

# Oracle® Database

## Database Development Guide



23ai  
F47572-17  
April 2025

ORACLE®

Copyright © 1999, 2025, Oracle and/or its affiliates.

Primary Author: Jiji Thomas

Contributing Authors: J. Sharma, A. Kumar, C. Murray, T. Kyte, D. Adams, L. Ashdown, S. Moore, E. Paapanen, R. Strohm, R. Ward

Contributors: D. Alpern, G. Arora, T. Chang, B. Cheng, R. Day, R. Decker, G. Doherty, A. Ganesh, M. Hartstein, Y. Hu, J. Huang, C. Iyer, N. Jain, V. Krishnaswamy, R. Kumar, S. Kumar, C. Lei, B. Llewellyn, K. Mohan, V. Moore, J. Muller, R. Murthy, R. Pang, B. Sinha, S. Vemuri, W. Wang, D. Wong

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

# Contents

## Preface

---

Audience	xxxiv
Documentation Accessibility	xxxiv
Related Documents	xxxiv
Conventions	xxxv

## Changes in This Release for Oracle Database Development Guide

---

New Features in 23ai	xxxvi
Deprecated Features	xxxix

## Part I Database Development Fundamentals

---

### 1 Design Basics

---

1.1	Design for Performance	1-1
1.2	Design for Scalability	1-2
1.3	Design for Extensibility	1-2
1.3.1	Data Cartridges	1-3
1.3.2	External Procedures	1-3
1.3.3	User-Defined Functions and Aggregate Functions	1-3
1.3.4	Object-Relational Features	1-3
1.4	Design for Security	1-4
1.5	Design for Availability	1-4
1.6	Design for Portability	1-5
1.7	Design for Diagnosability	1-5
1.8	Design for Special Environments	1-6
1.8.1	Data Warehousing	1-6
1.8.2	Online Transaction Processing (OLTP)	1-7
1.9	Features for Special Scenarios	1-7
1.9.1	SQL Analytic Functions	1-8
1.9.2	Materialized Views	1-9
1.9.3	Partitioning	1-10

## 2 Connection Strategies for Database Applications

---

2.1	Design Guidelines for Connection Pools	2-1
2.1.1	Connection Storms	2-1
2.1.2	Guideline for Preventing Connection Storms: Use Static Pools	2-2
2.2	Design Guideline for Login Strategy	2-3
2.3	Design Guideline for Preventing Programmatic Session Leaks	2-3
2.3.1	Drained Connection Pools	2-4
2.3.2	Checking for Session Leaks	2-4
2.3.3	Lock Leaks	2-4
2.3.4	Logical Corruption	2-4
2.4	Using Runtime Connection Load Balancing	2-5
2.4.1	About Runtime Connection Load Balancing	2-5
2.4.2	Enabling and Disabling Runtime Connection Load Balancing	2-6
2.4.2.1	OCI	2-6
2.4.2.2	OCCI	2-7
2.4.2.3	JDBC	2-7
2.4.2.4	ODP.NET	2-7
2.4.3	Receiving Load Balancing Advisory FAN Events	2-8

## 3 Performance and Scalability

---

3.1	Performance Strategies	3-1
3.1.1	Designing Your Data Model to Perform Well	3-1
3.1.1.1	Analyze the Data Requirements of the Application	3-2
3.1.1.2	Create the Database Design for the Application	3-2
3.1.1.3	Implement the Database Application	3-3
3.1.1.4	Maintain the Database and Database Application	3-4
3.1.2	Setting Performance Goals (Metrics)	3-4
3.1.3	Benchmarking Your Application	3-4
3.2	Tools for Performance	3-5
3.2.1	DBMS_APPLICATION_INFO Package	3-5
3.2.2	SQL Trace Facility (SQL_TRACE)	3-6
3.2.3	EXPLAIN PLAN Statement	3-7
3.3	Monitoring Database Performance	3-8
3.3.1	Automatic Database Diagnostic Monitor (ADDM)	3-8
3.3.2	Monitoring Real-Time Database Performance	3-9
3.3.3	Responding to Performance-Related Alerts	3-9
3.3.4	SQL Advisors and Memory Advisors	3-9
3.4	Testing for Performance	3-10

3.5	Using Client Result Cache	3-11
3.5.1	About Client Result Cache	3-11
3.5.2	Benefits of Client Result Cache	3-12
3.5.3	Guidelines for Using Client Result Cache	3-13
3.5.3.1	SQL Hints	3-14
3.5.3.2	Table Annotation	3-15
3.5.3.3	Session Parameter	3-16
3.5.3.4	Effective Table Result Cache Mode	3-16
3.5.3.5	Displaying Effective Table Result Cache Mode	3-16
3.5.3.6	Result Cache Mode Use Cases	3-17
3.5.3.7	Queries Never Result Cached in Client Result Cache	3-17
3.5.4	Setting Result Cache Integrity	3-18
3.5.5	Client Result Cache Consistency	3-19
3.5.6	Deployment-Time Settings for Client Result Cache	3-19
3.5.6.1	Server Initialization Parameters	3-20
3.5.6.2	Client Configuration Parameters	3-21
3.5.7	Client Result Cache Statistics	3-22
3.5.8	Validation of Client Result Cache	3-22
3.5.8.1	Measure Execution Times	3-22
3.5.8.2	Query V\$MYSTAT	3-23
3.5.8.3	Query V\$SQLAREA	3-23
3.5.9	Client Result Cache and Server Result Cache	3-24
3.5.10	Client Result Cache Demo Files	3-25
3.5.11	Client Result Cache Compatibility with Previous Releases	3-25
3.6	Statement Caching	3-25
3.7	OCI Client Statement Cache Auto-Tuning	3-26
3.8	Client-Side Deployment Parameters	3-26
3.9	Using Query Change Notification	3-27
3.10	Using Database Resident Connection Pool	3-27
3.10.1	About Database Resident Connection Pool	3-28
3.10.2	Configuring DRCP	3-29
3.10.2.1	Managing Permissions for Per-PDB DRCP	3-31
3.10.3	Using Multi-pool DRCP	3-31
3.10.3.1	Adding a DRCP Pool	3-32
3.10.3.2	Removing a DRCP Pool	3-33
3.10.3.3	About Authentication Pool in Multi-pool DRCP	3-33
3.10.3.4	Managing the Connection Broker in Multi-pool DRCP	3-34
3.10.4	Sharing Proxy Sessions	3-34
3.10.5	Using JDBC with DRCP	3-34
3.10.6	Using OCI Session Pool APIs with DRCP	3-35
3.10.7	Session Purity	3-35
3.10.8	Connection Class	3-36

3.10.8.1	Example: Setting the Connection Class as HRMS	3-36
3.10.9	Session Purity and Connection Class Defaults	3-37
3.10.10	Setting the Purity and Connection Class in the Connection String	3-37
3.10.11	Starting DRCP	3-38
3.10.12	Shut Down Connection Draining for DRCP	3-38
3.10.13	Enabling DRCP	3-39
3.10.14	Connecting to a Pool in Multi-pool DRCP	3-39
3.10.15	Implicit Connection Pooling	3-39
3.10.15.1	Implicit Stateful and Stateless Sessions	3-41
3.10.15.2	Statement and Transaction Boundary	3-41
3.10.15.3	Configuring Implicit Connection Pool Boundaries	3-42
3.10.15.4	Impact of Round-trip OCI Calls on Implicit Connection Pooling States	3-43
3.10.15.5	Deciding which Pool Boundary to Use	3-43
3.10.15.6	Implicit Connection Pooling with CMAN-TDM and PRCP	3-44
3.10.15.7	Setting or Resetting the Session State at the Boundaries During Deployment	3-44
3.10.15.8	Using the Session Cached Cursors with Implicit Connection Pooling	3-46
3.10.15.9	Security	3-46
3.10.16	Benefiting from the Scalability of DRCP in an OCI Application	3-46
3.10.17	Benefiting from the Scalability of DRCP in a Java Application	3-47
3.10.18	Best Practices for Using DRCP	3-47
3.10.19	Compatibility and Migration	3-48
3.10.20	Using DRCP with Oracle Database Native Network Encryption	3-49
3.10.21	DRCP Restrictions	3-49
3.10.22	Using DRCP with Custom Pools	3-50
3.10.23	Explicitly Marking Sessions Stateful or Stateless	3-51
3.10.24	Using DRCP with Oracle Real Application Clusters	3-52
3.10.25	DRCP with Data Guard	3-52
3.11	Memoptimize Pool	3-52
3.12	Oracle RAC Sharding	3-53

## 4 Designing Applications for Oracle Real-World Performance

---

4.1	Using Bind Variables	4-1
4.2	Using Instrumentation	4-2
4.3	Using Set-Based Processing	4-2
4.3.1	Iterative Data Processing	4-3
4.3.1.1	About Iterative Data Processing	4-3
4.3.1.2	Iterative Data Processing: Row-By-Row	4-3
4.3.1.3	Iterative Data Processing: Arrays	4-4
4.3.1.4	Iterative Data Processing: Manual Parallelism	4-5

## 5 Security

---

5.1	Enabling User Access with Grants, Roles, and Least Privilege	5-1
5.2	Automating Database Logins	5-2
5.3	Controlling User Access with Fine-Grained Access Control	5-3
5.4	Using Invoker's and Definer's Rights for Procedures and Functions	5-4
5.4.1	What Are Invoker's Rights and Definer's Rights?	5-4
5.4.2	Protecting Users Who Run Invoker's Rights Procedures and Functions	5-5
5.4.3	How Default Rights Are Handled for Java Stored Procedures	5-6
5.5	Managing External Procedures for Your Applications	5-6
5.6	Auditing User Activity	5-6

## 6 High Availability

---

6.1	Transparent Application Failover (TAF)	6-1
6.1.1	About Transparent Application Failover	6-1
6.1.2	Configuring Transparent Application Failover	6-2
6.1.3	Using Transparent Application Failover Callbacks	6-2
6.2	Oracle Connection Manager in Traffic Director Mode	6-3
6.3	About Fast Application Notification (FAN)	6-3
6.3.1	About Receiving FAN Event Notifications	6-5
6.4	About Fast Connection Failover (FCF)	6-5
6.5	About Application Continuity	6-6
6.5.1	Reset Database Session State to Prevent Application State Leaks, Use RESET_STATE	6-7
6.6	About Transaction Guard	6-10
6.7	About Service and Load Management for Database Clouds	6-10

## 7 Advanced PL/SQL Features

---

7.1	PL/SQL Data Types	7-1
7.2	Dynamic SQL	7-1
7.3	PL/SQL Optimize Level	7-2
7.4	Compiling PL/SQL Units for Native Execution	7-2
7.5	Exception Handling	7-2
7.6	Conditional Compilation	7-2
7.7	Bulk Binding	7-3

## Part II SQL for Application Developers

---

## 8 SQL Processing for Application Developers

---

8.1	Description of SQL Statement Processing	8-1
8.1.1	Stages of SQL Statement Processing	8-2
8.1.2	Shared SQL Areas	8-3
8.2	Grouping Operations into Transactions	8-4
8.2.1	Deciding How to Group Operations in Transactions	8-4
8.2.2	Improving Transaction Performance	8-5
8.2.3	Managing Commit Redo Action	8-6
8.2.4	Determining Transaction Outcome After a Recoverable Outage	8-7
8.2.4.1	Understanding Transaction Guard	8-8
8.2.4.2	Understanding DBMS_APP_CONT.GET_LTXID_OUTCOME	8-9
8.2.4.3	Using Transaction Guard	8-12
8.3	Ensuring Repeatable Reads with Read-Only Transactions	8-13
8.4	Locking Tables Explicitly	8-14
8.4.1	Privileges Required to Acquire Table Locks	8-15
8.4.2	Choosing a Locking Strategy	8-15
8.4.2.1	When to Lock with ROW SHARE MODE and ROW EXCLUSIVE MODE	8-16
8.4.2.2	When to Lock with SHARE MODE	8-16
8.4.2.3	When to Lock with SHARE ROW EXCLUSIVE MODE	8-17
8.4.2.4	When to Lock with EXCLUSIVE MODE	8-18
8.4.3	Letting Oracle Database Control Table Locking	8-18
8.4.4	Explicitly Acquiring Row Locks	8-18
8.4.5	Examples of Concurrency Under Explicit Locking	8-19
8.5	Using Oracle Lock Management Services (User Locks)	8-27
8.5.1	When to Use User Locks	8-27
8.5.2	Viewing and Monitoring Locks	8-28
8.6	Using Serializable Transactions for Concurrency Control	8-28
8.6.1	Transaction Interaction and Isolation Level	8-29
8.6.2	Setting Isolation Levels	8-31
8.6.3	Serializable Transactions and Referential Integrity	8-32
8.6.4	READ COMMITTED and SERIALIZABLE Isolation Levels	8-34
8.6.4.1	Transaction Set Consistency Differences	8-34
8.6.4.2	Choosing Transaction Isolation Levels	8-35
8.7	Nonblocking and Blocking DDL Statements	8-36
8.8	Autonomous Transactions	8-37
8.8.1	Examples of Autonomous Transactions	8-39
8.8.1.1	Ordering a Product	8-39
8.8.1.2	Withdrawing Money from a Bank Account	8-40
8.8.2	Declaring Autonomous Routines	8-43
8.9	Resuming Execution After Storage Allocation Errors	8-44
8.9.1	What Operations Have Resumable Storage Allocation?	8-44



8.9.2	Handling Suspended Storage Allocation	8-44
8.9.2.1	Using an AFTER SUSPEND Trigger in the Application	8-45
8.9.2.2	Checking for Suspended Statements	8-46
8.10	Using IF EXISTS and IF NOT EXISTS	8-47
8.10.1	Using IF NOT EXISTS with CREATE Command	8-47
8.10.2	Using IF EXISTS with ALTER Command	8-48
8.10.3	Using IF EXISTS with DROP Command	8-48
8.10.4	Supported Object Types	8-49
8.10.5	Limitations for CREATE OR REPLACE Statements	8-50
8.10.6	SQL*Plus Output Messages for DDL Statements	8-50

## 9 Using SQL Data Types in Database Applications

---

9.1	Using the Correct and Most Specific Data Type	9-1
9.1.1	How the Correct Data Type Increases Data Integrity	9-2
9.1.2	How the Most Specific Data Type Decreases Storage Requirements	9-2
9.1.3	How the Correct Data Type Improves Performance	9-3
9.2	Representing Character Data	9-6
9.3	Representing Numeric Data	9-7
9.3.1	Floating-Point Number Components	9-8
9.3.2	Floating-Point Number Formats	9-8
9.3.2.1	Binary Floating-Point Formats	9-9
9.3.3	Representing Special Values with Native Floating-Point Data Types	9-10
9.3.4	Comparing Native Floating-Point Values	9-11
9.3.5	Arithmetic Operations with Native Floating-Point Data Types	9-11
9.3.6	Conversion Functions for Native Floating-Point Data Types	9-12
9.3.7	Client Interfaces for Native Floating-Point Data Types	9-12
9.4	Representing Date and Time Data	9-13
9.4.1	Displaying Current Date and Time	9-14
9.4.2	Inserting and Displaying Dates	9-15
9.4.3	Inserting and Displaying Times	9-16
9.4.4	Arithmetic Operations with Datetime Data Types	9-18
9.4.5	Conversion Functions for Datetime Data Types	9-18
9.4.6	Importing, Exporting, and Comparing Datetime Types	9-19
9.5	Representing Specialized Data	9-19
9.5.1	Representing Spatial Data	9-19
9.5.2	Representing Large Amounts of Data	9-19
9.5.2.1	Large Objects (LOBs)	9-20
9.5.2.2	LONG and LONG RAW Data Types	9-21
9.5.3	Representing JSON Data	9-21
9.5.4	Representing Searchable Text	9-22
9.5.5	Representing XML Data	9-22

9.5.6	Representing Dynamically Typed Data	9-22
9.5.7	Representing ANSI, DB2, and SQL/DS Data	9-24
9.6	Identifying Rows by Address	9-25
9.7	Displaying Metadata for SQL Operators and Functions	9-25
9.7.1	ARGn Data Type	9-26
9.7.2	DISP_TYPE Data Type	9-26
9.7.3	SQL Data Type Families	9-27

## 10 Registering Application Data Usage with the Database

---

10.1	Data Use Case Domains	10-1
10.1.1	Overview of Use Case Domains	10-2
10.1.2	Use Case Domain Types and When to Use Them	10-2
10.1.3	Privileges Required for Use Case Domains	10-3
10.1.4	Using a Single-column Use Case Domain	10-4
10.1.4.1	Creating a Use Case Domain	10-4
10.1.4.2	Associating Use Case Domains with Columns at Table Creation	10-6
10.1.4.3	Associating Use Case Domains with Existing or New Columns	10-10
10.1.4.4	Altering a Use Case Domain	10-12
10.1.4.5	Disassociating a Use Case Domain from a Column	10-14
10.1.4.6	Dropping a Use Case Domain	10-16
10.1.5	Using a Multi-column Use Case Domain	10-18
10.1.5.1	Creating a Multi-column Use Case Domain	10-19
10.1.5.2	Associating a Multi-column Use Case Domain at Table Creation	10-20
10.1.5.3	Associating a Multi-column Use Case Domain with Existing Columns	10-22
10.1.5.4	Altering a Multi-column Use Case Domain	10-22
10.1.5.5	Disassociating a Multi-column Use Case Domain from a Column	10-23
10.1.5.6	Dropping a Multi-column Use Case Domain	10-24
10.1.6	Using a Flexible Use Case Domain	10-24
10.1.6.1	Creating a Flexible Use Case Domain	10-25
10.1.6.2	Associating a Flexible Use Case Domain at Table Creation	10-27
10.1.6.3	Associating a Flexible Domain with Existing Columns	10-30
10.1.6.4	Disassociating a Flexible Use Case Domain from Columns	10-30
10.1.6.5	Dropping a Flexible Use Case Domain	10-31
10.1.7	Using an Enumeration Use Case Domain	10-31
10.1.7.1	About Enumeration Type	10-31
10.1.7.2	Enumeration Domains Overview	10-32
10.1.7.3	Creating an Enumeration Domain	10-32
10.1.7.4	Associating an Enumeration Domain at Table Creation	10-35
10.1.7.5	Associating an Enumeration Domain with Existing Columns	10-38
10.1.8	Specifying a Data Type for a Domain	10-39
10.1.9	Changing the Use Case Domain Properties	10-43

10.1.10	SQL Functions for Use Case Domains	10-47
10.1.11	Viewing Domain Information	10-48
10.1.11.1	Dictionary Views for Use Case Domains	10-48
10.1.12	Built-in Use Case Domains	10-49
10.2	Schema Annotations	10-89
10.2.1	Overview of Annotations	10-90
10.2.2	Annotations and Comments	10-91
10.2.3	Supported Database Objects	10-91
10.2.4	Privileges Required for Using Annotations	10-91
10.2.5	DDL Statements for Annotations	10-92
10.2.5.1	Annotation Syntax	10-92
10.2.5.2	DDL Statements to Annotate a Table	10-93
10.2.5.3	DDL Statements to Annotate a Table Column	10-94
10.2.5.4	DDL Statements to Annotate Views and Materialized Views	10-95
10.2.5.5	DDL Statements to Annotate Indexes	10-96
10.2.5.6	DDL Statements to Annotate Domains	10-97
10.2.5.7	Dictionary Table and Views	10-98

## 11 Using Regular Expressions in Database Applications

---

11.1	Overview of Regular Expressions	11-1
11.2	Oracle SQL Support for Regular Expressions	11-2
11.3	Oracle SQL and POSIX Regular Expression Standard	11-4
11.4	Operators in Oracle SQL Regular Expressions	11-5
11.4.1	POSIX Operators in Oracle SQL Regular Expressions	11-5
11.4.2	Oracle SQL Multilingual Extensions to POSIX Standard	11-8
11.4.3	Oracle SQL PERL-Influenced Extensions to POSIX Standard	11-9
11.5	Using Regular Expressions in SQL Statements: Scenarios	11-10
11.5.1	Using a Constraint to Enforce a Phone Number Format	11-10
11.5.2	Example: Enforcing a Phone Number Format with Regular Expressions	11-11
11.5.3	Example: Inserting Phone Numbers in Correct and Incorrect Formats	11-11
11.5.4	Using Back References to Reposition Characters	11-12

## 12 Using Indexes in Database Applications

---

12.1	Guidelines for Managing Indexes	12-1
12.2	Managing Indexes	12-2
12.3	When to Use Domain Indexes	12-2
12.4	When to Use Function-Based Indexes	12-2
12.4.1	Advantages of Function-Based Indexes	12-3
12.4.2	Disadvantages of Function-Based Indexes	12-4
12.4.3	Example: Function-Based Index for Precomputing Arithmetic Expression	12-6

12.4.4	Example: Function-Based Indexes on Object Column	12-7
12.4.5	Example: Function-Based Index for Faster Case-Insensitive Searches	12-8
12.4.6	Example: Function-Based Index for Language-Dependent Sorting	12-8

## 13 Maintaining Data Integrity in Database Applications

---

13.1	Enforcing Business Rules with Constraints	13-2
13.2	Enforcing Business Rules with Both Constraints and Application Code	13-3
13.3	Creating Indexes for Use with Constraints	13-4
13.4	When to Use NOT NULL Constraints	13-5
13.5	When to Use Default Column Values	13-6
13.6	Choosing a Primary Key for a Table (PRIMARY KEY Constraint)	13-7
13.7	When to Use UNIQUE Constraints	13-8
13.8	Enforcing Referential Integrity with FOREIGN KEY Constraints	13-9
13.8.1	FOREIGN KEY Constraints and NULL Values	13-11
13.8.2	Defining Relationships Between Parent and Child Tables	13-11
13.8.3	Rules for Multiple FOREIGN KEY Constraints	13-12
13.8.4	Deferring Constraint Checks	13-12
13.9	Minimizing Space and Time Overhead for Indexes Associated with Constraints	13-15
13.10	Guidelines for Indexing Foreign Keys	13-15
13.11	Referential Integrity in a Distributed Database	13-15
13.12	When to Use CHECK Constraints	13-16
13.12.1	Restrictions on CHECK Constraints	13-16
13.12.2	Designing CHECK Constraints	13-17
13.12.3	Rules for Multiple CHECK Constraints	13-17
13.12.4	Choosing Between CHECK and NOT NULL Constraints	13-18
13.13	Using PRECHECK to Pre-validate a CHECK Constraint	13-18
13.13.1	PRECHECK Syntax and Definition	13-19
13.13.1.1	Supported Conditions for JSON Schema Validation	13-20
13.13.2	Enabling PRECHECK for a New Relational Table	13-26
13.13.3	Enabling PRECHECK for an Existing Table	13-29
13.13.4	Guidelines for Using PRECHECK	13-31
13.14	Examples of Defining Constraints	13-32
13.14.1	Privileges Needed to Define Constraints	13-33
13.14.2	Naming Constraints	13-34
13.15	Enabling and Disabling Constraints	13-34
13.15.1	Why Disable Constraints?	13-34
13.15.2	Creating Enabled Constraints (Default)	13-35
13.15.3	Creating Disabled Constraints	13-35
13.15.4	Enabling Existing Constraints	13-36
13.15.5	Disabling Existing Constraints	13-36
13.15.6	Guidelines for Enabling and Disabling Key Constraints	13-37

13.15.7	Fixing Constraint Exceptions	13-37
13.16	Modifying Constraints	13-38
13.17	Renaming Constraints	13-39
13.18	Dropping Constraints	13-40
13.19	Managing FOREIGN KEY Constraints	13-40
13.19.1	Data Types and Names for Foreign Key Columns	13-41
13.19.2	Limit on Columns in Composite Foreign Keys	13-41
13.19.3	Foreign Key References Primary Key by Default	13-41
13.19.4	Privileges Required to Create FOREIGN KEY Constraints	13-41
13.19.5	Choosing How Foreign Keys Enforce Referential Integrity	13-41
13.20	Viewing Information About Constraints	13-42

## Part III PL/SQL for Application Developers

---

### 14 Coding PL/SQL Subprograms and Packages

---

14.1	Overview of PL/SQL Subprograms	14-1
14.2	Overview of PL/SQL Packages	14-3
14.3	Overview of PL/SQL Units	14-4
14.3.1	PLSQL_OPTIMIZE_LEVEL Compilation Parameter	14-4
14.4	Creating PL/SQL Subprograms and Packages	14-6
14.4.1	Privileges Needed to Create Subprograms and Packages	14-7
14.4.2	Creating Subprograms and Packages	14-7
14.4.3	PL/SQL Object Size Limits	14-8
14.4.4	PL/SQL Data Types	14-9
14.4.4.1	PL/SQL Scalar Data Types	14-9
14.4.4.2	PL/SQL Composite Data Types	14-12
14.4.4.3	Abstract Data Types	14-12
14.4.5	Returning Result Sets to Clients	14-12
14.4.5.1	Advantages of Cursor Variables	14-13
14.4.5.2	Disadvantages of Cursor Variables	14-14
14.4.5.3	Returning Query Results Implicitly	14-17
14.4.6	Returning Large Amounts of Data from a Function	14-17
14.4.7	PL/SQL Function Result Cache	14-17
14.4.8	Overview of Bulk Binding	14-18
14.4.8.1	DML Statements that Reference Collections	14-19
14.4.8.2	SELECT Statements that Reference Collections	14-19
14.4.8.3	FOR Loops that Reference Collections and Return DML	14-20
14.4.9	PL/SQL Dynamic SQL	14-21
14.5	Altering PL/SQL Subprograms and Packages	14-22
14.6	Deprecating Packages, Subprograms, and Types	14-23

14.7	Dropping PL/SQL Subprograms and Packages	14-23
14.8	Compiling PL/SQL Units for Native Execution	14-23
14.9	Invoking Stored PL/SQL Subprograms	14-24
14.9.1	Privileges Required to Invoke a Stored Subprogram	14-25
14.9.2	Invoking a Subprogram Interactively from Oracle Tools	14-25
14.9.3	Invoking a Subprogram from Another Subprogram	14-27
14.9.4	Invoking a Remote Subprogram	14-28
14.9.4.1	Synonyms for Remote Subprograms	14-29
14.9.4.2	Transactions That Invoke Remote Subprograms	14-30
14.10	Invoking Stored PL/SQL Functions from SQL Statements	14-31
14.10.1	Why Invoke PL/SQL Functions from SQL Statements?	14-32
14.10.2	Where PL/SQL Functions Can Appear in SQL Statements	14-32
14.10.3	When PL/SQL Functions Can Appear in SQL Expressions	14-33
14.10.4	Controlling Side Effects of PL/SQL Functions Invoked from SQL Statements	14-33
14.10.4.1	Restrictions on Functions Invoked from SQL Statements	14-34
14.10.4.2	PL/SQL Functions Invoked from Parallelized SQL Statements	14-35
14.10.4.3	PRAGMA RESTRICT_REFERENCES	14-36
14.11	Analyzing and Debugging Stored Subprograms	14-39
14.11.1	PL/Scope	14-39
14.11.2	PL/SQL Hierarchical Profiler	14-40
14.11.3	Debugging PL/SQL and Java	14-40
14.11.3.1	Compiling Code for Debugging	14-40
14.11.3.2	Privileges for Debugging PL/SQL and Java Stored Subprograms	14-41
14.12	Package Invalidations and Session State	14-42
14.13	Example: Raising an ORA-04068 Error	14-42
14.14	Example: Trapping ORA-04068	14-43

## 15 Using PL/Scope

---

15.1	Overview of PL/Scope	15-1
15.2	Privileges Required for Using PL/Scope	15-2
15.3	Specifying Identifier and Statement Collection	15-2
15.4	How Much Space is PL/Scope Data Using?	15-3
15.5	Viewing PL/Scope Data	15-4
15.5.1	Static Data Dictionary Views for PL/SQL and SQL Identifiers	15-4
15.5.1.1	PL/SQL and SQL Identifier Types that PL/Scope Collects	15-4
15.5.1.2	About Identifiers Usages	15-6
15.5.1.3	Identifiers Usage Unique Keys	15-8
15.5.1.4	About Identifiers Usage Context	15-9
15.5.1.5	About Identifiers Signature	15-11
15.5.2	Static Data Dictionary Views for SQL Statements	15-12
15.5.2.1	SQL Statement Types that PL/Scope Collects	15-13

15.5.2.2	Statements Location Unique Keys	15-14
15.5.2.3	About SQL Statement Usage Context	15-15
15.5.2.4	About SQL Statements Signature	15-16
15.5.3	SQL Developer	15-17
15.6	Overview of Data Dictionary Views Useful to Manage PL/SQL Code	15-17
15.7	Sample PL/Scope Session	15-18

## 16 Using the PL/SQL Hierarchical Profiler

---

16.1	Overview of PL/SQL Hierarchical Profiler	16-1
16.2	Collecting Profile Data	16-2
16.3	Understanding Raw Profiler Output	16-4
16.3.1	Namespaces of Tracked Subprograms	16-7
16.3.2	Special Function Names	16-7
16.4	Analyzing Profile Data	16-8
16.4.1	Creating Hierarchical Profiler Tables	16-8
16.4.2	Understanding Hierarchical Profiler Tables	16-9
16.4.2.1	Hierarchical Profiler Database Table Columns	16-10
16.4.2.2	Distinguishing Between Overloaded Subprograms	16-12
16.4.2.3	Hierarchical Profiler Tables for Sample PL/SQL Procedure	16-12
16.4.2.4	Examples of Calls to DBMS_HPROF.analyze with Options	16-13
16.5	plshprof Utility	16-15
16.5.1	plshprof Options	16-15
16.5.2	HTML Report from a Single Raw Profiler Output File	16-16
16.5.2.1	First Page of Report	16-16
16.5.2.2	Function-Level Reports	16-17
16.5.2.3	Module-Level Reports	16-18
16.5.2.4	Namespace-Level Reports	16-19
16.5.2.5	Parents and Children Report for a Function	16-19
16.5.2.6	Understanding PL/SQL Hierarchical Profiler SQL-Level Reports	16-21
16.5.3	HTML Difference Report from Two Raw Profiler Output Files	16-21
16.5.3.1	Difference Report Conventions	16-22
16.5.3.2	First Page of Difference Report	16-22
16.5.3.3	Function-Level Difference Reports	16-23
16.5.3.4	Module-Level Difference Reports	16-25
16.5.3.5	Namespace-Level Difference Reports	16-25
16.5.3.6	Parents and Children Difference Report for a Function	16-25

## 17 Using PL/SQL Basic Block Coverage to Maintain Quality

---

17.1	Overview of PL/SQL Basic Block Coverage	17-1
17.2	Collecting PL/SQL Code Coverage Data	17-2

## 18 Developing PL/SQL Web Applications

---

18.1	Overview of PL/SQL Web Applications	18-1
18.2	Implementing PL/SQL Web Applications	18-2
18.2.1	PL/SQL Gateway	18-2
18.2.1.1	mod_plsql	18-2
18.2.1.2	Embedded PL/SQL Gateway	18-3
18.2.2	PL/SQL Web Toolkit	18-3
18.3	Using mod_plsql Gateway to Map Client Requests to a PL/SQL Web Application	18-4
18.4	Using Embedded PL/SQL Gateway	18-5
18.4.1	How Embedded PL/SQL Gateway Processes Client Requests	18-5
18.4.2	Installing Embedded PL/SQL Gateway	18-7
18.4.3	Configuring Embedded PL/SQL Gateway	18-7
18.4.3.1	Configuring Embedded PL/SQL Gateway: Overview	18-7
18.4.3.2	Configuring User Authentication for Embedded PL/SQL Gateway	18-9
18.4.4	Invoking PL/SQL Stored Subprograms Through Embedded PL/SQL Gateway	18-18
18.4.5	Securing Application Access with Embedded PL/SQL Gateway	18-19
18.4.6	Restrictions in Embedded PL/SQL Gateway	18-19
18.4.7	Using Embedded PL/SQL Gateway: Scenario	18-20
18.5	Generating HTML Output with PL/SQL	18-21
18.6	Passing Parameters to PL/SQL Web Applications	18-22
18.6.1	Passing List and Dropdown-List Parameters from an HTML Form	18-23
18.6.2	Passing Option and Check Box Parameters from an HTML Form	18-23
18.6.3	Passing Entry-Field Parameters from an HTML Form	18-24
18.6.4	Passing Hidden Parameters from an HTML Form	18-25
18.6.5	Uploading a File from an HTML Form	18-26
18.6.6	Submitting a Completed HTML Form	18-26
18.6.7	Handling Missing Input from an HTML Form	18-26
18.6.8	Maintaining State Information Between Web Pages	18-27
18.7	Performing Network Operations in PL/SQL Subprograms	18-27
18.7.1	Internet Protocol Version 6 (IPv6) Support	18-28
18.7.2	Sending E-Mail from PL/SQL	18-28
18.7.3	Getting a Host Name or Address from PL/SQL	18-29
18.7.4	Using TCP/IP Connections from PL/SQL	18-29
18.7.5	Retrieving HTTP URL Contents from PL/SQL	18-29
18.7.6	Using Tables, Image Maps, Cookies, and CGI Variables from PL/SQL	18-32



## 19 Using Continuous Query Notification (CQN)

---

19.1	About Object Change Notification (OCN)	19-2
19.2	About Query Result Change Notification (QRCN)	19-2
19.2.1	Guaranteed Mode	19-3
19.2.2	Best-Effort Mode	19-3
19.2.2.1	Example: Query Too Complex for QRCN in Guaranteed Mode	19-4
19.2.2.2	Example: Query Whose Simplified Version Invalidates Objects	19-4
19.3	Events that Generate Notifications	19-5
19.3.1	Committed DML Transactions	19-5
19.3.2	Committed DDL Statements	19-6
19.3.3	Deregistration	19-7
19.3.4	Global Events	19-7
19.4	Notification Contents	19-7
19.5	Good Candidates for CQN	19-8
19.6	Creating CQN Registrations	19-11
19.7	Using PL/SQL to Create CQN Registrations	19-11
19.7.1	PL/SQL CQN Registration Interface	19-12
19.7.2	CQN Registration Options	19-12
19.7.2.1	Notification Type Option	19-13
19.7.2.2	QRCN Mode (QRCN Notification Type Only)	19-13
19.7.2.3	ROWID Option	19-13
19.7.2.4	Operations Filter Option (OCN Notification Type Only)	19-14
19.7.2.5	Transaction Lag Option (OCN Notification Type Only)	19-15
19.7.2.6	Notification Grouping Options	19-15
19.7.2.7	Reliable Option	19-16
19.7.2.8	Purge-on-Notify and Timeout Options	19-16
19.7.3	Prerequisites for Creating CQN Registrations	19-17
19.7.4	Queries that Can Be Registered for Object Change Notification (OCN)	19-17
19.7.5	Queries that Can Be Registered for Query Result Change Notification (QRCN)	19-18
19.7.5.1	Queries that Can Be Registered for QRCN in Guaranteed Mode	19-18
19.7.5.2	Queries that Can Be Registered for QRCN Only in Best-Effort Mode	19-19
19.7.5.3	Queries that Cannot Be Registered for QRCN in Either Mode	19-20
19.7.6	Using PL/SQL to Register Queries for CQN	19-20
19.7.6.1	Creating a PL/SQL Notification Handler	19-21
19.7.6.2	Creating a CQ_NOTIFICATION\$_REG_INFO Object	19-22
19.7.6.3	Identifying Individual Queries in a Notification	19-25
19.7.6.4	Adding Queries to an Existing Registration	19-25
19.7.7	Best Practices for CQN Registrations	19-25
19.7.8	Troubleshooting CQN Registrations	19-26
19.7.9	Deleting Registrations	19-27
19.7.10	Configuring CQN: Scenario	19-27

19.7.10.1	Creating a PL/SQL Notification Handler	19-27
19.7.10.2	Registering the Queries	19-29
19.8	Using OCI to Create CQN Registrations	19-31
19.8.1	Using OCI for Query Result Set Notifications	19-32
19.8.2	Using OCI to Register a Continuous Query Notification	19-32
19.8.3	Using OCI for Client Initiated CQN Registrations	19-34
19.8.4	Using OCI Subscription Handle Attributes for Continuous Query Notification	19-34
19.8.5	OCI_ATTR_CQ_QUERYID Attribute	19-36
19.8.6	Using OCI Continuous Query Notification Descriptors	19-36
19.8.6.1	OCI_DTYPE_CHDES	19-36
19.8.7	Demonstrating Continuous Query Notification in an OCI Sample Program	19-38
19.9	Querying CQN Registrations	19-48
19.10	Interpreting Notifications	19-48
19.10.1	Interpreting a CQ_NOTIFICATION\$_DESCRIPTOR Object	19-48
19.10.2	Interpreting a CQ_NOTIFICATION\$_TABLE Object	19-49
19.10.3	Interpreting a CQ_NOTIFICATION\$_QUERY Object	19-50
19.10.4	Interpreting a CQ_NOTIFICATION\$_ROW Object	19-50

## Part IV Advanced Topics for Application Developers

---

### 20 Choosing a Programming Environment

---

20.1	Overview of Application Architecture	20-2
20.1.1	Client/Server Architecture	20-2
20.1.2	Server-Side Programming	20-2
20.1.3	Two-Tier and Three-Tier Architecture	20-3
20.2	Overview of the Program Interface	20-3
20.2.1	User Interface	20-3
20.2.2	Stateful and Stateless User Interfaces	20-4
20.3	Overview of PL/SQL	20-4
20.4	Overview of Oracle Database Java Support	20-4
20.4.1	Overview of Oracle JVM	20-5
20.4.2	Overview of Oracle JDBC	20-6
20.4.2.1	Oracle JDBC Drivers	20-7
20.4.2.2	Sample JDBC 2.0 Program	20-8
20.4.2.3	Sample Pre-2.0 JDBC Program	20-9
20.4.3	Overview of Oracle SQLJ	20-9
20.4.3.1	Benefits of SQLJ	20-10
20.4.4	Comparison of Oracle JDBC and Oracle SQLJ	20-11
20.4.5	Overview of Java Stored Subprograms	20-12
20.4.6	Overview of Oracle Database Web Services	20-12

20.5	Overview of JavaScript	20-13
20.5.1	Multilingual Engine Overview	20-14
20.5.2	MLE Concepts	20-14
20.5.3	Understanding MLE Execution Context and Runtime Isolation	20-15
20.5.4	MLE Environment Overview	20-16
20.5.5	JavaScript MLE Modules Overview	20-17
20.5.6	JavaScript MLE Call Specification Overview	20-17
20.5.7	Invoking JavaScript in the Database	20-18
20.5.8	Invoking JavaScript Using MLE Modules	20-18
20.5.8.1	Using MLE Module Contexts	20-19
20.5.8.2	Specifying an Environment for Call Specifications	20-19
20.5.8.3	Managing JavaScript MLE Modules	20-20
20.5.8.4	Running JavaScript Code Using MLE Modules	20-21
20.5.9	Invoking JavaScript Using Dynamic MLE Execution	20-23
20.5.9.1	Dynamic MLE Execution Overview	20-23
20.5.9.2	Using Dynamic MLE Execution contexts	20-24
20.5.9.3	Specifying an Environment for Dynamic MLE Contexts	20-24
20.5.9.4	Running JavaScript Code Using Dynamic MLE Execution	20-25
20.5.10	Privileges for Working with JavaScript in MLE	20-26
20.5.10.1	MLE User Privileges	20-27
20.5.11	Other Supported MLE Features	20-27
20.6	Choosing PL/SQL, Java, or JavaScript	20-29
20.6.1	Similarities of PL/SQL, Java, and JavaScript	20-31
20.6.2	Advantages of PL/SQL	20-31
20.6.3	Advantages of Java	20-32
20.6.4	Advantages of JavaScript	20-33
20.7	Overview of Precompilers	20-33
20.7.1	Overview of the Pro*C/C++ Precompiler	20-34
20.7.2	Overview of the Pro*COBOL Precompiler	20-36
20.8	Overview of OCI and OCCI	20-37
20.8.1	Advantages of OCI and OCCI	20-38
20.8.2	OCI and OCCI Functions	20-39
20.8.3	Procedural and Nonprocedural Elements of OCI and OCCI Applications	20-39
20.8.4	Building an OCI or OCCI Application	20-39
20.9	Comparison of Precompilers and OCI	20-40
20.10	Overview of Oracle Data Provider for .NET (ODP.NET)	20-41
20.11	Overview of OraOLEDB	20-42

## 21 Developing Applications with Multiple Programming Languages

---

21.1	Overview of Multilanguage Programs	21-1
21.2	What Is an External Procedure?	21-3

21.3	Overview of Call Specification for External Procedures	21-3
21.4	Loading External Procedures	21-4
21.4.1	Define the C Procedures	21-5
21.4.2	Set Up the Environment	21-6
21.4.3	Identify the DLL	21-8
21.4.4	Publish the External Procedures	21-9
21.5	Publishing External Procedures	21-10
21.5.1	AS LANGUAGE Clause for Java Class Methods	21-11
21.5.2	AS LANGUAGE Clause for External C Procedures	21-11
21.5.2.1	LIBRARY	21-11
21.5.2.2	NAME	21-11
21.5.2.3	LANGUAGE	21-12
21.5.2.4	CALLING STANDARD	21-12
21.5.2.5	WITH CONTEXT	21-12
21.5.2.6	PARAMETERS	21-12
21.5.2.7	AGENT IN	21-12
21.6	Publishing Java Class Methods	21-12
21.7	Publishing External C Procedures	21-13
21.8	Locations of Call Specifications	21-13
21.8.1	Example: Locating a Call Specification in a PL/SQL Package	21-14
21.8.2	Example: Locating a Call Specification in a PL/SQL Package Body	21-14
21.8.3	Example: Locating a Call Specification in an ADT Specification	21-15
21.8.4	Example: Locating a Call Specification in an ADT Body	21-15
21.8.5	Example: Java with AUTHID	21-15
21.8.6	Example: C with Optional AUTHID	21-15
21.8.7	Example: Mixing Call Specifications in a Package	21-16
21.9	Passing Parameters to External C Procedures with Call Specifications	21-17
21.9.1	Specifying Data Types	21-17
21.9.2	External Data Type Mappings	21-19
21.9.3	Passing Parameters BY VALUE or BY REFERENCE	21-22
21.9.4	Declaring Formal Parameters	21-22
21.9.5	Overriding Default Data Type Mapping	21-23
21.9.6	Specifying Properties	21-23
21.9.6.1	INDICATOR	21-25
21.9.6.2	LENGTH and MAXLEN	21-25
21.9.6.3	CHARSETID and CHARSETFORM	21-26
21.9.6.4	Repositioning Parameters	21-26
21.9.6.5	SELF	21-26
21.9.6.6	BY REFERENCE	21-29
21.9.6.7	WITH CONTEXT	21-29
21.9.6.8	Interlanguage Parameter Mode Mappings	21-29
21.10	Running External Procedures with CALL Statements	21-30

21.10.1	Preconditions for External Procedures	21-31
21.10.1.1	Privileges of External Procedures	21-31
21.10.1.2	Managing Permissions	21-31
21.10.1.3	Creating Synonyms for External Procedures	21-31
21.10.2	CALL Statement Syntax	21-31
21.10.3	Calling Java Class Methods	21-32
21.10.4	Calling External C Procedures	21-32
21.11	Handling Errors and Exceptions in Multilanguage Programs	21-33
21.12	Using Service Routines with External C Procedures	21-33
21.12.1	OCIExtProcAllocCallMemory	21-34
21.12.2	OCIExtProcRaiseExcp	21-38
21.12.3	OCIExtProcRaiseExcpWithMsg	21-39
21.13	Doing Callbacks with External C Procedures	21-39
21.13.1	OCIExtProcGetEnv	21-40
21.13.2	Object Support for OCI Callbacks	21-41
21.13.3	Restrictions on Callbacks	21-41
21.13.4	Debugging External C Procedures	21-42
21.13.5	Example: Calling an External C Procedure	21-43
21.13.6	Global Variables in External C Procedures	21-43
21.13.7	Static Variables in External C Procedures	21-44
21.13.8	Restrictions on External C Procedures	21-44

## 22 Using Oracle Flashback Technology

---

22.1	Overview of Oracle Flashback Technology	22-1
22.1.1	Application Development Features	22-2
22.1.2	Database Administration Features	22-4
22.2	Configuring Your Database for Oracle Flashback Technology	22-5
22.2.1	Configuring Your Database for Automatic Undo Management	22-5
22.2.2	Configuring Your Database for Oracle Flashback Transaction Query	22-7
22.2.3	Configuring Your Database for Flashback Transaction	22-7
22.2.4	Enabling Oracle Flashback Operations on Specific LOB Columns	22-7
22.2.5	Granting Necessary Privileges	22-8
22.3	Using Oracle Flashback Query (SELECT AS OF)	22-9
22.3.1	Example: Examining and Restoring Past Data	22-9
22.3.2	Guidelines for Oracle Flashback Query	22-10
22.4	Using Oracle Flashback Version Query	22-11
22.5	Using Oracle Flashback Transaction Query	22-13
22.6	Using Oracle Flashback Transaction Query with Oracle Flashback Version Query	22-14
22.7	Using DBMS_FLASHBACK Package	22-16
22.7.1	Using Flashback Version Query with DBMS_FLASHBACK	22-17
22.8	Using Flashback Transaction	22-18

22.8.1	Dependent Transactions	22-19
22.8.2	TRANSACTION_BACKOUT Parameters	22-19
22.8.3	TRANSACTION_BACKOUT Reports	22-20
22.8.3.1	*_FLASHBACK_TXN_STATE	22-20
22.8.3.2	*_FLASHBACK_TXN_REPORT	22-21
22.9	Using Flashback Time Travel	22-21
22.9.1	DDL Statements on Tables Enabled for Flashback Archive	22-23
22.9.2	Creating a Flashback Archive	22-24
22.9.3	Altering a Flashback Archive	22-25
22.9.4	Dropping a Flashback Archive	22-27
22.9.5	Specifying the Default Flashback Archive	22-27
22.9.6	Enabling and Disabling Flashback Archive	22-28
22.9.7	Viewing Flashback Archive Data	22-29
22.9.8	Transporting Flashback Archive Data between Databases	22-29
22.9.9	Flashback Time Travel Scenarios	22-29
22.9.9.1	Scenario: Using Flashback Time Travel to Enforce Digital Shredding	22-30
22.9.9.2	Scenario: Using Flashback Time Travel to Access Historical Data	22-30
22.9.9.3	Scenario: Using Flashback Time Travel to Generate Reports	22-30
22.9.9.4	Scenario: Using Flashback Time Travel for Auditing	22-31
22.9.9.5	Scenario: Using Flashback Time Travel to Recover Data	22-31
22.9.10	Protecting Flashback Archive Data	22-32
22.10	General Guidelines for Oracle Flashback Technology	22-34
22.11	Oracle Virtual Private Database Policies and Oracle Flashback Time Travel	22-36
22.12	Performance Guidelines for Oracle Flashback Technology	22-39
22.13	Multitenant Container Database Restrictions for Oracle Flashback Technology	22-40

## 23 Developing Applications with the Publish-Subscribe Model

---

23.1	Introduction to the Publish-Subscribe Model	23-1
23.2	Publish-Subscribe Architecture	23-2
23.2.1	Database Events	23-2
23.2.2	Oracle Advanced Queuing	23-2
23.2.3	Client Notification	23-2
23.3	Publish-Subscribe Concepts	23-3
23.4	Examples of a Publish-Subscribe Mechanism	23-5

## 24 Using the Oracle Database ODBC Driver

---

## 25 Using the Identity Code Package

---

25.1	Identity Concepts	25-1
25.2	What Is the Identity Code Package?	25-5
25.3	Using the Identity Code Package	25-6
25.3.1	Storing RFID Tags in Oracle Database Using MGD_ID ADT	25-6
25.3.1.1	Creating a Table with MGD_ID Column Type and Storing EPC Tag Encodings in the Column	25-6
25.3.1.2	Constructing MGD_ID Objects to Represent RFID Tags	25-7
25.3.1.3	Inserting an MGD_ID Object into a Database Table	25-9
25.3.1.4	Querying MGD_ID Column Type	25-10
25.3.2	Building a Function-Based Index Using the Member Functions of the MGD_ID Column Type	25-10
25.3.3	Using MGD_ID ADT Functions	25-11
25.3.3.1	Using the get_component Function with the MGD_ID Object	25-11
25.3.3.2	Parsing Tag Data from Standard Representations	25-11
25.3.3.3	Reconstructing Tag Representations from Fields	25-13
25.3.3.4	Translating Between Tag Representations	25-13
25.3.4	Defining a Category of Identity Codes and Adding Encoding Schemes to an Existing Category	25-13
25.3.4.1	Creating a Category of Identity Codes	25-14
25.3.4.2	Adding Two Metadata Schemes to a Newly Created Category	25-14
25.4	Identity Code Package Types	25-18
25.5	DBMS_MGD_ID_UTL Package	25-19
25.6	Identity Code Metadata Tables and Views	25-20
25.7	Electronic Product Code (EPC) Concepts	25-22
25.7.1	RFID Technology and EPC v1.1 Coding Schemes	25-22
25.7.2	Product Code Concepts and Their Current Use	25-23
25.7.2.1	Electronic Product Code (EPC)	25-23
25.7.2.2	Global Trade Identification Number (GTIN) and Serializable Global Trade Identification Number (SGTIN)	25-24
25.7.2.3	Serial Shipping Container Code (SSCC)	25-25
25.7.2.4	Global Location Number (GLN) and Serializable Global Location Number (SGLN)	25-25
25.7.2.5	Global Returnable Asset Identifier (GRAI)	25-25
25.7.2.6	Global Individual Asset Identifier (GIAI)	25-25
25.7.2.7	RFID EPC Network	25-25
25.8	Oracle Database Tag Data Translation Schema	25-25

## 26 Microservices Architecture

---

26.1	About Microservices Architecture	26-1
26.2	Features of Microservices Architecture	26-3
26.3	Challenges in a Distributed System	26-3

26.4	Solutions for Microservices	26-4
26.4.1	Two-Phase Commit Pattern	26-5
26.4.2	Saga Design Pattern	26-6
26.4.2.1	Why Use Sagas?	26-6
26.4.2.2	Saga Implementation Approaches	26-7
26.4.2.3	Successful and Unsuccessful Sagas	26-7
26.4.2.4	Saga Flow	26-7
26.4.3	Backend as a Service For The 12 Patterns For Microservices Success	26-8

## 27 Oracle Backend for Microservices and AI

---

27.1	About Oracle Backend for Microservices and AI	27-1
27.2	Getting Started with Oracle Backend for Microservices and AI	27-1
27.2.1	CloudBank Sample Application	27-3

## 28 Developing Applications with Sagas

---

28.1	Implementing Sagas with Oracle Database	28-1
28.2	Oracle Saga Framework Overview	28-2
28.3	Saga Framework Features	28-2
28.4	Saga Framework Concepts	28-3
28.5	Initializing the Saga Framework	28-6
28.6	Setting Up a Saga Topology	28-6
28.6.1	Adding a Message Broker	28-7
28.6.2	Adding a Coordinator	28-7
28.6.3	Adding a Participant	28-8
28.6.4	Managing Participants and Message Brokers	28-8
28.6.5	Message Propagation	28-9
28.6.6	About Dictionary Tables	28-9
28.6.7	Example: Saga Framework Setup	28-10
28.7	Managing a Saga Using the PL/SQL Interface	28-12
28.7.1	Example: Saga PL/SQL Program	28-12
28.8	Developing Java Applications Using Saga Annotations	28-13
28.8.1	LRA and Saga Annotations	28-14
28.8.2	Packaging	28-19
28.8.3	Configuration	28-19
28.8.4	Saga Interface and Classes	28-21
28.8.4.1	Saga Interface	28-21
28.8.4.2	SagaMessageContext Class	28-30
28.8.4.3	SagaParticipant Class	28-31
28.8.4.4	SagaInitiator Class	28-33
28.8.5	Example Program	28-35



28.9	Finalizing a Saga Explicitly	28-40
28.9.1	PL/SQL Callbacks for a PL/SQL Client	28-40
28.9.2	Integration with Lock-Free Reservation	28-42
28.10	AfterSaga Callbacks	28-43

## 29 Using Lock-Free Reservation

---

29.1	About Concurrency in Transaction Processing	29-1
29.2	Lock-Free Reservation Terminology	29-2
29.3	Lock-Free Reservation	29-3
29.3.1	Comparing Optimistic Locking and Lock-Free Reservation	29-5
29.3.2	Creating a Reservable Column at Table Creation	29-5
29.3.2.1	Reservation Journal Table Columns	29-6
29.3.3	Adding or Modifying Reservable Columns	29-7
29.3.4	About CHECK Constraints in Reservable Columns	29-8
29.3.5	Example: Conventional Locking and Lock-Free Reservation	29-9
29.3.6	Querying Reservable Column Views	29-10
29.4	Benefits of Using Lock-Free Reservation	29-11
29.5	Guidelines and Restrictions for Lock-Free Reservation	29-11
29.5.1	Guidelines and Restrictions for Reservable Columns	29-11
29.5.2	Guidelines and Restrictions for Update Statements	29-12
29.5.3	Guidelines for Inserts and Deletes	29-13
29.5.4	Guidelines for Concurrent DDL Statements	29-13
29.5.5	Restrictions for Reservation Journal Table	29-14

## 30 Developing Applications with Oracle XA

---

30.1	X/Open Distributed Transaction Processing (DTP)	30-2
30.1.1	DTP Terminology	30-3
30.1.2	Required Public Information	30-5
30.2	Oracle XA Library Subprograms	30-5
30.2.1	Oracle XA Library Subprograms	30-6
30.2.2	Oracle XA Interface Extensions	30-6
30.3	Developing and Installing XA Applications	30-7
30.3.1	DBA or System Administrator Responsibilities	30-7
30.3.2	Application Developer Responsibilities	30-8
30.3.3	Defining the xa_open String	30-8
30.3.3.1	Syntax of the xa_open String	30-9
30.3.3.2	Required Fields for the xa_open String	30-9
30.3.3.3	Optional Fields for the xa_open String	30-10
30.3.4	Using Oracle XA with Precompilers	30-12
30.3.4.1	Using Precompilers with the Default Database	30-12

30.3.4.2	Using Precompilers with a Named Database	30-13
30.3.5	Using Oracle XA with OCI	30-14
30.3.6	Managing Transaction Control with Oracle XA	30-14
30.3.7	Examples of Precompiler Applications	30-15
30.3.8	Migrating Precompiler or OCI Applications to TPM Applications	30-16
30.3.9	Managing Oracle XA Library Thread Safety	30-17
30.3.9.1	Specifying Threading in the Open String	30-17
30.3.9.2	Restrictions on Threading in Oracle XA	30-17
30.3.10	Using the DBMS_XA Package	30-18
30.4	Troubleshooting XA Applications	30-20
30.4.1	Accessing Oracle XA Trace Files	30-20
30.4.1.1	xa_open String DbgFl	30-21
30.4.1.2	Trace File Locations	30-22
30.4.2	Managing In-Doubt or Pending Oracle XA Transactions	30-22
30.4.3	Using SYS Account Tables to Monitor Oracle XA Transactions	30-22
30.5	Oracle XA Issues and Restrictions	30-23
30.5.1	Using Database Links in Oracle XA Applications	30-23
30.5.2	Managing Transaction Branches in Oracle XA Applications	30-24
30.5.3	Using Oracle XA with Oracle Real Application Clusters (Oracle RAC)	30-24
30.5.3.1	Oracle RAC XA Limitations	30-25
30.5.3.2	GLOBAL_TXN_PROCESSES Initialization Parameter	30-25
30.5.3.3	Managing Transaction Branches on Oracle RAC	30-25
30.5.3.4	Managing Instance Recovery in Oracle RAC with DTP Services (10.2)	30-27
30.5.3.5	Global Uniqueness of XIDs in Oracle RAC	30-27
30.5.3.6	Tight and Loose Coupling	30-28
30.5.4	SQL-Based Oracle XA Restrictions	30-28
30.5.4.1	Rollbacks and Commits	30-28
30.5.4.2	DDL Statements	30-28
30.5.4.3	Session State	30-29
30.5.4.4	EXEC SQL	30-29
30.5.5	Miscellaneous Restrictions	30-29

## 31 Developing Applications with Sessionless Transactions

---

31.1	Introduction to Managing Transactions and Sessionless Transactions	31-1
31.2	Sessionless Transactions Overview	31-3
31.3	Sessionless Transactions Capabilities	31-4
31.3.1	Example Use Cases	31-8
31.4	Benefits of Sessionless Transactions	31-12
31.5	Using Sessionless Transactions	31-13
31.5.1	Understanding Active Sessionless Transactions	31-14
31.5.2	Understanding the Lifecycle of Sessionless Transactions	31-14

31.5.3	Understanding Server Round Trips and Pre-call and Post-call Functions	31-14
31.5.4	Prerequisites for Using Sessionless Transactions	31-16
31.5.5	Setting a Global Transaction ID	31-16
31.5.6	Starting a New Sessionless Transaction	31-18
31.5.6.1	Running SQL Statements within Sessionless Transactions	31-22
31.5.7	Retrieving a Global Transaction ID	31-22
31.5.8	Suspending a Sessionless Transaction	31-24
31.5.8.1	Time-out of a Suspended Sessionless Transaction	31-27
31.5.9	Resuming a Suspended Sessionless Transaction	31-29
31.5.10	Finalizing a Sessionless Transaction	31-33
31.5.10.1	Committing a Sessionless Transaction	31-33
31.5.10.2	Rolling Back a Sessionless Transaction	31-36
31.5.11	Example: Sessionless Transactions with OCI API	31-40
31.5.12	Rules and Guidelines for Using Sessionless Transactions	31-46
31.5.13	Error Messages and Notifications	31-48
31.6	Sessionless Transactions and Oracle Coordinated Distributed Transaction Interoperability	31-50
31.7	Restrictions for Sessionless Transactions	31-50

## 32 Understanding Schema Object Dependency

---

32.1	Overview of Schema Object Dependency	32-1
32.1.1	Example: Displaying Dependent and Referenced Object Types	32-1
32.1.2	Example: Schema Object Change that Invalidates Some Dependents	32-2
32.1.3	Example: View That Depends on Multiple Objects	32-3
32.2	Querying Object Dependencies	32-4
32.3	Object Status	32-4
32.4	Invalidation of Dependent Objects	32-5
32.4.1	Session State and Referenced Packages	32-8
32.4.2	Security Authorization	32-8
32.5	Guidelines for Reducing Invalidation	32-9
32.5.1	Add Items to End of Package	32-9
32.5.2	Reference Each Table Through a View	32-9
32.6	Object Revalidation	32-9
32.6.1	Revalidation of Objects that Compiled with Errors	32-10
32.6.2	Revalidation of Unauthorized Objects	32-10
32.6.3	Revalidation of Invalid SQL Objects	32-10
32.6.4	Revalidation of Invalid PL/SQL Objects	32-10
32.7	Name Resolution in Schema Scope	32-10
32.8	Local Dependency Management	32-12
32.9	Remote Dependency Management	32-12
32.9.1	Dependencies Among Local and Remote Database Procedures	32-12

32.9.2	Dependencies Among Other Remote Objects	32-12
32.9.3	Dependencies of Applications	32-13
32.10	Remote Procedure Call (RPC) Dependency Management	32-13
32.10.1	Time-Stamp Dependency Mode	32-13
32.10.2	RPC-Signature Dependency Mode	32-14
32.10.2.1	Changing Names and Default Values of Parameters	32-16
32.10.2.2	Changing Specification of Parameter Mode IN	32-16
32.10.2.3	Changing Subprogram Body	32-16
32.10.2.4	Changing Data Type Classes of Parameters	32-17
32.10.2.5	Changing Package Types	32-18
32.10.3	Controlling Dependency Mode	32-19
32.10.3.1	Dependency Resolution	32-20
32.10.3.2	Suggestions for Managing Dependencies	32-20
32.11	Shared SQL Dependency Management	32-21

## 33 Using Edition-Based Redefinition

---

33.1	Overview of Edition-Based Redefinition	33-1
33.2	Editions	33-2
33.2.1	Editioned and Noneditioned Objects	33-2
33.2.1.1	Name Resolution for Editioned and Noneditioned Objects	33-3
33.2.1.2	Noneditioned Objects That Can Depend on Editioned Objects	33-4
33.2.1.3	Editionable and Noneditionable Schema Object Types	33-6
33.2.1.4	Enabling Editions for a User	33-7
33.2.1.5	EDITIONABLE and NONEDITIONABLE Properties	33-9
33.2.1.6	Rules for Editioned Objects	33-11
33.2.2	Creating an Edition	33-11
33.2.3	Editioned Objects and Copy-on-Change	33-12
33.2.3.1	Example: Editioned Objects and Copy-on-Change	33-12
33.2.3.2	Example: Dropping an Editioned Object	33-14
33.2.3.3	Example: Creating an Object with the Name of a Dropped Inherited Object	33-15
33.2.4	Making an Edition Available to Some Users	33-17
33.2.5	Making an Edition Available to All Users	33-17
33.2.6	Current Edition and Session Edition	33-17
33.2.6.1	Your Initial Session Edition	33-18
33.2.6.2	Changing Your Session Edition	33-19
33.2.6.3	Displaying the Names of the Current and Session Editions	33-20
33.2.6.4	When the Current Edition Might Differ from the Session Edition	33-20
33.2.7	Retiring an Edition	33-21
33.2.8	Dropping an Edition	33-22
33.3	Editions and Audit Policies	33-24

33.4	Editioning Views	33-24
33.4.1	Creating an Editioning View	33-25
33.4.2	Partition-Extended Editioning View Names	33-26
33.4.3	Changing the Writability of an Editioning View	33-26
33.4.4	Replacing an Editioning View	33-27
33.4.5	Dropped or Renamed Base Tables	33-27
33.4.6	Adding Indexes and Constraints to the Base Table	33-27
33.4.7	SQL Optimizer Index Hints	33-27
33.5	Crossedition Triggers	33-28
33.5.1	Forward Crossedition Triggers	33-28
33.5.2	Reverse Crossedition Triggers	33-29
33.5.3	Crossedition Trigger Interaction with Editions	33-29
33.5.3.1	Which Triggers Are Visible	33-29
33.5.3.2	What Kind of Triggers Can Fire	33-29
33.5.3.3	Firing Order	33-31
33.5.3.4	Crossedition Trigger Execution	33-32
33.5.4	Creating a Crossedition Trigger	33-32
33.5.4.1	Coding the Forward Crossedition Trigger Body	33-33
33.5.5	Transforming Data from Pre- to Post-Upgrade Representation	33-36
33.5.5.1	Preventing Lost Updates	33-37
33.5.6	Dropping the Crossedition Triggers	33-38
33.6	Displaying Information About EBR Features	33-39
33.6.1	Displaying Information About Editions	33-39
33.6.2	Displaying Information About Editioning Views	33-40
33.6.3	Displaying Information About Crossedition Triggers	33-41
33.7	Using EBR to Upgrade an Application	33-41
33.7.1	Preparing Your Application to Use Editioning Views	33-42
33.7.2	Procedure for EBR Using Only Editions	33-43
33.7.3	Procedure for EBR Using Editioning Views	33-45
33.7.4	Procedure for EBR Using Crossedition Triggers	33-46
33.7.5	Rolling Back the Application Upgrade	33-47
33.7.6	Reclaiming Space Occupied by Unused Table Columns	33-47
33.7.7	Example: Using EBR to Upgrade an Application	33-47
33.7.7.1	Existing Application	33-48
33.7.7.2	Preparing the Application to Use Editioning Views	33-49
33.7.7.3	Using EBR to Upgrade the Example Application	33-50

## 34 Using Transaction Guard

---

34.1	Problem That Transaction Guard Solves	34-1
34.2	Solution That Transaction Guard Provides	34-2
34.3	Transaction Guard Concepts and Scope	34-3

34.3.1	Logical Transaction Identifier (LTXID)	34-3
34.3.2	At-Most-Once Execution	34-4
34.3.3	Transaction Guard Coverage	34-4
34.3.4	Transaction Guard with XA Transactions	34-5
34.3.5	Transaction Guard Exclusions	34-6
34.4	Database Configuration for Transaction Guard	34-6
34.4.1	Configuration Checklist	34-7
34.4.2	Transaction History Table	34-7
34.4.3	Service Parameters	34-8
34.4.3.1	Example: Adding and Modifying a Service for a Server Pool	34-9
34.4.3.2	Example: Adding an Administrator-Managed Service	34-9
34.4.3.3	Example: Modifying a Service (PL/SQL)	34-9
34.5	Developing Applications That Use Transaction Guard	34-9
34.5.1	Typical Transaction Guard Usage	34-10
34.5.2	Details for Using the LTXID	34-10
34.5.3	Transaction Guard and Transparent Application Failover	34-11
34.5.4	Using Transaction Guard with ODP.NET	34-12
34.5.5	Connection-Pool LTXID Usage	34-12
34.5.6	Improved Commit Outcome for XA One Phase Optimizations	34-13
34.5.7	Additional Requirements for Transaction Guard Development	34-13
34.6	Transaction Guard and Its Relationship to Application Continuity	34-14
34.7	Transaction Guard Support during DBMS_ROLLING Operations	34-14
34.7.1	Rolling Upgrade Using Transient Logical Standby	34-15
34.7.2	Transaction Guard Support During Major Database Version Upgrades	34-15

## 35 Table DDL Change Notification

35.1	Overview of Table DDL Change Notification	35-1
35.2	Table DDL Change Notification Terminology	35-1
35.3	Benefits of Table DDL Change Notification	35-2
35.4	Features of Table DDL Change Notification	35-2
35.5	Using Table DDL Change Notification	35-3
35.6	Registering for Table DDL Change Notification	35-4
35.6.1	Table-level Registration	35-5
35.6.2	Schema-level Registration	35-5
35.7	Unregistering for Table DDL Change Notifications	35-5
35.8	Supported DDL Events and Commands	35-6
35.9	Monitoring Table DDL Change Notification	35-7

## Index

## List of Tables

---

3-1	Table Annotation Result Cache Modes	3-15
3-2	Effective Result Cache Table Mode	3-16
3-3	Client Configuration Parameters (Optional)	3-21
3-4	Setting Client Result Cache and Server Result Cache	3-24
3-5	Session Purity and Connection Class Defaults	3-37
8-1	COMMIT Statement Options	8-6
8-2	Examples of Concurrency Under Explicit Locking	8-20
8-3	Ways to Display Locking Information	8-28
8-4	ANSI/ISO SQL Isolation Levels and Possible Transaction Interactions	8-30
8-5	ANSI/ISO SQL Isolation Levels Provided by Oracle Database	8-30
8-6	Comparison of READ COMMITTED and SERIALIZABLE Transactions	8-35
8-7	Possible Transaction Outcomes	8-39
8-8	Object Types Supported for CREATE, ALTER, and DROP Commands	8-49
9-1	SQL Character Data Types	9-6
9-2	Range and Precision of Floating-Point Data Types	9-9
9-3	Binary Floating-Point Format Components	9-9
9-4	Summary of Binary Format Storage Parameters	9-9
9-5	Special Values for Native Floating-Point Formats	9-10
9-6	Values Resulting from Exceptions	9-12
9-7	SQL Datetime Data Types	9-13
9-8	SQL Conversion Functions for Datetime Data Types	9-18
9-9	Large Objects (LOBs)	9-20
9-10	Display Types of SQL Functions	9-27
9-11	SQL Data Type Families	9-27
10-1	Identifier Built-in Domains	10-49
10-2	Tech Built-in Domains	10-62
10-3	Numeric Built-in Domains	10-74
10-4	Miscellaneous Built-in Domains	10-79
11-1	Oracle SQL Pattern-Matching Condition and Functions	11-2
11-2	Oracle SQL Pattern-Matching Options for Condition and Functions	11-3
11-3	POSIX Operators in Oracle SQL Regular Expressions	11-5
11-4	POSIX Operators and Multilingual Operator Relationships	11-8
11-5	PERL-Influenced Operators in Oracle SQL Regular Expressions	11-9
11-6	Explanation of the Regular Expression Elements	11-11
11-7	Explanation of the Regular Expression Elements	11-12
15-1	Identifier Types that PL/Scope Collects	15-5

---

15-2	Usages that PL/Scope Reports	15-7
16-1	Raw Profiler Output File Indicators	16-5
16-2	Function Names of Operations that the PL/SQL Hierarchical Profiler Tracks	16-7
16-3	PL/SQL Hierarchical Profiler Database Tables	16-8
16-4	DBMSHP_RUNS Table Columns	16-10
16-5	DBMSHP_FUNCTION_INFO Table Columns	16-10
16-6	DBMSHP_PARENT_CHILD_INFO Table Columns	16-11
17-1	DBMSPPCC_RUNS Table Columns	17-3
17-2	DBMSPPCC_UNITS Table Columns	17-3
17-3	DBMSPPCC_BLOCKS Table Columns	17-3
18-1	Commonly Used Packages in the PL/SQL Web Toolkit	18-3
18-2	Mapping Between mod_plsql and Embedded PL/SQL Gateway DAD Attributes	18-8
18-3	Mapping Between mod_plsql and Embedded PL/SQL Gateway Global Attributes	18-9
18-4	Authentication Possibilities for a DAD	18-12
19-1	Continuous Query Notification Registration Options	19-12
19-2	Attributes of CQ_NOTIFICATION\$_REG_INFO	19-22
19-3	Quality-of-Service Flags	19-24
19-4	Attributes of CQ_NOTIFICATION\$_DESCRIPTOR	19-49
19-5	Attributes of CQ_NOTIFICATION\$_TABLE	19-50
19-6	Attributes of CQ_NOTIFICATION\$_QUERY	19-50
19-7	Attributes of CQ_NOTIFICATION\$_ROW	19-51
20-1	PL/SQL Packages and Their Java and JavaScript Equivalents	20-29
21-1	Parameter Data Type Mappings	21-18
21-2	External Data Type Mappings	21-19
21-3	Properties and Data Types	21-23
22-1	Oracle Flashback Version Query Row Data Pseudocolumns	22-12
22-2	Flashback TRANSACTION_BACKOUT Options	22-20
22-3	Static Data Dictionary Views for Flashback Archive Files	22-29
25-1	General Structure of EPC Encodings	25-2
25-2	Identity Code Package ADTs	25-18
25-3	MGD_ID ADT Subprograms	25-18
25-4	DBMS_MGD_ID_UTL Package Utility Subprograms	25-19
25-5	Definition and Description of the MGD_ID_CATEGORY Metadata View	25-21
25-6	Definition and Description of the USER_MGD_ID_CATEGORY Metadata View	25-21
25-7	Definition and Description of the MGD_ID_SCHEME Metadata View	25-21
25-8	Definition and Description of the USER_MGD_ID_SCHEME Metadata View	25-21
28-1	LRA and Saga Annotations	28-15



28-2	saga_finalization\$ table entries	28-43
29-1	Reservation Table Columns	29-6
30-1	Required XA Features Published by Oracle Database	30-5
30-2	XA Library Subprograms	30-6
30-3	Oracle XA Interface Extensions	30-6
30-4	Required Fields of xa_open string	30-9
30-5	Optional Fields in the xa_open String	30-10
30-6	TX Interface Functions	30-14
30-7	TPM Replacement Statements	30-17
30-8	Sample Trace File Contents	30-21
30-9	Tightly and Loosely Coupled Transaction Branches	30-24
31-1	Errors Summary	31-48
32-1	Database Object Status	32-4
32-2	Operations that Cause Fine-Grained Invalidation	32-6
32-3	Data Type Classes	32-17
33-1	*_ Dictionary Views with Edition Information	33-39
33-2	*_ Dictionary Views with Editioning View Information	33-40
34-1	LTXID Condition or Situation, Application Actions, and Next LTXID to Use	34-11
34-2	Transaction Manager Conditions/ Situations and Actions	34-13

# Preface

*Oracle Database Development Guide* explains topics of interest to experienced developers of databases and database applications. Information in this guide applies to features that work the same on all supported platforms, and does not include system-specific information.

## Preface Topics:

## Audience

This guide is intended primarily for application developers who are either developing applications or converting applications to run in the Oracle Database environment.

This guide might also help anyone interested in database or database application development, such as systems analysts and project managers.

This guide assumes that you are familiar with the concepts and techniques relevant to your job. To use this guide most effectively, you also need a working knowledge of:

- Structured Query Language (SQL)
- Object-oriented programming

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

## Related Documents

For more information, see these documents in the Oracle Database documentation set:

- *Oracle Database PL/SQL Language Reference*
- *Oracle Call Interface Programmer's Guide*
- *Oracle Database JSON Developer's Guide*
- *Oracle Database SODA for PL/SQL Developer's Guide*
- *Oracle Database Security Guide*
- *Pro\*C/C++ Programmer's Guide*
- *Oracle Database SQL Language Reference*

- *Oracle Database Administrator's Guide*
- *Oracle Database Concepts*
- *Oracle Database Java Developer's Guide*
- *Oracle Database JDBC Developer's Guide*
- *Oracle XML Developer's Kit Programmer's Guide*
- *Oracle XML DB Developer's Guide*
- *Oracle Database Migration Guide*
- *Oracle Database Globalization Support Guide*
- *Oracle Database Sample Schemas*

**See Also:**

- <https://www.oracle.com/database/technologies/application-development.html>

## Conventions

This guide uses these text conventions:

Convention	Meaning
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Also:

- `*_view` means all static data dictionary views whose names end with `view`. For example, `*_ERRORS` means `ALL_ERRORS`, `DBA_ERRORS`, and `USER_ERRORS`. For more information about any static data dictionary view, or about static dictionary views in general, see *Oracle Database Reference*.
- Table names not qualified with schema names are in the sample schema `HR`. For information about the sample schemas, see *Oracle Database Sample Schemas*.