for your communications packages. You may find IMS Connect Guide and Reference IBM documentation useful.

Note: The title of the IBM publication was accurate at the time of the printing of this guide. Titles and structure of these documents are subject to change. For other Operating System and SNA communication package and TCP/IP package references, refer to the appropriate vendor documentation for your system

Installing the Gateway

This chapter describes how to install and configure the Oracle Procedural Gateway for APPC. It contains the following sections:

- [Section 4.1, "Before You Begin"](#)
- [Section 4.2, "Planning to Upgrade or Migrate the Gateway"](#)
- [Section 4.3, "Preinstallation Steps for Upgrade"](#)
- [Section 4.4, "Installing the Gateway Software"](#)
- [Section 4.5, "Installation Steps"](#)
- [Section 4.6, "Deinstalling Oracle Procedural Gateway for APPC"](#)

4.1 Before You Begin

You can perform certain actions on online transaction processing (OLTP) and some components of the host operating system to configure the OLTP to enable access by the gateway. You also need access to the host system and knowledge of the host system and the OLTP. Although this chapter includes some information about host system and OLTP installation steps, you must ensure that you have the applicable OLTP and host system documentation available.

Some configuration actions on the OLTP might require you to restart the OLTP. Ensure that your host system programmer or DBA review the instructions for your OLTP before you restart it.

To install and configure the gateway with a single Oracle Integrating Server and a single OLTP, perform the procedures described in this chapter.

Note: If your gateway uses the SNA communication protocol, then follow the instructions for installation and configuration in this chapter, in [Chapter 5, "Configuring Your Oracle Network"](#), and in [Chapter 12, "Gateway Configuration Using SNA Communication Protocol"](#).

If your gateway uses the TCP/IP communication protocol, then follow the instructions for installation and configuration in this chapter, in [Chapter 5, "Configuring Your Oracle Network"](#), and in [Chapter 13, "Gateway Configuration Using TCP/IP Communication Protocol"](#).

4.2 Planning to Upgrade or Migrate the Gateway

This section is only for customers who have a previous release of Oracle Procedural Gateway for APPC. If you have a previous gateway installation, then you need to perform some tasks before you can install release 10.2.0.1.0 of the Oracle Procedural Gateway for APPC.

Note: After reading this section, you must read [Chapter 15, "Migration From Existing Gateways"](#) to determine the specific actions you must perform to prepare for upgrade or migration of your gateway. If you are migrating to Oracle Procedural Gateway for APPC release 10.2.0.1.0 from version 4.01 or earlier, then you will find content related to migrating the gateway in [Chapter 15, "Migration From Existing Gateways"](#).

You may also find it helpful to refer to [Appendix D, "Summary of Changes in Previous Versions"](#) for information on changes or corrected problems from earlier releases of the gateway.

If you are installing Oracle Procedural Gateway for APPC for the first time, then begin with ["Preinstallation Steps for Upgrade"](#).

This section includes the following topics:

- [Section 4.2.1, "Preupgrade Procedures"](#)
- [Section 4.2.2, "Upgrade and Migration Considerations"](#)
- [Section 4.2.3, "Restoration"](#)

4.2.1 Preupgrade Procedures

Perform the following steps to prepare for upgrading the previous versions of Oracle Procedural Gateway for APPC to current versions:

1. Make backups of altered PGA shipped files.
2. Remove or rename any old gateway directories.

4.2.2 Upgrade and Migration Considerations

Upgrade considerations are as follows:

- PGAU control files from Gateway release 8 or 9 are upward compatible and you do not need to change them.
- After upgrade, the PG Data Dictionary (PG DD) contains all of its earlier entries without modification. New PGAU control information is added along with some columns to support new features, but no customer entries are altered by the upgrade.
- All TIPs from Oracle Procedural Gateway for APPC release 4.0.1 or earlier must be recompiled because of the changes in the following:
 - PL/SQL compatibility
 - Gateway server RPC interface
 - UTL_PG interface

- If you have existing TIPs that were generated previously on a gateway using the SNA communication protocol and you want to utilize the new TCP/IP feature, then TIPs will have to be regenerated by PGAU with mandatory `NLS_LANGUAGE` and Side Profile Settings. Specify the appropriate ASCII character set in the `DEFINE TRANSACTION` command.

This is because the gateway assumes that the user exit in IMS Connect is being used, which would translate between the ASCII and EBCDIC character sets.

Caution: An upgraded PG Data Dictionary (PG DD) cannot be accessed by an earlier release of PGAU.

4.2.3 Restoration

If you want to restore a previous release of gateway, then you must restore the following components to their previous versions:

- PGAU
- PG DD
- Gateway server

4.3 Preinstallation Steps for Upgrade

Before you install the gateway, perform the following pre-installation procedures:

- Ensure that your system meets all of the hardware and software requirements specified in [Chapter 3, "System Requirements"](#).
- A CD-ROM drive is required.

The Oracle Procedural Gateway for APPC is currently shipped on CD-ROM media. The hardware required for the medium on which the gateway was ordered must be installed on your system.

- Ensure that your security requirements are met.

Refer to [Chapter 3, "System Requirements"](#) for more information about the security requirements for connections and data access on your OLTP.

- Fill out the worksheet identifying unique parameter names needed to configure your system and your chosen communication protocol (either SNA or TCP/IP), which is located in [Chapter E, "Configuration Worksheet"](#).
- Decide on a SID (system identifier) for your gateway. This SID is used in [Section 12.7, "Configuring the Gateway"](#) on page 12-10.

The SID must be unique and must not be used by any other gateway or Oracle Integrating Server on the system.

- SNA only: Your SNA package must be installed and configured before you can proceed with installation of the gateway. Ensure that your system can communicate with the OLTP using the SNA communication package appropriate for your platform.

For more information about setting up and configuring the SNA communication package your platform needs to run the Oracle Procedural Gateway for APPC, refer to the appropriate chapter in this guide from the following list:

- For Linux for Intel Pentium-based 32-bit systems, refer to [Chapter 6, "Configuring the SNA Communication Package on Linux"](#).

- For AIX-Based Systems, refer to [Chapter 7, "Configuring the SNA Communication Package on AIX-Based Systems"](#).
- For HP-UX, refer to [Chapter 8, "Configuring the SNA Communication Package on HP-UX"](#).
- For Solaris Operating System (SPARC 64-bit), refer to [Chapter 9, "Configuring the SNA Communication Package on Solaris"](#).
- TCP/IP only: Your TCP/IP package must be installed and configured before you can proceed with installation of the gateway. Refer to [Chapter 10, "Configuring the TCP/IP Communications Package for IMS Connect"](#) for more information about setting up and configuring TCP/IP for your platform.

Ensure that your system can communicate with the OLTP using the TCP/IP communication package for your platform.

If you need general information about installing Oracle products and using the Oracle Universal Installer, then refer to the *Oracle Database Installation Guide*.

4.3.1 Gateway Installation Methods

You can install the gateway in any of the following ways:

- On the same system as the existing Oracle Integrating Server but in a different directory.
All tasks for this type of installation or upgrade are discussed in this section.
- On a system different from a local Oracle Integrating Server.
- On the same system as the Oracle Integrating Server, and in the same Oracle home directory. Note that in this case, the Oracle Integrating Server and the gateway must be at the same Release level.

4.4 Installing the Gateway Software

For general information about installing Oracle products and how to use the Oracle Universal Installer, refer to the *Oracle Database Installation Guide* and perform all necessary tasks there first.

If your server release is different than your gateway release, *do not* install the gateway in the same Oracle home directory as the Oracle Integrating Server. This is required to isolate the gateway from the Oracle Integrating Server upgrades that might cause incompatibilities if the gateway executables were relinked with later versions of the Oracle server libraries.

4.5 Installation Steps

If you want to install the gateway in the same Oracle home as the Oracle Integrating Server, then the release number of both products must be the same. This section provides the steps for installing the gateway. It contains the following topics:

- [Section 4.5.1, "Step 1: Login as DBA and Create Login User ID"](#)
- [Section 4.5.2, "Step 2: Create the Product Installation Directory"](#)
- [Section 4.5.3, "Step 3: Setting the Environment Variables"](#)
- [Section 4.5.4, "Step 5: Mount the CD-ROM"](#)
- [Section 4.5.5, "Step 6: Start the Oracle Universal Installer"](#)

- [Section 4.5.6, "Step 7: Step through the Oracle Universal Installer"](#)

4.5.1 Step 1: Login as DBA and Create Login User ID

Log in as the Oracle database administrator (DBA) user. If you are not currently a DBA user, then contact your system administrator to create a DBA login user ID. .

See Also: *Oracle Database Installation Guide* for login information

4.5.2 Step 2: Create the Product Installation Directory

When you create a new directory, Oracle recommends that you use the version number as part of the path name. Doing this enables different versions of the same Oracle product to be installed under one Oracle directory tree. The product installation directory is also known as \$ORACLE_HOME for the gateway.

Next create the `/oracle/pga/10.2` product installation directory.

For example, enter:

```
$ mkdir /oracle/pga
$ mkdir /oracle/pga/10.2
$ chown oracle:dba /oracle/pga/10.2
$ chmod 755 /oracle/pga/10.2
```

4.5.3 Step 3: Setting the Environment Variables

The ORACLE_HOME, LD_LIBRARY_PATH, and TNSADMIN environment variables must point to the directory that you created in Step 2. The command to set the environment variable depends on the shell that you are using.

For example, if you are a Bourne or Korn shell user, then enter:

```
$ ORACLE_HOME=/oracle/pga/10.2; export ORACLE_HOME
$ LD_LIBRARY_PATH=$ORACLE_HOME/lib;export LD_LIBRARY_PATH
$ TNS_ADMIN=$ORACLE_HOME/network/admin;export TNS_ADMIN
$ DISPLAY=machine:0; export DISPLAY
```

and, if you are using SNA, then enter:

```
$NLS_LANG=american_america.WE8EBCDIC37C;export NLS_LANG
```

or, if you are using TCP/IP, then enter:

```
$NLS_LANG=american_america.us7ascii;export NLS_LANG
```

If you are a C shell user, then enter:

```
$ setenv ORACLE_HOME /oracle/pga/10.2
$ setenv LD_LIBRARY_PATH /oracle/pga/10.2/lib
$ setenv TNS_ADMIN $ORACLE_HOME/network/admin
$ setenv DISPLAY machine:0
```

and, if you are using SNA, then enter:

```
$ setenv NLS_LANG american_america.WE8EBCDIC37C
```

or, if you are using TCP/IP, then enter:

```
$ setenv NLS_LANG american_america.us7ascii
```

Note: Ensure that the `TNS_ADMIN` environment variable points to the path where your `tnsnames.ora` and `listener.ora` files are. These files are usually in the `$ORACLE_HOME/network/admin` directory.

Set the `DISPLAY` environment variable to the X-server display where the Oracle Universal Installer is displayed.

See Also: *Oracle Database Installation Guide* for more information about this issue

4.5.4 Step 5: Mount the CD-ROM

The following sections describe how to mount the CD-ROM for each platform:

- [Section 4.5.4.1, "Mounting the CD-ROM for Linux"](#)
- [Section 4.5.4.2, "Mounting the CD-ROM for AIX-Based Systems"](#)
- [Section 4.5.4.3, "Mounting the CD-ROM for HP-UX"](#)
- [Section 4.5.4.4, "Mounting the CD-ROM for Solaris Operating System \(SPARC 64-bit\)"](#)

4.5.4.1 Mounting the CD-ROM for Linux

To mount the CD-ROM on Linux:

1. Insert the CD in your CD-ROM drive and enter the following commands:

```
$ su root
# mkdir /cdrom
# mount /dev/hdb /cdrom
# exit
$ cd /cdrom
```

2. Proceed to [Step 6: Start the Oracle Universal Installer For Linux](#) in [Section 4.5.5.1](#)

4.5.4.2 Mounting the CD-ROM for AIX-Based Systems

1. Insert the CD in your CD-ROM drive and enter:

```
$ su root
# mkdir /cdrom
# mount -r -v cdrfs /dev/cd0 /cdrom
# exit
```

2. Proceed to [Step 6: Start the Oracle Universal Installer For AIX Systems](#) in [Section 4.5.5.2](#) on page 4-8, to start the Oracle Universal Installer.

4.5.4.3 Mounting the CD-ROM for HP-UX

To mount the installation, perform the following:

1. Use a system editor to add the following line to the `/etc/pfs_fstab` file.

```
device_file mount_point filesystem_type translation_method
```

For example:

```
/dev/dsk/c5t2d0 /SD_CDRROM pfs-rrip xlat=unix 10
```

The first entry is the CD-ROM device file; the second is the mount point. The third entry indicates that the CD-ROM to be mounted is in ISO9660 format with RockRidge extension.

2. Log in as the root user:

```
$ su root
```

3. Enter the following command:

```
# nohup/usr/sbin/pfs_mountd &
```

4. Enter the following file command:

```
# nohup/usr/sbin/pfsd &
```

5. Insert the CD into the tray and enter the following command to mount the CD-ROM:

```
# /usr/sbin/pfs_mount /SD_CDROM
```

6. Exit the root account.

```
# exit
```

Change directories to /SD_CDROM where you can see a lowercase listing of the directories and files on the CD-ROM. The mounted CD-ROM should appear as another read-only file system.

7. Now proceed to [Step 6: Start the Oracle Universal Installer For HP-UX](#) in [Section 4.5.5.3](#) on page 4-8 to start the Oracle Universal Installer.

4.5.4.4 Mounting the CD-ROM for Solaris Operating System (SPARC 64-bit)

1. Insert the CD in your CD-ROM drive and enter the following command:

```
$ cd /cdrom
```

2. Now proceed to [Step 6: Start the Oracle Universal Installer For Solaris](#) in [Section 4.5.5.4](#) on page 4-8 to start the Oracle Universal Installer.

4.5.5 Step 6: Start the Oracle Universal Installer

The Oracle Universal Installer is provided on the distribution installation media with the gateway. For general information about installing Oracle products and how to use the Oracle Universal Installer, refer to the *Oracle Database Installation Guide*.

The following sections describe how to start the Oracle Universal Installer for each of the platforms:

- [Section 4.5.5.1, "For Linux"](#)
- [Section 4.5.5.2, "For AIX Systems"](#)
- [Section 4.5.5.3, "For HP-UX"](#)
- [Section 4.5.5.4, "For Solaris"](#)

4.5.5.1 For Linux

To start Oracle Universal Installer, enter the following commands:

```
$ cd cdrom_mount_point_directory
$ ./runInstaller
```

4.5.5.2 For AIX Systems

1. Start the Oracle Universal Installer with the following command:

```
$ cd cdrom_mount_point_directory
$ ./runInstaller
```

A window appears with the question "Has the `rootpre.sh` script been run by root?" Before you can proceed with the installation, this script must have been run on the install system of the gateway.

2. If you have already run this script (either during a previous installation of this version of the gateway, or during an installation of the Oracle Database 10g Server), then type "Y" and begin stepping through the Oracle Universal Installer. If you have not run the script, then perform step 3 .
3. If the `rootpre.sh` script has not yet been run, type "N" and complete the following steps:
 - a. Log on as the `root` user.
 - b. Change directory to the install directory on the installation media.
 - c. Run the `rootpre.sh` script.
 - d. Exit from the `root` user.
 - e. Restart the Oracle Universal Installer.

4.5.5.3 For HP-UX

Start the Oracle Universal Installer, as follows:

```
$ cd cdrom_mount_point_directory
$ ./runInstaller
```

4.5.5.4 For Solaris

Start the Oracle Universal Installer, as follows:

```
$ cd cdrom_mount_point_directory
$ ./runInstaller
```

4.5.6 Step 7: Step through the Oracle Universal Installer

Caution: Oracle Universal Installer automatically installs the Oracle-supplied version of the Java Runtime Environment (JRE). This version is required to run the Oracle Universal Installer and several Oracle assistants. Do not modify the JRE except by using a patch provided by Oracle Support Services. The Oracle Universal Installer also installs JDK .

Oracle Universal Installer is a menu-driven utility that guides you through installing the gateway by prompting you with action items. The action items and the sequence in which they appear depend on your platform.

The following section describes how use the Oracle Universal Installer to install the gateway on your platform:

4.5.6.1 Oracle Universal Installer on UNIX platforms

Use [Table 4-1](#) as a guide to step through the Oracle Universal Installer. At each prompt from the Oracle Universal Installer, perform the actions described in the Response column of the table to install the gateway on your UNIX platform.

Table 4-1 The Oracle Universal Installer: Steps for Installing the Gateway

Prompt	Response
1. Oracle Universal Installer: Welcome	Click Next .
2. Oracle Universal Installer: Specify Inventory Directory and credentials	<ul style="list-style-type: none"> a. Specify the full path of the inventory directory. b. Specify the operating system group name. c. Click Next.
3. Oracle Universal Installer: Specify Home Details	<ul style="list-style-type: none"> a. Specify the name of the installation. b. Specify the full path where you want to install the product. c. Click Next.
4. Oracle Universal Installer: Available Product Components	<ul style="list-style-type: none"> a. Deselect the checked products. b. Select "Oracle Transparent Gateway 10.2.0.1.0", open up this row. c. Select "Oracle Procedural Gateway for APPC 10.2.0.1.0". d. Click Next.
5. Oracle Universal Installer: Network Software	Specify your network package and click Next .
6. Oracle Universal Installer: Summary	Click Install .
7. Oracle Net Congiguration Assistance: Welcome	Click Next .
8. Oracle Net Configuration Assistance: Listener Configuration, Listener Name	Specify the name of Listener you want to create and click Next .
9. OracleNetConfiguration Assistance: Listener Configutration, Select Protocols	Select the protocols and click Next .
10. Oracle Net Configuration Assistance: Listener Configuration, TCP/IP Protocol	Specify a port number and click Next .
11. Oracle Net Configuration Assistance:Listener Configuration, More Listeners?	Click "No" and then click Next .
12. Oracle Net Configuration Assistance: Listener Configuration Done	Click Next .
13. Oracle net Configuration Assistance: Naming Methods Configuration	Click No and then click Next .

Table 4–1 (Cont.) The Oracle Universal Installer: Steps for Installing the Gateway

Prompt	Response
14. Oracle Net Configuration Assistance: Done	Click Finish .
15. Execute Configuration scripts	You must run the <code>root.sh</code> configuration script from the <code>\$ORACLE_HOME</code> directory at this point. Leave the installation open, run the script as the <code>root</code> user from another window, then come back to the installation screen and click OK to continue.
16. Oracle Universal Installer: End of Installation	Click Exit .

When the Oracle Universal Installer confirms that the installation is complete, verify that the installation procedure was successful. To do this, read the contents of the installation log file, which is located in the `$ORACLE_HOME/install` directory. The default file name is `make.log`.

Attention: Print the contents of the `$ORACLE_HOME/pg4appc/doc/README.doc` file and read the entire document; it contains important information about the installation. After reading the `README.doc` file, proceed with configuration of the gateway.

4.6 Deinstalling Oracle Procedural Gateway for APPC

If you decide to deinstall the gateway, perform the following steps:

1. Drop the PG DD (packages and procedures):
 - a. Use SQL*Plus to connect to the Oracle Integrating Server as user `PGAADMIN`.
 - b. From SQL*Plus, run the `pgdddel.sql` script in the `$ORACLE_HOME/pg4appc/admin` directory.
2. Log in as DBA. Refer to "[Step 1: Login as DBA and Create Login User ID](#)" for more information.
3. Set the `ORACLE_HOME` environment variable. Refer to "[Step 3: Setting the Environment Variables](#)" for more information.
4. Verify the `DISPLAY` and `ORACLE_HOME` environment variables. Refer to "[Step 3: Setting the Environment Variables](#)" for more information.
5. Mount the CD-ROM. Refer to "[Step 5: Mount the CD-ROM](#)" for more information.
6. Start the Oracle Universal Installer by running the following command:


```
$ ./runInstaller
```
7. Step through the Oracle Universal Installer. Use the prompts listed in [Table 4–2](#), "[Steps to DeInstall the Gateway Using Oracle Universal Installer](#)" as a guide for deinstallation, following the instructions in the Response column.

Table 4–2 Steps to DeInstall the Gateway Using Oracle Universal Installer

Prompt	Response
1. Oracle Universal Installer	Click Deinstall Products .

Table 4–2 (Cont.) Steps to Deinstall the Gateway Using Oracle Universal Installer

Prompt	Response
2. Inventory	Check "Oracle Procedural Gateway for APPC 10.2.0.1.0" and click Remove .
3. Confirmation	Click Yes .
4. Inventory	Click Close .
5. Oracle Universal Installer	Click Cancel .
6. Exit	Click Yes .

8. The gateway is now deinstalled.

When the Oracle Universal Installer confirms that the deinstallation has ended, verify that the deinstallation procedure was successful. To do this, read the contents of the deinstallation log file, which is located in the `$ORACLE_HOME/install` directory. The default file name is `install.log`.

9. The only files that are removed are those that were copied to the `$ORACLE_HOME` directory during the gateway installation. You must remove any other related files manually, including deleting `listener.ora` and `tnsnames.ora` entries relating to the gateway, dropping database links and the `PGAADMIN` user ID and deleting the TIPs.

Configuring Your Oracle Network

This chapter describes how to configure the network.

5.1 Configuring the Oracle Network for the Gateway

The instructions in this section will show you how to configure the network if the gateway is utilizing the SNA or TCP/IP communication protocol.

The gateway must be defined to the TNS listener, and a service name must be defined for accessing the gateway. To do this, perform the following steps:

1. Add an entry for the gateway to the `listener.ora` file:

- If you are using SNA:

```
(SID_DESC=
  (SID_NAME=PGA)
  (ORACLE_HOME=/oracle/pga/10.2)
  (PROGRAM=pg4asrv)
)
```

where: `/oracle/pga/10.2` is your gateway Oracle home and `PGA` is the gateway SID name.

- Or, if you are using TCP/IP:

```
(SID_DESC=
  (SID_NAME=PGA)
  (ORACLE_HOME=/oracle/pga/10.2)
  (PROGRAM=pg4t4ic)
)
```

where: `/oracle/pga/10.2` is the gateway Oracle home and `PGA` is the gateway SID name.

2. Add a service name for the gateway to the `tnsnames.ora` file on the system where your Oracle Integrating Server is located. The service name is specified in the `USING` parameter of the database link defined for accessing the gateway from the Oracle Integrating Server. For example, if you are using the IPC protocol adapter and your gateway `sid` is `PGA`, then add the following entry to `tnsnames.ora`:

```
pgaipc=
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = ipc) (KEY=key))
    (CONNECT_DATA = (SID=PGA))
    (HS=)
  )
```

In this example, *key* is the IPC key defined in the `listener.ora` file for the IPC protocol. You can use the IPC protocol only if the Oracle Integrating Server and the gateway are on the same system.

If you are using the TCP/IP protocol adapter, and if your gateway *sid* is PGA, then add the following entry to `tnsnames.ora`:

```
pgatcp=
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL= TCP) (Host= gateway) (Port= port))
    (CONNECT_DATA = (SID=PGA))
    (HS=)
  )
```

In this example, *port* is the TCP port defined in the `listener.ora` file for the TCP protocol, and *gateway* is the TCP/IP host name of the system where the gateway is located.

Note: For the following cases:

- If your gateway and Oracle Integrating Server are not on the same system,
- If the gateway and the Oracle Integrating Server are on the same system but the Oracle Integrating Server Listener is different than the gateway listener

You must define the Oracle Integrating Server to PGAU by adding a service name to `tnsnames.ora` on the system where your gateway is located. For example:

```
ora_server =
  (DESCRIPTION=
    (ADDRESS =
      (PROTOCOL= TCP)
      (PORT= port)
      (HOST= ora_srv)
    )
    (CONNECT_DATA= (SID= ora_server))
  )
```

In this example:

- *port* is the TCP port defined in the Oracle Integrating Server `listener.ora` for the TCP protocol;
 - *ora_srv* is the TCP/IP host name of the system where the Oracle Integrating Server is located; and
 - *ora_server* is the SID of the Oracle Integrating Server.
-
-

Ensure that you start the defined listener(s).

See Also: *Oracle Net Services Administrator's Guide* for more information about configuring the network

Proceed with Configuring the Communication Package for the Gateway

If your communication protocol is SNA, you must now configure the SNA communication package profiles for APPC connections.

Configure the profiles to define LU6.2 conversations with the OLTP. Refer to the appropriate chapter from the following list to read about the SNA communication package or the TCP/IP package for your platform.

- For Linux for Intel Pentium-based 32-bit systems, refer to [Chapter 6, "Configuring the SNA Communication Package on Linux"](#).
- For AIX-Based Systems, refer to [Chapter 7, "Configuring the SNA Communication Package on AIX-Based Systems"](#).
- For HP-UX, refer to [Chapter 8, "Configuring the SNA Communication Package on HP-UX"](#).
- For Solaris Operating System (SPARC 64-bit), refer to [Chapter 9, "Configuring the SNA Communication Package on Solaris"](#).
- For configuring your TCP/IP communication for the gateway, refer to [Chapter 10, "Configuring the TCP/IP Communications Package for IMS Connect"](#).

In addition, if your communication protocol is SNA, refer to ["Sample listener.ora file for a Gateway Using SNA"](#) and ["Sample tnsnames.ora file for a Gateway Using SNA"](#) in [Appendix A](#).

If your communication protocol is TCP/IP, refer to ["Sample listener.ora File for a Gateway Using TCP/IP"](#) and ["Sample tnsnames.ora File for a Gateway Using TCP/IP"](#) in [Appendix B](#).

Configuring the SNA Communication Package on Linux

The Oracle Procedural Gateway for APPC uses the SNA Advanced Program to Program Communication (APPC/LU6.2) protocol to communicate with an OLTP. Linux for Intel Pentium-based 32-bit system support for APPC is provided by IBM Communications Server for Linux.

This chapter describes how to configure your SNA Server on a Linux for Intel Pentium-based 32-bit system to run the Oracle Procedural Gateway for APPC, using IBM Communications Server for Linux.

This chapter contains the following sections:

- [Section 6.1, "Using SNA Security Validation"](#)
- [Section 6.2, "Processing Inbound Connections"](#)
- [Section 6.3, "Independent Versus Dependent LUs"](#)
- [Section 6.5, "Creating IBM Communications Server Definitions for the Gateway"](#)
- [Section 6.6, "Testing the Connection"](#)
- [Section 6.7, "Resume Configuration of the Gateway"](#)

Note: The term SNA Server is used in this guide to generally refer to the IBM Communications Server.

6.1 Using SNA Security Validation

When the gateway receives an RPC request to start a remote transaction program, the gateway attempts to start an APPC conversation with online transaction processing (OLTP). Before the conversation can begin, a session must start between the Linux Logical Unit (LU) and the OLTP LU.

SNA and its various access method implementations (including SNA Server and VTAM) provide security validation at session initiation time, enabling each LU to authenticate its partner. This validation is carried out entirely by network software before the gateway and OLTP application programs begin their conversation and process conversation-level security data. If session-level security is used, then correct password information must be established in the gateway SNA Server definitions and in similar parameter structures in the OLTP to be accessed. Refer to the appropriate communications software product documentation for detailed information about this subject.

6.2 Processing Inbound Connections

Many OLTPs provide options for manipulating the security conduct of an inbound (client) APPC session request. Refer to the appropriate OLTP documentation for detailed information about this topic.

Note: For CICS, one security option is not supported by the gateway: `ATTACHSEC=PERSISTENT`, specified in the `CICS CONNECTION` definition, requires capability that is not yet available in the gateway.

However, the `ATTACHSEC=LOCAL`, `ATTACHSEC=IDENTIFY`, `ATTACHSEC=VERIFY`, and `ATTACHSEC=MIXIDPE` security options are fully supported by the gateway.

6.3 Independent Versus Dependent LUs

Oracle recommends that you use independent LUs for the Oracle Procedural Gateway for APPC because they support multiple parallel sessions or conversations. This means that multiple Oracle client applications can be active simultaneously using the same OLTP through the independent LU.

Dependent LUs support only a single active session. The CP (Control Point for the node) queues additional conversation requests from the gateway server behind an already active conversation.

In other words, conversations are single threaded for dependent LUs. If a gateway LU is correctly defined, then you do not need to alter the configuration for the Oracle Procedural Gateway for APPC and the gateway server.

The operational impact of dependent LUs is that the first client application can initiate a conversation through the Procedural Gateway with the gateway server. While that transaction is active (which could be seconds to minutes to hours, depending on how the client application and transaction are designed), any other client application initiating a session with the same gateway server appears to hang as it waits behind the previous session.

If a production application uses only a single conversation at any one time, then there should be no impact.

However, additional concurrent conversations might be required for testing or for other application development. Each conversation requires that you define additional dependent LUs on the remote host, plus additional IBM Communication Server configuration entries, which define the additional dependent LUs on the gateway system. Additional Side Information Profiles should be defined to use the new dependent LUs. New gateway instances should be created and configured to use these new Side Information Profiles.

See Also: Refer to `PGAU DEFINE TRANSACTION SIDEPROFILE` and `LUNAME` parameters in Chapter 2, "Procedural Gateway Administration Utility," in *Oracle Procedural Gateway for APPC User's Guide*

6.4 Definition Types

Several types of IBM Communications Server definitions are relevant to gateway APPC/LU6.2 operation. Each definition can be created and edited using a corresponding SNA Node Configuration menu.

The definitions relevant to the gateway are presented in [Section 6.5](#) in hierarchical order. This section provides an overview of IBM Communications Server definitions in relation to the Oracle Procedural Gateway for APPC. Those definition types that are lowest in the hierarchy are discussed first. This matches the logical sequence in which to create the definitions.

Refer to the IBM Communications Server online documentation for a complete discussion of IBM Communications Server definitions.

6.5 Creating IBM Communications Server Definitions for the Gateway

IBM Communications Server definitions are created using the SNA Node Configuration tool, while the operation of the server is done using the SNA Node Operations tool, both of which are provided with IBM Communications Server. Maintenance of SNA definitions is normally done by a user with Administrator privileges.

The following sections describe the process of creating SNA definitions for IBM Communications Server using the SNA Node Configuration tool. All of the tasks described in this section are performed within SNA Node Configuration.

The section includes the following topics:

- [Section 6.5.1, "Creating the Configuration"](#)
- [Section 6.5.2, "Creating the Node"](#)
- [Section 6.5.3, "Creating Devices"](#)
- [Section 6.5.4, "Choosing the Device Type"](#)
- [Section 6.5.5, "Configuring a LAN Device"](#)
- [Section 6.5.6, "Creating Peer Connections"](#)
- [Section 6.5.7, "Defining the Link Station"](#)
- [Section 6.5.8, "Defining the Adjacent Node"](#)
- [Section 6.5.9, "Creating Local LUs"](#)
- [Section 6.5.10, "Defining Local LUs"](#)
- [Section 6.5.11, "Creating Partner LUs"](#)
- [Section 6.5.12, "Defining Partner LUs"](#)[Section 6.5.13, "Creating the CPI-C Side Information Profile"](#)
- [Section 6.5.13, "Creating the CPI-C Side Information Profile"](#)

6.5.1 Creating the Configuration

The SNA Node Configuration screen prompts you to specify if you are creating a new configuration or loading an existing configuration. These tasks are based on the assumption that a new configuration is being created.

SNA Node Configuration next prompts you for a configuration scenario.

6.5.2 Creating the Node

Each SNA Server must have a Control Point defined. This is typically called the Node definition. To define the node:

1. Click **Node**.

2. Click **Create**.

In the Define the Node dialog box:

- a. Select the **Basic** tab.
- b. Enter information in the Control Point, Local Node ID, and Node Type boxes.

You can select **Advanced** tab options depending on your SNA network configuration.

3. Click **OK**.

6.5.3 Creating Devices

To configure communication devices:

1. Click **Devices**.
2. Click **Create**.

6.5.4 Choosing the Device Type

Select the type of device to use for communication. The LAN type is typical for either Ethernet or Token Ring attached network devices.

6.5.5 Configuring a LAN Device

To configure a LAN device:

1. Select the **Basic** tab.
2. Choose the Adapter to use and the Local SAP. The other tabs provide options for network tuning parameters.
3. Click **OK**.

6.5.6 Creating Peer Connections

To create peer connections:

1. Click **Peer Connections**.
2. Click **Create**.

6.5.7 Defining the Link Station

To define the link station:

1. In the Basic tab, enter a Link Station name for this connection.
2. Choose the device for the connection.
3. Enter the Destination address and Remote SAP.

6.5.8 Defining the Adjacent Node

To define the adjacent node:

1. Select the **Adjacent Node** tab.
2. Enter the Adjacent CP name of the remote system and select its CP type.

You might have to choose a different transmission group (TG) than the default. Consult your SNA network administrator for details.

Other tabs provide options on tuning and reactivation.

3. Click **OK**.

6.5.9 Creating Local LUs

To create local LUs for the node:

1. Click **Local LU 6.2 LUs**.
2. Click **Create**.

6.5.10 Defining Local LUs

To define local LUs:

1. In the **Basic** tab, enter the name of the Local LU, and, optionally, an alias. The name must match the Local LU definition of the remote host for this Node. You can examine the other tab for synchronization support and for LU session limits.
2. Click **OK**.

6.5.11 Creating Partner LUs

To create partner LUs:

1. Click **Partner LU 6.2 LUs**.
2. Click **Create**.

6.5.12 Defining Partner LUs

To define partner LUs:

1. In the **Basic** tab, enter the name of the Remote or Partner LU, and, optionally, an alias.
2. Choose the Fully Qualified CP from the Existing list. You can examine the other tab for logical record limits and security support.
3. Click **OK**.

6.5.13 Creating the CPI-C Side Information Profile

To define the CPI-C profile that will be used to create the gateway:

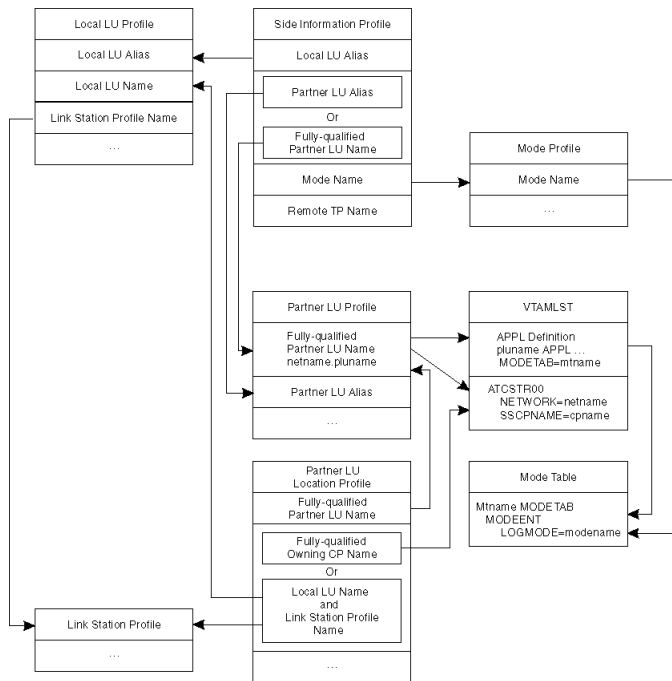
1. Click the **CPI-C Side Information Definitions**.
2. Click **Create**.

6.6 Testing the Connection

Before proceeding with the gateway configuration tasks, ensure that your connection is working.

[Figure 6–1](#) shows the relationship between SNA Server definitions and the VTAM definitions on the host.

Figure 6–1 SNA server defianctions and VTAM



6.7 Resume Configuration of the Gateway

When you have finished configuring the SNA communication package for your Linux for Intel Pentium-based 32-bit System, proceed to [Chapter 11, "Configuring the OLTP"](#) to continue configuring the network.

Configuring the SNA Communication Package on AIX-Based Systems

The Oracle Procedural Gateway for APPC uses the SNA Advanced Program to Program Communication (APPC/LU6.2) protocol to communicate with online transaction processing (OLTP). All AIX-based system support for APPC is provided by IBM SNA server product.

This product requires a stored set of definitions, called profiles, to support connections between the gateway and the applications using OLTP. Each profile consists of a profile name, a profile type, and a set of fields describing the profile. The fields in a profile type are generally a mixture of operating parameter values and names of other SNA profiles relevant to the profile.

Refer to this chapter if your gateway uses the SNA communication protocol. This chapter describes how to create and activate SNA server profiles.

Note: When you finish following the instructions in this chapter to configure your communication protocol, refer to [Chapter 5, "Configuring Your Oracle Network"](#) to continue your network configuration.

This chapter contains the following sections:

- [Section 7.1, "Processing Inbound Connections"](#)
- [Section 7.2, "Independent Versus Dependent LUs"](#)
- [Section 7.3, "Creating SNA Profiles for the Gateway"](#)
- [Section 7.4, "Profile Types"](#)
- [Section 7.5, "SNA Server Profiles"](#)
- [Section 7.6, "Activating Profiles"](#)
- [Section 7.7, "Resume Configuration of the Gateway"](#)

7.1 Processing Inbound Connections

Many OLTPs provide options for manipulating the security conduct of an inbound (client) APPC session request. Refer to the appropriate OLTP documentation for detailed information about this topic.

Note: For CICS, one security option is not supported by the gateway. `ATTACHSEC=PERSISTENT`, specified on the CICS `CONNECTION` definition, requires capability that is not yet available in the gateway.

However, `ATTACHSEC=LOCAL`, `ATTACHSEC=IDENTIFY`, `ATTACHSEC=VERIFY`, and `ATTACHSEC=MIXIDPE` are fully supported by the gateway.

7.2 Independent Versus Dependent LUs

Oracle recommends independent LUs for Oracle Procedural Gateway for APPC because they support multiple parallel sessions or conversations. This means that multiple Oracle client applications can be active simultaneously with the same OLTP through the independent LU.

Dependent LUs support only a single active session. The CP (SNA server for AIX, in this case) queues additional conversation requests from the gateway server behind an already active conversation. In other words, conversations are single threaded for dependent LUs.

If a dependent LU is correctly defined, then you do not need to alter the Oracle Procedural Gateway for APPC configuration, the host transaction, or how the OLTP is started.

The operational impact of dependent LUs is that the first client application can initiate a conversation through the Procedural Gateway with the OLTP. While that transaction is active (which could be seconds to minutes to hours depending on how the client application and transaction are designed), any other client application initiating a conversation with the same OLTP instance appears to hang as it waits behind the previous conversation.

If a production application really uses only a single conversation or transaction at a time, then there should be no impact.

However, additional concurrent conversations or transactions might be required for testing or for other application development. Each requires that additional dependent LUs be defined on the remote host, plus additional SNA server profiles, which define the additional dependent LUs on the IBM pSeries workstation. The TIP that initiates the conversation must specify the different Partner LU through a different side information profile or by overriding the LU name.

See Also: Refer to `PGAU DEFINE TRANSACTION SIDPROFILE` and `LUNAME` parameters in Chapter 2, "Procedural Gateway Administration Utility," in the *Oracle Procedural Gateway for APPC User's Guide* and the SNA server-side information profile discussed in the section, [Section 7.5.7, "Side Information Profile"](#)

7.3 Creating SNA Profiles for the Gateway

You can create and modify the SNA server profile definitions using menus in the AIX System Management Interface Tool (`smit`).

Maintenance of SNA server profiles is normally done by a user with `root` access. The following information is intended for the user who creates profiles for the gateway. You should have some knowledge of SNA before reading this section.

By using `smit`, you should be able to accept most of the defaults. The default values assigned to many of the fields in a new set of profiles are acceptable for the gateway.

The `$ORACLE_HOME/pg4appc/sna` subdirectory contains a sample set of profiles for the gateway in the `pgasna.export` file.

Before building the SNA server profiles, examine the appropriate sample export file to determine the profiles needed, their contents, and their interrelationships. The export file format is text-oriented, and each field of each profile is clearly labeled. You can print a copy of the export file to use while working with the profiles in a smit session.

7.4 Profile Types

Several types of SNA server profiles are relevant to gateway APPC/LU6.2 operation. Each profile can be created and edited using a corresponding smit menu that can be reached from the Communications Applications and Services primary menu choice.

The profiles are presented in hierarchical order. The profile types that are lowest in the hierarchy are discussed first. This matches the logical sequence in which to create the profiles. You can use the smit list menu to specify the profile names.

7.5 SNA Server Profiles

Refer to the appropriate vendor documentation for a complete discussion of SNA profiles. This section is an overview of SNA server profiles in relation to the Oracle Procedural Gateway for APPC. It includes the following topics:

- [Section 7.5.1, "SNA Node Profile"](#)
- [Section 7.5.2, "Link Station Profile"](#)
- [Section 7.5.3, "Mode Profile"](#)
- [Section 7.5.4, "Local LU Profile"](#)
- [Section 7.5.5, "Partner LU Profile"](#)
- [Section 7.5.6, "Partner LU Location Profile"](#)
- [Section 7.5.7, "Side Information Profile"](#)

7.5.1 SNA Node Profile

The SNA node profile defines miscellaneous SNA system defaults. Set the "Maximum number of sessions" and "Maximum number of conversations" fields to values large enough to handle the maximum number of concurrent gateway conversations anticipated, plus any non-gateway sessions and conversations that are in use on your system by other applications.

Set the "Recovery Resource Manager (RRM) enabled?" field to `no`, unless you already have other applications running on your AIX system that require this field to be set to `yes`. For example, CICS pSeries and Encina need this field set to `yes`.

7.5.2 Link Station Profile

The Link Station Profile and the related DLC profile describe and control the connection of the IBM pSeries to the network. Details on profile contents are not discussed here because Oracle Procedural Gateway for APPC does not impose special requirements on these profiles. The sample profile distributed in `pgasna.export` includes a profile created for a Token-Ring network connection. The Link Station Profile name is specified later in the Partner LU Location Profile, if one is necessary.

7.5.3 Mode Profile

The Mode Profile specifies parameters that determine:

- APPC/LU6.2 parallel session limits
- send and receive pacing values
- SNA RU size
- the mode name that is sent to OLTP at session initiation

The mode name that you specify must be defined to the OLTP communications software. Choose the mode name in addition to the other mode parameters after consulting the user who is responsible for configuring the OLTP communications software.

The parameters that are related to parallel session limits determine the maximum number of concurrent conversations allowed between a gateway instance and the OLTP. This equates to the maximum number of concurrently active remote transaction program invocations through the gateway instance.

The mode name, for example, `ORAPLU62`, is specified later in the Side Information Profile.

Note: Do not confuse the Mode Profile with the mode name.

7.5.4 Local LU Profile

The Local LU (Logical Unit) Profile describes the SNA LU through which the gateway communicates.

An LU name must be assigned to the gateway. The LU name assigned to the gateway might be required elsewhere in the SNA network. Contact the SNA network administrator to determine the correct LU name to specify in the profile.

Set the "Local LU name" to the LU name assigned to the gateway.

An alias should be assigned to the LU using the "Local LU alias" field. This alias is used later in the side information profile.

Set the "Local LU is dependent" field to `no`.

The Local LU Profile name is specified later in the side information profile.

7.5.5 Partner LU Profile

The Partner LU Profile describes the SNA LU of the OLTP system with which the gateway communicates. The name of the OLTP LU and the name of your SNA network must be specified in this profile. Contact your SNA network administrator to determine the correct LU and network names.

Set the "Fully qualified partner LU name" field to the network name, followed by a period, followed by the OLTP LU name. For example, `network.oltplu`.

You can assign an alias to the partner LU name by setting the "Partner LU alias" to the value of your choice. This enables you to reference the partner LU without knowing the fully qualified partner LU name and minimizes the change if the partner LU name is changed.

Set the "Parallel sessions supported?" field to `yes` unless your OLTP does not support parallel sessions.

If you plan to use SNA session or conversation security, then set the "Session security supported?" and "Conversation security supported?" fields as required. These settings require the Session Security and Conversation Security Profiles.

See Also: The vendor documentation for more information

7.5.6 Partner LU Location Profile

The Partner LU Location Profile is used when the remote host where the Partner LU is located is not an APPN-capable node. Many mainframe systems do not have APPN capability. For example, z/OS systems running VTAM versions before version 4 do not support APPN. Also, if your hardware connection is through a front-end processor running NCP versions before version 5, then APPN is not supported. In these cases, the Partner LU Location Profile can be used to specify the name of the System Services Control Point (SSCP) or Control Point (CP), which owns the network connection to the partner LU.

Set the "Fully qualified partner LU name" field to the network name, followed by a period, followed by the OLTP LU name. For example, `network.oltp.lu`.

Set the "Partner LU location method" and associated fields as required by your network configuration. If you use the `owning cp` option, then the "Fully qualified owning Control Point (CP) name" field should be set to the SSCP or CP name, which owns the network connection to the partner LU. For VTAM, the SSCP name is the value of the VTAM `NETID` start parameter, usually found in `VTAMLST` member `ATCSTR00`.

7.5.7 Side Information Profile

The side information profile is a required profile, which is used to identify target OLTP systems to be accessed through Oracle Procedural Gateway for APPC.

The side information profile identifies the following:

- the local LU alias
- the partner LU alias or fully-qualified name
- the remote transaction program name (optional)
- the mode name

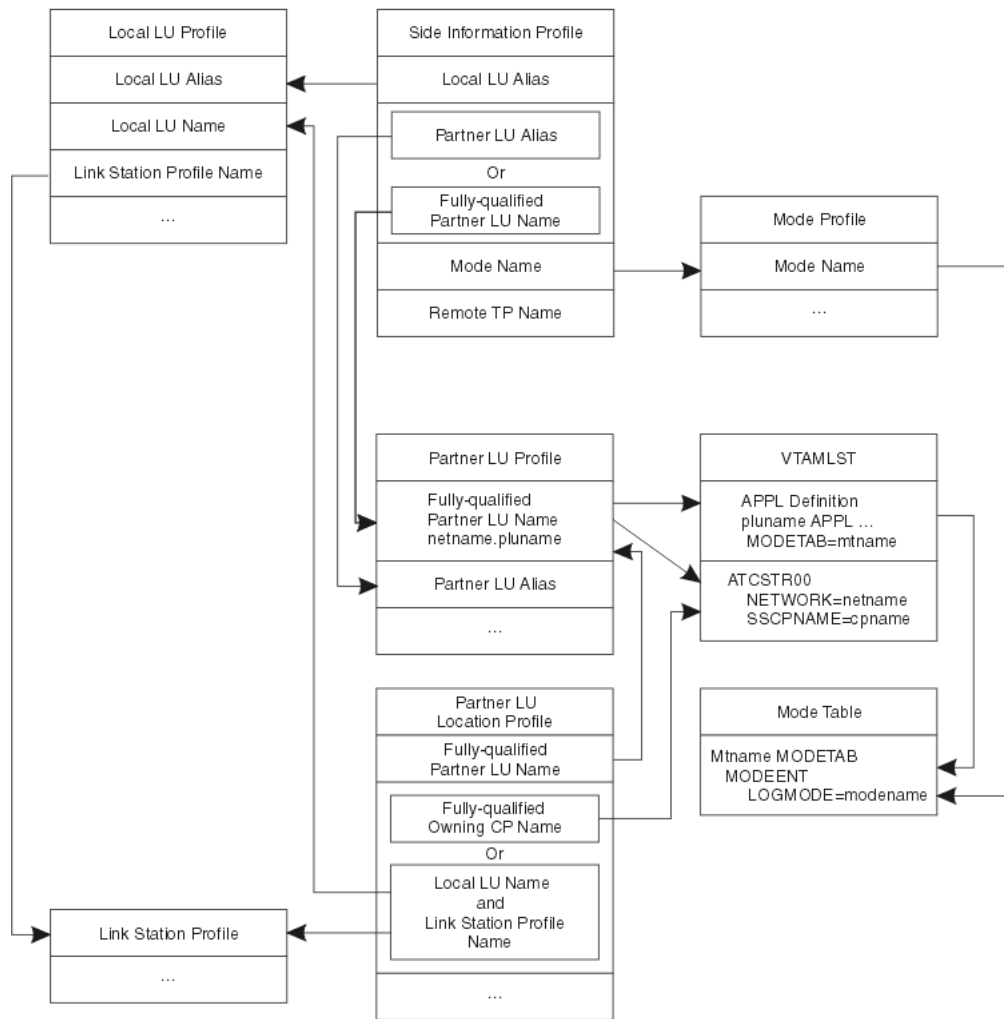
Set the profile information as follows for each side information profile field:

- Set the Local LU or Control Point alias to the alias assigned to the local LU in the Local LU Profile.
- Set the Mode name to the actual mode name as specified in the Mode Profile.
- Set the Remote transaction program name (RTPN) using the actual remote TP name or a dummy name to be overridden at execution time.
- Set the "RTPN in hexadecimal?" field to `yes`, if the remote TP name is hexadecimal.
- If there is a field for "Partner LU alias", then add the alias assigned to the partner LU in the Partner LU Profile.
- If there is a field for "Fully qualified partner LU name", then add the fully qualified partner LU name of the partner LU.

Figure 7–1 shows the relationship between SNA server profiles and the VTAM definitions on the host.

Figure 7-1 Relationship Between SNA Profiles and Host VTAM Definitions

SNA profiles and host VTAM on AIX-based systems



7.6 Activating Profiles

After you have built all the necessary SNA server profiles for communicating with the remote host, you must verify the profiles. Use the "Verify Configuration Profiles" option under the "Advanced Configuration" option of the smit SNA server menu. Then use smit to start the link station profile.

See Also: the vendor documentation for more information about using smit to start link stations

7.7 Resume Configuration of the Gateway

When you finish configuring the SNA communication package for your AIX-based system, proceed to [Chapter 11, "Configuring the OLTP"](#) to continue configuring the network.

Configuring the SNA Communication Package on HP-UX

Oracle Procedural Gateway for APPC uses the SNA Advanced Program to Program Communication (APPC/LU6.2) protocol to communicate with an online transaction processing (OLTP). HP-UX system support for APPC is provided by the SNAPPlus2 product.

This chapter describes how to set up and configure SNAPPlus2 on a HP-UX system to run Oracle Procedural Gateway for APPC.

Note: When you finish following the instructions in this chapter, refer to [Chapter 5, "Configuring Your Oracle Network"](#) to continue network configuration.

This chapter contains the following sections:

- [Section 8.1, "Processing Inbound Connections"](#)
- [Section 8.2, "SNAPPlus2 Configuration Tool"](#)
- [Section 8.3, "Creating SNAPPlus2 Profiles for the Gateway"](#)
- [Section 8.4, "Independent Versus Dependent LUs"](#)
- [Section 8.5, "Creating SNA Definitions for the Gateway"](#)
- [Section 8.6, "Sample SNAPPlus2 Definitions"](#)
- [Section 8.7, "Configuring SNAPPlus2"](#)
- [Section 8.8, "Testing the Connection"](#)
- [Section 8.9, "Resuming Gateway Configuration"](#)

8.1 Processing Inbound Connections

Many OLTPs provide options for manipulating the security conduct of an inbound (client) APPC session request. Refer to the appropriate documentation for your OLTP for detailed information about this topic.

Note that for CICS, one security option is not supported by the gateway: `ATTACHSEC=PERSISTENT`, specified on the CICS `CONNECTION` definition, requires capability that is not yet available in the gateway.

`ATTACHSEC=LOCAL`, `ATTACHSEC=IDENTIFY`, `ATTACHSEC=VERIFY`, and `ATTACHSEC=MIXIDPE` are fully supported by the gateway.

8.1.1 Steps for Configuring the Communications Interfaces

To configure the communications interfaces, do the following:

1. Create SNAPPlus2 profiles for the gateway.
2. Create SNA definitions for the gateway.
3. Test the configuration.

8.2 SNAPPlus2 Configuration Tool

All SNAPPlus2 product configuration is done using the `xsnapadmin` program. `xsnapadmin` is an X-Windows application that provides a graphical interface to view and modify the current SNAPPlus2 configuration and the current running state of the host SNA node. Refer to vendor documentation for more information about using `xsnapadmin`.

8.3 Creating SNAPPlus2 Profiles for the Gateway

Oracle Procedural Gateway for APPC requires a stored set of definitions, called Side Information Profiles, to support connections between the gateway and remote server. Each profile consists of a profile name and a profile type, which is a set of fields describing the profile. The fields in a profile type are generally a mix of operating parameter values and names of other SNA profiles relevant to the profile. Each functional part of APPC, such as Mode, Remote Transaction Program name and Logical Unit (LU), is described by a distinct profile type.

8.4 Independent Versus Dependent LUs

Oracle recommends independent LUs for the gateway because they support multiple parallel sessions or conversations. This means multiple Oracle client applications can be active simultaneously with the same OLTP through the independent LU.

Dependent LUs only support a single active session. SNAPPlus2 queues additional conversation requests from Oracle Procedural Gateway for APPC behind an active conversation. That is, conversations are single-threaded for dependent LUs.

If a dependent LU is correctly defined, no alterations are needed for the gateway configuration, to the host transaction, or to how the OLTP is started.

The operational impact of dependent LUs is that the first client application can initiate a conversation through the gateway with the OLTP, and while that transaction is active (which could be for seconds, minutes, or hours, depending on how the client application and transaction are designed), any other client application initiating a conversation with the same OLTP instance appears to hang as it waits behind the previous conversation.

If a production application really only uses a single conversation or transaction at any one time, there should be no impact.

However, additional concurrent conversations or transactions might be required for testing or other application development. Each requires that additional dependent LUs be defined on the remote host in addition to an SNAPPlus2 configuration file entry that defines the additional dependent LUs on the HP-UX workstation. The TIP that initiates the conversation must specify the different SNAPPlus2 Partner LU through a different Side Information Profile. Refer to the `PGAU DEFINE TRANSACTION SIDEPROFILE` parameter in Chapter 2 of *Oracle Procedural Gateway for APPC User's*

Guide, and the SNAPPlus2 Symbolic Destination Name discussed in the section, "[Sym Dest Name](#)" on page 8-3.

In some uses of the gateway, independent LUs cannot be used. For example, with the IMS LU6.1 Adapter for LU6.2, parallel sessions are not supported. In this case, multiple concurrent sessions with the IMS LU6.1 Adapter for LU6.2 can be achieved by defining a pool of dependent LUs. For each dependent LU, select the "LU in the pool of default LUs" option. When a conversation is requested, an available local LU from the default LU pool is assigned automatically by SNAPPlus2. For more information, refer to vendor documentation.

8.4.1 Sym Dest Name

This option lets you enter the side information associated with a particular symbolic destination name. You can use an alphanumeric string up to 8 characters as the `Sym Dest Name`. The symbolic destination name is referred to as the side information profile in other parts of this guide. This name is specified by the `SIDEPROFILE` keyword in the `DEFINE TRANSACTION` statement used to define your transaction to PGAU.

The Partner TP name field specifies the name of the transaction to be run on the OLTP side of the conversation. This field must be specified, but the TP name can be overridden by the gateway at conversation startup.

The Partner LU field specifies the LU name of the OLTP on the remote host. The Mode Name field specifies the mode name to be used for conversations with the specified OLTP.

The security information that can be specified in this menu is not usable for the gateway. The security parameters are always set by the gateway based on gateway initialization parameters.

8.5 Creating SNA Definitions for the Gateway

SNAPPlus2 definitions are stored in the following files, which are located in the `/etc/opt/sna` directory:

- SNA node definitions: `sna_node.cfg`
- SNA domain definitions: `sna_domn.cfg`

These files are created and maintained with the `xsnapadmin` tool. Maintenance of the SNA definitions is normally done by a user with administrative authority. The following information is intended for the person creating SNA definitions for the gateway. You should have some knowledge of SNA before reading this section.

8.6 Sample SNAPPlus2 Definitions

The gateway `$ORACLE_HOME/pg4appc/sna` subdirectory contains a set of sample SNAPPlus2 definitions for the gateway, created with the `xsnapadmin`. SNA definitions are very specific to the HP 9000 host and SNA network. Because of this, these sample definitions will not work without being tailored for the local host and SNA network.

8.7 Configuring SNAPPlus2

This section describes the process of creating SNA definitions for SNAPPlus2 using the `xsnapadmin` tool. All configuration is done using the various pulldown menus and panels in `xsnapadmin`. The configuration descriptions in the steps below follow the

samples provided. You must tailor the various SNA values for your local host and SNA network.

Step 1 Starting xsnapadmin

Use the following commands to run `xsnapadmin`. The `DISPLAY` environment variable must be appropriately set. If you are running `xsnapadmin` from the local HP 9000 console, then `DISPLAY` should already be set. If you are running `xsnapadmin` from a remote X display, then set `DISPLAY` to the host name or IP address of that display.

```
$ DISPLAY=xstation10.us.oracle.com:0
$ export DISPLAY
$ xsnapadmin &
```

At `xsnapadmin` startup, the main screen opens and displays the current configuration of the local SNA node.

Step 2 Configuring the SNA Node

On the main screen of `xsnapadmin`, pull down the Services menu and select **Configure Node Parameters**.

1. In the Node Parameters dialog box, enter the APPN support type, the Control Point Name, Control Point, and Node ID as needed. The Control Point Name is composed of the SNA Network Name and the CP name of the local host.
2. Click **OK**.

Step 3 Adding a Port

From the Service menu, select **Connectivity** and then select **Add Port**.

1. In the Add to *nodename* dialog box, select the Port and type you are using and click **OK**.
2. In the subsequent SAP dialog box, enter a Port name and network card number. The Port name is used to logically name the physical network card you are using and to bind a Service Access Port to the card for SNA protocols. Usually, you can accept the values provided in the dialog box. If a different network card is needed, however, enter the card number as reported with the `lanscan` command.
3. Click **OK**.

Step 4 Create a Link Station

When the Port has been defined, you need to create a Link Station. The Link Station represents the SNA node of the remote host of the APPC server. But before you can create the Link Station, you must create a Remote Node definition.

1. From the Services menu, select **APPC** and select **Add Remote Node**.
2. In the Remote Node dialog box, enter the SNA CPNAME of the remote node and click **OK**.
3. Now you are ready to create the Link Station. From the Services menu, select **Connectivity** and select **Add Link Station**. In the dialog box that is displayed, select the Port previously defined and click **OK**.
4. In the Link Station dialog box, enter a name for the Link Station, choose the SNA Port name, and choose the type of link activation.
5. Choose the LU Traffic type. For maximum flexibility, choose the **Any** option.

6. For Independent LU traffic, specify the Remote Node name.
7. Click on **Remote Node** and select the node you previously created. Click **OK**.
8. For Dependent LU traffic, specify the Local Node ID, and optionally, Remote Node ID. Then, specify the Contact Information. Contact information contains the MAC address of the remote host as well as the SAP number. Click **Advanced** for additional parameters of the Link Station.
9. The Ethernet Parameters screen shows additional parameters of the Link Station. These parameters affect initial XID contact and retransmission times and limits. You need not change the default values. Click **OK**.

Step 5 Create Local LUs

When the Remote Node definitions have been made, create the Local LU names for the local host. From the Services menu, select **APPC** and **Add Local LU**.

1. In the Local LU dialog box, enter the name of the local LU and an alias. This name must correspond to the VTAM definitions on the remote server host for the HP 9000 host.
2. Click **OK**.

Step 6 Create Partner LUs

Now define a Partner LU that represents the LU that the remote server is using to communicate. From the Services menu, select **APPC** and **Add Partner LUs and Partner LU on Remote Node**.

1. In the dialog box that is displayed, enter the Partner LU name and characteristics. The Partner LU name will contain the SNA Network Name as well as the LU name of the remote LU.
2. Enable parallel session support. The location is the name as the Remote Node name. You can click on **Location** for a list.
3. Click **OK**.

Step 7 Create Mode and CPI-C Profiles

Once the local and remote LU definitions have been made, create the necessary Mode and CPI-C definitions. From the Services menu, select **APPC** and **Modes**.

1. In the Modes dialog box, click on **Add** to add a new mode.
2. In the subsequent Mode dialog box, enter the Mode Name and other session parameters. The recommended name for an APPC mode is ORAPLU62. Contact your Remote Host system administrator for an appropriate mode parameter name.
3. Click **OK**.
4. Now that the Mode has been defined, create the CPI-C Side Information Profile, which the gateway will use as a connection name. From the menu, select **APPC** and **CPI-C**. In the CPI-C destination names dialog box, click on **Add** to add a new profile.
5. In the CPI-C destination dialog box, enter the Profile name, Partner TP, Partner LU, mode, and Security option. The partner TP name is the name of the host transaction program or a dummy value to be overridden in the TIP.
6. For the Partner LU, enter either the full LU name or the alias created previously. Enter ORAPLU62 for the mode name.

7. Lastly, choose the type of security these sessions will use. This affects how session authorization is done.
8. Click OK.

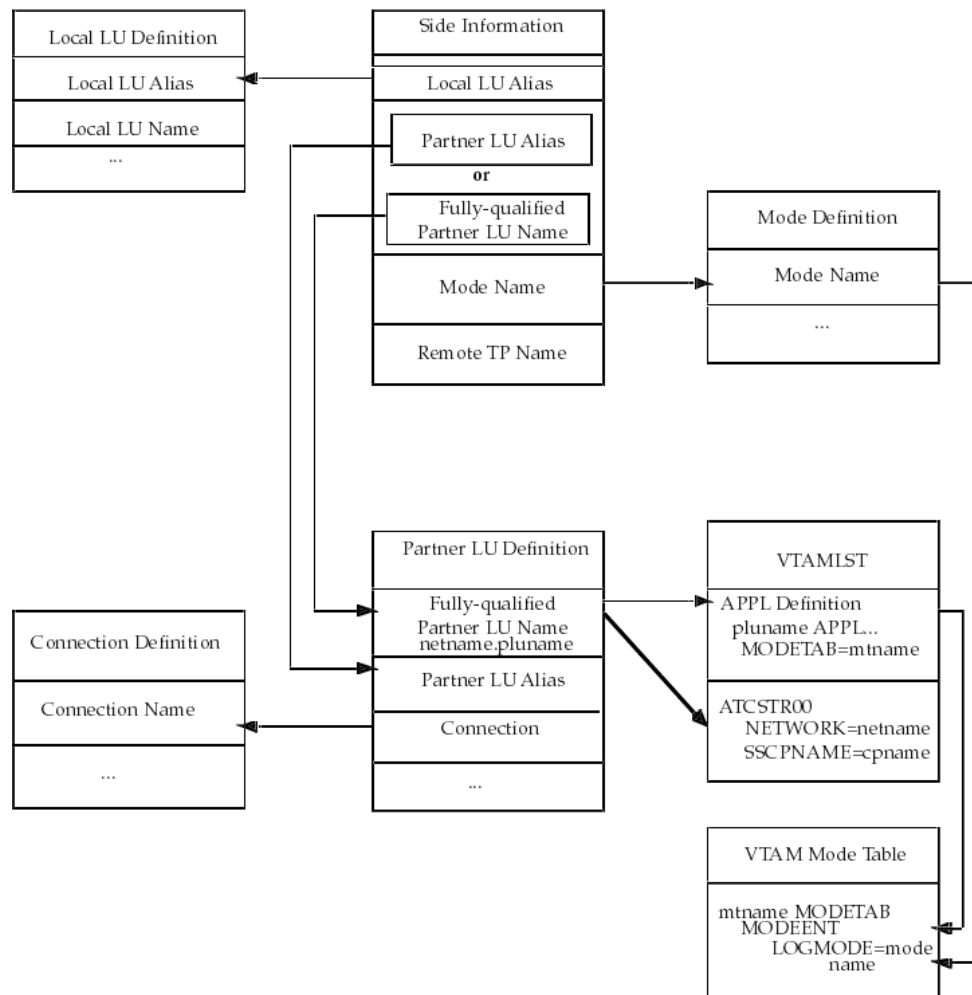
8.8 Testing the Connection

Ensure that your connection is working. Do this by starting the SNAPPlus2 Node and then starting the individual link stations.

Figure 8–1 shows the relationship between SNAPPlus2 definitions and the VTAM definitions on the remote host.

Figure 8–1 Relationship Between SNAPPlus2 Definitions and Remote Host Definitions

HP-UX SNAPPlus2 profiles and host VTAM



8.9 Resuming Gateway Configuration

When you have finished configuring SNAPPlus2 for HP-UX, proceed to [Chapter 11, "Configuring the OLTP"](#) to continue configuring the network.

Configuring the SNA Communication Package on Solaris

Oracle Procedural Gateway for APPC uses the SNA Advanced Program to Program Communication (APPC/LU6.2) protocol to communicate with an OLTP. APPC support on Solaris Operating System (SPARC 64-bit) is provided by the SNAP-IX product.

This chapter describes how to configure SNAP-IX on a Solaris system to run Oracle Procedural Gateway for APPC.

Note: When you finish following the instructions in this chapter, refer to [Chapter 5, "Configuring Your Oracle Network"](#) to continue network configuration.

This chapter contains the following sections:

- [Section 9.1, "Processing Inbound Connections"](#)
- [Section 9.2, "Configuring SNAP-IX Version 6"](#)
- [Section 9.3, "Resuming Gateway Configuration"](#)

9.1 Processing Inbound Connections

Many OLTPs provide options for manipulating the security conduct of an inbound (client) APPC session request. Refer to the appropriate documentation for your OLTP for detailed information about this issue.

Note that for CICS, the gateway provides the following support:

- ATTACHSEC=LOCAL, ATTACHSEC=IDENTIFY, ATTACHSEC=VERIFY, and ATTACHSEC=MIXIDPE are fully supported by the gateway.
- ATTACHSEC=PERSISTENT, specified in the CICS CONNECTION definition, requires capability that is not yet available in the gateway.

9.2 Configuring SNAP-IX Version 6

The following sections describe how to configure SNAP-IX version 6.

9.2.1 Before You Begin

This section requires you to specify parameters that are unique to your system to configure SNAP-IX version 6 properly. Before you begin, request these parameters from your network administrator.

9.2.2 SNAP-IX Configuration Tool

All of the SNAP-IX product configuration is done using the `xsnaadmin` program. This tool is an X-Windows application that provides a graphical interface to view and modify the current SNAP-IX configuration and the current running state of the host SNA node.

9.2.3 Creating SNAP-IX Profiles for the Gateway

Oracle Procedural Gateway for APPC requires a stored set of definitions, called Side Information Profiles, to support connections between the gateway and gateway servers. Each profile consists of a profile name and a profile type, which is a set of fields describing the profile. The fields in a given profile type are generally a mix of operating parameter values and names of other SNA profiles relevant to the profile. Each functional part of APPC, such as the Mode, Remote Transaction Program name, and Logical Unit (LU), is described by a distinct profile type.

9.2.4 Independent Versus Dependent LUs

The gateway configuration can accommodate either independent or dependent LUs. If you choose to use dependent LUs, or are restricted to using dependent LUs, the gateway functions properly. If a dependent LU is correctly defined, then you do not need to make changes to the configuration of the gateway, nor should any changes be needed to the gateway server. However, Oracle recommends that you use independent LUs for the gateway because they support multiple parallel sessions or conversations. This means that multiple Oracle client applications can be active simultaneously with the same gateway server through the independent LU.

In contrast to independent LUs, dependent LUs support only a single active session. The CP (Control Point for the Node) queues each additional conversation request from the gateway behind an already active conversation. In other words, conversations are single-threaded for dependent LUs.

The operational impact of dependent LUs is that the first client application can initiate a conversation through the gateway with the gateway server, but while that session is active (which could be for seconds, minutes, or hours, depending on how the client application and transaction are designed), any other client application initiating a session with the same gateway server appears to hang as it waits behind the previous session.

If a production application really uses only a single conversation at any one time, then there should not be a problem. However, at some point, you might require additional concurrent conversations for testing or for other application development. Having more than one conversation requires that additional dependent LUs be defined on the remote host. Additional configuration entries must be added to SNAP-IX. Additional Side Information Profiles should be defined to use the new dependent LUs. Gateway instances should be created and configured to use these new Side Information Profiles.

9.2.5 Creating SNA Definitions for the Gateway

SNAP-IX definitions are stored in the following files, which are located in the `/etc/opt/sna` directory:

- SNA node definitions: `sna_node.cfg`
- SNA domain definitions: `sna_domn.cfg`

These files are created and maintained with the `xsnaadmin` tool. Maintenance of SNA definitions is usually done by a user with administrative authority. The following information is intended for a user creating SNA definitions for the gateway. You must have some knowledge of SNA before reading this section.

9.2.6 Sample SNAP-IX Definitions

The `$ORACLE_HOME/pg4appc/sna` subdirectory contains a set of sample SNAP-IX definition files for the gateway, which are created with the `xsnaadmin`. These sample files are `sna_domn.cfg` and `sna_node.cfg`. SNA definitions are very specific to the host and SNA network. As such, the sample definitions provided will not work without being tailored for the local host and SNA network.

9.2.7 Configuring SNAP-IX

This section describes the process of creating SNA definitions for SNAP-IX, using `xsnaadmin`. All configuration is done using the various pulldown menus and panels in `xsnaadmin`. The following configuration descriptions follow the samples provided. Please tailor the various SNA values for your local host and SNA network.

9.2.8 Starting `xsnaadmin`

Use the following commands to invoke `xsnaadmin`. The `DISPLAY` environment variable must be set correctly. If you are running `xsnaadmin` from the local console, then `DISPLAY` should already be set. If you are running `xsnaadmin` from a remote X display, then set `DISPLAY` to the host name or IP address of that display.

```
$ DISPLAY=<your_display>:0
$ export DISPLAY
$ xsnaadmin &
```

On startup of `xsnaadmin`, the main screen opens and displays the current configuration of the local SNA node.

9.2.8.1 Configuring the SNA node

To configure the SNA node, you need to do the following:

1. From the Services menu, select **Configure Node Parameters**.
2. In the Node Parameters dialog box, enter the APPN support type, Control Point Name, Control Point Alias, and Node ID as needed. The Control Point Name is composed of the SNA Network Name and the CP name of the local host.
3. Click **OK**.

9.2.8.2 Adding a Port

To add a new port, from the Services menu, select **Connectivity and New Port**.

1. In the Add to *Nodename* dialog box, select the Port type and click **OK**.

2. In the SAP dialog box, enter a Port name and network card number. The Port name is used to logically name the physical network card that you are using and is used to bind a Service Access Port to the card for SNA protocols. Usually, you can accept the values provided in the dialog box. If a different network card is needed, however, enter the card number as reported by the `dmesg` command.
3. Click **OK**.

9.2.8.3 Create a Link Station

When the Port has been defined, you must create a Link Station. The Link Station represents the SNA node of the remote host of the gateway server. But before you create the Link Station, you must create a Remote Node definition as described in the following procedure:

1. From the Services menu, select **APPC and Add Remote Node**.
2. In the dialog box, enter the SNA CPNAME of the remote node and click **OK**.

Now you can create a Link Station as follows:

1. From the Services menu, select **Connectivity and New Link Station**. In the dialog box, select the Port previously defined and click **OK**.
2. In the Link Station dialog box, enter a name for the Link Station, choose the SNA Port name, the type of link activation, and the LU Traffic type. For maximum flexibility, select the **Any** option.
3. For Independent LU traffic, specify the Remote Node name. Click **Remote Node** and select the node you previously created, and then click **OK**. Choose the type of the Remote node, typically a Network node.
4. For Dependent LU traffic, specify the role of the Remote node, typically 'host', the Local Node ID, and optionally, Remote Node ID.
5. Specify the Contact Information. Contact information contains the MAC address of the remote host as well as the SAP number.
6. Click **Advanced** for additional parameters of the Link Station. The Token Ring Parameters dialog box shows additional parameters of the Link Station. These parameters affect initial XID contact and retransmission times and limits. You usually do not need to change the default values.
7. Click **OK**.

9.2.8.4 Creating Local LUs

When the Remote Node definitions have been made, create the Local LU names for the local host as follows:

1. From the Services menu, select **APPC and New Local LU**. In the local LU dialog box, enter the name of the local LU and an alias. This name must correspond to the VTAM definitions on the remote gateway server host for the UNIX host.
2. Click **OK**.

9.2.8.5 Creating Partner LUs

Now define a Partner LU that represents the LU that the gateway server is using to communicate.

1. From the Services menu, select **APPC and New Partner LUs and Partner LU on Remote Node**.

2. In the Partner LU dialog box, enter the Partner LU name and characteristics. The Partner LU name contains the SNA Network Name as well as the LU name of the remote LU. Enable parallel session support. The location is the name as the Remote Node name. You can click **Location** for a list.
3. Click **OK**.

9.2.8.6 Creating Mode and CPI-C Profiles

When the local and remote LU definitions have been created, create the necessary Mode and CPI-C definitions.

1. From the Services menu, select **APPC and Modes**. In the Modes dialog box, click **New** to add a new mode.
2. In the Mode dialog box, enter the Mode Name and other session parameters. The recommended name for a gateway mode is C1CSPGA. Contact your Remote Host system administrator for appropriate mode parameters.
3. Click **OK**.
4. Now that the Mode has been defined, create the CPI-C Side Information Profile, which the gateway uses as a connection name. From the menu, select **APPC and CPI-C**.
5. In the CPI-C destination names dialog box, click **New** to add a new Profile.
6. In the CPI-C destination dialog box, enter the Profile name, Local LU name, Partner TP, Partner LU and mode, and Security option. The partner TP name is the name of the host transaction program or a dummy value to be overridden in the TIP.
7. For the Local LU, you may specify a specific LU or choose the default LU. For the Partner LU, enter either the full LU name or the alias created previously.
8. Enter ORAPLU62 for the mode name. Choose the type of security for these sessions to use. This affects how session authorization is done.
9. Click **OK**.

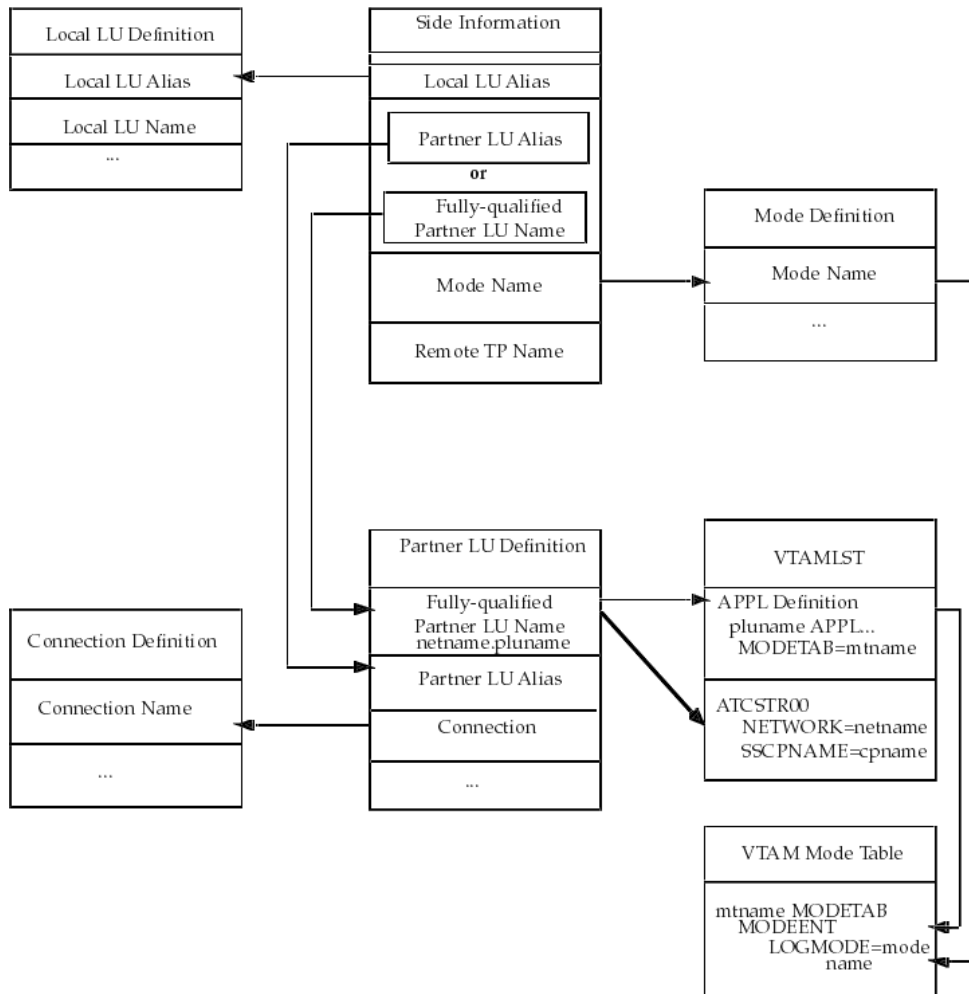
9.2.9 Testing the Connection

Before proceeding with gateway configuration tasks, ensure that your connection is working. Perform this by starting the SNAP-IX Node and then starting the individual link stations.

[Figure 9–1](#) shows the relationship between SNAP-IX definitions and the VTAM definitions on the remote host.

Figure 9-1 Relationship Between SNAP-IX Definitions and Host VTAM Definitions

Solaris SNAP-IX profiles and host VTAM



9.3 Resuming Gateway Configuration

When you have finished configuring the SNA communication package for Solaris, proceed to [Chapter 11, "Configuring the OLTP"](#) to continue configuring the network.

Configuring the TCP/IP Communications Package for IMS Connect

Read this chapter if your gateway uses the TCP/IP communication protocol. It describes how to configure the TCP/IP for IMS Connect feature for the various UNIX platforms supported by the Oracle Procedural Gateway for APPC. TCP/IP is a communication protocol that is already a part of your operating system. No third-party protocol software is required. This chapter contains the following section:

- ["Configuring TCP/IP for IMS Connect on UNIX Platforms"](#) on page 10-1

10.1 Configuring TCP/IP for IMS Connect on UNIX Platforms

The configuration process requires you to enter parameters that are unique to your system to properly configure TCP/IP. Ask your network administrator to provide you with a list of the installation parameters that you require to complete the configuration process.

Basic configuration consists of assigning a host name, an IP address, and a Network Mask to a given network interface. This basic configuration must already be done by your systems administrator. If not, contact your systems administrator to have this configuration completed before you continue.

Additional configuration consists of defining a Name Server IP Address or creating entries in the Hosts file on the local system. Name servers translate host names into IP addresses when queried for a particular host name. The hosts file provides this same functionality, but in a nonnetwork participating manner.

For local configuration (the gateway is on the same system), you might want to use the loopback address. The IP address is 127.0.0.1 and is typically given the local name (*localhost* or *loopback*) in the Hosts file. Using the loopback address reduces the amount of network overhead by handling the traffic internally without actually *talking* to the network.

Note: Oracle Procedural Gateway for APPC using TCP/IP support for IMS Connect has specific initialization file parameters. These parameters are located in `$ORACLE_HOME/pg4appc/admin/initsid.ora`. For information about these parameters, refer to ["Gateway Initialization Parameter File Using TCP/IP"](#) on page B-1.

For more information about configuring TCP/IP for a particular host operating system, refer to the appropriate platform and operating system installation and configuration guides.

When you have finished configuring TCP/IP for IMS Connect for the gateway, proceed to [Chapter 11, "Configuring the OLTP"](#) to complete and verify the configuration process.

Configuring the OLTP

The following sections describe how to configure the online transaction processing OLTP.

- **If your communications protocol is SNA:**
Proceed to [Section 11.1, "Configuring the OLTP for an SNA Environment"](#).
- **If your communications protocol is TCP/IP:**
Proceed to [Section 11.2, "Configuring the OLTP for a TCP/IP Environment"](#).

Note: On a gateway using TCP/IP support for IMS Connect, you must specify `EDIT=ULC` in the IMS TRANSACT macro if you need input case sensitivity. When you are using SNA support, you do not need to specify `EDIT=ULC` in the IMS TRANSACT macro.

11.1 Configuring the OLTP for an SNA Environment

The steps for configuring OLTP to communicate with the Oracle Procedural Gateway for APPC vary depending on which OLTP you are using and on which platform the OLTP is running. CICS Transaction Server for z/OS, IMS/TM, APPC/MVS, IDMS-DC, and z/OS are the currently supported OLTPs. Choose the instructions corresponding to your OLTP from the following sections:

- [Section 11.1.1, "Configuring CICS Transaction Server for z/OS"](#)
- [Section 11.1.2, "Configuring IDMS-DC"](#)
- [Section 11.1.3, "Configuring IMS/TM"](#)
- [Section 11.1.4, "Configuring APPC/MVS"](#)

Note: You need to perform the configuration steps for an OLTP only if this is the first time that you are configuring that OLTP.

11.1.1 Configuring CICS Transaction Server for z/OS

If your OLTP is CICS Transaction Server for z/OS, then perform the following steps to configure it for communication with the gateway:

1. Configure MVS VTAM for the SNA communication package that will make the APPC connection to your system. At least one independent LU must be available to the gateway.
2. Check the VTAM logmode table used by the CICS Transaction Server for z/OS. (The table name is specified in the `MODETAB` parameter in the VTAM APPL

definition for CICS.) Ensure that an entry exists for APPC sessions with parallel session and sync-level support.

The `oraplu62.asm` file in the `$ORACLE_HOME/pg4appc/sna` directory contains a sample mode entry, including comments that indicate the required values in the mode entry.

3. Using a file transfer facility, transfer the following files from the `$ORACLE_HOME/pg4appc/demo/CICS` directory to the z/OS system on which you run CICS Transaction Server for z/OS:
 - `dfhcsdup.jcl`: JCL to run the CICS DFHCS DUP utility
 - `pgaflip.asm`: Assembler source for the CICS FLIP transaction
 - `pgaflip.jcl`: JCL to assemble and linkedit the CICS FLIP transaction
4. Using the comments in the `dfhcsdup.jcl` file, tailor the JCL and input statements to match your system setup, and submit it for batch execution. Performing this step updates your Transaction Server for z/OS system definitions.
5. Using the instructions in the `pgaflip.jcl` file comments, tailor the JCL to match your system setup, and submit it for batch execution. Performing this step assembles and linkedit the `pgaflip.asm` file into a load module library accessible to your Transaction Server for z/OS through the `DFHRPL DD` statement in the CICS startup procedure.
6. Log on to your CICS Transaction Server for z/OS and enter the following transaction:

```
CEDA INSTALL GROUP(ORAPGA)
```

This transaction installs the CICS connection and session definitions for APPC communication with the gateway on UNIX. It also installs definitions for the sample CICS programs and transactions provided with the gateway.

Your CICS Transaction Server for z/OS configuration is now complete.

11.1.2 Configuring IDMS-DC

If your OLTP is IDMS-DC, perform the following steps to configure IDMS-DC and z/OS for communication with the gateway:

1. Configure MVS VTAM for the communication interface APPC connection to UNIX. At least one independent LU must be available for use by the gateway.
2. If your IDMS-DC system does not have APPC support, then set up a separate MVS VTAM APPL definition for use by the IDMS-DC LU6.2 interface. IDMS-DC cannot use the same VTAM APPL for both VTAM terminal and VTAM APPC support.

For more information, refer to vendor documentation.

3. Check the VTAM logmode table used by IDMS-DC. (The table name is specified in the `MODETAB` parameter in the VTAM APPL definition for IDMS-DC.) Ensure that an entry exists for APPC sessions with parallel session and sync level support. The `oraplu62.asm` file in the `$ORACLE_HOME/pg4appc/sna` directory contains a sample mode entry, including comments that indicate the required values in the mode entry.
4. Modify the Mode Profile to be used for your IDMS-DC connections.

To bypass a bug in IDMS-DC, set the Auto ACTIVATIONS Limit field to the same value as the Maximum Number of SESSIONS field. Without this setting, IDMS-DC rejects the first data packet received from UNIX on each conversation, making it impossible for the gateway to communicate with an IDMS-DC transaction.

5. Using *CA-IDMS System Generation* as a guide, perform the following IDMS tasks:
 - a. Enable the IDMS-DC multiple session service manager, RHCCNOS.
 - b. Define a VTAM line for APPC use if one is not already defined.
 - c. Define PTERMs and LTERMs for communications with the UNIX independent LU, as defined to VTAM on z/OS.

Set up two PTERM/LTERM definitions to use the SNASVCMG mode entry for communications between the SNA Service Managers on MVS and UNIX. Set up one of these definitions as a contention winner and set up the other one as a contention loser. Set up additional PTERM/LTERM definitions to use the mode entry defined in the SNA communication package. Profile DEFINE/MODE entry for use by the gateway sessions. One PTERM/LTERM definition is required for each concurrent session with the gateway. Set up these definitions with contention off.

In the `$ORACLE_HOME/pg4appc/demo/IDMS` directory, the `appcdef.doc` file provides sample IDMS-DC definitions for a VTAM line with PTERMs and LTERMs.

- d. Define the IVP and sample programs and transactions to IDMS-DC.

In the `$ORACLE_HOME/pg4appc/demo/IDMS` directory, the `trandef.doc` file provides sample IDMS-DC definitions for the IVP and sample programs and transactions. These definitions should not be modified.

6. Using a file transfer facility, transfer the following files from the `$ORACLE_HOME/pg4appc/demo/IDMS` directory to the z/OS system on which you run IDMS-DC:
 - `pgaf1ip.asm`: The assembler source for the IDMS-DC FLIP program
 - `pgaf1ip.jcl`: JCL to assemble and linkedit the IDMS-DC FLIP program
7. Using the comments in the `pgaf1ip.jcl` file, tailor the JCL to match your system setup and submit it for batch execution. Performing this step assembles and linkedit the `pgaf1ip.asm` file into a load module library accessible to your IDMS-DC system through the CDMSLIB DD statement in the IDMS-DC startup procedure.

The IDMS-DC configuration is now complete.

11.1.3 Configuring IMS/TM

If your OLTP is IMS/TM, then perform the following steps to configure IMS/TM and z/OS for communication with the gateway:

1. Configure your IMS system for the APPC.
2. Configure MVS VTAM for the SNA APPC connection to UNIX. At least one independent LU must be available for use by the gateway, unless you are using the IMS LU6.1 Adapter for LU6.2 applications. In this case, you must have one dependent LU defined for each concurrent session. For example, if you want to support 10 concurrent sessions, then you must have 10 dependent LUs defined.

3. Check the VTAM logmode table used by IMS/TM. The table name is specified by the `MODETAB` parameter in the VTAM APPL definition. For APPC/IMS, ensure that an entry exists for APPC sessions with sync-level support and parallel session support. The `oralu62.asm` and `oraplu62.asm` files in the `$ORACLE_HOME/pg4appc/sna` directory contain sample mode entries for single session and parallel session support, respectively. The samples include comments that indicate the required values in the mode entries.
4. Using your file transfer facility, transfer the following files from the `$ORACLE_HOME/pg4appc/demo/IMS` directory to the z/OS system on which you run IMS/TM:
 - `pgaf1ip.asm` is assembler source for IMS FLIP transaction
 - `pgaf1ip.jcl` is JCL to assemble and linkedit IMS FLIP transaction
 - `imsgen.asm` is IMS stage 1 gen definitions for the IMS FLIP transaction
5. Add the statements in the `imsgen.asm` file to your IMS stage 1 gen and run your IMS stage 1 and stage 2 gens. Use the online change utility to enable the new transaction definition.
6. Using the comments in the `pgaf1ip.jcl` file, tailor the JCL to match your system setup and submit it for batch execution. This assembles and linkedit the `pgaf1ip.asm` file into a load module library that is accessible to your IMS/TM system and creates an PSB and an ACB for the FLIP transaction.
7. Perform the tasks necessary on your system to make the new transaction available to IMS/TM. Depending on your system setup, you might have to restart IMS.

The IMS/TM configuration is now complete.

11.1.4 Configuring APPC/MVS

If your OLTP is APPC/MVS, then perform the following steps to configure APPC/MVS for communication with the gateway:

1. Configure MVS VTAM for the SNA APPC connection to UNIX. At least one independent LU must be available for use by the gateway.
2. Check the VTAM logmode table used by APPC/MVS. (The table name is specified by the `MODETAB` parameter in the VTAM APPL definition for APPC/MVS.) Ensure that an entry exists for APPC sessions with sync level and parallel session support. The `oraplu62.asm` file in the `$ORACLE_HOME/pg4appc/sna` directory contains a sample mode entry, including comments that indicate the required values in the mode entry.
3. Allocate a partitioned data set (PDS) on your z/OS system where the sample files are placed. The PDS should be allocated with `RECFM=FB`, `LRECL=80`, and a `BLKSIZE` appropriate for the device type on which it is located. Approximately two tracks of 3390 disk space are required with one directory block. Oracle suggests naming this partitioned data set (PDS) `ORAPGA.APPCMVS.SAMPLIB`.
4. Using a file transfer facility, transfer the following files from the `$ORACLE_HOME/pg4appc/demo/MVS` directory to the z/OS PDS you allocated in the previous step, using the following specified member names:
 - `pgaf1ip.jcl`: JCL to add an APPC/MVS TP profile and to define the execution environment for the transaction. Store this file in your z/OS PDS as member `PGAFLIPJ`.

- `pgaflip.rex`: The REXX source for the APPC/MVS PGAFLIP transaction. Store this file in your z/OS PDS as member PGAFLIP.
5. Using the comments in the `pgaflip.jcl` file, tailor the JCL to match your system setup and submit it for batch execution. Performing this step defines the APPC/MVS TP profile for the PGAFLIP transaction and stores it in the APPC/MVS profile data set. Ensure that you change the data set name in the JCL to match the name of the z/OS PDS allocated in Step 3.

The APPC/MVS configuration is now complete.

Now that you have completed configuration of the network on a gateway using the SNA protocol, refer to [Chapter 12, "Gateway Configuration Using SNA Communication Protocol"](#). Refer to [Section 12.8, "Configuring Commit-Confirm"](#) for more information on configuring commit-confirm.

11.2 Configuring the OLTP for a TCP/IP Environment

These are the steps for configuring OLTP to communicate with Oracle Procedural Gateway for APPC using TCP/IP for IMS Connect. IMS/TM, through IMS Connect, is the only supported OLTP for this release of the gateway.

Perform the following steps to configure IMS/TM and z/OS for communication with the gateway:

1. Configure your IMS system.
2. Configure IMS Connect

For information on how to configure IMS Connect, refer to the IBM manual, *IMS Connect Guide and Reference*.

3. Using a file transfer facility, transfer the following files from the `$ORACLE_HOME/pg4appc/demo/IMS` directory to the z/OS system on which you run IMS/TM:
 - `pgaflip.asm`: Assembler source for IMS FLIP transaction
 - `pgaflip.jcl`: JCL to assemble and linkedit IMS FLIP transaction
 - `imgen.asm`: IMS stage 1 gen definitions for the IMS FLIP transaction
4. Add the statements in the `imgen.asm` file to your IMS stage 1 gen and run your IMS stage 1 and stage 2 gens. Use the online change utility to enable the new transaction definition.
5. Using the comments in the `pgaflip.jcl` file, tailor the JCL to match your system setup and submit it for batch execution. This assembles and linkedit the `pgaflip.asm` file into a load module library that is accessible to your IMS/TM system and creates an PSB and an ACB for the FLIP transaction.
6. Perform the tasks necessary on your system to make the new transaction available to IMS/TM. Depending on your system setup, you might have to restart IMS.

The IMS/TM configuration is now complete.

- At this point, proceed to [Chapter 13, "Gateway Configuration Using TCP/IP Communication Protocol"](#) to complete configuration of the gateway and its components.

Gateway Configuration Using SNA Communication Protocol

This chapter describes how to configure the Oracle Integrating Server for a gateway using the SNA protocol on your UNIX platform. It also shows you how to configure commit-confirm, should you choose to implement it.

This chapter provides the steps necessary to verify installation and configuration of the gateway components, including optional commit-confirm. It contains the following sections:

- [Section 12.1, "Before You Begin"](#)
- [Section 12.2, "Preparing to Configure a Gateway Installation/Upgrade"](#)
- [Section 12.3, "Integrating Server Configuration: First-Time Gateway Installations"](#)
- [Section 12.4, "Upgrading or Migrating the Oracle Server from Previous Gateways"](#)
- [Section 12.5, "Configuring the Oracle Integrating Server for Gateways to Coexist"](#)
- [Section 12.6, "Optional Configuration Steps to Permit Multiple Users"](#)
- [Section 12.7, "Configuring the Gateway"](#)
- [Section 12.8, "Configuring Commit-Confirm"](#)
- [Section 12.9, "Verifying the Gateway Installation and OLTP Configuration"](#)
- [Section 12.10, "Performing Postinstallation Procedures"](#)

Configuring the Oracle Procedural Gateway for APPC using SNA involves working with the following components:

- Oracle Integrating Server
- UNIX system
- Network
- OLTP

12.1 Before You Begin

This chapter requires you to input parameters unique to your system to properly configure the gateway and SNA communications interface.

Refer to [Appendix E, "Configuration Worksheet"](#) for a worksheet listing the installation parameters that you need to know before you can complete the configuration process. Ask your network administrator to provide you with these unique parameter names before you begin.

12.2 Preparing to Configure a Gateway Installation/Upgrade

There are three ways to establish the gateway-Oracle Integrating Server relationship when you are installing or upgrading or migrating the gateway:

- [When Oracle Integrating Server and the Gateway Are Installed in the Same ORACLE_HOME](#)
- [When Oracle Integrating Server and the Gateway Are Installed on Separate Systems](#)
- [When Oracle Integrating Server and the Gateway Are on the Same System but in Different Directories](#)

Depending on the location of the gateway and the Oracle Integrating Server, you might need to transfer some of the gateway administrative files to the location where Oracle Integrating Server is installed.

Follow the instructions corresponding to your combination of the gateway-Oracle Integrating Server locations listed below.

When Oracle Integrating Server and the Gateway Are Installed in the Same ORACLE_HOME

You do not need to transfer files. Proceed to [Section 12.3, "Integrating Server Configuration: First-Time Gateway Installations"](#) on page 12-3.

When Oracle Integrating Server and the Gateway Are Installed on Separate Systems

You need to perform the following tasks if Oracle Integrating Server and the gateway are installed on separate systems:

1. Locate the gateway administrative files in the gateway `$ORACLE_HOME/pg4appc/admin` directory. All files in this directory that have the `.sql`, `.pkh`, or `.pkb` suffixes must be copied into a similarly-named directory in the Oracle Integrating Server Oracle home directory.
2. Now locate the gateway demo files and subdirectories in the `$ORACLE_HOME/pg4appc/demo` directory of the gateway. Copy the `pgavsn.sql` and `pgaecho.sql` files into a similarly named directory in Oracle Integrating Server.
3. Copy the other subdirectories and files related to your installed OLTP on your remote host. For example, if you have CICS as your only OLTP, then copy the `$ORACLE_HOME/pg4appc/demo/CICS` gateway files into a similarly named directory in the Oracle Integrating Server.

Note: Before transferring the files from the `$ORACLE_HOME/pg4appc/demo` directory, ensure that you have generated your required TIPs. You need to transfer the TIPs as well.

Refer to the *Oracle Procedural Gateway for APPC User's Guide* for information about generating TIPs using Procedural Gateway Administrative Utility (PGAU).

When Oracle Integrating Server and the Gateway Are on the Same System but in Different Directories

You must change your gateway Oracle home to the Oracle home directory of Oracle Integrating Server.

1. For example, if your gateway Oracle home is set as follows:

```
$ echo $ORACLE_HOME
/oracle/pga/10.2
```

and your server Oracle home is located in the `/oracle/pga/10.2` directory, then you need to do the following:

```
$ ORACLE_HOME=/oracle/pga/10.2; export ORACLE_HOME
```

2. Now create the directories with the following commands:

```
$ cd $ORACLE_HOME
$ mkdir pg4appc
$ mkdir pg4appc/admin
```

3. Use whatever file transfer mechanism is available on your system to copy all of the `.sql`, `.pkh`, and `.pkb` files from the gateway Oracle home `$ORACLE_HOME/pg4appc/admin` directory to the Oracle Integrating Server Oracle home `$ORACLE_HOME/pg4appc/admin` directory.
4. You might also transfer the demo files from the gateway directory to the Oracle Integrating Server directory. Copy the files and directory recursively from the gateway Oracle home `$ORACLE_HOME/pg4appc/demo` directory to the Oracle Integrating Server `$ORACLE_HOME/pg4appc/demo` directory.

For example:

```
$ cp -p -R /oracle/pga/10.2/pg4appc/demo $ORACLE_HOME/pg4appc
```

Note: Before transferring the files from the `$ORACLE_HOME/pg4appc/demo` directory, ensure that you have generated your required TIPs. You need to transfer the TIPs as well.

Refer to the *Oracle Procedural Gateway for APPC User's Guide* for information about generating TIPs using PGAU.

If this is a first-time installation, proceed with [Section 12.3, "Integrating Server Configuration: First-Time Gateway Installations"](#).

If this is an upgrade, proceed with [Section 12.4, "Upgrading or Migrating the Oracle Server from Previous Gateways"](#).

Following those steps, you might want to perform the [Section 12.6, "Optional Configuration Steps to Permit Multiple Users"](#).

12.3 Integrating Server Configuration: First-Time Gateway Installations

Follow these steps to configure your Oracle Integrating Server if you have installed Oracle Procedural Gateway for APPC for the first time:

1. Ensure that the UTL_RAW PL/SQL package has been installed on your Oracle Integrating Server. All PGAU-generated TIP specifications use UTL_RAW, which provides routines for manipulating raw data.

- a. Use SQL*Plus to connect to the Oracle Integrating Server as the SYS user.
- b. From SQL*Plus, enter the following command:

```
SQL> DESCRIBE UTL_RAW
```

The DESCRIBE statement produces output on your screen. If you browse through the output, you should see some functions, including a compare function. If you do not see this output, then continue the UTL_RAW installation by performing step d below.

If the DESCRIBE statement indicates success, then your Oracle Integrating Server has UTL_RAW installed and you can proceed to Step 2.

- c. Use SQL*Plus to connect to the Oracle Integrating Server as the SYS user.
- d. From SQL*Plus, run the `utlraw.sql` and `prvtrawb.plb` scripts in the Oracle Integrating Server `$ORACLE_HOME/rdbms/admin` directory, in the following order:

```
SQL> @$ORACLE_HOME/rdbms/admin/utlraw.sql  
SQL> @$ORACLE_HOME/rdbms/admin/prvtrawb.plb
```

2. Ensure that the DBMS_OUTPUT standard PL/SQL package is enabled on Oracle Integrating Server. The sample programs and installation verification programs on the distribution media use this standard package.

- a. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as the SYS user.
- b. From SQL*Plus, enter the following command:

```
SQL> DESCRIBE DBMS_OUTPUT
```

The DESCRIBE statement produces output on your screen. If you browse through that output, you should see some functions, including a `put_line` function.

If you do not see this output, then you must create the DBMS_OUTPUT package. Refer to the *Oracle Database Application Developer's Guide* for more information about creating the DBMS_OUTPUT package. After successful installation of the DBMS_OUTPUT package, issue the DESCRIBE statement.

If the DESCRIBE statement indicates success, then your Oracle Integrating Server has DBMS_OUTPUT created, and you can proceed to Step 3.

3. Install the UTL_PG PL/SQL package. All PGAU-generated TIP specifications use UTL_PG, which provides routines for performing numeric conversions to and from raw data.

- a. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as the SYS user.
- b. From SQL*Plus, run the `utlpg.sql` and `prvtpgb.plb` scripts in the Oracle Integrating Server `$ORACLE_HOME/rdbms/admin` directory, in the following order:

```
SQL> @$ORACLE_HOME/rdbms/admin/utlpg.sql  
SQL> @$ORACLE_HOME/rdbms/admin/prvtpgb.plb
```

4. Install the Heterogeneous Services (HS) catalogs.

- a. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as the SYS user.
- b. Enter the following command:


```
SQL> DESCRIBE HS_FDS_CLASS
```

The DESCRIBE statement produces output on your screen. If the DESCRIBE statement indicates success, then heterogeneous services catalogs have been created on your Oracle Integrating Server and you can proceed to Step 5, otherwise follow the next step only if the DESCRIBE statement does not indicate success. Step c creates the Heterogeneous Services catalog.

- c. If it is necessary to create the Heterogeneous Services catalog, enter the following command:

```
SQL> $ORACLE_HOME/rdbms/admin/caths.sql
```

5. Create a public database link to access Oracle Procedural Gateway for APPC:

Use SQL*Plus to connect to the Oracle Integrating Server as the SYSTEM user. You can use the following SQL*Plus sample whether the Oracle Integrating Server and the gateway are on the same system or on different systems. In the following sample, *pgasrv* is the *tns_name_entry* that will be assigned to the gateway when you modify the *tnsnames.ora* file later in this chapter.

```
SQL> CREATE PUBLIC DATABASE LINK PGA USING 'PGASRV'
```

6. Create the gateway administrator user PGAADMIN and install the PG DD.

- a. Use SQL*Plus to connect to the Oracle Integrating Server as the SYSTEM user.

- b. From SQL*Plus, run the *pgacr8au.sql* script in the *\$ORACLE_HOME/pg4appc/admin* directory. This script creates the PGAADMIN user ID.

The initial password defined for PGAADMIN is PGAADMIN. Use the ALTER USER command to change the password. For more information about password issues, refer to the *Oracle Database SQL Reference*.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgacr8au.sql
```

- c. Use SQL*Plus to connect to the Oracle Integrating Server as user PGAADMIN.

- d. From SQL*Plus, run the *pgddcr8.sql* script in the *\$ORACLE_HOME/pg4appc/admin* directory. This script installs the PG DD.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddcr8.sql
```

- e. From SQL*Plus, connect to the Oracle Integrating Server as the SYS user.

- f. Grant execution privileges on DBMS_PIPE to PGAADMIN:

```
SQL> GRANT EXECUTE ON DBMS_PIPE TO PGAADMIN
```

7. Install the TIP trace access PL/SQL routines. These routines require that the DBMS_PIPE standard PL/SQL package is installed and that PGAADMIN has execute privileges on it. For more information on DBMS_PIPE, refer to the *Oracle Database Application Developer's Guide*.

- a. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as user PGAADMIN.

- b. From SQL*Plus, run the *pgatiptr.sql* script in the *\$ORACLE_HOME/pg4appc/admin* directory. This script creates PL/SQL routines that can be called to read and purge trace information created by PGAU-generated TIP specifications. It also creates public synonyms for these routines. The script prompts you for the necessary user IDs and passwords.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgatiptr.sql
```

8. Install the GPGLOCAL package. This package is required for compilation and execution of all PGAU-generated TIP specifications. TIP developers should be granted execute privileges on GPGLOCAL (refer to Step 3 of "[Optional Configuration Steps to Permit Multiple Users](#)" on page 12-7).
 - a. Use SQL*Plus to connect to the Oracle Integrating Server as user PGAADMIN.
 - b. From SQL*Plus, run the `gpglocal.pkh` script in the `ORACLE_HOME/pg4appc/admin` directory. This script compiles the GPGLOCAL package specification.

```
SQL> @$ORACLE_HOME/pg4appc/admin/gpglocal.pkh
```

- c. From SQL*Plus, run the `gpglocal.pkb` script in the `ORACLE_HOME/pg4appc/admin` directory. This script compiles the GPGLOCAL package body.

```
SQL> @$ORACLE_HOME/pg4appc/admin/gpglocal.pkb
```

12.4 Upgrading or Migrating the Oracle Server from Previous Gateways

Follow these instructions only if you have a previous version of the Oracle Procedural Gateway for APPC installed on your system and need to configure it for Release 10.1.0.2.0 of the gateway.

1. Upgrade your Oracle Procedural Gateway for APPC to current version levels as follows:
 - a. Use SQL*Plus to connect to the Oracle Integrating Server as the SYS user.
 - b. Install the UTL_RAW package body. From SQL*Plus, run the `prvtrawb.plb` script from the `ORACLE_HOME/rdbms/admin` directory. This script upgrades the UTL_RAW package body.

```
SQL> @$ORACLE_HOME/rdbms/admin/prvtrawb.plb
```

- c. Install the UTL_PG package body. From SQL*Plus, run the `prvtpgb.plb` script from the `ORACLE_HOME/rdbms/admin` directory. This script upgrades the UTL_PG package body.

```
SQL> @$ORACLE_HOME/rdbms/admin/prvtpgb.plb
```

The `prvtrawb.plb` and `prvtpgb.plb` scripts should complete successfully. If they fail because specifications do not exist or were invalidated, then consider reinstalling the package specifications as directed in the following section.

12.4.1 If You Must Reinstall Package Specifications

If the UTL_RAW or UTL_PG package has been invalidated or deinstalled, the `prvtrawb.plb` and `prvtpgb.plb` scripts might not complete successfully and you might have to reinstall the package specifications.

If you do reinstall the package specifications, any dependent objects (such as existing user TIPs and client applications) are invalidated and will subsequently need to be recompiled. The impact of this is a one-time performance delay while recompilation of the TIPs and dependent client applications proceeds.

Important: Before proceeding with reinstallation of the package scripts, make sure that you are in the `ORACLE_HOME/pg4appc/admin` directory.

TIPs were split into separate specification and body files in release 3.3 to avoid cascaded recompilations in later releases.

Step 1 Run the Following Scripts before Proceeding with the PGAU Upgrade

From SQL*Plus, run the `utlraw.sql` script:

1. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as the `SYS` user.
2. From SQL*Plus, run the `utlraw.sql` and `utlpg.sql` scripts in Oracle Integrating Server `ORACLE_HOME/rdbms/admin` directory, in the following order, to upgrade their respective package specifications:

```
SQL> @$ORACLE_HOME/rdbms/admin/utlraw.sql
SQL> @$ORACLE_HOME/rdbms/admin/utlpg.sql
```

Step 2 Repeat Installation of UTL_RAW and UTL_PG Package Body

After the scripts have run, repeat Steps 1b and 1c in [Section 12.4, "Upgrading or Migrating the Oracle Server from Previous Gateways"](#) on page 12-6. Then proceed to the section titled ["Upgrading PGAU From Previous Gateway Releases"](#) on page 12-7.

Note: TIPs and dependent client applications must be recompiled after reinstallation of the package specifications. Refer to the "Compiling a TIP" section in Chapter 3 of the *Oracle Procedural Gateway for APPC User's Guide* for information about compiling TIPs.

12.4.2 Upgrading PGAU From Previous Gateway Releases

Upgrade the PG DD as follows before executing the new PGAU:

1. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as user `PGAADMIN`.
2. From SQL*Plus, run the `pgddupgr.sql` script in the `ORACLE_HOME/pg4appc/admin` directory. This script upgrades the PG DD.

12.5 Configuring the Oracle Integrating Server for Gateways to Coexist

You might have an older version of the gateway already installed. Be aware that although a version 10 gateway can communicate with a version 9 data dictionary (PGDD), a version 9 gateway cannot communicate with a version 10 data dictionary. Thus, if you upgrade your data dictionary to a version 10, no gateways which were configured with a version 9 data dictionary will be able to communicate with it.

12.6 Optional Configuration Steps to Permit Multiple Users

The following configuration steps are optional. Perform these steps if you want to allow users other than `PGAADMIN` to perform PG DD operations using PGAU.

1. Create public synonyms for the PG DD to allow other users to access the tables:

- a. Use SQL*Plus to connect to the Oracle Integrating Server as the SYSTEM user.
- b. From SQL*Plus, run the `pgddcr8s.sql` script in the `$ORACLE_HOME/pg4appc/admin` directory. This script creates public synonyms for the PG DD.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddcr8s.sql
```

2. Create roles for accessing the PG DD, performing definitions of transactions, and generating TIP specifications. The PGAADMIN user can grant these roles to other users as necessary.

- a. Use SQL*Plus to connect to the Oracle Integrating Server as user PGAADMIN.
- b. From SQL*Plus, run the `pgddcr8r.sql` script in the `$ORACLE_HOME/pg4appc/admin` directory. This script creates two roles, PGDDDEF and PGDDGEN. The PGDDDEF role provides SELECT, INSERT, UPDATE, and DELETE privileges against some of the PG DD tables, and SELECT privileges against others, and allows execution of the PGAU DEFINE, GENERATE, REDEFINE, REPORT, and UNDEFINE statements. The PGDDGEN role provides select privileges against the PG DD tables, and allows execution of the PGAU GENERATE and REPORT statements only.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddcr8r.sql
```

3. Grant access to PGA required packages.

TIP developers require access to the following PL/SQL packages, which are shipped with the Oracle Integrating Server:

- DBMS_PIPE in the `$ORACLE_HOME/rdbms/admin` directory
- UTL_RAW in the `$ORACLE_HOME/rdbms/admin` directory
- UTL_PG in the `$ORACLE_HOME/rdbms/admin` directory

Explicit grants to execute these packages must be made to TIP developers.

These grants can be private as in the following example:

```
$ sqlplus SYS/pw@database_specification_string
SQL> GRANT EXECUTE ON UTL_RAW TO tip_developer;
SQL> GRANT EXECUTE ON UTL_PG TO tip_developer;
SQL> GRANT EXECUTE ON DBMS_PIPE TO tip_developer;
SQL> CONNECT PGAADMIN/pw@database_specification_string
SQL> GRANT EXECUTE ON PGAADMIN.PURGE_TRACE TO tip_developer;
SQL> GRANT EXECUTE ON PGAADMIN.READ_TRACE TO tip_developer;
SQL> GRANT EXECUTE ON PGAADMIN.GPGLOCAL TO tip_developer;
SQL> exit
```

Alternatively, these grants can be public as in the following example:

```
$ sqlplus SYS/pw@database_specification_string
SQL> GRANT EXECUTE ON UTL_RAW TO PUBLIC;
SQL> GRANT EXECUTE ON UTL_PG TO PUBLIC;
SQL> GRANT EXECUTE ON DBMS_PIPE TO PUBLIC;
SQL> CONNECT PGAADMIN/pw@database_specification_string
SQL> GRANT EXECUTE ON PGAADMIN.PURGE_TRACE TO PUBLIC;
SQL> GRANT EXECUTE ON PGAADMIN.READ_TRACE TO PUBLIC;
SQL> GRANT EXECUTE ON PGAADMIN.GPGLOCAL TO PUBLIC;
SQL> EXIT
```

You can use either private or public grants. Both are sufficient for using PGA. Public grants are easier and can be performed now. If you use private grants, then they must be issued each time a new TIP developer user ID is created.

SQL scripts for performing these grants are provided in the `$ORACLE_HOME/pg4appc/admin` directory. The `pgddapub.sql` script performs these grants for public access to the packages. The `pgddadev.sql` script performs the grants for private access to the packages by a single TIP developer. If you are going to use private grants, then you must run the `pgddadev.sql` script once for each TIP developer user ID:

- a. Use SQL*Plus to connect to the Oracle Integrating Server as user PGAADMIN.
- b. From SQL*Plus, run the appropriate script (`pgddapub.sql` or `pgddadev.sql`) from the `$ORACLE_HOME/pg4appc/admin` directory. The script performs the necessary grants as described earlier. You are prompted for the required user IDs, passwords, and database specification strings.

If you are using private grants, then repeat this step for each user ID requiring access to the packages.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddapub.sql
SQL> @$ORACLE_HOME/pg4appc/admin/pgddadev.sql
```

4. If you are upgrading from a previous release of the gateway, and if you want to upgrade your existing TIPs with new function and maintenance, then regenerate existing TIP specifications using the PGAU GENERATE statement.

Note: The PGAU has been enhanced to automatically upgrade existing PG DD entries with a new attribute when a PGAU GENERATE command is executed. To support this enhancement, add a new privilege to the PGDDGEN role. To do this, as the PGAADMIN user, use SQL*Plus to connect to the Oracle Integrating Server where the PG DD is stored. Then issue the following SQL command:

```
SQL> GRANT INSERT ON PGA_DATA_VALUES TO PGDDGEN
```

- a. Invoke PGAU in the directory path where the PGAU control files are generated and where TIPs are stored:

```
$ pgau
PGAU> CONNECT PGAADMIN/pgaadmin@database_specification_string
PGAU> GENERATE tranname
PGAU> EXIT
```

For more information about the GENERATE command, refer to the PGAU GENERATE command section in Chapter 2, of the *Oracle Procedural Gateway for APPC User's Guide*.

Note that it is not necessary to define the PG DD entries again.

5. Invoke SQL*Plus in the same directory path where the newly-generated TIP specifications are stored.

```
$ sqlplus tip_owner/pw@database_specification_string
SQL> @tipname.pkh
SQL> @tipname.pkb
SQL> exit
```

PGAU GENERATE produces the TIP in two output files: a specification and a body. You must compile both, first the specification and then the body.

For more information about the GENERATE command, refer to the PGAU GENERATE command section in Chapter 2, of the *Oracle Procedural Gateway for APPC User's Guide*.

12.7 Configuring the Gateway

To configure the gateway, perform the following steps:

1. Tailor the Oracle Procedural Gateway for APPC parameters.

Parameters specific to the gateway are supplied in the gateway parameter file, `initsid.ora`, which is in the `$ORACLE_HOME/pg4appc/admin` directory. A sample gateway parameter file, `initPGA.ora` is provided in this subdirectory.

Note: In the `initsid.ora` file, substitute your `pg4appc` SID name for "`sid`" in this file name.

The `initsid.ora` file contains both APPC and TCP/IP parameters, separated by a description. You must modify the `initsid.ora` file by deleting the TCP/IP parameters. Refer to [Appendix A, "Gateway Initialization Parameters for SNA Protocol"](#) for the valid APPC parameters.

The parameters fall into two categories:

- Gateway initialization parameters

These parameters control the general operation of the gateway in the Oracle environment.

Important: Before performing the following step, refer to [Appendix A, "Gateway Initialization Parameters for SNA Protocol"](#) for information about tailoring gateway initialization and PGA parameters. Pay special attention to the information about using the `PGA_CAPABILITY` parameter.

- PGA parameters

PGA parameters control the APPC interface portion of the gateway. Use the `SET` gateway initialization parameter to specify PGA parameters. Oracle recommends that you group all `SET` commands for PGA parameters at the end of the `initsid.ora` file.

Note: Misspelled parameters are ignored. However, if the `$ORACLE_HOME/pg4appc/admin/initsid.ora` file is missing, all calls to the gateway fail and return a PGA-20928 error.

12.8 Configuring Commit-Confirm

Note: If you are planning to implement commit-confirm, read the detailed explanation of commit-confirm's capabilities in Chapter 5 of the *Oracle Procedural Gateway for APPC User's Guide*, "Implementing Commit-Confirm (SNA Only) before proceeding."

Follow these steps to configure the commit-confirm components. The steps for configuring commit-confirm include:

- Configuring the Oracle Integrating Server where the gateway server will store its transaction log information
- Configuring the gateway initialization parameters
- Configuring the OLTP

All of these steps must be performed before attempting to use any applications that use commit-confirm.

12.8.1 Configuring the Oracle Integrating Server for Commit-Confirm

The Oracle Integrating Server installation where the gateway server will store its transaction log information should ideally be on the same system where the gateway runs. The configuration of the server consists of creating the gateway DBA user, creating the commit-confirm log tables and creating the PL/SQL stored procedure used by the gateway server for logging transactions.

The `pgaccou.sql` script from the `$ORACLE_HOME/pg4appc/admin` directory creates the gateway DBA user ID. The default user ID is `PGADBA` with the initial password set to `PGADBA`. If you want to change the user ID or initial password, you must modify the script.

1. Use SQL*Plus to connect to the Oracle Integrating Server as the `SYSTEM` user.
2. From SQL*Plus, run the `pgaccou.sql` script from the `$ORACLE_HOME/pg4appc/admin` directory. This script creates the gateway DBA user ID. If you want to change the password at any time after running this script, you can use the `ALTER USER` command to change the password. For further information, refer to the *Oracle Database SQL Reference*.
3. Use SQL*Plus to connect to Oracle Integrating Server as the `PGADBA` user.
4. From SQL*Plus, run the `pgaccpnd.sql` script from the `$ORACLE_HOME/pg4appc/admin` directory. This script creates the `PGA_CC_PENDING` table used by the gateway server for its commit-confirm transaction log.
5. From SQL*Plus, run the `pgacclog.sql` script from the `$ORACLE_HOME/pg4appc/admin` directory. This script creates the `PGA_CC_LOG` PL/SQL stored procedure used by the gateway server for updating the `PGA_CC_PENDING` table.
6. Disconnect from Oracle Integrating Server.

12.8.2 Configuring Gateway Initialization Parameters for Commit-Confirm

The gateway initialization parameters are described in [Appendix A, "Gateway Initialization Parameters for SNA Protocol"](#). The parameters necessary for commit-confirm support in the gateway are:

- PGA_CAPABILITY
- PGA_LOG_DB
- PGA_LOG_USER
- PGA_LOG_PASS
- PGA_RECOVERY_USER
- PGA_RECOVERY_PASS
- PGA_RECOVERY_TPNAME

These parameters should be added to your `initsid.ora` file, where `sid` is the gateway SID for your commit-confirm gateway.

12.8.3 Configuring the OLTP for Commit-Confirm

Configuration of the OLTP includes defining and installing the following:

- Commit-confirm transaction log database
- Commit-confirm forget or recovery transaction
- Sample commit-confirm applications provided with the gateway

Note: A restart of the OLTP may be necessary to implement the changes required for commit-confirm support. You should plan for this with your OLTP system administrator.

Detailed instructions for configuring the Transaction Server for z/OS and IMS/TM are provided in the `$ORACLE_HOME/pg4appc/demo/CICS/README.doc` and `$ORACLE_HOME/pg4appc/demo/IMS/README.doc` files, respectively.

Refer to "Implementing Commit-Confirm (SNA Only)" in the *Oracle Procedural Gateway for APPC User's Guide* for detailed information about commit-confirm. You will take steps to verify configuration of commit-confirm later in this chapter, in [Section 12.9.3, "Verifying OLTP Configuration for Commit-Confirm"](#) on page 12-16.

12.9 Verifying the Gateway Installation and OLTP Configuration

To verify the gateway installation and the OLTP configuration, perform the following procedures after installing Oracle Procedural Gateway for APPC. In addition, if you chose to configure commit-confirm, follow the steps to verify the OLTP configuration for commit-confirm.

Note: If your database link name is not PGA, modify the demonstration .sql files to give them the particular database link name that you created in Step 5 of [Section 12.3, "Integrating Server Configuration: First-Time Gateway Installations"](#). You must modify the following .sql files:

- pgavsn.sql
 - pgaecho.sql
 - pgacics.sql
 - pgaidms.sql
 - pgaims.sql
 - pgamvs.sql
-
-

12.9.1 Verifying the Gateway Installation

To verify the gateway software installation using the database link PGA previously created, perform the following steps:

1. Using SQL*Plus, connect to your Oracle Integrating Server as PGAADMIN.
2. Run `$ORACLE_HOME/pg4appc/demo/pgavsn.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/pgavsn.sql
```

The server version number banner appears at your terminal. The following output appears:

```
Oracle Procedural Gateway for APPC.  
Version 10.2.0.1.0 Thu Feb 13 13:12:49 2005
```

```
Copyright (c) Oracle Corporation 1979,  
2005. All rights reserved.
```

```
PL/SQL procedure successfully completed.
```

3. Run `$ORACLE_HOME/pg4appc/demo/pgaecho.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/pgaecho.sql
```

You will receive the following output:

```
==> Congratulations, your installation was successful. <==
```

12.9.2 Verifying the OLTP Configuration

The procedure for verifying your OLTP configuration varies, depending on which OLTP you are using and depending upon which platform the OLTP is running on. CICS Transaction Server for z/OS, IMS/TM, APPC/MVS, IDMS-DC, and z/OS are the currently supported OLTPs. Follow the instructions in the following sections for verifying installation:

- [Section 12.9.2.1, "CICS Verification"](#)
- [Section 12.9.2.2, "IDMS-DC Verification"](#)
- [Section 12.9.2.3, "IMS/TM Verification"](#)
- [Section 12.9.2.4, "APPC/MVS Verification"](#)

Attention: If you have not completed the file transfers detailed in [Section 12.2, "Preparing to Configure a Gateway Installation/Upgrade"](#), complete them now, before proceeding to the next step.

12.9.2.1 CICS Verification

If your OLTP is CICS Transaction Server for z/OS, perform the following steps to verify the CICS configuration.

1. To verify that the FLIP transaction is installed correctly, log on to your CICS Transaction Server for z/OS and enter the following transaction, replacing `FLIP` with the transaction ID you chose for `FLIP` when you configure your CICS Transaction Server for z/OS for the gateway:

```
FLIP THIS MESSAGE
```

The following output appears:

```
EGASSEM SIHT PILF
```

2. Log on to UNIX.
3. Modify the `pgacics.sql` file, which is located at `$ORACLE_HOME/pg4appc/demo/CICS/pgacics.sql`. Customize the following three items used for accessing the gateway and the CICS Transaction Server for z/OS as described in the comments at the beginning of the file:

- CICS transaction ID
- Side Profile name
- Logmode entry name

4. Ensure that the SNA communication package on your system has been started.
5. Log on to your CICS Transaction Server for z/OS and run this transaction, where *name* is the name of the `CONNECTION` definition installed by the `DFHCSDUP` job run in the CICS configuration steps:

```
CEMT SET CONNECTION(name) ACQUIRED
```

This transaction activates the CICS connection to UNIX.

6. Use `SQL*Plus` to connect to your Oracle Integrating Server as `PGAADMIN`.
7. Run `pgacics.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/CICS/pgacics.sql
```

The following message appears:

```
==> Congratulations, your gateway is communicating with CICS <==
```

Your CICS Transaction Server for z/OS installation verification is complete.

12.9.2.2 IDMS-DC Verification

If your OLTP is IDMS-DC, perform the following steps to verify the IDMS-DC configuration:

1. Log on to UNIX.

2. Modify the `pgaidms.sql` file, which is located at `$ORACLE_HOME/pg4appc/demo/IDMS/pgaidms.sql`. Customize the following three items used for accessing the gateway and the IDMS-DC system as described in the comments at the beginning of the file:

- IDMS-DC transaction ID
- Side Profile Name
- Logmode entry name

3. Ensure that the SNA communication package on your system has been started.
4. Using the IDMS-DC DCMT transaction, display the LU6.2 line to check that sessions have been started with your system. Enter the following command, where `linename` is the name of the LINE defined for LU6.2 communications with your system:

```
DCMT DIS LINE linename
```

Each defined LTERM/PTERM should show a status of INSRV. Any other status indicates a problem with the IDMS-DC APPC interface

5. Use SQL*Plus to connect to Oracle Integrating Server as PGAADMIN.
6. Run `pgaidms.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/IDMS/pgaidms.sql
```

The following message appears:

```
==> Congratulations, your gateway is communicating with IDMS-DC<==
```

Your IDMS-DC installation verification is now complete.

12.9.2.3 IMS/TM Verification

If your OLTP is IMS/TM, then perform the following steps to verify the IMS/TM configuration:

1. To verify that the FLIP transaction is installed correctly, log on to your IMS/TM system and enter the following transaction, replacing `FLIP` with the transaction ID you chose for `FLIP` when you configured your IMS/TM system for the gateway:

```
FLIP THIS MESSAGE
```

The following output appears on your terminal:

```
EGASSEM SIHT PILF
```

2. Log on to UNIX.
3. Modify the `pga ims.sql` file, which is located at `$ORACLE_HOME/pg4appc/demo/IMS/pga ims.sql`. Customize the following three items used for accessing the gateway and the IMS/TM system as described in the comments at the beginning of the file:
 - IMS/TM transaction ID
 - Side Profile Name
 - Logmode entry name
4. Ensure that the SNA communication package on your system has been started.
5. Use SQL*Plus to connect to Oracle Integrating Server as PGAADMIN.

6. Run `pgaims.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/IMS/pgaims.sql
```

The following message appears:

```
==> Congratulations, your gateway is communicating with IMS/TM <==
```

Your IMS/TM installation verification is now complete.

12.9.2.4 APPC/MVS Verification

If your OLTP is APPC/MVS, perform the following steps to verify the APPC/MVS configuration:

1. Verify that your APPC/MVS subsystem is active.
2. Log on to the UNIX system
3. Modify the `pgamvs.sql` file, which is located at `ORACLE_HOME/pg4appc/demo/MVS/pgamvs.sql`. Customize the following three items used for accessing the gateway and the APPC/MVS system as described in the comments at the beginning of the file:
 - APPC/MVS transaction ID
 - Side Profile Name
 - Logmode entry name
4. Ensure that the SNA communication package on your system has been started.
5. Use SQL*Plus to connect to your Oracle Integrating Server as `PGAADMIN`.
6. Run `pgamvs.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/MVS/pgamvs.sql
```

The following message appears:

```
=> Congratulations, your gateway is communicating with APPC/MVS <=
```

Your APPC/MVS installation verification is now complete.

12.9.3 Verifying OLTP Configuration for Commit-Confirm

If you chose to configure commit-confirm in [Section 12.8, "Configuring Commit-Confirm"](#), the following section will assist you in verifying the configuration.

Note: Refer to Chapter 5, "Implementing Commit-Confirm" in the *Oracle Procedural Gateway for APPC User's Guide* for background information on the components and capabilities of commit-confirm.

Samples are provided with the gateway for Transaction Server for z/OS and IMS/TM for implementing commit-confirm support. They are in the following directories, respectively: `ORACLE_HOME/pg4appc/demo/CICS` and `ORACLE_HOME/pg4appc/demo/IMS`. A `README.doc` file in each directory provides detailed information about installing and using the samples. JCL files for compiling and linking the sample programs are provided as well. The samples included with the gateway assist you with the following:

- Creating and initializing the commit-confirm transaction log databases and defining those databases to the OLTP

For Transaction Server for z/OS, the sample uses a VSAM file for the log database. For IMS/TM, a SHISAM/VSAM database is used.

- Using subroutines for receiving the Oracle Global Transaction ID from the gateway and logging it into the commit-confirm transaction log database

These subroutines are provided in the `pgacc1g.asm` files. They can be used in your applications to reduce the complexity of the code changes to your programs. For Transaction Server for z/OS, the subroutine provided is called using the EXEC CICS LINK interface. For IMS/TM, the subroutine provided is called using the standard CALL statement or its equivalent in the application programming language. Both of these subroutines are written in 370 assembler to eliminate any interlanguage interface complexities and compiler dependencies.

- Forget and recovery transactions

These are provided in the `pgareco.asm` files. Forget and recovery transactions must be installed into your OLTP and accessible through APPC so that the gateway can invoke them to forget a transaction once it has been successfully committed, and to query transaction state during recovery processing. These transactions delete the entry for a particular Oracle Global Transaction ID from the OLTP commit-confirm transaction log database during forget processing, and query the entry for a particular Oracle Global Transaction ID from the OLTP commit-confirm transaction log database during recovery processing. For both Transaction Server for z/OS and IMS/TM, these transactions are written in 370 assembler.

- Using the sample commit-confirm transaction log databases and subroutines

For Transaction Server for z/OS, a sample DB2 update transaction, `DB2C`, is provided in the `pgadb2c.cob` file. This is a COBOL example that updates the DB2 sample `EMP` table. For IMS/TM, a sample DLI update transaction, `PGAIMSU`, is provided in the `pgaimesu.cob` file. This is a COBOL example that updates the DLI sample `PARTS` database.

12.10 Performing Postinstallation Procedures

The following are optional steps that you can perform as necessary. Installation of the sample applications for your OLTP is recommended to help you to fully understand how the gateway works and how it interfaces with your OLTP.

12.10.1 Installing Sample Applications

Oracle Procedural Gateway for APPC package contains the following sample PL/SQL procedures and OLTP transaction programs that demonstrate the capabilities of the gateway.

APPC/MVS

- z/OS data set information

CICS Transaction Server for z/OS

- ADABAS inquiry
- DB2 inquiry
- DB2 multi-row inquiry

- DB2 update
- VSAM inquiry
- VSAM update
- DLI inquiry
- FEPI DB2 inquiry
- FEPI VSAM inquiry
- IDMS-DC
- IDMS/R inquiry

IMS/TM

- IMS inquiry using IVTNO and IVTNV sample transactions
- IMS PARTS inquiry (CPI-C)
- IMS PARTS update (CPI-C)

Additional samples are added to the distribution media in later releases of the product. Wherever possible, sample applications use the sample databases provided with the database products.

For this release, full documentation on installing and using the sample applications is available in the `README.doc` files in the following directories:

- `$ORACLE_HOME/pg4appc/demo`
- `$ORACLE_HOME/pg4appc/CICS`
- `$ORACLE_HOME/pg4appc/IMS`
- `$ORACLE_HOME/pg4appc/MVS`
- `$ORACLE_HOME/pg4appc/IDMS`

Gateway Configuration Using TCP/IP Communication Protocol

This chapter describes how to configure Oracle Integrating Server for a gateway using TCP/IP for IMS Connect on your UNIX platform. It also provides the steps necessary to verify installation and configuration of the gateway and OLTP components. This chapter contains the following sections:

- [Section 13.1, "Before You Begin"](#)
- [Section 13.2, "Preparing to Configure a Gateway Installation/Upgrade"](#)
- [Section 13.3, "Configuring Oracle Integrating Server: First-Time Installation"](#)
- [Section 13.5, "Optional Configuration Steps to Permit Multiple Users"](#)
- [Section 13.6, "Configuring TCP/IP for the Gateway"](#)
- [Section 13.7, "Configuring the Gateway"](#)
- [Section 13.8, "Loading the PGA_TCP_IMSC Table"](#)
- [Section 13.9, "Verifying the Gateway Installation and OLTP Configuration"](#)
- [Section 13.10, "Performing Postinstallation Procedures"](#)

Configuring Oracle Procedural Gateway for APPC using TCP/IP support for IMS Connect involves working with the following components:

- Oracle Integrating Server
- UNIX system
- Network
- OLTP

13.1 Before You Begin

This chapter requires you to input parameters unique to your system to correctly configure the gateway and TCP/IP communications interface.

Refer to [Appendix E, "Configuration Worksheet"](#) for a worksheet listing the installation parameters that you will need to know before you can complete the configuration process. Ask your network administrator to provide you with these unique parameter names before you begin.

13.2 Preparing to Configure a Gateway Installation/Upgrade

There are three ways to establish the gateway and Oracle Integrating Server relationship when you are installing, upgrading, or migrating the gateway:

- [When Oracle Integrating Server and the Gateway Are Installed in the Same ORACLE_HOME](#)
- [When Oracle Integrating Server and the Gateway Are Installed on Separate Systems](#)
- [When Oracle Integrating Server and the Gateway Are on the Same System but in Different Directories](#)

Depending on the location of your gateway and your Oracle Integrating Server, you might need to transfer some of the gateway administrative files to the location where Oracle Integrating Server is installed.

Follow the instructions applicable to your combination of the gateway-Oracle Integrating Server locations listed below.

When Oracle Integrating Server and the Gateway Are Installed in the Same ORACLE_HOME

You do not need to transfer files. Proceed to [Configuring Oracle Integrating Server: First-Time Installation](#) on page 13-3.

When Oracle Integrating Server and the Gateway Are Installed on Separate Systems

When Oracle Integrating Server and the gateway are installed on separate systems, you need to perform the following tasks:

1. Locate the gateway administrative files in the gateway `$ORACLE_HOME/pg4appc/admin` directory. All files in this directory that have the suffix `.sql`, `.pkh`, and `.pkb` must be copied into a similarly named directory in the Oracle Integrating Server Oracle home directory.
2. Now, locate the gateway demo files and subdirectories in the gateway `$ORACLE_HOME/pg4appc/demo` directory. Copy the `pgavsn.sql` and `pgaecho.sql` files into a similarly named directory on Oracle Integrating Server.
3. Copy the `pgaaims.sql` file from the gateway Oracle home `$ORACLE_HOME/pg4appc/demo/IMS` directory to the Oracle Integrating Server Oracle home `$ORACLE_HOME/pg4appc/demo/IMS` directory.

Optional Steps: If you want to run IVTNV and IVTNO, then you will need to copy the `ivtnoctl`, `ivtnod.sql`, `ivtnvctl`, and `ivtnvd.sql` files into the Oracle Integrating Server Oracle home at `$ORACLE_HOME/pg4appc/demo/IMS` directory as well. Ensure that you generate the required TIPs and transfer them as well.

When Oracle Integrating Server and the Gateway Are on the Same System but in Different Directories

You must change your gateway Oracle home to the Oracle Integrating Server Oracle home directory.

1. For example, if your gateway Oracle home is set as follows:

```
$ echo $ORACLE_HOME
/oracle/pga/10.2
```


and the server Oracle home is located in the `/oracle/pga/10.2` directory, then you need to do the following:

```
$ ORACLE_HOME=/oracle/pga/10.2; export ORACLE_HOME
```

- Now, create the directories with the following commands:

```
$ cd $ORACLE_HOME
$ mkdir pg4appc
$ mkdir pg4appc/admin
$ mkdir pg4appc/demo
$ mkdir pg4appc/demo/IMS
```

- Use whatever file transfer mechanism is available on your system to copy all of the `.sql`, `.pkh`, and `.pkb` files from the gateway Oracle home `$ORACLE_HOME/pg4appc/admin` directory to the Oracle Integrating Server Oracle home `$ORACLE_HOME/pg4appc/admin` directory.
- You might also transfer the demo files from the gateway directory to the Oracle Integrating Server directory. Copy the `pgavsn.sql` and `pgaecho.sql` files and directory recursively from the gateway Oracle home `$ORACLE_HOME/pg4appc/demo` directory to the Oracle Integrating Server `$ORACLE_HOME/pg4appc/demo` directory.
- You might also copy the `pgaims.sql` file from the gateway Oracle home `$ORACLE_HOME/pg4appc/demo/IMS` directory to the Oracle Integrating Server Oracle home `$ORACLE_HOME/pg4appc/demo/IMS` directory.

Optional Steps: If you want to run IVTNV and IVTNO, then you will need to copy the `ivtnoctl`, `ivtnod.sql`, `ivtnvctl`, and `ivtnvd.sql` files into the Oracle Integrating Server Oracle home `$ORACLE_HOME/pg4appc/demo/IMS` directory as well. Ensure that you generate the required TIPs and transfer them as well.

Proceed with [Configuring Oracle Integrating Server: First-Time Installation](#) on page 13-3. Following those steps, you may want to perform the [Optional Configuration Steps to Permit Multiple Users](#), on page 13-8.

13.3 Configuring Oracle Integrating Server: First-Time Installation

Follow these steps to configure your Oracle Integrating Server after installing the Oracle Procedural Gateway for APPC.

- Ensure that the UTL_RAW PL/SQL package has been installed on Oracle Integrating Server. All TIP specifications generated by Procedural Gateway Administrative Utility (PGAU) use UTL_RAW, which provides routines for manipulating raw data.
 - Use SQL*Plus to connect to Oracle Integrating Server as the SYS user.
 - From SQL*Plus, enter the following command:

```
SQL> DESCRIBE UTL_RAW
```

The DESCRIBE statement produces output on your screen. If you browse through the output, then you should see some functions, including a compare function. If you do not see this output, then continue the UTL_RAW installation by performing Step d below.

If the DESCRIBE statement indicates success, then Oracle Integrating Server has UTL_RAW installed and you can proceed to Step 2.

- c. Use SQL*Plus to connect to Oracle Integrating Server as *SYS*.
- d. From SQL*Plus, run the `utlraw.sql` and `prvtrawb.plb` scripts in the Oracle Integrating Server `$ORACLE_HOME/rdbms/admin` directory in the following order:

```
SQL> @$ORACLE_HOME/rdbms/admin/utlraw.sql
SQL> @$ORACLE_HOME/rdbms/admin/prvtrawb.sql
```

2. Ensure that the `DBMS_OUTPUT` standard PL/SQL package is enabled on Oracle Integrating Server. The sample programs and installation verification programs on the distribution media use this standard package.

- a. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as the *SYS* user.

- b. Enter the following command:

```
SQL> DESCRIBE DBMS_OUTPUT
```

The `DESCRIBE` statement produces output on screen. If you browse through that output, then you should see some functions, including a `put_line` function.

If you do not see this output, then you must create the `DBMS_OUTPUT` package. Refer to the *Oracle Database Application Developer's Guide* for more information about creating the `DBMS_OUTPUT` package. After installing the `DBMS_OUTPUT` package successfully, run the `DESCRIBE` statement.

If the `DESCRIBE` statement indicates success, then Oracle Integrating Server has `DBMS_OUTPUT` created and you can proceed to Step 3.

3. Install the `UTL_PG` PL/SQL package. All PGAU-generated TIP specifications use `UTL_PG`, which provides routines for performing numeric conversions to and from raw data.

- a. If necessary, use SQL*Plus to connect to Oracle Integrating Server as the *SYS* user.

- b. From SQL*Plus, run the `utlpg.sql` and `prvtpgb.plb` scripts in the Oracle Integrating Server `$ORACLE_HOME/rdbms/admin` directory in the following order:

```
SQL> @$ORACLE_HOME/rdbms/admin/utlpg.sql
SQL> @$ORACLE_HOME/rdbms/admin/prvtpgb.plb
```

4. Install the Heterogeneous Services (HS) catalogs.

- a. If necessary, use SQL*Plus to connect to Oracle Integrating Server as the *SYS* user.

- b. Enter the following command:

```
SQL> DESCRIBE HS_FDS_CLASS
```

The `DESCRIBE` statement produces output on your screen. If the `DESCRIBE` statement indicates success, then heterogeneous services catalogs have been created on Oracle Integrating Server and you can proceed to Step 5.

If the `DESCRIBE` statement does not indicate success, then you must create Heterogeneous Services catalogs, and you must follow Step c below:

- c. If it is necessary to create the Heterogeneous Services catalog, then enter the following command:

```
SQL> @$ORACLE_HOME/rdbms/admin/caths.sql
```

5. Create a public database link to access the Oracle Procedural Gateway for APPC:

Use SQL*Plus to connect to the Oracle Integrating Server as the SYSTEM user. You can use the following SQL*Plus sample whether the Oracle Integrating Server and the gateway are on the same system or on different systems. In the following sample, `pgasrv` is the `tns_name_entry` that will be assigned to the gateway when you modify the `tnsnames.ora` file later in this chapter.

```
SQL> CREATE PUBLIC DATABASE LINK PGA USING 'PGASRV'
```

6. Create the gateway administrator user (PGAADMIN) and install the PG DD.

a. Use SQL*Plus to connect to Oracle Integrating Server as the SYSTEM user.

b. From SQL*Plus, run the `pgacr8au.sql` script in the `ORACLE_HOME/pg4appc/admin` directory. This script creates the PGAADMIN user ID.

The initial password defined for PGAADMIN is PGAADMIN. Use the ALTER USER command to change the password. For further information about password issues, refer to the *Oracle Database SQL Reference*.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgacr8au.sql
```

c. Use SQL*Plus to connect to the Oracle Integrating Server as PGAADMIN.

d. From SQL*Plus, run the `pgddcr8.sql` script in the `ORACLE_HOME/pg4appc/admin` directory. This script installs the PG DD.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddcr8.sql
```

e. From SQL*Plus, connect to the Oracle Integrating Server as the SYS user.

f. Grant execute privileges on DBMS_PIPE to PGAADMIN:

```
SQL> GRANT EXECUTE ON DBMS_PIPE TO PGAADMIN
```

7. Ensure that the `pg4tcpmap` package has been installed on your Oracle Integrating Server. Follow Steps a through c to test for proper installation of `pg4tcpmap`.

Refer to [Output for the `pg4tcpmap` Tool](#) in [Appendix B](#) for a sample of the output from the `pg4tcpmap` tool, and refer to Chapter 6 of the *Oracle Procedural Gateway for APPC User's Guide* for details about the commands needed to run the tool.

a. Use SQL*Plus to connect to Oracle Integrating Server as the SYSTEM user.

b. Enter the following command:

```
SQL> select owner, table_name
from dba_tables
where table_name = 'PGA_TCP_IMSC',
and owner = 'PGAADMIN';
SQL> column owner format a 10
SQL> column index_name format a 18
SQL> column table_name format a 14
SQL> select owner, index_name, table_name
from dba_indexes
where index_name = 'PGA_TCP_IMSC_IND';
```

Each SELECT statement must produce one row. The following output is the result of the first SELECT statement:

TABLE_NAME	OWNER
PGA_TCP_IMSC	PGAADMIN

The following output is the result of the second SELECT statement:

OWNER	INDEX_NAME	TABLE_NAME	UNIQUENESS
PGAADMIN	PGA_TCP_IMSC_IND	PGA_TCP_IMSC	UNIQUE

If the SELECT statements produce the preceding output on your screen, then you can skip Step 1. If the SELECT statement produces no output or more than one row, then the result is not the same as the output described above, and it is necessary for you to perform Step 1.

- c. From SQL*Plus, run the `pgaimsc.sql` script in the Oracle Integrating Server `$ORACLE_HOME/pg4appc/admin` directory:

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgaimsc.sql
```

- 8. Install the TIP trace access PL/SQL routines. These routines require that the DBMS_PIPES standard PL/SQL package is installed and that PGAADMIN has execute privileges on it. For more information on DBMS_PIPES, refer to the *Oracle Database Application Developer's Guide*.

- a. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as user PGAADMIN.

- b. From SQL*Plus, run the `pgatiptr.sql` script in the `$ORACLE_HOME/pg4appc/admin` directory. This script creates PL/SQL routines that can be called to read and purge trace information created by PGAU-generated TIP specifications. It also creates public synonyms for these routines. The script prompts you for the necessary user IDs and passwords.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgatiptr.sql
```

- 9. Install the GPGLOCAL package. This package is required for compilation and execution of all PGAU-generated TIP specifications. TIP developers should be granted execute privileges on GPGLOCAL (refer to [Optional Configuration Steps to Permit Multiple Users](#) on page 13-8).

- a. Use SQL*Plus to connect to Oracle Integrating Server as PGAADMIN.

- b. From SQL*Plus, run the `gpglocal.pkh` script in the `$ORACLE_HOME/pg4appc/admin` directory. This script compiles the GPGLOCAL package specification.

```
SQL> @$ORACLE_HOME/pg4appc/admin/gpglocal.pkh
```

- c. From SQL*Plus, run the `gpglocal.pkb` script in the `$ORACLE_HOME/pg4appc/admin` directory. This script compiles the GPGLOCAL package body.

```
SQL> @$ORACLE_HOME/pg4appc/admin/gpglocal.pkb
```

Note: Recompile TIPs when changing communication protocol from SNA to TCP/IP:

If you have existing TIPs that were generated previously on a gateway using the SNA communication package protocol and you want to utilize the new TCP/IP feature, then the TIPs have to be regenerated by PGAU with certain mandatory NLS_LANGUAGE and Side Profile Settings. Specify the correct ASCII character set in the DEFINE TRANSACTION command.

This is because the gateway assumes that the correct user exit in IMS Connect is being used, which would translate between the correct ASCII and EBCDIC character sets.

13.4 Upgrading or Migrating the Oracle Server from Previous Gateways

Follow these instructions only if you have a previous version of the Oracle Procedural Gateway for APPC installed on your system and need to configure it for Release 10.2.0.1.0 of the gateway.

1. Upgrade your Oracle Procedural Gateway for APPC to current version levels as follows:

- a. Use SQL*Plus to connect to Oracle Integrating Server as the SYS user.
- b. Install the UTL_RAW package body. From SQL*Plus, run the prvtrawb.plb script from the \$ORACLE_HOME/rdbms/admin directory. This script upgrades the UTL_RAW package body.

```
SQL> @$ORACLE_HOME/rdbms/admin/prvtrawb.plb
```

- c. Install the UTL_PG package body. From SQL*Plus, run the prvtpgb.plb script from the \$ORACLE_HOME/rdbms/admin directory. This script upgrades the UTL_PG package body.

```
SQL> @$ORACLE_HOME/rdbms/admin/prvtpgb.plb
```

The prvtrawb.plb and prvtpgb.plb scripts should complete successfully. If they fail because specifications do not exist or were invalidated, then consider reinstalling the package specifications as directed in the following section.

13.4.1 If You Must Reinstall Package Specifications

If the UTL_RAW or UTL_PG package has been invalidated or deinstalled, then the prvtrawb.plb and prvtpgb.plb scripts might not complete successfully. You may have to reinstall the package specifications.

If you do reinstall the package specifications, then any dependent objects (such as existing user TIPs and client applications) are invalidated and subsequently need to be recompiled. The impact of this is a one-time performance delay while recompilation of the TIPs and dependent client applications proceeds.

Important: Before proceeding with reinstallation of the package scripts, ensure that you are in the \$ORACLE_HOME/pg4appc/admin directory.

TIPs were split into separate specification and body files in release 3.3 to avoid cascaded recompilations in later releases.

Step 1 Run the Following Scripts Before Proceeding with the PGAU Upgrade

From SQL*Plus, run the `utlraw.sql` script:

1. If necessary, use SQL*Plus to connect to the Oracle Integrating Server as the `SYS` user.
2. From SQL*Plus, run the `utlraw.sql` and `utlpg.sql` scripts in the Oracle Integrating Server `$ORACLE_HOME/rdbms/admin` directory, in the following order, to upgrade their respective package specifications:

```
SQL> @$ORACLE_HOME/rdbms/admin/utlraw.sql
SQL> @$ORACLE_HOME/rdbms/admin/utlpg.sql
```

Step 2 Repeat Installation of UTL_RAW and UTL_PG Package Body

After the scripts have run, repeat Steps b and c in [Section 13.4, "Upgrading or Migrating the Oracle Server from Previous Gateways"](#) on page 13-7. Then, proceed to the section titled [Upgrading PGAU from Previous Gateway Releases](#) on page 13-8.

Note: TIPs and dependent client applications must be recompiled after reinstallation of the package specifications. Refer to the "Compiling a TIP" section in Chapter 3 of the *Oracle Procedural Gateway for APPC User's Guide* for information about compiling TIPs.

13.4.2 Upgrading PGAU from Previous Gateway Releases

Upgrade the PG DD as follows before running the new PGAU:

1. If necessary, use SQL*Plus to connect to Oracle Integrating Server as the `PGAADMIN` user.
2. From SQL*Plus, run the `pgddupgr.sql` script in the `$ORACLE_HOME/pg4appc/admin` directory. This script upgrades the PG DD.

13.5 Optional Configuration Steps to Permit Multiple Users

The following configuration steps are optional. Perform these steps if you want to allow users other than `PGAADMIN` to perform PG DD operations using PGAU.

1. Create public synonyms for the PG DD to allow other users to access the tables:
 - a. Use SQL*Plus to connect to Oracle Integrating Server as the `SYSTEM` user.
 - b. From SQL*Plus, run the `pgddcr8s.sql` script in the `$ORACLE_HOME/pg4appc/admin` directory. This script creates public synonyms for the PG DD.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddcr8.sql
```

2. Create roles for accessing the PG DD, performing definitions of transactions, and generating TIP specifications. The `PGAADMIN` user can grant these roles to other users as necessary.
 - a. Use SQL*Plus to connect to Oracle Integrating Server as user `PGAADMIN`.

- b. From SQL*Plus, run the `pgddcr8r.sql` script in the `$ORACLE_HOME/pg4appc/admin` directory. This script creates two roles, `PGDDDEF` and `PGDDGEN`. The `PGDDDEF` role provides `SELECT`, `INSERT`, `UPDATE`, and `DELETE` privileges against some of the PG DD tables, and `SELECT` privileges against others, and allows execution of the `PGAU DEFINE`, `GENERATE`, `REDEFINE`, `REPORT`, and `UNDEFINE` statements. The `PGDDGEN` role provides select privileges against the PG DD tables and allows execution of the `PGAU GENERATE` and `REPORT` statements only.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddcr8r.sql
```

3. Grant access to PGA required packages.

TIP developers require access to the following PL/SQL packages, which are shipped with Oracle Integrating Server:

- `DBMS_PIPE` in the `$ORACLE_HOME/rdbms/admin` directory
- `UTL_RAW` in the `$ORACLE_HOME/rdbms/admin` directory
- `UTL_PG` in the `$ORACLE_HOME/rdbms/admin` directory

Explicit grants to run these packages must be made to TIP developers.

These grants can be private as in the following example:

```
$ sqlplus SYS/ pw@database_specification_string
SQL> GRANT EXECUTE ON UTL_RAW TO tip_developer;
SQL> GRANT EXECUTE ON UTL_PG TO tip_developer;
SQL> GRANT EXECUTE ON DBMS_PIPE TO tip_developer;
SQL> CONNECT PGAADMIN/ pw@database_specification_string
SQL> GRANT EXECUTE ON PGAADMIN.PURGE_TRACE TO tip_developer;
SQL> GRANT EXECUTE ON PGAADMIN.READ_TRACE TO tip_developer;
SQL> GRANT EXECUTE ON PGAADMIN.GPGLocal TO tip_developer;
SQL> exit
```

Alternatively, these grants can be public as in the following example:

```
$ sqlplus SYS/pw@ database_specification_string
SQL> GRANT EXECUTE ON UTL_RAW TO PUBLIC;
SQL> GRANT EXECUTE ON UTL_PG TO PUBLIC;
SQL> GRANT EXECUTE ON DBMS_PIPE to PUBLIC;
SQL> CONNECT PGAADMIN/ pw@database_specification_string
SQL> GRANT EXECUTE ON PGAADMIN.PURGE_TRACE TO PUBLIC;
SQL> GRANT EXECUTE ON PGAADMIN.READ_TRACE TO PUBLIC;
SQL> GRANT EXECUTE ON PGAADMIN.GPGLocal TO PUBLIC;
SQL> EXIT
```

You can use either private or public grants. Both are sufficient for using PGA. Public grants are easier and can be performed now. If you use private grants, then they must be given each time a new TIP developer user ID is created.

SQL scripts for performing these grants are provided in the `$ORACLE_HOME/pg4appc/admin` directory. The `pgddapub.sql` script performs these grants for public access to the packages. The `pgddadev.sql` script performs the grants for private access to the packages by a single TIP developer. If you are going to use private grants, then you must run the `pgddadev.sql` script once for each TIP developer user ID:

- a. Use SQL*Plus to connect to Oracle Integrating Server as `PGAADMIN`.
- b. From SQL*Plus, run the appropriate script (`pgddapub.sql` or `pgddadev.sql`) from the `$ORACLE_HOME/pg4appc/admin` directory. The

script performs the necessary grants as described earlier. You are prompted for the required user IDs, passwords, and database specification strings. If you are using private grants, then repeat this step for each user ID requiring access to the packages.

If you are using private grants, then repeat this step for each user ID requiring access to the packages.

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddapub.sql
```

or

```
SQL> @$ORACLE_HOME/pg4appc/admin/pgddadev.sql
```

4. If you are upgrading from a previous release of the gateway when the communication protocol was SNA, to the current gateway using TCP/IP, and if you want to upgrade your existing TIPs with new function and maintenance, then regenerate existing TIP specifications using the PGAU GENERATE statement.

Note: The PGAU has been enhanced to automatically upgrade existing PG DD entries with a new attribute when a PGAU GENERATE command is executed. To support this enhancement, add a new privilege to the PGDDGEN role. To do this as the PGAADMIN user, use SQL*Plus to connect to the Oracle Integrating Server installation where the PG DD is stored. Then issue the following SQL command:

```
SQL> GRANT INSERT ON PGA_DATA_VALUES TO PGDDGEN
```

- a. Start PGAU from the directory path where the PGAU control files are generated and where TIPs are stored:

```
$ pgau
PGAU> CONNECT PGAADMIN/pgadmin@database_specification_string
PGAU> GENERATE traname
PGAU> EXIT
```

For more information about the GENERATE command, refer to the PGAU GENERATE command section in Chapter 2 of the *Oracle Procedural Gateway for APPC User's Guide*.

Note that it is not necessary to define the PG DD entries again.

5. Start SQL*Plus in the same directory path where the newly-generated TIP specifications are stored.

```
$ sqlplus tip_owner/pw@database_specification_string
SQL> @tipname.pkh
SQL> @tipname.pkb
SQL> exit
```

PGAU GENERATE produces the TIP in two output files: a specification and a body. You must compile both, first the specification and then the body.

For more information about the GENERATE command, refer to the PGAU GENERATE command section in Chapter 2, of the *Oracle Procedural Gateway for APPC User's Guide*.

13.6 Configuring TCP/IP for the Gateway

You must now configure the TCP/IP for IMS Connect communication package profiles for TCP/IP connections.

Configure the profiles to define the TCP/IP conversations with the OLTP. Refer to [Chapter 10.1, "Configuring TCP/IP for IMS Connect on UNIX Platforms"](#) for the appropriate instructions.

When you have finished configuring your communications package, return to the following section, [Section 13.7, "Configuring the Gateway"](#) on page 13-11.

13.7 Configuring the Gateway

To configure the gateway, perform the following:

1. Tailor the gateway parameters.

There are a number of parameters specific to Oracle Procedural Gateway for APPC when it is using TCP/IP for IMS Connect. These are supplied in the gateway parameter file, *initsid.ora*, which is in the `$ORACLE_HOME/pg4appc/admin` directory. A sample gateway parameter file, *initPGA.ora* is provided in this subdirectory.

Note: In the *initsid.ora* file, substitute your gateway SID name for "*sid*" in this file name.

The *initsid.ora* file contains both APPC and TCP/IP parameters separated by a description. You must modify the *initsid.ora* file by deleting the APPC parameters. Refer to [Appendix B.1, "Gateway Initialization Parameter File Using TCP/IP"](#) for the valid TCP/IP parameters.

The parameters fall into two categories:

- Gateway initialization parameters

These parameters control the general operation of the gateway in the Oracle environment.

Important: Before performing the following step, refer to [Appendix B.1, "Gateway Initialization Parameter File Using TCP/IP"](#) for information about tailoring gateway initialization and PGA parameters. Pay special attention to the information about using the `PGA_CAPABILITY` parameter.

- PGA parameters

PGA parameters control the TCP/IP interface portion of the gateway. Use the `SET` gateway initialization parameter to specify PGA parameters. Oracle recommends that you group all `SET` commands for PGA parameters at the end of the *initsid.ora* file.

Note: Misspelled parameters are ignored. However, if the `$ORACLE_HOME/pg4appc/admin/initsid.ora` file is missing all calls to the gateway fail and return a PGA-20928 error..

13.8 Loading the PGA_TCP_IMSC Table

Gateway users who want to employ the TCP/IP protocol do so by using the `pg4tcpmap` tool.

The `pg4tcpmap` tool is located on the gateway. Its function is to map the Side Profile Name to TCP/IP and IMS Connect attributes. You must run this tool before executing the PL/SQL gateway statements (such as `SQL> @$ORACLE_HOME/pg4appc/demo/IMS/pgaims.sql`).

In PGAINIT TIP, for example, the user must specify a Side Profile Name. The SNA protocol recognizes and utilizes the parameter. In this release of the gateway, the `pg4tcpmap` tool uses the original PGAINIT TIP format to map the relevant SNA parameters to TCP/IP. The `pg4tcpmap` tool inserts the values of these parameters into a table called `PGA_TCP_IMSC`.

Before executing `pg4tcpmap`, you must specify the `ORACLE_HOME`, Oracle SID, and modify `initsid.ora`. Refer to [Appendix B, "Gateway Initialization Parameters for TCP/IP Communication Protocol"](#) in this guide and Chapter 6 in the *Oracle Procedural Gateway for APPC User's Guide* for complete information about the `pg4tcpmap` commands.

Chapter 6 of the *Oracle Procedural Gateway for APPC User's Guide* contains a list of the `pg4tcpmap` commands and instructions for using them, as well as an example of the table. Refer to "Problem Determination" in the *Oracle Procedural Gateway for APPC User's Guide* for information about the trace file for the executed `pg4tcpmap` tool.

To operate this tool, execute the following command:

```
$ $ORACLE_HOME/bin/pg4tcpmap
```

Refer to [Output for the pg4tcpmap Tool](#) on page B-1 for a sample of the `pg4tcpmap` output.

13.9 Verifying the Gateway Installation and OLTP Configuration

To verify the gateway installation and the OLTP configuration, perform the following procedures after installing the gateway.

Note: If your database link name is not "PGA," modify the demonstration `.sql` files to give them the particular database link name that you created in Step 5 of [Section 13.3, "Configuring Oracle Integrating Server: First-Time Installation"](#) on page 13-3. You must modify the following `.sql` files:

- `pgavsn.sql`
 - `pgaecho.sql`
 - `pgaims.sql`
-
-

13.9.1 Verifying the Gateway Installation

To verify the gateway software installation using the database link PGA previously created, perform the following steps:

1. Using SQL*Plus, connect to your Oracle Integrating Server as PGAADMIN.
2. Run `SQL> @$ORACLE_HOME/pg4appc/demo/pgavsn.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/pgavsn.sql
```

The server version number banner appears at your terminal. The following output appears:

```
Oracle Procedural Gateway for APPC (extension TCP/IP for IMS connect).
Version 10.2.0.1.0 Thu Feb 13 13:12:49 2005
```

```
Copyright (c) Oracle Corporation 1979,
2005. All rights reserved.
```

PL/SQL procedure successfully completed.

3. Run `$ORACLE_HOME/pg4appc/demo/pgaecho.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/pgaecho.sql
```

You will receive the following output:

```
==> Congratulations, your installation was successful. <==
```

13.9.2 Verifying the OLTP Configuration

Use the following procedure to verify your OLTP configuration.

Attention: If you have not completed the file transfers detailed in [Preparing to Configure a Gateway Installation/Upgrade](#) on page 13-2, then complete them now before proceeding to the next step.

13.9.2.1 IMS/TM Verification

Perform the following steps to verify the IMS/TM configuration. Be certain that you have installed and configured the IMS Connect and that it is up and running before you begin this procedure. Refer to the *IBM IMS Connect Guide and Reference* for information about how to perform the installation and configuration tasks.

Note: **TIPs must be recompiled when changing communication protocol to TCP/IP.**

TCP/IP only: If you have existing TIPs that were generated previously on a gateway using the SNA communication package protocol and you want to utilize the new TCP/IP feature, then the TIPs will have to be regenerated by PGAU with mandatory NLS_LANGUAGE and Side Profile Settings. Specify the appropriate ASCII character set in the `DEFINE TRANSACTION` command.

This is because the gateway assumes that the appropriate user exit in IMS Connect is being used, which would translate between the appropriate ASCII and EBCDIC character sets.

1. To verify that the FLIP transaction is installed correctly, log on to your IMS/TM system and enter the following transaction (replacing *FLIP* with the transaction ID you chose for FLIP when you configured your IMS/TM system for the gateway):

```
FLIP THIS MESSAGE
```

The following output appears on your terminal:

```
EGASSEM SIHT PILF
```

2. Log on to UNIX.

Attention: If you have not completed the file transfers detailed in [Section 13.2 Preparing to Configure a Gateway Installation/Upgrade](#) on page 13-2, complete them now, before proceeding to the next step.

3. Modify the `pgaims.sql` file, which is located at `$ORACLE_HOME/pg4appc/demo/IMS/pgaims.sql`. Customize the following three items used for accessing the gateway and the IMS/TM system as described in the comments at the beginning of the file:
 - IMS/TM transaction ID
 - Side Profile Name
 - Logmode entry name
4. Ensure that the TCP/IP communication protocol on your system has been started.
5. Using SQL*Plus, connect to your Oracle Integrating Server from PGAADMIN.
6. Run `pgaims.sql`.

```
SQL> @$ORACLE_HOME/pg4appc/demo/IMS/pgaims.sql
```

The following message appears:

```
==> Congratulations, your gateway is communicating with IMS/TM <==
```

Your IMS/TM installation verification is now complete.

13.10 Performing Postinstallation Procedures

The following are optional steps that you can perform as necessary. Installation of the sample applications for your OLTP is recommended to help you to fully understand how the gateway works and how it interfaces with your OLTP.

13.10.1 Installing Sample Applications

Your Oracle Procedural Gateway for APPC featuring TCP/IP for IMS Connect contains sample PL/SQL procedures and OLTP transaction programs that demonstrate the gateway's capabilities.

Note: When you call a gateway that uses TCP/IP as the communication protocol, and you use EBCDIC as the language in the control files, then you must change the language from EBCDIC to ASCII. Some examples of control files that might be in EBCDIC language are `ivtno.ctl` and `ivtnv.ctl`.

For more information, refer to the `$ORACLE_HOME/pg4appc/demo/IMS/ivtno.ctl` and `$ORACLE_HOME/pg4appc/demo/IMS/ivtnv.ctl` files.

Samples are provided for IMS/TM:

- IMS inquiry using IVTNO and IVTNV sample transactions

Additional samples are added to the distribution media in newer releases of the product. Wherever possible, the sample applications use the sample databases provided with the database products.

For this release, the entire documentation about installing and using the sample applications is available in the `README.doc` files in the following directory:

`$ORACLE_HOME/pg4appc/demo/IMS`

Security Requirements

The gateway architecture involves multiple systems, database servers, and communications facilities, each having distinct security capabilities and limitations. To effectively plan and implement your security scheme, you must understand these capabilities and limitations, in addition to knowing your installation security requirements.

Read this chapter to learn about the capabilities and limitations of the Oracle Procedural Gateway for APPC. This chapter contains the following sections:

- [Section 14.1, "Overview of Security Requirements"](#)
- [Section 14.2, "Authenticating Application Logons"](#)
- [Section 14.3, "Defining and Controlling Database Links"](#)
- [Section 14.4, "Using SNA Security Validation"](#)
- [Section 14.5, "TCP/IP Security"](#)
- [Section 14.6, "Passwords in the Gateway Initialization File"](#)

14.1 Overview of Security Requirements

Before implementing your security scheme, you must understand the existing security requirements and expectations in your environment. Because you are enabling application access to different databases on different systems, you may need to merge multiple security cultures. When you connect several different systems into an operating whole, the system with the strictest security requirements generally dictates what the other systems can and cannot do.

Gateway security includes two main concerns:

- Users and applications that are permitted access to a particular gateway instance and OLTP
- OLTP transactions that users and applications are able to execute

You can control access at several points in the gateway architecture. The primary options are discussed in the following sections. Control over remote transaction program access is provided by each OLTP with native authorization mechanisms based on user ID. These facilities are described in the product documentation for your OLTP. Information in this chapter includes how the gateway facilities determine the user ID that is in effect for a particular OLTP connection.

When the gateway is involved in an RPC request, security mechanisms are in effect for each system component encountered by the gateway. The first system component that

is encountered is the application tool or third-generation language (3GL) program. The last system component that is encountered is the OLTP.

Each of the following sections identifies the component and the type of security processing that is available in that component. Each section offers a summary of key features and parameters. Refer to product-specific documentation for detailed information about the non-gateway components for Oracle and non-Oracle products.

14.2 Authenticating Application Logons

An application must connect to an Oracle Integrating Server before using the gateway. The type of logon authentication that you use determines the resulting Oracle user ID and can affect gateway operation.

The following types of authentication are available:

1. Oracle authentication

With Oracle authentication, each Oracle user ID has an associated password that is known to Oracle. When an application connects to the server, it supplies a user ID and password. Oracle confirms that the user ID exists and that the password matches the one stored in the database.

2. Operating system authentication

With operating system authentication, the underlying server operating system is responsible for authentication. An Oracle user ID that is created with the `IDENTIFIED EXTERNALLY` attribute (instead of a password) is accessed with operating system authentication. To log on with such a user ID, the application supplies a forward slash (/) for a user ID and does not supply a password.

To perform operating system authentication, the server determines the requester operating system user ID, optionally adds a fixed prefix to it, and uses the result as the Oracle user ID. The server confirms that the user ID exists and is `IDENTIFIED EXTERNALLY`, but no password checking is done. The underlying assumption is that users are authenticated when they log on to the operating system.

Operating system authentication is not available on all platforms and is not available in some Oracle Net (client-server) and multithreaded server configurations. Refer to your platform-specific Oracle server documentation and *Oracle Net Services Administrator's Guide* to determine the availability of this feature in your configuration.

For more information about authenticating application logons, refer to the *Oracle Database Administrator's Guide*.

14.3 Defining and Controlling Database Links

The following sections discuss database links for users of the gateway employing either TCP/IP or SNA communications protocols.

14.3.1 Link Accessibility

The first point of control for a database link is simply if it is accessible to a given user. A public database link can be used by any user ID. A private database link can be used only by the user who created it. Database link usability is determined by its ability to open a session to the gateway. Oracle Integrating Server makes no distinction as to the type of use (such as read-only versus update or write) or which remote objects can be accessed. These distinctions are the responsibility of the OLTP that is accessed.

14.3.2 Links and CONNECT Clauses

The `CONNECT` clause is another security-related attribute of a database link. You can use the `CONNECT` clause to specify an explicit user ID and password, which can differ from the Oracle user ID and password. This `CONNECT` user ID and password combination is sent to the gateway when the database link connection is first opened. Depending on gateway-specific options, the gateway might send that user ID and password to the OLTP to be validated.

If a database link is created without a `CONNECT` clause using Oracle authentication, then the Oracle user ID and password of the user are sent to the gateway when the connection is opened. If the user logs on to Oracle Integrating Server with operating system authentication, then the gateway receives no user ID or password from Oracle Integrating Server. It is impossible for operating system-authenticated Oracle users to use a gateway database link defined without a `CONNECT` clause. However, if your OLTP provides user ID mapping facilities based on the gateway LU name from which the user is connecting, then such a connection is possible if all users on the same gateway instance can use the same OLTP user ID.

For more information about database links, refer to the *Oracle Database Administrator's Guide*.

14.4 Using SNA Security Validation

The information in [Section 14.4](#) applies only to the security needs of gateway users employing the SNA communications protocol. When an RPC request to start a remote transaction program is received by the gateway, the gateway attempts to start an APPC conversation with the OLTP. Before the conversation can begin, a session must start between the platform's Logical Unit (LU) and the OLTP LU.

APPC support for your platform is provided by a SNA communication package (SNAP-IX for Solaris Operating System (SPARC), SNAPPlus2 for HP-UX and SNA Server for AIX-based systems).

SNA and its various access method implementations, including VTAM and the SNA communication package for your platform, provide security validation at session initiation time, allowing each LU to authenticate its partner. This validation is carried out entirely by network software before the gateway and OLTP application programs begin their conversation and process conversation-level security data. If session-level security is used, then correct password information must be established in your platform's SNA profiles and in similar parameter structures in the OLTP to be accessed. Refer to the appropriate communications software product documentation for detailed information about this subject.

14.4.1 Specifying SNA Conversation Security

The `PGA_SECURITY_TYPE` parameter of the gateway initialization file allows you to specify one of three options that determine the security conduct of the LU6.2 conversation that is allocated with the OLTP. These options are part of the SNA LU6.2 architecture, but their precise behavior might vary depending on the particular OLTP system.

14.4.1.1 SNA Security Option `SECURITY=NONE`

If you specify `PGA_SECURITY_TYPE=NONE`, then the gateway performs no processing of the client user ID and password. The conversation is allocated with SNA option `SECURITY=NONE`.

14.4.1.2 SNA Security Option SECURITY=PROGRAM

If you specify `PGA_SECURITY_TYPE=PROGRAM`, then the gateway allocates the conversation with SNA option `SECURITY=PROGRAM`, and the following information is sent to the OLTP:

- If the TIP user ID and password overrides are used, then the specified user ID and password are sent regardless of the database link specification.
- If the database link has explicit `CONNECT` information, then the specified user ID and password are sent.
- If the database link has no `CONNECT` clause, and if the application logged on to Oracle with an explicit user ID and password, then the Oracle user ID and password are sent.
- If the application logs on to Oracle with operating system authentication, and if the database link lacks explicit `CONNECT` information, then no user ID and password are sent. If no user ID and password are sent, and if the OLTP is not configured to assign a default user ID, then the connection fails.

In general, SNA option `SECURITY=PROGRAM` tells the OLTP to authenticate the user ID/password combination using whatever authentication mechanisms are available. For example, if CICS Transaction Server for z/OS is the OLTP, then RACF can be used. This is not always the case, however, because each OLTP can be configured to process inbound user IDs in other ways.

14.4.1.3 SNA Security Option SECURITY=SAME

If you specify `PGA_SECURITY_TYPE=SAME`, the gateway allocates the conversation with SNA option `SECURITY=SAME` and sends only a user ID, without a password, to the OLTP. In this case, your SNA communication package sends the owning user ID of the gateway server executable, `$ORACLE_HOME/bin/pg4asrv`, as the user ID. The user ID that is sent is not the Oracle user ID. This user ID can be viewed with the `UNIX ls` command and can be changed by an authorized user with the `chown` command. Because this user ID is the same for all users of a given gateway instance, this option is of limited use.

Note: The user ID sent is not translated to uppercase by your SNA communication package. If your OLTP is running on a system which does not allow lowercase user IDs (z/OS, for example), you must set up an uppercase user ID on your platform to be the owner of the gateway executable file.

`SECURITY=SAME` is similar to your platform operating system authentication. It tells the OLTP that the user has already been authenticated at the originating side of the conversation. There might be configuration parameters or options on the server side that affect whether `SECURITY=SAME` conversations are accepted. When properly configured, the OLTP only confirms that the user ID itself is valid and then accepts the connection. As with `SECURITY=PROGRAM`, you can change this using configuration options in many OLTPs.

14.5 TCP/IP Security

The security information in this section applies only to users of the Oracle Procedural Gateway for APPC using the TCP/IP for IMS Connect feature. When an RPC request to start a remote transaction program is received by the gateway, the gateway

attempts to start the TCP/IP conversation with IMS Connect. IMS Connect would contact the OLTP (IMS) through OTMA and XCF. Refer to the *IBM IMS Connect Guide and Reference* for more information. The conversation between the gateway and IMS Connect occurs when the network uses the TCP/IP address or host name and port number to connect from the gateway to IMS Connect.

Note: As the gateway is using PGAU to generate TIPs, the TIPs contain SNA information. When using the Oracle Procedural Gateway for APPC with TCP/IP support for IMS Connect, you need to map the SNA names to the TCP/IP host name and port number in order for the gateway to talk to IMS Connect. Use the `pg4tcpmap` tool to map the information from SNA to TCP/IP. Refer to Chapter 6, "*pg4tcpmap Commands*," of the *Oracle Procedural Gateway for APPC User's Guide* for more information.

IMS Connect provides validation at session initiation time, allowing each connection to authenticate its partner. This validation is carried out entirely by network software before the gateway and OLTP application programs at IMS begin their conversation and process conversation-level security data. If session-level security is used, then correct password information must be established in your platform and in similar parameter structures in the OLTP to be accessed.

14.5.1 Specifying TCP/IP Conversation Security

The `PGA_SECURITY_TYPE` parameter of the gateway initialization file enables you to specify the security conduct for the conversation that is allocated by the gateway for OLTP. Refer to [Appendix B, "Gateway Initialization Parameters for TCP/IP Communication Protocol"](#).

14.5.1.1 TCP/IP Security Option SECURITY=NONE

If you specify `PGA_SECURITY_TYPE=NONE`, then the gateway performs no processing of the client user ID and password.

14.5.1.2 TPC/IP Security Option SECURITY=PROGRAM

If you specify `PGA_SECURITY_TYPE=PROGRAM`, then the following information is sent to the OLTP:

- If the TIP user ID and password overrides are used, then the specified user ID and password are sent regardless of the database link specification.
- If the database link has explicit `CONNECT` information, then the specified user ID and password are sent.
- If the database link has no `CONNECT` clause, and if the application logged on to Oracle with an explicit user ID and password, then the Oracle user ID and password are sent.
- If the application logs on to Oracle with operating system authentication, and if the database link lacks explicit `CONNECT` information, then no user ID and password are sent. If no user ID and password are sent, and if the OLTP is not configured to assign a default user ID, then the connection fails.

RACF is the only authentication mechanism available when the Oracle Procedural Gateway for APPC using TCP/IP for IMS Connect talks to IMS Connect.

Important: You must specify your RACF group name through the pg4tcpmap tool if you have set your PGA security option to SECURITY=PROGRAM. For more information about this issue, refer to the *Oracle Procedural Gateway for APPC User's Guide*.

14.6 Passwords in the Gateway Initialization File

Initialization parameters may contain sensitive information, such as user IDs or passwords. Initialization parameters are stored in plain text files and may be deemed insecure. An encryption feature has been added to Heterogenous Services making it possible to encrypt parameters values. This is done through the tg4pwd utility. For more information on this utility refer to "Oracle® Database Heterogenous Connectivity Administrator's Guide" 10g Release 2 (10.2).

Migration From Existing Gateways

Migrating to new instances of Oracle Procedural Gateway for APPC from an existing installation is straightforward, provided you follow some guidelines. This chapter provides information to make these new installations as easy as possible. It also provides the parameters you need if you are using the TCP/IP for IMS Connect communication protocol on your gateway.

This chapter provides information that is specific to this release of Oracle Procedural Gateway for APPC for UNIX. It contains the following sections:

- [Section 15.1, "Migrating An Existing Gateway Instance to a New Release Using SNA Protocol"](#)
- [Section 15.2, "Migrating from an Existing Gateway to TCP/IP Using SNA"](#)

15.1 Migrating An Existing Gateway Instance to a New Release Using SNA Protocol

Follow these steps to migrate an existing gateway to Release 10.2.0 of the gateway using the SNA communication protocol.

Note that if you are using the gateway TCP/IP support for IMS Connect, you will not be migrating an existing release to the current release of the gateway. However, you will need to place valid Heterogeneous Services parameters into your `initsid.ora` file. Proceed to [Section 15.1.5, "Parameter Changes: Version 4 to Release 10.2.0 of the Gateway"](#).

15.1.1 Step 1: Install the New Release

Install the new release of the gateway in a separate directory as outlined in [Chapter 4, "Installing the Gateway"](#).

Caution: Do not install the gateway over a previously existing gateway installation. Doing this corrupts the existing installation.

15.1.2 Step 2: Transferring the `initsid.ora` Gateway Initialization File Parameters

Copy the `initsid.ora` file from the old gateway instance to the new instance.

If you are migrating from Release 9.0.1 or earlier of the gateway, `PGA_TRACE` is not supported. You need to modify the parameter to `TRACE_LEVEL` instead.

Note: If you are using `TRACE_LEVEL`, you must set the path for the `LOG_DESTINATION` parameter.

If you have encrypted some of the parameters at file `initsid.ora` by using `pg4pwd`, copy the `$ORACLE_HOME/pg4appc/admin/initsid.pwd` file from the old gateway instance to the new instance.

Note: If you are migrating from Release 9.0.1 or earlier of the gateway, you do not need to copy this file into the new instance.

15.1.3 Backout Considerations When Migrating to New Releases

Oracle recommends that you keep the old gateway Oracle home directory and instance configurations intact and operational when you are installing a new release of the gateway and upgrading existing instances, in case there are problems with the upgrade. This helps to ensure minimum down time between changes to different gateway instances.

15.1.4 Oracle Net Considerations

Oracle Procedural Gateway for APPC uses the Heterogeneous Services (HS) facilities of Oracle and Oracle Net. If you are upgrading from a version 4 gateway, then you need to slightly modify the gateway service name entries in the `tnsnames.ora` file. Add an `(HS=)` clause to tell Oracle Net that the gateway uses HS facilities. For more information, refer to [Section 5.1, "Configuring the Oracle Network for the Gateway"](#).

15.1.5 Parameter Changes: Version 4 to Release 10.2.0 of the Gateway

This release of Oracle Procedural Gateway for APPC introduces new and changed initialization parameters if you are migrating from a Version 4 gateway to Release 10.2.0 of the gateway.

Note: The "[Parameter Changes: Version 4 to Release 10.2.0 of the Gateway](#)" section does not apply to you if you are migrating to Release 10.2.0 from Version 8 of the Oracle Procedural Gateway for APPC.

If you are using the gateway TCP/IP support for IMS Connect, you will not be *migrating* from Version 4 to the current release of the gateway. However, you need to place valid Heterogeneous Services parameters into your `initsid.ora` file.

This chapter contains references to the particular HS parameters that you need to run the gateway.

Note: Refer to the *Oracle Connectivity Guide* for a complete list and descriptions of all HS parameters used in Oracle products.

New Gateway Initialization Parameters

The following parameters are in the gateway initialization file (`initsid.ora`):

- `FDS_CLASS`

- FDS_INSTANCE
- HS_FDS_FETCH_ROWS
- LOG_DESTINATION
- TRACE_LEVEL
- PGA_TCP_DB (TCP/IP only)
- PGA_TCP_USER (TCP/IP only)
- PGA_TCP_PASS (TCP/IP only)

Note: The HS_ parameters are specific to Oracle Heterogeneous Services. For details on HS parameters, refer to the *Oracle Database Administrator's Guide*.

Renamed Gateway Initialization File Parameters

Following is a list of the gateway initialization file (*init.sid.ora*) parameters that have been renamed in this release of the gateway. The former names of the parameters are shown in parentheses.

- HS_COMMIT_STRENGTH_POINT (formerly COMMIT_STRENGTH_POINT)
- HS_DB_DOMAIN (formerly DB_DOMAIN)
- HS_DB_INTERNAL_NAME (formerly DB_INTERNAL_NAME)
- HS_DB_NAME (formerly DB_NAME)
- HS_DESCRIBE_CACHE_HWM (formerly DESCRIBE_CACHE_HWM)
- HS_LANGUAGE (formerly LANGUAGE)
- HS_NLS_DATE_FORMAT (formerly NLS_DATE_FORMAT)
- HS_NLS_DATE_LANGUAGE (formerly NLS_DATE_LANGUAGE)
- HS_OPEN_CURSORS (formerly OPEN_CURSORS)
- HS_ROWID_CACHE_SIZE (formerly ROWID_CACHE_SIZE)

Obsolete Parameters

The following parameters are obsolete. If necessary, remove them from your configuration files:

- MODE
- SERVER_PATH
- ERROR_LOGGING
- ERROR_REPORTING
- ERRORTAG
- GATEWAY_SID
- GROUP_BY_OFF
- GTWDEBUG
- INCREMENT_CURSORS
- INIT_CURSORS

- LIST
- MAX_LOG_SIZE
- OPTIMIZE_FILE_OPEN
- ORDER_BY_OFF
- RESOLVE_BINDS
- RETRY
- SET
- SNMP_SUPPORT
- SQL_TRACE
- TRIM_CURSORS
- D_OPEN_CURSORS
- D_INIT_CURSORS
- D_INCREMENT_CURSORS
- D_TRIM_CURSORS
- PGA_TRACE

15.1.6 Parameter Changes: Version 8 or Earlier to Gateway Release 10.2.0

The following startup shell script parameter must be added to `initsid.ora` if you are migrating from a Version 4 or Version 8 gateway to Release 10.2.0 of the Oracle Procedural Gateway for APPC:

- FDS_CLASS_VERSION

15.1.7 Migrating from Gateway Release 9.0.1 or 9.2.0 or 10.1.0 to Gateway Release 10.2.0

No new parameters were added between release 9.0.1 and this release of the gateway.

15.2 Migrating from an Existing Gateway to TCP/IP Using SNA

The following sections are for those users who have an existing release of the gateway using the SNA protocol, but who want to switch to using TCP/IP support for IMS Connect.

The TCP/IP support for IMS Connect feature in this release of the gateway enables you to continue to use existing TIPs.

15.2.1 Using Existing TIPs with Existing Side Profile Definitions

Follow these instructions:

1. Make sure you have used the `pg4tcpmap` tool to insert valid parameter values into the `PGA_TCP_IMSC` table.
[Refer to Chapter 13, "Gateway Configuration Using TCP/IP Communication Protocol"](#) for instructions to load the `PGA_TCP_IMSC` table.
2. Make sure the `LANGUAGE` parameter in your TIPs is set to `american_american_us7ascii`.

3. Use PGAU to regenerate the IMS TIPs.
4. Add the following new TCP/IP parameters to the `initsid.ora` file:
 - PGA_TCP_DB
 - PGA_TCP_USER
 - PGA_TCP_PASS

You will find descriptions and information about adding these parameters in [Section 15.1.5, "Parameter Changes: Version 4 to Release 10.2.0 of the Gateway"](#). You will also find descriptions of the parameters in [Appendix B, "Gateway Initialization Parameters for TCP/IP Communication Protocol"](#).

Note: If your TIPs from a previous version of the gateway were already defined using a SideProfileName and the NLS_ LANGUAGE parameter has been set to a value of `american_america_us7ascii`, then you will not need to recompile these TIPs. You still need to map your parameter values using the `pg4tcpmap` tool.

15.2.2 Using Existing TIPs with LU/TP/LOGMODE Names

The new TCP/IP support for IMS Connect feature of the gateway only supports Side Profile Name coding in the IMS TIPs. If you have been using SNA on a gateway using LU, TP, or LOGMODE combinations, perform the following:

1. Edit the IMS TIPs. Create an 8-character, unique identifier and insert it into the TIP as the Side Profile.

This identifier does not need to be a system-defined SNA Side Profile Name, and it is simply a unique name.

2. Make sure you have used the `pg4tcpmap` tool to insert the Side Profile name into the `PGA_TCP_IMSC` table.

Refer to [Chapter 13, "Gateway Configuration Using TCP/IP Communication Protocol"](#) for instructions on loading the `PGA_TCP_IMSC` table.

3. Make sure the LANGUAGE in your TIPs is set to `american_america_us7ascii`.
4. Use PGAU to regenerate the IMS TIPs.
5. Add the following new TCP/IP parameters to the `initsid.ora` file:
 - PGA_TCP_DB
 - PGA_TCP_USER
 - PGA_TCP_PASS

Descriptions and information about adding these parameters is available in [Section 15.1.5, "Parameter Changes: Version 4 to Release 10.2.0 of the Gateway"](#). You will also find descriptions of the parameters in [Appendix B, "Gateway Initialization Parameters for TCP/IP Communication Protocol"](#).

Note: TIPs must be recompiled when changing communication protocol from SNA to TCP/IP.

If you have existing TIPs that were generated previously on a gateway using the SNA communication package protocol, and you want to use the new TCP/IP feature, then the TIPs have to be regenerated by PGAU with mandatory NLS_LANGUAGE and Side Profile Settings. Specify the appropriate ASCII character set in the DEFINE TRANSACTION command.

This is because the gateway assumes that the appropriate user exit in IMS Connect is being used, which would translate between the appropriate ASCII and EBCDIC character sets.

Gateway Initialization Parameters for SNA Protocol

This appendix describes the gateway initialization file location and lists the gateway initialization parameters supported by Oracle Procedural Gateway for APPC, specifically for the SNA protocol. These parameters are fully documented in [Section 15.1, "Migrating An Existing Gateway Instance to a New Release Using SNA Protocol"](#) on page 15-1. In addition, this appendix contains sample `listener.ora` and `tnsnames` files for a gateway using SNA.

This appendix contains the following sections:

- ["Gateway Initialization Parameter File Using SNA Protocol"](#) on page A-1

A.1 Gateway Initialization Parameter File Using SNA Protocol

The parameter file for the gateway is located in the `$ORACLE_HOME/pg4appc/admin` directory and is called `initsid.ora`.

This section covers various aspects of the gateway initialization file in the following sections:

- [Section A.1.1, "PGA Parameters"](#)
- [Section A.1.2, "PGA_CAPABILITY Parameter Considerations"](#)
- [Section A.1.3, "PGA_CONFIRM Parameter Considerations"](#)
- [Section A.1.4, "Sample listener.ora file for a Gateway Using SNA"](#)
- [Section A.1.5, "Sample tnsnames.ora file for a Gateway Using SNA"](#)

Note: The `initsid.ora` file contains both SNA and TCP/IP parameters. You must modify these files with suitable parameters.

A.1.1 PGA Parameters

The PGA parameters control the APPC interface portion of the gateway. PGA parameters are specified using the `SET` gateway initialization parameter. For example:

```
SET pga_parm=value
```

where:

- `pga_parm` is one of the PGA parameter names in the list that follows
- `value` is a character string with contents that depend on `pga_parm`

Table A-1 provides a list of PGA parameters and their descriptions.

Table A-1 PGA Parameters for Gateway Using SNA

Parameter	Description
LOG_DESTINATION= <i>logpath</i>	<p><i>logpath</i> specifies the destination at which STDERR is reopened. LOG_DESTINATION specifies a directory only and stderr is reopened to <i>logpath/sid_pid.log</i></p> <p>where:</p> <ul style="list-style-type: none"> ▪ <i>sid</i> is the sid name ▪ <i>pid</i> is the process ID assigned to the gateway
PGA_CAPABILITY	<p>PGA transaction capability. This controls whether updates are allowed through the gateway. The following are valid values:</p> <p>READ_ONLY or RO: Read-only capabilities.</p> <p>SINGLE_SITE or SS: Single-site update only. This indicates that in a distributed environment, only the gateway can perform updates. No other database updates can occur within the Oracle transaction.</p> <p>COMMIT_CONFIRM or CC: Commit-confirm. This indicates that in a distributed environment, updates can be performed by both the gateway and other participants within the Oracle transaction. The gateway is always committed first in this mode, and no other commit-confirm sites are allowed to participate in the Oracle transaction.</p> <p>The default is SINGLE_SITE.</p>
PGA_CONFIRM	<p>Incoming APPC CONFIRM request handling option. This controls what the gateway does when an APPC CONFIRM request is received from the remote transaction program. This parameter has meaning only when the conversation is running with SYNCLEVEL set to a value higher than 0. The following are valid values:</p> <p>ACCEPT - Respond to incoming APPC CONFIRM requests with APPC CONFIRMED responses.</p> <p>REJECT - Treat incoming APPC CONFIRM requests as errors causing the conversation to be deallocated and an error message to be issued.</p> <p>The default is REJECT.</p>
PGA_LOG_DB	<p>The Oracle Net service name for the Oracle server in which the gateway maintains its transaction log. This parameter can be from 1 to 255 characters long. This parameter is required only when PGA_CAPABILITY is set to COMMIT_CONFIRM.</p> <p>There is no default value.</p>
PGA_LOG_PASS	<p>The Oracle password to be used by the gateway when connecting to the Oracle server specified by the PGA_LOG_DB parameter. The password can be from 1 to 30 characters long. This parameter is required only when PGA_CAPABILITY is set to COMMIT_CONFIRM. The password can be encrypted. For more information about encrypting the password, refer to Passwords in the Gateway Initialization File on page 14-6.</p> <p>There is no default value.</p>

Table A-1 (Cont.) PGA Parameters for Gateway Using SNA

Parameter	Description
PGA_LOG_USER	<p>The Oracle user ID to be used by the gateway when connecting to the Oracle server specified by the PGA_LOG_DB parameter. The user ID can be from 1 to 30 characters long. This parameter is required only when PGA_CAPABILITY is set to COMMIT_CONFIRM.</p> <p>There is no default value.</p>
PGA_RECOVERY_PASS	<p>The password to be used by the gateway when allocating an APPC conversation with the transaction specified by the PGA_RECOVERY_TPNAME parameter. The password can be from 1 to 8 characters long. This parameter is required only when PGA_CAPABILITY is set to COMMIT_CONFIRM and PGA_SECURITY_TYPE is set to PROGRAM. The password can be encrypted. For more information about encrypting the password, refer to Passwords in the Gateway Initialization File on page 14-6.</p> <p>There is no default value.</p>
PGA_RECOVERY_TPNAME	<p>The TP name of the transaction installed in the OLTP for commit-confirm FORGET and RECOVERY processing. The TP name can be from 1 to 64 characters long. For CICS Transaction Server for z/OS, the TP name is limited to four characters. For IMS/TM, the TP name is limited to eight characters. Other OLTPs might have other limits on the length of the TP name. This parameter is required only when PGA_CAPABILITY is set to COMMIT_CONFIRM.</p> <p>The default value is RECO.</p>
PGA_RECOVERY_USER	<p>The user ID to be used by the gateway when allocating an APPC conversation with the transaction specified by the PGA_RECOVERY_TPNAME parameter. The user ID can be from 1 to 8 characters long. This parameter is required only when PGA_CAPABILITY is set to COMMIT_CONFIRM and PGA_SECURITY_TYPE is set to PROGRAM or SAME.</p> <p>There is no default value.</p>
PGA_SECURITY_TYPE	<p>APPC conversation security option. This controls what security parameters are sent to the OLTP in the FMH-5 at conversation allocation. The following are valid values:</p> <p>NONE: Sends no security parameters</p> <p>SAME: Sends only a user ID</p> <p>PROGRAM: Sends a user ID and password</p> <p>The default is NONE.</p> <p>For further information on these options, refer to Chapter 14, "Security Requirements".</p>
PGA_SIGDANGER (AIX only parameter)	<p>Action to take on receipt of a SIGDANGER signal from the system indicating a shortage of paging space. The following is a valid value:</p> <p>IGNORE: Ignores the signal</p> <p>The default is IGNORE.</p>
TRACE_LEVEL	<p>PGA trace level. This controls tracing output written to stderr (the target of the LOG_DESTINATION parameter). The value must be an integer between 0 and 255.</p> <p>The default is 0, indicating no tracing.</p> <p>Any value between 1 and 255 will turn tracing on.</p>

A.1.2 PGA_CAPABILITY Parameter Considerations

When choosing a setting for the `PGA_CAPABILITY` parameter, take care to ensure that the correct setting is used based on what the remote transaction programs will be doing.

The `READ_ONLY` setting should always be used when the remote transaction programs are read-only, that is, when the remote transaction programs perform no database updates. `READ_ONLY` should never be used when the remote transaction programs perform database updates. For example, if the `READ_ONLY` setting is chosen and if a remote transaction program invoked by the gateway performs updates to a foreign database, then Oracle Integrating Server does not provide any integrity protection for those updates. Furthermore, the `READ_ONLY` mode enables a gateway transaction to be part of a distributed transaction that might update several other databases. If the gateway invokes a remote transaction program that performs updates in this situation and if a failure occurs, then the database updated by the remote transaction program is not synchronized with the other databases.

In cases where the remote transaction programs perform updates to foreign databases, there are two options for the value of `PGA_CAPABILITY`:

- `SINGLE_SITE`
- `COMMIT_CONFIRM`

Each of these options provides protection against data integrity problems by allowing `COMMIT` and `ROLLBACK` requests to be forwarded to the remote transaction program and by informing Oracle Integrating Server about the distributed update and recovery capabilities of the gateway. The particular option depends on the design of the remote transaction programs and on the capabilities of the OLTP (online transaction processor) where they execute.

If the OLTP has LU6.2 `SYNCLEVEL 1` or `2` support, then the `COMMIT_CONFIRM` capability provides limited two-phase commit between the Oracle Integrating Server and the OLTP, with the restriction that no other commit-confirm site (gateway or Oracle) can be part of the distributed transaction. If it is not possible to use `COMMIT_CONFIRM`, then the `SINGLE_SITE` capability provides update capability between Oracle Integrating Server and the OLTP, with the restriction that only the OLTP can perform updates and no updates can occur on the Oracle side.

Each of the `PGA_CAPABILITY` options for update control imposes specific requirements on the remote transaction program and on the OLTP. For `COMMIT_CONFIRM` capability, these requirements are discussed in detail in Chapter 5, "Implementing Commit-Confirm," of the *Oracle Procedural Gateway for APPC User's Guide*. Also refer to [Section 12.8.3, "Configuring the OLTP for Commit-Confirm"](#) on page 12-12 in this guide. For `SINGLE_SITE` capability, the remote transaction program is responsible for performing the required tasks in response to `COMMIT` and `ROLLBACK` requests received from the gateway on behalf of the Oracle Integrating Server. The gateway uses the `APPC_CONFIRM` and `SEND_ERR` requests to implement `COMMIT` and `ROLLBACK`, respectively. On receipt of a `CONFIRM`, the remote transaction program must perform `COMMIT` processing and then respond to the gateway with an `APPC_CONFIRMED` response. On receipt of `SEND_ERR`, the remote transaction program must perform `ROLLBACK` processing.

Because the distributed transaction capability of the Oracle Integrating Server is affected by the `PGA_CAPABILITY` option used by the gateway, it is desirable to separate inquiry and update applications by using different gateway instances for each. One gateway can be defined with `PGA_CAPABILITY` set to `READ_ONLY` and others with `PGA_CAPABILITY` set to `SINGLE_SITE` or `COMMIT_CONFIRM`.

This allows read-only transaction programs to participate in distributed transactions under the control of the Oracle Integrating Server. For example, data from DB2 can be retrieved through the `READ_ONLY` gateway by an inquiry-only remote transaction program and can then be used as input to database updates on the Oracle Integrating Server, all in one Oracle transaction. A `SINGLE_SITE` gateway can be used only for accessing remote transaction programs which perform updates to foreign databases outside the scope of the Oracle Integrating Server control. Data can be read from any databases accessible to the Oracle Integrating Server, and that data can be used to perform updates through the gateway.

When it is necessary to update resources on both the Oracle side and the OLTP side, a `COMMIT_CONFIRM` gateway can be used, provided that the OLTP and the remote transaction programs are set up to implement commit-confirm.

All that is necessary to set up multiple gateway instances is to set up the following for each instance:

- An entry in the `listener.ora` file defining the `sid` of the gateway instance
- An entry in the `tnsnames.ora` file defining an alias to be used to connect to the gateway instance defined in `listener.ora`
- A database link in Oracle Integrating Server that specifies the alias defined in the `tnsnames.ora` file in its `USING` parameter

Note that the gateway instances can share one common directory structure and use the same executables.

For example, to set up two gateways, `PGAI` and `PGAU` (for inquiry and update use, respectively), the following steps are required:

1. Define entries in `listener.ora` for two `sids`, `PGAI` and `PGAU`.
2. Define two aliases in `tnsnames.ora` that connect to the two new `sids`, `PGAI` and `PGAU`.
3. Define two database links in Oracle Integrating Server, one connecting to `PGAI` and the other connecting to `PGAU`.
4. Finally, create the initialization files `initPGAI.ora` and `initPGAU.ora`.

In `initPGAI.ora`, set `PGA_CAPABILITY` to `READ_ONLY`, and in `initPGAU.ora`, set `PGA_CAPABILITY` to `SINGLE_SITE` or `COMMIT_CONFIRM`. Then, use the `PGAI` gateway for inquiry-only transactions, and use the `PGAU` gateway for update transactions.

The same steps can be used to set up additional gateway instances.

A.1.3 PGA_CONFIRM Parameter Considerations

When deciding on the setting for the `PGA_CONFIRM` parameter, it is important to understand the effects of each setting. First, keep in mind that this parameter affects only those conversations running at `SYNCLEVEL 1`. The default setting, `PGA_CONFIRM=REJECT`, is correct for most applications. With this setting, the gateway generates an error if a `CONFIRM` request is received from the remote transaction program. If you have a remote transaction that uses `CONFIRM` to verify that data was received by the gateway, then you must use `PGA_CONFIRM=ACCEPT` to allow the gateway to respond to those incoming `CONFIRM` requests with `CONFIRMED` responses. You must be aware that the gateway sends `CONFIRM` requests to the remote transaction when the Oracle application has sent a `COMMIT` request. For the `COMMIT` processing to work correctly, the remote transaction must be written to perform its local commit processing whenever a `CONFIRM` request is received from the gateway,

and respond to the gateway with `CONFIRMED` after the commit processing has successfully completed. If an error occurs during commit processing, then the remote transaction must respond to the gateway with `SEND_ERR` to indicate that the commit failed.

One special case for the use of `PGA_CONFIRM=ACCEPT` is with IMS/TM version 7. When using the "implied APPC" support that is provided by IMS/TM version 7, conversations that run at `SYNCLEVEL 1` are handled differently than conversations that run at `SYNCLEVEL 0`. IMS/TM automatically generates `CONFIRM` requests after each APPC `SEND` when the conversation is at `SYNCLEVEL 1`. On the gateway side, if `PGA_CONFIRM=ACCEPT` is not specified, then the `CONFIRM` requests sent by IMS/TM result in errors generated by the gateway. Using `PGA_CONFIRM=ACCEPT` alleviates this problem, allowing the gateway to respond to incoming `CONFIRM` requests with `CONFIRMED` responses. The only limitation with running this way is that the implied APPC support provided by IMS does not notify the application when a `CONFIRM` request is received from the gateway. This means that the gateway cannot use `CONFIRM` to implement `COMMIT`, thereby disabling the use of `COMMIT` and `ROLLBACK` to control updates on the IMS side of the conversation.

A.1.4 Sample listener.ora file for a Gateway Using SNA

```
LISTENER =
  (ADDRESS_LIST =
    (ADDRESS=
      (COMMUNITY= TCP.world)
      (Host = bay)
      (PROTOCOL= TCP)
      (Port= 2621)
    )
    (ADDRESS=
      (COMMUNITY= TCP.world)
      (Host = bay)
      (PROTOCOL= TCP)
      (Port= 2623)
    )
  )

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PGA)
      (ORACLE_HOME = /oracle/pga/10.2)
      (PROGRAM = pg4asrv)
    )
  )
```

A.1.5 Sample tnsnames.ora file for a Gateway Using SNA

```
ORA920 =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = bay.us.oracle.com)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = ORA920.bay)
    )
  )
```



```
PGA =  
  (DESCRIPTION =  
    (ADDRESS_LIST =  
      (ADDRESS = (PROTOCOL = TCP) (HOST = bay) (PORT = 2621))  
    )  
    (CONNECT_DATA =  
      (SID = PGA)  
    )  
    (HS = OK)  
  )
```

Gateway Initialization Parameters for TCP/IP Communication Protocol

This appendix describes the parameters needed for a gateway featuring the TCP/IP for IMS Connect communication protocol. It also provides a sample output of the `pg4tcpmap` tool. In addition, this appendix contains sample `listener.ora` and `tnsnames.ora` files for a gateway using TCP/IP. It contains the following sections:

- ["Gateway Initialization Parameter File Using TCP/IP"](#) on page B-1
- ["Output for the pg4tcpmap Tool"](#) on page B-3

B.1 Gateway Initialization Parameter File Using TCP/IP

The parameter file for the Oracle Procedural Gateway for APPC using TCP/IP for IMS Connect is located in the `$ORACLE_HOME/pg4appc/admin` directory and is called `initsid.ora`.

Note: The `initsid.ora` file contains both SNA and TCP/IP parameters. You will need to modify these files with the suitable parameters.

B.1.1 PGA Parameters

The PGA parameters control the TCP/IP interface portion of the gateway. PGA parameters are specified using the `SET` gateway initialization parameter. For example:

```
SET pga_parm=value
```

where:

- `pga_parm` is one of the PGA parameter names in the list that follows
- `value` is a character string with contents that depend on `pga_parm`

[Table B-1](#) provides a list of PGA parameters and their descriptions.

Table B-1 PGA Parameters on Gateway Using TCP/IP for IMS Connect

Parameter	Description
LOG_DESTINATION= <i>logpath</i>	<p><i>logpath</i> specifies the destination at which STDERR is reopened. LOG_DESTINATION specifies a directory only and stderr is reopened to <i>logpath/sid_pid.log</i></p> <p>where:</p> <ul style="list-style-type: none"> ▪ <i>sid</i> is the sid name ▪ <i>pid</i> is the process ID assigned to the gateway <p>NOTE: This parameter will be used for the <code>pg4tcpmap</code> tool when you set the Trace Level to 255. The log file for this tool will reside in the same place as the gateway log file, in <code>pg4tcpmap_sid.log</code>.</p>
PGA_CAPABILITY	<p>PGA transaction capability. The following are valid values:</p> <p>READ_ONLY or RO: Read-only capabilities.</p> <p>SINGLE_SITE or SS: Single-site update only. This indicates that in a distributed environment, only the gateway can perform updates. No other database updates can occur within the Oracle transaction.</p> <p>The default is SINGLE_SITE.</p>
PGA_SECURITY_TYPE	<p>TCP/IP conversation security option. This controls what security parameters are sent to the OLTP. The following are valid values:</p> <p>NONE: Sends no security parameters</p> <p>PROGRAM: Sends a user ID and password</p> <p>The default is NONE.</p> <p>For further information on these options, refer to Chapter 14, "Security Requirements".</p> <p>Important: You must specify your RACF group name through the <code>pg4tcpmap</code> tool if you have set your PGA security option to SECURITY=PROGRAM. For more information about this issue, refer to the <i>Oracle Procedural Gateway for APPC User's Guide</i>.</p> <p>If you have already loaded the table <code>pga_tcp_ims</code> and you did not first specify the RACF group name, delete the row and reinsert it with the value for the RACF group name.</p>
PGA_TCP_DB	<p>The Oracle Net service name for the Oracle server in which the gateway receives its TCP/IP for IMS Connect information, such as host name and port number. This parameter can be from 1 to 255 characters long. This parameter is required.</p> <p>There is no default value.</p>
PGA_TCP_PASS	<p>The Oracle password to be used by the gateway when connecting to the Oracle server specified by the PGA_TCP_DB parameter. The password can be from 1 to 30 characters long. This parameter is required. The password can be encrypted. For more information about encrypting the password, refer to Passwords in the Gateway Initialization File on page 14-6.</p> <p>There is no default value.</p>

Table B-1 (Cont.) PGA Parameters on Gateway Using TCP/IP for IMS Connect

Parameter	Description
PGA_TCP_USER	The Oracle user ID to be used by the gateway when connecting to the Oracle server specified by the PGA_TCP_DB parameter. The user ID can be from 1 to 30 characters long. This parameter is required. There is no default value.
TRACE_LEVEL	PGA trace level. This controls tracing output written to stderr (the target of the LOG_DESTINATION parameter). The value must be an integer from 0 to 255. The default is 0, indicating no tracing. NOTE: This parameter is used in the pg4tcpmap tool as well as the gateway.

B.2 Output for the pg4tcpmap Tool

The following output illustrates the results from executing the pg4tcpmap tool when running TCP/IP for IMS Connect on the gateway. Refer to [Section 13.8, "Loading the PGA_TCP_IMSC Table"](#) on page 13-12 of this guide and to Chapter 6 of the *Oracle Procedural Gateway for APPC User's Guide* for detailed information about the function and parameters of the pg4tcpmap tool.

Note that input in this sample is shown within angle brackets (<>).

```
$ pg4tcpmap

PG4TCPMAP: Release 10.2.0.1.0 - Production on Fri Jun 3 14:57:41 2005

Copyright(c) 1982,2005 Oracle. All rights reserved.

This tool takes the IMS Connect TCP/IP information, such as host name
and port number and maps them to your TIPS.

You may use this tool to insert or delete IMS Connect TCP/IP
information.
If you want to insert a row, type I
If you want to delete a row, type D
<i>
Enter the Unique Side Profile.
pgaimst
Enter either the remote hostname or its TCP/IP address.
<mvs08>
Enter the IMS Connect port number.
<9900>
Do you want to select a Conversational Protocol?(Y|N)
The default is NO, 'no request for acknowledgment or deallocation'
<n>
Enter one of the following letters for Timer.
For .25 second, enter 'D'.
For .01 to .25 second, enter 'S'.
For 'does not set the timer, no wait occurs', enter 'N'.
For Receive waits indefinitely, enter 'I'.
The default is 'D'.
<d>
Enter one of the following letters for 'socket connection type'.
For transaction socket, enter 'T'.
For persistent socket, enter 'P'.
```

```

For non-persistent socket, enter 'N'.
The default is 'T'.
<t>
Do you want to enter the client ID name? (Y|N)
If NO, IMS Connect (user exit) will generate it.
<n>
Enter one of the following letters for 'commit mode'.
For Commit Mode set to 0, enter '0'.
For Commit Mode set to 1, enter '1'.
The default is '1'.
<1>
Enter the Datastore name (IMS Destination ID).
The maximum string length is 8 and the Datastore name must be
specified.
<imse>
Do you want to enter the LTERM? (Y|N)
If NO, the default is blank.
<n>
Do you want to enter the RACF group name? (Y|N)
If NO, the default is blank.
<n>
Requested to INSERT a row.
'Side Profile name' is 'PGAIMST'
'remote host name' is 'MVS08'
'IMS Connect port number' is '9900'
'conversational protocol' is ' '
'Timer' is 'D'
'socket connection type' is 'T'
'client ID' is ' '
'commit mode' is '1'
'Datastore name (IMS destination ID)' is 'IMSE '
'IMS LTERM override' is ' '
'RACF group name' is ' '
pg4tcpmap is complete.

```

B.2.1 Sample listener.ora File for a Gateway Using TCP/IP

The following is an example of a listener.ora file for a gateway using TCP/IP:

```

LISTENER =
  (ADDRESS_LIST =
    (ADDRESS=
      (COMMUNITY= TCP.world)
      (Host = bay)
      (PROTOCOL= TCP)
      (Port= 2621)
    )
    (ADDRESS=
      (COMMUNITY= TCP.world)
      (Host = bay)
      (PROTOCOL= TCP)
      (Port= 2623)
    )
  )

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PGA)

```

```
(ORACLE_HOME = /oracle/pga/10.2)
(PROGRAM = pg4t4ic)
)
)
```

B.2.2 Sample tnsnames.ora File for a Gateway Using TCP/IP

The following is an example of a `tnsnames.ora` file for a gateway using TCP/IP:

```
ORA920 =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = bay.us.oracle.com) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = ORA920.bay)
    )
  )

PGA =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = bay) (PORT = 2623))
    )
    (CONNECT_DATA =
      (SID = PGA)
    )
    (HS = OK)
  )
```

Gateway Terminology

This appendix contains a list of terms and definitions pertaining to the gateway and its components and function.

For a list of other terms and definitions associated with the gateway, refer to Chapter 1 of the *Oracle Procedural Gateway for APPC User's Guide*.

This appendix contains the following section:

- ["Gateway Term Definitions"](#) on page C-1

C.1 Gateway Term Definitions

This section includes the following terms and definitions used in this edition of the *Oracle Procedural Gateway for APPC Installation and Configuration Guide*:

- [Gateway Initialization File](#)
- [Gateway Remote Procedure](#)
- [tg4pwd](#)
- [pg4tcpmap tool](#)
- [PGA \(Procedural Gateway Administration\)](#)
- [PGDL \(Procedural Gateway Definition Language\)](#)
- [PL/SQL Stored Procedure Specification \(PL/SQL package\)](#)
- [UTL_RAW PL/SQL Package \(the UTL_RAW Functions\)](#)
- [UTL_PG PL/SQL Package \(the UTL_PG Functions\)](#)

Gateway Initialization File

This file is known as `initsid.ora` and it contains parameters that govern the operation of the gateway. If you are using the SNA protocol, refer to [Appendix A, "Gateway Initialization Parameters for SNA Protocol"](#) for more information. Refer to [Appendix B, "Gateway Initialization Parameters for TCP/IP Communication Protocol"](#) if your protocol is TCP/IP.

Gateway Remote Procedure

The Oracle Procedural Gateway for APPC provides prebuilt remote procedures. In general, the following three remote procedures are used:

- `PGAINIT`, which initializes transactions
- `PGAXFER`, which transfers data

- PGATERM, which terminates transactions

Refer to ["RPC Functions"](#) in this guide and to Appendix C, "Gateway RPC Interface" in the *Oracle Procedural Gateway for APPC User's Guide* for more information about gateway remote procedures.

tg4pwd

tg4pwd is a utility which encrypts passwords that are normally stored in the gateway initialization file. Passwords are stored in an encrypted form in the password file, making the information more secure. Refer to ["Passwords in the Gateway Initialization File"](#) for detailed information about how the tg4pwd utility works.

pg4tcpmap tool

This gateway mapping tool is applicable only when the gateway is using TCP/IP support for IMS Connect. Its function is to map the Side Profile Name to TCP/IP and IMS Connect attributes into the PGA_TCP_IMSC table.

PGA (Procedural Gateway Administration)

PGA is a general reference within this guide to all or most components constituting the Oracle Procedural Gateway for APPC. This term is used when references to a specific product or component are too narrow.

PGDL (Procedural Gateway Definition Language)

PGDL is the collection of statements used to define transactions and data to the PGAU.

PL/SQL Stored Procedure Specification (PL/SQL package)

This is a precompiled PL/SQL procedure that is stored in Oracle Integrating Server.

UTL_RAW PL/SQL Package (the UTL_RAW Functions)

This component of the gateway represents a series of data conversion functions for PL/SQL RAW variables and remote host data. The types of conversions performed depend on the language of the remote host data. Refer to "UTL_RAW Functions" in Appendix D of the *Oracle Procedural Gateway for APPC User's Guide* for more information.

UTL_PG PL/SQL Package (the UTL_PG Functions)

This component of the gateway represents a series of COBOL numeric data conversion functions. Refer to NUMBER_TO_RAW and RAW_TO_NUMBER argument values in Appendix D of the *Oracle Procedural Gateway for APPC User's Guide* for supported numeric data type conversions.

Summary of Changes in Previous Versions

This appendix provides a summary of changes in previous versions of the Oracle Procedural Gateway for APPC. This appendix contains the following sections:

- ["Changes and Enhancements in Previous Versions"](#) on page D-1
- ["Corrected Problems in Previous Versions"](#) on page D-4
- ["Known Restrictions in Previous Releases"](#) on page D-7

D.1 Changes and Enhancements in Previous Versions

The following sections list changes and enhancements that were made to previous releases of the gateway.

D.1.1 Release 10.2.0.1.0

The following section lists changes and enhancements that were made to Release 10.2.0.1.0 of the gateway:

D.1.1.1 pg4hoa1.sh Bootstrap Shell Script No Longer Used When Migrating

Users migrating to Release 10.1.0.2.0 from a previous release of the gateway will no longer use the `pg4hoa1.sh` bootstrap shell script when defining `listener.ora`. In earlier releases, the value used when defining `listener.ora` was `pg4hoa1.sh`. You will now use the `pg4asrv` program directly.

D.1.1.2 SunLink Peer-to-Peer No Longer Supported on Solaris

SunLink cannot support 64-bit architecture and is no longer regarded as a strategic product for the Solaris platform. Therefore, with this release of the Oracle Procedural Gateway for APPC, the SunLink P2P SNA protocol is no longer supported for the Solaris Operating System. Instead, you must use the SNAP-IX SNA communication product.

D.1.1.3 AIX: IBM Comm. Server on AIX 5L Does Not Support SNA 32-bit Libraries

APPC API calls are not supported for an SNA Communication Server 6.0.1 running on AIX 5L.

Therefore, with this release of the gateway:

1. Two-Phase Commit is no longer supported. As a result:
 - the password encryption utility for Resource Recovery Manager (`pg4rrmpwd`) is no longer supported;

- the Resource Recovery Manager (pg4arm) is no longer supported.
2. In addition, with this release of the gateway, the value DEALLOCATE is no longer supported for the PGA_SIGDANGER parameter.

D.1.1.4 TCP/IP Support for Connecting PG4APPC to IMS Connect Release 1.2

This release of the gateway includes TCP/IP support for IMS Connect. IMS Connect is an IBM product which allows TCP/IP clients to trigger execution of IMS transactions. The gateway can use a TCP/IP communication protocol to access IMS Connect, which triggers execution of IMS transactions. There is no SNA involvement with this configuration. Related to this new feature of the gateway is:

- The pg4tcpmap tool. This release of the gateway includes a new tool whose purpose is to map the information from your Side Profile Name to TCP/IP and IMS Connect attributes. You might turn on a trace for this tool as well. For more information about the gateway mapping tool, refer to Chapter 6 of the *Oracle Procedural Gateway for APPC User's Guide* and to [Chapter 13, "Gateway Configuration Using TCP/IP Communication Protocol"](#) in this guide.

D.1.1.5 Improved Gateway Log File

The log file for the gateway remote application has been improved.

D.1.2 Release 9.2.0.1.0

The following section lists changes and enhancements that were made to Release 9.2.0.1.0 of the gateway:

- Oracle Call Interface: Because of changes to the Oracle Call Interface (OCI), Oracle Procedural Gateway for APPC internal codes for calling COMMIT/CONFIRM have changed. This does not affect user functions.
- APPC log file: The Oracle Procedural Gateway for APPC log file is improved and provides more information about transactions. For a sample of the new log file, refer to Chapter 6 of the *Oracle Procedural Gateway for APPC User's Guide for UNIX, Release 9.2.0.1.0*.
- Password encryption utility for gateway initialization file: This release of the gateway includes a utility to support encryption of plain text passwords in the gateway initialization file.
- AIX-only: password encryption utility for Resource Recovery Manager: This release of the gateway includes another utility to support encryption of plain text passwords, which is called LOG_PASS in the initialization file for local LU. For more information, refer to "AIX-Based Systems Only: Using the pg4rrmpwd Utility" on page 5-6 in *Oracle Procedural Gateway for APPC Installation and Configuration Guide, Release 9.2.0.1.0*.

D.1.3 Release 9.0.1.0.1

The following section lists changes and enhancements that were made to Release 9.0.1.0.1 of the gateway:

- PGA_TRACE parameter no longer supported: With this release of Oracle Procedural Gateway for APPC, the PGA_TRACE parameter is no longer supported. If you are copying the `initsid.ora` file from an older version of the gateway, then you must remove this parameter.

- `TRACE_LEVEL` and `LOG_DESTINATION` parameters moved: With this release of Oracle Procedural Gateway for APPC, the `TRACE_LEVEL` and `LOG_DESTINATION` parameters were moved from the `pg4hoa1.sh` file to the `initsid.ora` file.
- `FDS_CLASS_VERSION` parameter added: You need to add the `FDS_CLASS_VERSION` parameter to your startup shell script file (`pg4hoa1.sh`). A default value is specified in `pg4hoa1.sh`.

D.1.4 Release 8.0.6.1.0 and 8.0.4.1.0

The following section lists changes and enhancements that were made to Releases 8.0.6.1.0 and 8.0.4.1.0 of the gateway:

- Heterogeneous Services architecture

This release of Oracle Procedural Gateway for APPC utilizes the Oracle Heterogeneous Services component within the Oracle8 server. Heterogeneous Services is the building block for the next generation of Oracle Open Gateways.
- Performance enhancements

Oracle Procedural Gateway for APPC contains several internal performance enhancements. This product has shown major improvements over the version 4 gateways, particularly in response time and CPU utilization for all relevant address spaces for a variety of workloads. The actual performance improvement at your site may vary, depending on your installation type and workload.
- PL/SQL V3 compatibility

Before Release 8.0.6.1.0, you had to use the `PLSQL_V2_COMPATIBILITY=TRUE` parameter to compile PGAU-generated TIP specifications. This is no longer necessary.
- Oracle server dependencies

This release of Oracle Procedural Gateway for APPC requires that the Oracle Integrating Server be an Oracle8 server Version 8.0.6.2.0 (Oracle database server release 8.0.6.2.0) or later.
- Gateway initialization parameters

In previous versions of the gateway, the initialization parameters were stored in files named `initsid.gtwboot` and `initsid.ora`, both found in the gateway instance directories. With version 8, most parameters that were in `initsid.gtwboot` have been moved to the `initsid.ora` file. The syntax of `initsid.ora` has been simplified.

In addition, Oracle Procedural Gateway for APPC's executable (`pg4asrv`) calls the gateway indirectly through a startup shell script sample file called `pg4hoa1.sh`. This startup shell script is also used for setting SNA environmental variables, such as `APPCLU`.
- `tnsnames.ora`: The service name definition (from the Oracle Integrating Server to the gateway) must contain an extra parameter, (`HS=`).

D.1.5 Release 4.0.1.1.0

The following section lists changes and enhancements that were made to Release 4.0.1.1.0 of the gateway:

- Gateway components now based on Oracle7 Server release 7.3.2

The gateway components are now built using the Oracle7 server release 7.3.2 product libraries as a base. All product components shipped with the gateway, including SQL*Net, are now at the release 7.3.2 level.

- Gateway supports Oracle8 Server release 8.0.3
The gateway now supports Oracle8 server release 8.0.3 as an integrating server.
- UTL_PG now shipped with Oracle Server
The scripts for installing the UTL_PG PL/SQL package are now shipped with the Oracle7 or Oracle8 server and are no longer shipped with the gateway.
- Gateway no longer supports Oracle7 Server release 7.1.6 and 7.2
The gateway no longer supports Oracle7 server release 7.1.6 and 7.2 as integrating servers.
- User ID and password overrides now supported by gateway (Enh. No. 423300)
A new TIP override is now available to allow the user ID and password sent by the gateway to the OLTP (online transaction processor) to be specified by the user. For complete information on using this override, refer to "Overriding TIP Initializations" in Chapter 2 of the *Oracle Procedural Gateway for APPC User's Guide*.
- New "LENGTH IS" clause supported by PGAU (Enh. No. 428783)
A new "LENGTH IS" clause in COBOL data definitions is now recognized by PGAU and allows the definition of variable-length character fields without using OCCURS DEPENDING ON. For complete information on using this clause, refer to "Format Conversion" in Appendix G of the *Oracle Procedural Gateway for APPC User's Guide*.
- Date/time stamp in trace output
All trace and debug output from the gateway server now contains a date/time stamp in the beginning of each line.
- Installer improvements: The installer prompts not relevant to the gateway have been eliminated to streamline the installation process.

D.2 Corrected Problems in Previous Versions

The following tables list the numbers and descriptions of each bug fixed in previous releases of the gateway.

D.2.1 Bugs Fixed in Release 10.2.0.1.0

Table D-1 lists the bugs that were fixed in release 10.2.0.1.0 of the gateway.

Table D-1 Bugs Fixed in Release 10.0.1 of the Gateway

Bug Number	Description
1310776	PG4APPC to support TCP/IP connectivity to IMS.
1472845	Receiving message ORA-1401: handling multiple items in one call to PGAXFER causes error.
1847587	Receiving message ORA-2047: executing procedures against read-only gateways.
1853035	The gateway generates the wrong message (PGA-20936) instead of generating message number PGA-20938.

Table D-1 (Cont.) Bugs Fixed in Release 10.0.1 of the Gateway

Bug Number	Description
1919413	Receiving an ORA-28528 error (Heterogeneous Services data type conversion error) when sending a buffer of more than 4K of data to CICS.
1920013	Test case tgpsm#34 fails with error message PGA-20910, rc=219, and errno = 9 instead of PGA-20934.
2088612	Receiving message ORA-1401: inserted value too large for column.
2222386	Transferring more than 32760 bytes in a RAW field. Customer receives error message ORA-01401: "inserted value too large for column". When the gateway attempts to transfer 32760 or more bytes back to Oracle in a single transfer, the result is ORA-02055: "distributed update operation failed; rollback required".
2505566	PGDDVCHK.SQL does not require spooling file.
2526386	CMDEAL not called in certain error situations in PGAXFER.
2691119	ORA-06510: PL/SQL: Unhandled user-defined exception when using PGATRAC.
3067506	Savepoint and rollback savepoint are not supported by PG4APPC.

D.2.2 Release 9.2.0.1.0

Table D-2 lists the bugs that were fixed in Release 9.2.0.1.0 of the gateway, along with their descriptions.

Table D-2 Bugs Fixed in Release 9.2.0.1.0

Bug Number	Description
1276298	When using an invalid value for receive parameter for the pgaxfer procedure, the user receives an ORA 28511 error message.
1302253	Oracle Procedural Gateway for APPC procedure statements executed on the Oracle integrating servers would hang in SQL*Plus, without giving any messages.
1336805	When the PGA_SECURITY_TYPE parameter was set to PROGRAM and the PGA_CAPABILITY parameter was set to COMMIT_CONFIRM, the user received an authentication error but could not see who was complaining about it.
1404454	The pg4appc log file should be improved and have the received buffer in the log file.
1411694	The user receives message ORA-28527 when the PGA_CAPABILITY is set to READ_ONLY.
1472800	Multi-row queries failed on the following error messages: ORA-01401 and ORA-06512.
1519088	User received sporadic abends when inserting CICS records to VSAM file.
1677939	Oracle Procedural Gateway for APPC would partially transfer low values to VSAM files.
1722467	When the PGA_SECURITY_TYPE parameter was set to PROGRAM and the user specified the user ID and password through database link explicit CONNECT information, the query failed with message PGA-20910 RC=6.

Table D–2 (Cont.) Bugs Fixed in Release 9.2.0.1.0

Bug Number	Description
1724988	When the programmer used RPC PGAINIT_SEC, no matter what synclevel was being used, the following message was received: "invalid SYNCHLEVEL, 152, specified; valid range is 0:1."
2092204	When using PGATCTL procedure with specifying the function code as "D", the user receives the message PGA-20932 with an invalid function code.
2128785	Oracle Server Heterogeneous Services place unwanted trace files into /tmp directory.

D.2.3 Release 9.0.1.0.1

Table D–3 lists the bugs that were fixed in Release 9.0.1.0.1 of the gateway, along with their descriptions.

Table D–3 Bugs Fixed in Release 9.0.1.0.1

Bug Number	Description
1276298	When using an invalid value for the receive parameter for the pgaxfer procedure, the user receives an ORA-28511 error message.
1302253	Oracle Procedural Gateway for APPC procedure statements carried out on the Oracle integrating servers would suspend in SQL*Plus, without giving any messages.
1336805	When the PGA_SECURITY_TYPE parameter was set to PROGRAM and the PGA_CAPABILITY parameter was set to COMMIT_CONFIRM, the user received an authentication error but could not see who was complaining about it.
1404454	The pg4appc log file should be improved and have the received buffer in the log file.
1411694	The user receives message ORA-28527 when the PGA_CAPABILITY is set to READ_ONLY.
1472800	Multirow queries failed on the following error messages: ORA-01401 and ORA-06512.
1519088	User received sporadic abends when inserting CICS records to VSAM file.
1677939	Oracle Procedural Gateway for APPC would partially transfer low values to VSAM files.
1722467	When the PGA_SECURITY_TYPE parameter was set to PROGRAM and the user specified the user ID and password through database link explicit CONNECT information, the query failed with message PGA-20910 RC=6.
1724988	When the programmer used RPC PGAINIT_SEC, no matter what synclevel was being used, the following message was received: "invalid SYNCHLEVEL, 152, specified; valid range is 0:1."

D.2.4 Release 8.0.6.1.0 and 8.0.4.1.0

Table D–4 lists the bugs that were fixed in Releases 8.0.6.1.0 and 8.0.4.1.0 of the gateway, along with their descriptions.

Table D-4 Bugs Fixed in releases 8.0.6.1.0 and 8.0.4.1.0

Bug Number	Description
1329386 (fixed in Release 8.0.6.1.0)	PGAU core dumped on large packages. When users tried to generate large packages with PGAU, they got a core dump.
561128	The PLSTYPE attribute of binary integers (COBOL COMP data items) had the wrong precision when the data was defined without COMPOPTS('TRUNC(BIN)') specified.
599696	PGAU COBOL parser did not accept OCCURS clause appearing before data type as valid syntax.
603934	When COBOL data definitions contained OCCURS and REDEFINES, the PGAU-generated .pkb file was invalid.
689304	This bug occurred when a CICS transaction abended between the time that the Oracle server called the gateway to perform a commit and the time that the CICS transaction was successfully committed. When this occurred, the Oracle recovery process (RECO) would call the gateway's recovery function (GTARECO) to resolve the transaction. The GTARECO function mistakenly relayed that it had successfully rolled back the CICS transaction, but in fact it had failed to delete the row from PGA_CC_PENDING.

D.2.5 Release 4.0.1.1.0

Table D-5 lists the bugs that were fixed in Release 4.0.1.1.0 of the gateway, along with their descriptions.

Table D-5 Bugs Fixed in Release 4.0.1.1.0

Bug Number	Description
420391	PGAU generated invalid PL/SQL in the TIP when a COBOL definition contained an OCCURS clause and was defined as an IN OUT parameter. A PLS-103 error was generated when attempting to compile the TIP.
460960	ORA-9199 errors were generated by the gateway after some gateway exceptions due to a memory overlay.
473360	PGAU encountered a segment fault when executing a GENERATE command for a transaction with a very large number of fields defined in its COBOL data.
506777	An ORA-1017 error was generated by the gateway when RECO attempted to log onto the gateway during recovery of a distributed transaction, though the gateway was running in the READ-ONLY mode.
444471	The installation of the gateway failed while linking the gateway executables if the C compiler was not installed on the UNIX system.
474889	The gateway server encountered a segment fault during installation verification if the patch for bug number 444771 was installed. The patch contained incorrect linker parameters.

D.3 Known Restrictions in Previous Releases

The following section lists restrictions that were known to exist in previous releases of the gateway.

D.3.1 Release 9.2.0.1.0

AIX-Based systems only: two-phase commit provides no automatic recovery with CICS/ESA: When an update transaction with CICS/ESA at LU6.2 Sync Level 2 fails, CICS always rolls back or commits the transaction (based on the INDOUBT parameter of the CICS transaction definition), making automatic recovery by the Oracle Integrating Server impossible.

D.3.2 Release 8.0.6.1.0 and 4.0.1.1.0

The following problems in other products are known to affect the operation of products in this release:

- UTL_PG Does Not Handle Numeric Data Values Between 0 and +1 (Bug No. 360427)

The UTL_PG.NUMBER_TO_RAW function erroneously issues messages ORA-08414 and ORA-08467 when converting an Oracle number between 0 and +1 into a COBOL numeric value. Contact Oracle Support Services to order the patch correcting this bug.
- PGA Grows Excessively When Using Indexed PL/SQL Tables (Bug No. 373700)

When TIPs are executed that use large PL/SQL tables, the memory usage grows excessively. Contact Oracle Support Services to order the patch correcting this bug.
- UTL_PG Does Not Handle Leading or Trailing Signs Correctly for DISPLAY Data (Bug No. 383510)

The UTL_PG.RAW_TO_NUMBER and UTL_PG.NUMBER_TO_RAW functions do not handle leading or trailing signs correctly for numeric data defined with USAGE IS DISPLAY. Contact Oracle Support Services to order the patch correcting this bug.

Configuration Worksheet

Table E-1 lists the parameter names and the reasons you will need them to configure the gateway and the communications interface you have chosen (either SNA or TCP/IP). You can use this table as a worksheet to gather the specific information you need before you begin the configuration process.

Ask your systems administrator to provide you with any parameter names you do not know.

Table E-1 Parameters for Configuring Gateway and Communication Protocols

Name of Parameter Needed	Purpose	Your Specific Parameters Here
ORACLE_HOME	For: gateway's Oracle home	_____
ORACLE_SID	For: gateway's system ID	_____
Any Security Options Needed	For: SNA remote LU properties options	_____
Appropriate Name for Each Side Information Profile	for: SNA creating CPI-C symbolic destination names (side information profiles), general information	_____
Appropriate Mode	For: SNA	_____
TP Name	For: SNA partner information in CPI-C name properties	_____
Partner LU Name Alias	For: SNA partner information in CPI-C name properties	_____
Unique Side Profile Name	For: Configuring TCP/IP support for IMS Connect	_____
Remote Host name or TCP/IP Address	For: Configuring TCP/IP support for IMS Connect, pg4tcmap tool	_____
IP Address	For: Configuring TCP/IP support for IMS Connect, pg4tcmap tool	_____

Table E-1 (Cont.) Parameters for Configuring Gateway and Communication Protocols

Name of Parameter Needed	Purpose	Your Specific Parameters Here
IMS Connect Port Number	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	_____
Conversational Protocol (Y/N)	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	_____
Timer choice:		
a) .25	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	
b) .01 to .25		
c) Does not exist		
d) Receives wait		_____
Socket Connection Type Choice:		
a) Transaction	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	
b) Persistent		
c) Nonpersistent		_____
IMS Client ID Name	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	_____
IMS Commit Mode Choice:		
a) 0	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	
b) 1		_____
IMS Destination ID, data store name	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	_____
LTERM	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	_____
RACF Group Name	For: Configuring TCP/IP support for IMS Connect, pg4tcpmap tool	_____

Symbols

\$ cd \$ORACLE_HOME \$ mkdir pg4appc, 12-3, 13-3
\$ORACLE_HOME, 4-11, 11-2, 11-3, 11-4, 12-10,
12-15, 12-16, 12-18, 13-11
\$ORACLE_HOME/bin/pg4asrv, 14-4
(HS=) (TNSNAMES parameter for Oracle Net), 5-1
with TCP/IP protocol, 5-2

A

action items
for installing the gateway, 4-8
activating and verifying SNA server profiles, 7-6
AIX
activating profiles, 7-6
configuring SNA, 5-2
creating mode profiles, 7-4
processing inbound connections, 7-1
SNA conversation security, 14-3
SNA security option SECURITY=SAME, 14-4
SNA security option SECURITY=NONE, 14-3
SNA security option
SECURITY=PROGRAM, 14-4
SNA security validation, 14-3
System Management Interface Tool, 7-2
AIX-based communications package
SNA server, 7-1
ALTER USER command, 12-5, 13-5
american_america_us7ascii, 15-4, 15-5
APPC, 1-1
conversation security option, A-3, B-2
appdef.doc file, 11-3
APPC/MVS
installation verification, 12-16
verification of configuration, 12-16
architecture
components of the gateway, 1-5
ASCII
automatic conversion, 1-3
authentication
for operating system, 14-2
for Oracle, 14-2
types, security, 14-2

B

backout possibilities during migration, 15-2
Bourne shell, 4-5
bugs
bypassing a bug in IDMS-DC, 11-3
fixed in previous releases, D-6
fixed in this release, 2-1, D-4

C

CDMSLIB DD statement, 11-3
CD-ROM drive requirements, 3-2
changes
in this release, 2-1
release 4.0.1.1.0, D-3
release 8.0.6.1.0, D-3
release 9.0.1.0.1, D-2
release 9.2.0.1.0, D-2
choosing the device type
on Linux, 6-4
CICS, 1-9, 3-4
ATTACHSEC parameter
on HP-UX, 8-1
on Solaris, 9-1
installation verification
on gateway using SNA, 12-14
security options not supported by the gateway
on HP-UX, 8-1
on Solaris, 9-1
transaction ID, 12-14
verifying configuration
on gateway using SNA, 12-14
CICS Transaction Server for z/OS
authentication mechanism
on all platforms, 14-4
OLTP
configuration verification, 12-14
TP name length, A-3
command
SET, 12-10, 13-11
COMMIT, A-4
COMMIT_CONFIRM, A-2, A-4, B-2
and PGA_CAPABILITY parameter, A-4
capability, A-4
commit-confirm, A-4

- configuring, 12-11, 12-12
 - gateway initialization parameters, 12-12
 - OLTP, 12-12
 - Oracle server, 12-11
 - sample applications, 12-16
 - transaction log, 12-17
- COMMIT/ROLLBACK, 3-4
- communications
 - between server, gateway and remote host, 1-7
 - needed for HP-UX, 3-3
 - needed for Solaris, 3-3
- components
 - in this release, 2-1
- configuration
 - gateway directories, 13-2
- configuration verification
 - OLTP
 - on gateway using SNA, 12-12
 - on gateway using TCP/IP for IMS Connect, 13-12, 13-13
- configuring
 - commit-confirm, 12-11, 12-12
 - gateway
 - optional steps to allow multiple users using SNA, 12-7
 - optional steps to allow multiple users using TCP/IP for IMS Connect, 13-8
 - gateway directories, 12-2
 - Oracle integrating server, 12-11
 - upgrading from previous releases, 12-6, 13-7
 - Oracle integrating server for gateway using TCP/IP for IMS Connect
 - pre-configuration steps, 13-2
 - SNAP-IX, 9-1
 - TCP/IP for IMS Connect, 10-1
 - on the gateway, 13-11
 - the gateway
 - for TCP/IP for IMS Connect, 13-1
 - using SNA, 12-1
 - the OLTP, 11-1
 - your network
 - using SNA, 5-1
- configuring a LAN device
 - on Linux, 6-4
- Configuring APPC/MVS, 11-4
- Configuring CICS Transaction Server for z/OS, 11-1
- Configuring IDMS-DC, 11-2
- configuring your network, 5-1
- CONNECT clause, 14-4
 - for database link security, 14-3
 - in TCP/IP security, 14-5
- corrected problems
 - release 4.0.1.1.0, D-7
 - release 8.0.6.1.0 & 8.0.4.1.0, D-6
 - release 9.0.1.0.1, D-6
 - release 9.2.0.1.0, D-5
- CPI-C, 12-18
- CPI-C profiles
 - creating
 - on HP-UX, 8-5

- on Solaris, 9-5
 - creating, on Solaris, 9-4
- creating
 - mode profiles, on AIX, 7-4
 - new product installation directory, 4-5
 - public database link, 12-5, 13-5
 - SNAPPlus2 definitions, 8-3
 - SNAPPlus2 profiles, 8-2
- creating devices
 - on Linux systems, 6-4
- creating IBM communications server definitions for
 - the gateway
 - on Linux, 6-3
- creating local LUs
 - on Linux, 6-5
- creating partner LUs
 - on Linux, 6-5
- creating peer connections
 - on Linux, 6-4
- creating the configuration
 - on Linux systems, 6-3
- creating the CPI-C side information profile
 - on Linux, 6-5
- creating the node
 - on Linux systems, 6-3

D

- data dictionary
 - See PG DD
- data exchange
 - PGAXFER function, 1-8
- database link, 1-7, A-5
 - creating, 12-5, 13-5
 - when configuring integrating server, 12-5, 13-5
 - dropping, when deinstalling the gateway, 4-11
 - in configuring the network, 5-1
 - in verifying gateway installation
 - on gateway using SNA, 12-13
 - on gateway using TCP/IP for IMS Connect, 13-12
 - public and private, 14-2
 - security, CONNECT clause, 14-3
- database link name
 - modifying .sql files, 12-13, 13-12
- datastores
 - gateway access to, 1-2
- DBMS_OUTPUT packages, 12-4, 13-4
- DBMS_PIPE, 12-5, 12-8, 13-5, 13-6, 13-9
- DEFINE TRANSACTION statement, 8-3
- defining local LUs
 - on Linux, 6-5
- defining partner LUs
 - on Linux, 6-5
- defining the adjacent node
 - on Linux, 6-4
- defining the link station
 - on Linux, 6-4
- de-installing

- the gateway, 4-10
- dependent LU
 - and OLTP, on HP-UX, 8-2
 - on AIX, 7-2
 - on HP-UX, 8-2
 - on Solaris, 9-2
- DESCRIBE statement, 12-4, 13-3
 - DBMS_OUTPUT, 12-4, 13-4
 - UTL_RAW, 12-4, 13-3
- describe statement
 - DBMS_OUTPUT, 13-4
- dfhcsdup.jcl file, 11-2
- DFHRPL DD statement, 11-2
- directories
 - for installing gateway and OIS files, 12-2, 13-2
- disk space requirements, 3-2
- DISPLAY datatypes, 2-2

E

- EBCDIC language, 1-3
 - gateway know restrictions pertaining to, 2-2
 - necessary to change to ASCII when using TCP/IP, 13-13
- enhancements
 - in this release, 2-1
 - release 4.0.1.1.0, D-3
 - release 8.0.6.1.0, D-3
 - release 9.0.1.0.1, D-2
 - release 9.2.0.1.0, D-2
 - using PGAU to automatically upgrade PG DD entries, 12-9, 13-10
- environmental variables
 - ORACLE_HOME, 4-5
- error
 - during commit processing, A-6
 - obsolete parameters, 15-3
 - parameter name misspelled, 12-10, 13-11
 - treating incoming APPC CONFIRM requests as errors, A-2, A-5

F

- FDS_CLASS parameter, 15-2
- FDS_CLASS_VERSION
 - parameter added, 15-4
- FDS_INSTANCE parameter, 15-3
- file
 - appcdef.doc, 11-3
 - dfhcsdup.jcl, 11-2
 - imsgen.asm, 11-5
 - initPGAL.ora, A-5
 - initPGA.ora, 12-10, 13-11
 - initPGAU.ora, A-5
 - initsid.ora, 1-6, 12-10, 12-12, 13-11, 15-1, A-1, B-1, C-1, D-3
 - gateway parameters for gateway using SNA, 12-10
 - gateway parameters for gateway using TCP/IP for IMS Connect, 13-11

- new parameters on gateway using SNA, 15-2
- new startup shell parameters, 15-4
- parameters changed since V4, 15-3
- install.log, 4-11
- listener.ora, 4-6, 4-11, 5-1, 5-2, A-5
- oraplu62.asm, 11-2, 11-4
- pg4hoa1.sh, D-3
- pgaccou.sql, 12-11
- pgacclg.asm, 12-17
- pgacics.sql, 12-13, 12-14
- pgadb2c.cob, 12-17
- pgaecho.sql, 12-13, 13-12
- pgaflipl.asm, 11-2, 11-3, 11-5
- pgaflipl.jcl, 11-2, 11-5
- pgaidms.sql, 12-13, 12-15
- pgaim.sql, 12-16, 13-14
- pgaims.sql, 12-13, 13-12, 13-14
- pgamvs.sql, 12-13, 12-16
- pgareco.asm, 12-17
- pgasna.export, 7-3
- PGAU control files, 4-2
- pgavsn.sql, 12-13, 13-12
- pgddaddev.sql, 12-9, 13-9
- pgddapub.sql, 12-9, 13-9
- prvtpgb.plb, 12-6, 13-7
- README.doc, 12-18, 13-15
- tnsnames.ora, 4-6, 4-11, 5-1, 5-2, 12-5, 13-5, A-5
- trandef.doc, 11-3
- utlpg.sql, 12-7, 13-8
- utlraw.sql, 12-7, 13-8
- FLIP transaction
 - OLTP configuration
 - and verification for APPC/MVS, 12-16
 - and verification for CICS Transaction Server for z/OS, 12-14
 - and verification for IDMS-DC, 12-14
 - and verification for IMS/TM on gateway using SNA, 12-15
 - and verification for IMS/TM on gateway using TCP/IP, 11-5, 13-13
- function
 - put_line, 12-4, 13-4
- functions
 - see RPC (remote procedural call)
 - See UTL_PG
 - see UTL_RAW

G

- gateway, 1-5
 - access to IBM datastores, 1-2
 - and SNAPplus2 on HP-UX, 8-1
 - communication overview, 1-7
 - communications with all platforms, 1-2
 - compatibility with other SNA-enabled products, 3-4
 - component release number, 2-1
 - components, 1-5, 12-1
 - for SNA and TCP/IP for IMS Connect, 1-6
 - configuring, 12-1

- for multiple users, on gateway using TCP/IP
 - for IMS Connect, 13-8
 - for SNA, 5-2
 - for TCP/IP for IMS Connect, 13-1, 13-11
- configuring for multiple users
 - on gateway using SNA, 12-7
- creating
 - SNA server profiles for, 7-2
 - SNAP-IX definitions for, 9-3
 - SNAP-IX profiles for, 9-2
 - SNAPPlus2 definitions for, 8-3
 - SNAPPlus2 profiles on HP-UX, 8-2
- de-installing, 4-10
- directory locations for configuration, 12-2, 13-2
- factors affecting memory requirements, 3-2
- features
 - application transparency, 1-2
 - code generator, 1-3
 - fast interface, 1-2
 - flexible interface, 1-2
 - location transparency, 1-2
 - Oracle server integration, 1-3
 - performs automatic conversions, 1-3
 - site autonomy and security, 1-3
 - support for tools, 1-3
- functions, using SNA, 1-9
- initialization files, C-1
- initialization parameters
 - also see PGA parameters
 - described, 12-10, 13-11
 - for gateway using SNA, A-1
 - new and changed since Version 4
 - gateway, 15-2
 - renamed since V4, 15-3
 - SET, 12-10, 13-11, A-1, B-1
 - stored, D-3
- installation
 - first-time install, configuring the integrating
 - server, 12-3, 13-3
 - pre-installation procedures, 4-3
 - steps, 4-4
 - verification, 12-13, 13-12
 - with Oracle Universal Installer action
 - items, 4-8
- installing, 4-1
- known restrictions, when using SNA, 2-2
- migrating to new release, using SNA, 15-1
- network attachment requirements, 3-2
- overview, 1-1
- parameter files, 12-10, 13-11
 - also see gateway initialization parameters, and
 - PGA parameters
 - initPGA.ora, 12-10, 13-11
- pre-installation steps for TPC/IP, 4-4
- remote procedure, definition, C-1
- remote transaction initiation
 - using SNA, 1-8
 - using TCP/IP, 1-8
- remote transaction termination
 - using SNA, 1-8

- using TCP/IP, 1-8
- requirements
 - hardware, 3-1
- restoring to previous releases, 4-3
- security options and overview, 14-1
 - also see, security
- server
 - restoring previous version, 4-3
- setting up multiple gateway instances, A-5
- SNA
 - SNAP-IX configuration on Solaris, 9-1, 14-3
 - SNA security validation, 14-3
 - startup shell parameters
 - FDS_CLASS_VERSION, 15-4
 - steps to install, via Oracle Universal Installer, 4-8
 - support for using TCP/IP for IMS Connect, D-2
- upgrading
 - from previous release, 4-2
 - preparing to upgrade, 4-2
- gateway initialization parameters
 - for commit-confirm support, 12-12
 - new, 15-2
 - See gateway, also see, PGA parameters
- gateway security requirements, 14-1
- gateway using TCP/IP for IMS Connect
 - configuring communications package for, 10-1
 - description, D-2
 - gateway initialization parameters needed, B-1
 - transaction types, 1-9
- gpglocal, 12-6, 13-6
 - needed to compile PGAU-generated TIP
 - specifications, 12-6, 13-6
- gpglocal package, 13-6
- gpglocal.pkb script, 12-6, 13-6
- gpglocal.pkh script, 12-6, 13-6
- grant
 - access, 12-8, 13-9
 - authorization, 12-6, 13-6
 - execute, 12-5, 13-5
 - explicit, 12-8, 13-9
 - private, 12-8, 13-9
 - public, 12-8, 13-9

H

- hardware requirements, 3-1
- Heterogeneous Services (HS), 15-3
 - and Oracle Net considerations, on gateway using
 - SNA, 15-2
- catalogs
 - installing, on gateway using SNA, 12-4
 - installing, on gateway using TCP/IP for IMS
 - Connect, 13-4
- definition, D-3
- parameters needed for gateway using
 - TCP/IP, 15-2
- HP-UX
 - and security, 14-3
 - and xsnapadmin, 8-2
 - APPC communication support provided by

- SNAPPlus2, 8-1
- communication protocol needed, 3-3
- configuring SNA, 5-2
- creating SNAPPlus2 definitions for the gateway, 8-3
- dependent LUs, 8-2
- in gateway architecture, 1-5
- in verifying CICs, 12-14
- operating system, 1-5
- processing inbound connections, 8-1
- SNA conversation security, 14-3
- SNA security option SECURITY=SAME, 14-4
- SNA security option SECURITY=NONE, 14-3
- SNA security option SECURITY=PROGRAM, 14-4
- SNA security validation, 14-3
- HS parameters, 15-3
 - description, 15-1, 15-2
 - see also, (HS=)
- HS_COMMIT_STRENGTH_POINT parameter, 15-3
- HS_DB_DOMAIN parameter, 15-3
- HS_DB_INTERNAL_NAME parameter, 15-3
- HS_DB_NAME parameter, 15-3
- HS_DESCRIBE_CACHE_HWM parameter, 15-3
- HS_FDS_FETCH_ROWS parameter, 15-3
- HS_LANGUAGE parameter, 15-3
- HS-NLS_DATE_FORMAT parameter, 15-3
- HS-NLS_DATE_LANGUAGE parameter, 15-3
- HS_OPEN_CURSORS parameter, 15-3
- HS_ROWID_CACHE_SIZE parameter, 15-3

I

- IBM mainframe requirements, 3-4
- IDMS-DC
 - installation verification, 12-14
 - OLTP configuration verification, 12-14
 - verification of configuration, 12-14
- IDMS-DC FLIP program, 11-3
- implementation of the gateway
 - for SNA and TCP/IP for IMS Connect, 1-6
- implied APPC, A-6
- IMS Connect
 - and security, 14-5
 - mainframe requirements, for gateway using TCP/IP, 3-4
 - TCP/IP support for, D-2
- IMS FLIP transaction, 11-5
- msgen.asm file, 11-5
- IMS/TM
 - installation verification
 - on gateway using SNA, 12-15
 - on gateway using TCP/IP for IMS Connect, 13-13
 - mainframe requirements for gateway using TCP/IP, 3-4
 - TP name length, A-3
 - verification of configuration
 - on gateway using SNA, 12-15
 - on gateway using TCP/IP for IMS

- Connect, 13-13
- independent LU, 11-1, 11-2, 11-3, 11-4
 - vs. dependent LU, on HP-UX, 8-2
- independent LU, on AIX, 7-2
- independent versus dependent LUs
 - on Linux, 6-2
- initialization files
 - See gateway initialization files, also see PGA parameters
- initiating remote transactions, 1-8
- initPGAL.ora file, A-5
- initPGA.ora file, 12-10, 13-11
- initPGAU.ora file, A-5
- initsid.ora file, 1-6, 12-10, 12-12, 13-11, 15-1, A-1, B-1, C-1
 - gateway parameters on gateway using SNA, 12-10
 - gateway parameters on gateway using TCP/IP for IMS Connect, 13-11
 - has LOG_DESTINATION parameter, D-3
 - has TRACE_LEVEL parameter, D-3
 - HS parameter descriptions, 15-1, 15-2
 - new parameters, on gateway using SNA, 15-2
 - new startup shell parameters, 15-4
 - parameters changed since V4, 15-3
- installation
 - steps, 4-4
- installation verification
 - CICS on gateway using SNA, 12-14
 - gateway
 - with SNA, 12-13
 - with TCP/IP for IMS Connect, 13-12
 - IDMS-DC, 12-14
 - IMS/TM
 - on gateway using SNA, 12-15
 - on gateway using TCP/IP for IMS Connect, 13-13
 - OLTP, 12-13
- installing
 - and configuring the gateway, 4-1
 - preinstallation steps, 4-3
 - sample applications
 - on gateway for SNA protocol, 12-17
 - on gateway with TCP/IP for IMS Connect, 13-14
- install.log file, 4-11
- IPC
 - key, 5-2
 - protocol, 5-2
- ivtnoctl, 13-2, 13-3
- ivtnod.sql, 13-2, 13-3
- ivtnvctl, 13-2, 13-3
- ivtnvd.sql, 13-2, 13-3

K

- key
 - IPC, 5-2
- known restrictions
 - for PG4APPC, 2-2

- for PGAU, 2-2
- prior gateway releases, D-7

Korn shell, 4-5

L

Link Station

- creating, 8-4

Link Station profiles

- on AIX, 7-3
- using smit to start, on AIX, 7-6

listener.ora file, 4-6, 4-11, 5-1, 5-2, A-5

- defining program pg4asrv, D-1
- sample file for gateway using SNA, A-6
- sample for gateway using TCP/IP, B-4

Local LUs

- creating
 - on AIX, 7-4
 - on HP-UX, 8-5
 - on Solaris, 9-4

LOG_DESTINATION parameter, 15-3

- for gateway using SNA, 15-2
- for gateway using TCP/IP, B-2
- moved to initsid.ora, D-3

logmode entry name, 12-14

LTERM, 11-3

LU name

- assigning to the gateway, on AIX, 7-4

LU6.2

- and specifying SNA conversation security, 14-3

LUs

- (Logical Units), 7-4
- alias identified by side information profile, on AIX, 7-5
- and gateway security, 14-3
- and Side Information Profiles, 8-2
- in SNA security validation, 14-3
- independent
 - in configuring APPC/MVS on the gateway, 11-4
 - in configuring CICS Transaction Server for z/OS, 11-1
 - in configuring IDMS-DC, 11-2
 - vs. dependent, on AIX, 7-2
 - vs. dependent, on HP-UX, 8-2
 - vs. dependent, on Solaris, 9-2
- local
 - profile, on AIX, 7-4
- on AIX, 7-2
- Partner
 - location profile, on AIX, 7-5
 - on AIX, 7-4
 - on HP-UX, 8-2
 - on Solaris, 9-4
 - profile, 7-4
- Partner LU name
 - assigning alias, on AIX, 7-4
- recommended for the gateway, on HP-UX, 8-2

M

mainframe requirements, 3-4

memory requirements, 3-2

migrating

- an existing gateway to use TCP/IP, 15-4
- backout considerations when migrating to new release, 15-2
- existing gateway instance to new release, using SNA, 15-1
- to 10.1.0.2.0
 - new listener.ora value, D-1
 - special parameters, 15-2

Mode

- and Side Information Profiles, 8-2

mode definitions

- creating
 - on HP-UX, 8-5
 - on Solaris, 9-5

mode profiles

- creating on AIX server, 7-4
- creating on Solaris, 9-4

mounting the CD-ROM

- for AIX, 4-6
- for HP-UX, 4-6
- for Solaris, 4-7

multi-conversational transaction type, for gateway using TCP/IP, 1-9

N

network

- configuring
 - with SNA, 5-1
 - reconfiguring, 5-1
- networking products required, 3-4
- node profile, on AIX, 7-3
- non-persistent socket transaction type for TCP/IP for IMS Connect, 1-9

O

obsolete parameters, in gateway using SNA, 15-3

OLTP, 5-3, A-4

- and dependent LUs
 - on AIX, 7-2
 - on HP-UX, 8-2
 - on Solaris, 9-4
- and SECURITY=PROGRAM option, on all platforms, 14-4
- and SECURITY=PROGRAM option, on all platforms using TCP/IP, 14-5
- and SNA SECURITY=SAME option, on all platforms, 14-4
- configuration, 11-1
- configuration verification
 - APPC/MVS, 12-16
 - CICS Transaction Server for z/OS, 12-14
 - IDMS-DC, 12-14
 - on gateway using SNA, 12-12
 - on gateway using SNA IMS/TM, 12-15

- on gateway using TCP/IP for IMS
 - Connect, 13-12
- configuring
 - for gateway using TCP/IP for IMS
 - Connect, 13-11
 - to allow gateway access, 4-1
 - currently supported types, 11-1, 12-13
 - defining PTERM and LTERMs (configuration of IDMS-DC), 11-3
 - definition, 1-3
 - for TCP/IP for IMS Connect, 1-6
 - in gateway architecture featuring SNA, 1-5
 - in gateway using TCP/IP, 1-6
 - installation verification, 12-13, 13-13
 - mode name specification, on AIX, 7-4
 - post-installation steps
 - on gateway using SNA, 12-17
 - on gateway using TCP/IP for IMS
 - Connect, 13-14
 - remote, 1-1
 - requirements, 3-4
 - security and inbound APPC session requests
 - on HP-UX, 8-1
 - on Solaris, 9-1
 - security on the gateway, 14-1
 - SNA security option
 - on all platforms, 14-4
 - user ID mapping, 14-3
 - verifying configuration
 - on gateway using SNA, 12-13
 - on gateway using TCP/IP for IMS
 - Connect, 13-13
- OLTP for SNA
 - mainframe requirements, 3-4
- OLTP for TCP/IP
 - IBM mainframe requirements, 3-4
- one-shot transaction types, for gateway using SNA, 1-8
- online transaction processor
 - See OLTP
- Oracle directory tree, 4-5
- Oracle global transaction ID, 12-17
- Oracle Heterogeneous Services
 - See Heterogeneous Services
- Oracle integrating server, 1-8, 4-1, 4-3, 4-4, 5-2, 12-5, 13-5, 13-6, 14-2, A-4, A-5
 - and gateway security, 14-3
 - and TCP/IP for IMS Connect
 - pre-configuration steps, 13-2
 - component of the gateway, 1-5
 - configuring for commit-confirm, 12-11
 - definition, 1-3
 - enabling DBMS_OUTPUT PL/SQL package, 12-4, 13-4
 - logon authentication needed, 14-2
 - multiple servers on the gateway
 - using TPC/IP, 1-6
 - multiple servers on the gateway using SNA, 1-5
 - precompiles PL/SQL package, 1-2
 - READ_ONLY mode, A-4

- role
 - in gateway communication, 1-7
 - in logon security, 14-3
 - in starting the gateway, 1-7
- shipped with PL/SQL packages, 12-8, 13-9
- stores PL/SQL, C-2
- upgrading
 - from previous releases, 12-6, 13-7
- verifying
 - APPC/MVS configuration, 12-16
 - gateway installation with SNA, 12-13
 - gateway installation with TCP/IP for IMS
 - Connect, 13-12
 - IMS/TM, on gateway using SNA, 12-15
 - IMS/TM, on gateway using TCP/IP for IMS
 - Connect, 13-14
 - version requirements, 3-3
- Oracle Integrating Server Listener, 5-2
- Oracle Net, 1-3, A-2
 - considerations, when migrating gateway featuring SNA, 15-2
 - function in the gateway architecture, 1-5
 - Heterogeneous services/tnsnames.ora
 - considerations, on gateway using SNA, 15-2
 - on gateway using TCP/IP, B-2
 - security considerations, 14-2
 - to start the gateway, 1-7
 - version needed, 2-1
- Oracle Procedural Gateway for APPC
 - also see gateway
 - de-installing, 4-10
 - development environment, 1-3
 - functions, 1-9
- Oracle Universal Installer, 4-7
 - using, 4-9
 - for AIX, 4-7
 - to de-install the gateway, 4-10
 - to install gateway, 4-8
- ORACLE_HOME environmental variable, 4-5
- Oracle10g Database Server
 - and networking products needed, 3-4
- ORAPGA.APPCMVS.SAMPLIB, 11-4
- oraplu62.asm file, 11-2, 11-4
- override
 - user ID and password, 14-4, 14-5

P

- package
 - DBMS_OUTPUT, 12-4, 13-4
 - gpglocal, 13-6
 - UTL_PG, 12-4, 12-8, 13-4, 13-9
 - invalidated or deinstalled, 12-6, 13-7
 - UTL_RAW, 12-3, 13-3
 - invalidated or deinstalled, 12-6, 13-7
- package specifications
 - avoid reinstalling, 12-6, 13-7
 - reinstalling, 12-6, 13-7
- parameter files
 - See gateway initialization files

- See PGA parameters
- See RRM parameters
- parameters
 - changed since Release 4, on gateway using SNA or TCP/IP, 15-2
 - FDS_CLASS_VERSION, 15-4
 - for configuring TCP/IP for IMS Connect, 10-1
 - gateway initialization parameter,
 - described, 12-10, 13-11
 - needed for commit-confirm support, 12-12
 - new
 - FDS_CLASS (startup shell), 15-2
 - FDS_INSTANCE (startup shell), 15-3
 - obsolete, in gateway using SNA, 15-3
 - PGA
 - described for SNA, 12-10
 - described for TCP/IP, 13-11
 - renamed since version 4 (gateway initialization), 15-3
 - see PGA parameters and gateway USING, 5-1
- Partner LU
 - locations profile, on AIX, 7-5
 - on HP-UX, 8-2, 8-5
 - on Solaris, 9-4
 - profile, 7-4
 - see also LUs
 - Partner
- password
 - change using ALTER USER command, 12-5, 13-5
 - Oracle authentication, 14-2
 - Oracle password to be used by gateway, A-2
 - using TCP/IP, B-2
 - overrides, 14-4, 14-5
 - with operating system authentication, 14-2
- PDS (partitioned dataset), 11-4
- persistent socket transaction type
 - for TCP/IP for IMS Connect, 1-9
- persistent transaction type, for gateway using SNA, 1-9
- PG Data Dictionary
 - See PG DD
- PG DD
 - after upgrade, 4-2
 - allowing multiple users, 12-7, 13-8
 - creating public synonyms for multiple users, 12-7, 13-8
 - definition, 1-4
 - install script, 12-5, 13-5
 - installing for gateway configuration, 12-5, 13-5
 - no access from earlier PGAU versions, 4-3
 - packages
 - dropping after de-installing the gateway, 4-10
 - restoring previous version, 4-3
 - tables, 12-8, 13-9
 - upgrade when upgrading gateway, 12-7, 13-8
 - using PGAU to upgrade existing entries, 12-9, 13-10
- PG4APPC
 - known restrictions, 2-2
 - see gateway
- PG4APPC support for TCP/IP
 - see gateway using TCP/IP for IMS Connect
- pg4asrv
 - defining to listener.ora, D-1
- pg4hoa1.sh, D-3
 - no longer used in release 10.1.0.2.0
 - file, D-1
 - parameter added in past release, D-3
- pg4tcpmap table, 15-4, 15-5
 - see PGA_TCP_IMSC table
- pg4tcpmap tool, 13-12, 14-5, 14-6, 15-4, 15-5, B-2, B-3
 - definition, C-2
 - function, 1-2, 13-12
 - in remote transaction initiation, 1-8
 - in the gateway, D-2
 - on gateway using TCP/IP, 13-5, B-2
 - output sample, B-3
- PGA
 - definition, C-2
 - initialization files
 - initPGAI.ora and initPGAU.ora, A-5
- PGA parameters
 - described, 12-10, 13-11
 - list of, for gateway using TCP/IP, B-2
 - LOG_DESTINATION, A-2
 - on gateway using
 - SNA, A-1
 - TCP/IP, B-1
 - PGA_CAPABILITY, 12-10, 13-11, A-2, B-2
 - choosing settings, A-4
 - options for updating foreign databases, A-4
 - protections against data problems, A-4
 - PGA_CAPABILITY, for gateway using TCP/IP, B-2
 - PGA_CONFIRM, A-2
 - choosing settings, A-5
 - PGA_LOG_DB, A-2, B-2
 - PGA_LOG_PASS, A-2, B-2
 - on gateway using TCP/IP, B-2
 - PGA_LOG_USER, A-3, B-3
 - PGA_RECOVERY_PASS, A-3
 - PGA_RECOVERY_TPNAME, A-3
 - PGA_RECOVERY_USER, A-3
 - PGA_SECURITY_TYPE, 14-4, 14-5, A-3, B-2
 - PGA_SECURITY_TYPE=NONE, on all platforms, 14-3, 14-5
 - PGA_SECURITY_TYPE=SAME, on all platforms, 14-4
 - PGA_TCP_PASS
 - for gateway using TCP/IP, B-2
 - TRACE_LEVEL, A-3
 - on gateway using TCP/IP, B-3
- PGA_CAPABILITY
 - See PGA parameters
- PGA_CC_PENDING
 - table
 - unsuccessful rollback, D-7
- PGA_CONFIRM
 - See PGA parameters for gateway using SNA or

TCP/IP

PGA_SECURITY_TYPE
See PGA parameters

PGA_SECURITY_TYPE parameter
and TCP/IP security, 14-5

PGA_SECURITY_TYPE=NONE, on all
platforms, 14-3, 14-5

PGA_SECURITY_TYPE=SAME, on all
platforms, 14-4

PGA_SIGDANGER parameter
DEALLOCATE value not supported, D-2

PGA_TCP_DB
PGA parameter
for gateway using TCP/IP, B-2

PGA_TCP_DB parameter (TCP/IP only), 15-3

PGA_TCP_DB PGA parameter (TCP/IP only), B-2

PGA_TCP_IMSC table, 15-4, 15-5
for mapping SNA parameters to TCP/IP, 13-12
loading, on gateway using TCP/IP, 13-12

PGA_TCP_PASS parameter (TCP/IP only), 15-3

PGA_TCP_PASS PGA parameter (TCP/IP only), B-2

PGA_TCP_USER parameter (TCP/IP only), 15-3,
B-3

PGAADMIN, 12-5, 12-6, 12-13, 13-6, 13-12
creating the gateway administrator user ID, 12-5,
13-5
granting access to additional users, 12-7, 13-8
granting execution privileges on DBMS_
PIPE, 12-5, 13-5
initial password during creation, 12-5, 13-5

pgaccu.sql file, 12-11

pgacclg.asm file, 12-17

pgacics.sql file, 12-13, 12-14

pgacr8au.sql script, 12-5, 13-5

pgadb2c.cob file, 12-17

pgaecho.sql file, 12-13, 13-12, 13-13

pgaflipl.asm file, 11-2, 11-3, 11-5

pgaflipl.jcl file, 11-2, 11-5

PGAID
setting up, A-5

pgaidms.sql file, 12-13, 12-15

pgaimsc.sql, 13-6

pgaims.sql, 13-2, 13-3

pgaims.sql file, 12-13, 12-15, 12-16, 13-12
on gateway using TCP/IP, 13-14

PGAINIT, 1-8

PGAINIT function, 1-8, C-1

PGAINIT TIP, 13-12

pgamvs.sql file, 12-13, 12-16

pgareco.asm file, 12-17

pgasna.export file, 7-3

PGATERM function, C-1

pgatiptr.sql script, 12-5, 13-6

PGAU, 5-2
accesses definitions in PG DD, 1-4
control files, 4-2
definition
used to generate TIP specifications, 1-4
-generated TIP specifications, 1-7
-generated TIP specifications use UTL_PG, 12-4,
13-4
-generated TIP specifications use UTL_
RAW, 12-3, 13-3
generating TIP specifications, PL/SQL v3
compatibility, D-3
known restrictions in this release, 2-2
purpose of PGDL, C-2
restoring previous versions, 4-3
setting up, A-5
upgrading existing PG DD entries, 12-9, 13-10

PGAU commands
DEFINE DATA
COBOL COPY REPLACE restrictions, 2-2

GENERATE, 12-9, 13-10
produces TIP in output files, 12-10, 13-10
to upgrade existing TIPs, 12-9, 13-10

pgavsn.sql file, 12-13, 13-12

PGAXFER function, 1-8, C-1

PGDD
compatibility issues between new and older
gateways, 12-7

pgddadev.sql
file, 12-9, 13-9
script, 12-9, 13-9

pgddapub.sql
file, 12-9, 13-9
script, 12-9, 13-9

pgddcr8r.sql script, 12-8, 13-9

pgddcr8.sql script, 12-5, 13-5

pgddcr8s.sql script, 12-8, 13-8

PGDDDEF role, 12-8, 13-9

pgdddel.sql script, 4-10

PGDDGEN role, 12-8, 13-9
adding a privilege for upgrading PG DD
entries, 12-9, 13-10

pgddupgr.sql script, 12-7, 13-8

PGDL (Procedural Gateway Definition Language)
definition, C-2

PL/SQL, 1-4
code generator, 1-3
datatypes, 1-7
converted to RAW, 1-8
function in the gateway, 1-1, 1-7
running pgatiptr.sql script to create
routines, 12-5, 13-6
UTL_PG package function, C-2
UTL_RAW function, C-2
UTL_RAW package installation
on gateway using SNA, 12-3
on gateway using TCP/IP for IMS
Connect, 13-3

PL/SQL package
definition, 1-4, C-2
developer access to, 12-8, 13-9
enabled, 12-4, 13-4
functions, 1-7
See TIP

PL/SQL stored procedure, 12-11
used for logging transactions, 12-11

PL/SQL stored procedure specification

- also called "TIP"
- See PL/SQL package
- post-installation steps for OLTP
 - on gateway using SNA, 12-17
 - on gateway using TCP/IP, 13-14
- pre-installation steps
 - for SNA, 4-3
- privileges
 - needed to create TIPs, 12-9
- Procedural Gateway Administration
 - See PGA
- Procedural Gateway Administration Utility
 - see PGAU, 1-7
- processing inbound connections
 - Linux, 6-2
- processor OLTP
 - configuring, for commit-confirm, 12-12
- processor requirements, 3-1
- product installation directory, 4-5
 - see ORACLE_HOME and pg4appc directory
- product set in this release, 2-1
- profiles
 - see SNA server profiles
- protocol
 - IPC, 5-2
 - TCP, 5-2
- prvtpgb.plb
 - file, 12-6, 13-7
 - script, 12-4, 12-6, 13-4, 13-7
- prvtrawb.plb script, 12-4, 12-6, 13-4, 13-7
- PTERM, 11-3
- public synonyms for multiple PG DD users, 12-7, 13-8
- put_line function, 12-4, 13-4

R

- RACF, 14-5, 14-6, B-2
- READ_ONLY
 - PGA_COMPATIBILITY setting, A-4
- README.doc file, 12-18, 13-15
- RECO
 - Oracle recovery process, D-7
- recompiling TIPs
 - See TIP
- reinstallation of package specifications, 12-6, 13-7
- release-specific information
 - restrictions, 2-2
- remote host transactions (RHT)
 - types, 1-8
- Remote Node definition
 - creating, 8-4
- remote procedural call
 - See RPC
- remote procedure
 - definition, C-1
- remote transaction initiation
 - on gateway using SNA, 1-8
 - on gateway using TC/IP, 1-8
- remote transaction program

- See RTP
- remote transaction termination
 - on gateway using SNA, 1-8
 - on gateway using TCP/IP for IMS Connect, 1-8
- requirements
 - documentation, 3-4
 - hardware, 3-1
 - network attachments, 3-2
 - software, 3-2
 - system, 3-1
- Resource Recovery Manager
 - not supported on AIX5L, D-2
- restoring a previous release of the gateway, 4-3
- restrictions
 - on gateway using SNA, 2-2
- resume configuration of the gateway
 - on Linux, 6-6
- RHDCCNOS multiple session service manager, 11-3
- role
 - PGDDDEF, 12-8, 13-9
 - PGDDGEN, 12-8, 13-9
- ROLLBACK, 3-4, A-4
- rollback
 - in bug no. 689304, D-7
- RPC
 - definition, 1-4
 - function
 - PGAINIT, 1-8, C-1
 - PGATERM, C-1
 - PGAXFER, 1-8, C-1
 - within the gateway, 1-1, 1-7
 - processing, 1-1
- RTP
 - and Side Information Profiles, 8-2
 - definition, 1-4
 - executing, 1-3
 - function in the gateway, 1-1
 - PGA_CAPABILITY settings for read-only RTPs, A-4

S

- sample applications
 - included, 12-17, 13-14
 - installing, 12-17, 13-14
- savepoint
 - not supported, D-5
- script
 - gpglocal.pkb, 12-6, 13-6
 - gpglocal.pkh, 12-6, 13-6
 - pgacr8au.sql, 12-5, 13-5
 - pgddadev.sql, 12-9, 13-9
 - pgddapub.sql, 12-9, 13-9
 - pgddcr8r.sql, 12-8, 13-9
 - pgddcr8.sql, 12-5, 13-5
 - pgddcr8s.sql, 12-8, 13-8
 - pgdddel.sql, 4-10
 - pgddupgr.sql, 12-7, 13-8
 - prvtpgb.plb, 12-4, 12-6, 13-4, 13-7
 - prvtrawb.plb, 12-4, 12-6, 13-4, 13-7

- utlpg.sql, 12-4, 12-7, 13-4, 13-8
- utlraw.sql, 12-4, 12-7, 13-4, 13-8
- security
 - and database links, 14-2
 - and SNA validation, 14-3
 - authenticating application logons, 14-2
 - authentication mechanisms
 - in SNA security, 14-4
 - for TCP/IP for IMS Connect, 14-4
 - link accessibility for public and private databases, 14-2
 - links and CONNECT clauses, 14-3
 - overview of gateway security requirements, 14-1
 - processing inbound connections
 - on AIX, 7-1
 - on HP-UX, 8-1
 - on Solaris, 9-1
 - specifying SNA conversation security, on all platforms, 14-3
- SECURITY=NONE SNA security option, on all platforms, 14-3
- SECURITY=NONE TCP/IP security option, 14-5
- SECURITY=PROGRAM SNA security option, on all platforms, 14-4
- SECURITY=SAME SNA security option, on all platforms, 14-4
- SET
 - gateway initialization parameter, 12-10, 13-11
- SET command, 12-10, 13-11
- shells
 - see Bourne, Korn, and C shells, 4-1
- side information
 - and Sym Dest Name option, on HP-UX, 8-3
- Side Information Profiles, 7-4, 8-2, 8-5
 - description, 8-2
 - function, on AIX, 7-5
 - function, on HP-UX, 8-2
 - function, on Solaris, 9-2
- side profile name, 12-14
- SINGLE_SITE
 - and PGA_CAPABILITY parameter, A-4
- smit, 7-2
 - System Management Interface Tool for AIX, 7-2
- SNA
 - and gateway components, 1-5
 - implementation of the gateway, 1-6
 - location of gateway initialization parameters, A-1
 - migrating existing gateway to the new release, 15-1
 - new gateway initialization parameters, 15-2
 - parameters, 1-8
 - PGA parameters, A-1
 - preinstallation procedures, 4-3
 - remote transaction initiation, 1-8
 - remote transaction termination on the gateway, 1-8
 - required for HP-UX, 3-3
 - required for Solaris, 3-3
 - transaction types, 1-8
- SNA APPC
 - also see APPC
 - function in the gateway, 1-1
- SNA communication package, 11-3, 12-14, 12-15, 12-16, 14-4
 - and SNA security option, on all platforms, 14-4
 - and SNA security validation, all platforms, 14-3
 - configuration for the gateway, 5-2
- SNA definitions
 - creating SNAP-IX definitions, 9-3
- SNA node
 - and Link Station, 8-4
 - profile, on AIX, 7-3
- SNA profiles, 14-3
 - See SNA server profiles
- SNA protocol
 - gateway initialization parameters, A-1
- SNA security options
 - SECURITY=NONE, on all platforms, 14-3
 - SECURITY=PROGRAM, on all platforms, 14-4
 - SECURITY=SAME, on all platforms, 14-4
- SNA server, 7-1
 - dependent LUs, 7-2
 - independent LUs, on AIX, 7-2
 - profiles, 7-3, 7-5
 - activating and verifying, 7-6
 - creating, on AIX, 7-2
 - definition, 7-1
 - for non APPN-capable nodes, on AIX, 7-5
 - Link Station, 7-3
 - local LU profile, 7-4
 - mode profile, 7-4
 - Partner LU, 7-5
 - Partner LU location profile, 7-4, 7-5
 - Side Information Profiles, 7-5
 - SNA node profile, 7-3
 - types, on AIX, 7-3
 - verify and activate profiles, on AIX, 7-6
- SNAP-IX
 - communication support for Solaris, 9-1, 14-3
 - configuring, 9-1, 9-3
 - definitions stored, 9-3
 - function in gateway communication, 9-1, 14-3
 - profiles creating, 9-2
 - supported SNA protocol for Solaris, D-1
- SNAPPlus2
 - and LUs, 8-2
- APPC menu
 - Sym Dest Names option, 8-3
- configuring, 8-3
- creating definitions for the gateway, on HP-UX, 8-3
- creating profiles for the gateway, 8-2
- function, 8-1
- testing the connection, 8-6
- using xsnadmin to configure, 8-2

- SNASVCMG mode entry, 11-3
- socket file descriptor
- returned by TCP/IP network to PGAINIT, 1-8
- software requirements, 3-2
- Solaris

- APPC support provided by SNAP-IX, 9-1
- communication protocol needed, 3-3
- configuring SNA, 5-2
- configuring SNAP-IX, 9-1, 9-3
- LU types, 9-2
- SNA conversation security, 14-3
- SNA security option SECURITY=SAME, 14-4
- SNA security option SECURITY=NONE, 14-3
- SNA security option SECURITY=PROGRAM, 14-4
- SNA security validation, 14-3
- storing SNAP-IX SNA definitions, 9-3
- specifying LUs
 - on Solaris, 9-5
- SQL*Plus
 - sample, used when gateway and Oracle integrating server share machine, 12-5, 13-5
 - to configure Oracle integrating server, 12-4, 13-3, 13-5
 - use in configuring Oracle integrating server for Commit-Confirm, 12-11
 - using to connect to Oracle integrating server, 12-5, 13-5
- startup shell script file
 - new parameter added, D-3
- statement
 - CDMSLIB DD, 11-3
 - DESCRIBE, 12-4
 - describe, 13-4
 - DFHRPL DD, 11-2
- SunLink P2P
 - no longer supported, D-1
- Sym Dest Name
 - symbolic destination name, on HP-UX, 8-3
- system identifier
 - choosing, 4-3
- system requirements, 3-1

T

- TCP protocol, 5-2
- TCP/IP
 - specifying conversation security, 14-5
- TCP/IP for IMS Connect
 - and Remote Transaction Initiation, 1-8
 - configuration parameters needed, 10-1
 - configuring
 - for the gateway, 13-11
 - gateway to permit multiple users, 13-8
 - function in the gateway, 1-1
 - gateway initialization parameters, list, B-1
 - gateway preinstallation procedures, 4-4
 - gateway support for, description, 1-2
 - Heterogeneous Services parameters needed, 15-2
 - HS parameter descriptions, 15-1, 15-2
 - implementation of the gateway, 1-6
 - IMS Connect release required on IBM mainframe, 3-4
 - installing sample applications, 13-14

- loading the PGA_TCP_IMSC table, 13-12
- mapping SNA parameters to, 13-12
- migrating existing gateway using SNA to TCP/IP, 15-4
- necessary to recompile TIPs when changing communication protocol, 13-13
- new gateway initialization parameters, 15-2
- non-persistent socket transaction type, 1-9
- OLTP in gateway architecture, 1-6
- parameter files
 - also see gateway initialization parameters, and PGA parameters
- performing post-installation procedures, 13-14
- persistent socket transaction type, 1-9
- pg4tcpmap tool output sample, B-3
- PGA parameters, 13-11, B-1
- PGA_TCP_DB parameter, B-2
- PGA_TCP_USER parameter, B-3
- remote transaction initiation, 1-8
- remote transaction termination, 1-8
- security, 14-4
- security options
 - SECURITY=NONE, 14-5
 - SECURITY=NONE, on all platforms, 14-5
 - SECURITY=PROGRAM, on all platforms, 14-5
- support for IMS connect, D-2
- TIP recompile needed on upgrade, 4-3, 13-13
- TRACE_LEVEL parameter, B-3
- transaction types, 1-9
- using pg4tcpmap tool, B-2
- verifying
 - gateway installation, 13-12
 - OLTP configuration, 13-13, 13-14
- TCP/IP protocol adapter
 - for SNA, 5-2
- terms, gateway terms defined, 1-3
- testing the connection
 - on Linux, 6-5
- testing the SNAplus2 connection, 8-6
- tg4pwd, C-2
- tg4pwd utility
 - definition, C-2
 - recommended security utility feature, on gateway using SNA, C-2
- TIP
 - Also called PL/SQL package
 - body output files, 12-10, 13-10
 - conversions, 1-3
 - converting PL/SQL datatypes to RAW, 1-8
 - definition, 1-4
 - developer access to PL/SQL packages, 12-8, 13-9
 - developer authorization on gpglocal, 12-6, 13-6
 - functions, 1-7
 - in Oracle integrating server, 1-8
 - invalidated if package specifications reinstalled, 12-6, 13-7
 - override, on all platforms, 14-4
 - override, on all platforms, in security for TCP/IP, 14-5
 - recompile

- after reinstalling package specifications, 12-6, 13-7
- on upgrade from Release 4.0.1, 4-2
- recompile upon upgrade from SNA to TCP/IP, 4-3, 13-13
- recompiling when changing from SNA to TCP/IP, 13-13
- regenerate to upgrade function and maintenance, 12-9, 13-10
- remote transaction initiation (PGAINIT), 1-8
- specification output files, 12-10, 13-10
- specifications, 12-5, 12-6, 12-8, 12-9, 13-6, 13-8, 13-10
 - generated by PGAU, 1-7, D-3
 - PL/SQL v3 compatibility, D-3
 - use UTL_PG, 12-4, 13-4
 - use UTL_RAW, 12-3, 13-3
- specifying LUs
 - on HP-UX, 8-2
- trace access PL/SQL routines, 12-5, 13-6
- upgrade considerations from previous versions, 4-2
- TNS listener, 5-1
- tnsnames.ora file, 4-6, 4-11, 5-1, 5-2, 12-5, 13-5, A-5
 - and Oracle Net considerations, 15-2
 - sample file for gateway using SNA, A-6
 - sample for gateway using TCP/IP, B-5
- TP name, A-3
- trace access, 12-5, 13-6
- TRACE_LEVEL parameter, 15-3, A-3
 - on gateway using SNA, A-3
 - on gateway using TCP/IP for IMS Connect, B-3
- trandef.doc file, 11-3
- Transaction Interface Package
 - See TIP
- transaction socket
 - transaction type for TCP/IP, 1-9
- transaction types
 - for TCP/IP for IMS/Connect, 1-9
 - one-shot, persistent and multi-conversational, for SNA, 1-8
- transferring
 - initsid.ora gateway initialization file parameters, 15-1
- transparency
 - (application), 1-2
 - (location), on gateway using SNA, 1-2
- two-phase commit
 - no longer supported, D-1

U

- upgrading
 - considerations, 4-2
 - existing TIP specifications, 12-9, 13-10
- user ID
 - as security authentication, 14-2
 - uppercase translation, OLTP and SNA communications packages, on all platforms, 14-4

- user ID mapping
 - OLTP, 14-3
- USING parameter, 5-1
- using SNA security validation
 - Linux, 6-1
- utility
 - tg4pwd, C-2
- UTL_PG, 12-8, 13-9
 - installing, 12-4, 13-4
 - package, 12-8, 13-9
 - definition, C-2
 - invalidated or deinstalled, 12-6, 13-7
- UTL_RAW, 12-3, 12-4, 12-8, 13-3, 13-9
 - interface
 - PL/SQL package, 12-8, 13-9
 - package
 - invalidated or deinstalled, 12-6, 13-7
 - PL/SQL package
 - definition, C-2
- utlpg.sql
 - file, 12-7, 13-8
 - script, 12-4, 12-7, 13-4, 13-8
- utlraw.sql
 - file, 12-7, 13-8
 - script, 12-4, 12-7, 13-4, 13-8

V

- VTAM, 11-4
 - and Side Information Profiles, 8-2
 - configuring for connection to the gateway, 11-1, 11-2
- VTAM logmode table, 11-1, 11-2, 11-4

X

- xснаadmin, 9-3
 - invoking on Solaris, 9-3
 - with SNAP-IX, 9-3
- xsnapadmin, 8-4
 - invoking, 8-4
 - to configure SNAPPlus2, 8-2
 - to create SNA definitions, on HP-UX, 8-3

Z

- z/OS, 1-2, 1-3, 3-4, 11-1, 11-2

