

Oracle® XML DB

Developer's Guide



23ai
F46732-05
July 2024

ORACLE®

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	xlili
Documentation Accessibility	xlili
Diversity and Inclusion	xliv
Related Documents	xliv
Conventions	xlvi
Code Examples	xlvi
Syntax Descriptions	xlvi

Part I Oracle XML DB Basics

1 Introduction to Oracle XML DB

Overview of Oracle XML DB	1-1
Oracle XML DB Benefits	1-2
Data and Content Unified	1-4
Database Capabilities for Working with XML	1-5
Advantages of Storing Data as XML in the Database	1-6
Data Duality: XML and Relational	1-7
Use XMLType Views If Your Data Is Not XML	1-8
Efficient Storage and Retrieval of Complex XML Documents	1-8
Oracle XML DB Architecture	1-9
Oracle XML DB Features	1-10
XMLType Data Type	1-11
XMLType Storage Models	1-12
XML Schema Support in Oracle XML DB	1-13
DTD Support in Oracle XML DB	1-15
Static Data Dictionary Views Related to XML	1-16
SQL/XML Standard Functions	1-16
Programmatic Access to Oracle XML DB (Java, PL/SQL, and C)	1-17
Oracle XML DB Repository: Overview	1-17
Standards Supported by Oracle XML DB	1-19
Oracle XML DB Technical Support	1-20

Oracle XML DB Examples	1-20
Oracle XML DB Case Studies and Demonstrations on OTN	1-21

2 Getting Started with Oracle XML DB

Oracle XML DB Installation	2-1
Oracle XML DB Use Cases	2-1
Application Design Considerations for Oracle XML DB	2-2
XML Data Storage	2-3
The Structure of Your XML Data	2-4
Languages Used to Implement Your Application	2-5
XML Processing Options	2-5
Oracle XML DB Repository Access	2-6
Oracle XML DB Cooperates with Other Database Options and Features	2-6

3 Overview of How To Use Oracle XML DB

Creating XMLType Tables and Columns	3-2
Creating XMLType Columns in Shared and Duplicated Tables	3-3
Creating Virtual Columns on XMLType Data Stored as Binary XML	3-4
Partitioning Tables That Contain XMLType Data Stored as Binary XML	3-5
Enforcing XML Data Integrity Using the Database	3-7
Enforcing Referential Integrity Using SQL Constraints	3-8
Loading XML Content into Oracle XML DB	3-12
Loading XML Content Using SQL or PL/SQL	3-12
Loading XML Content Using Java	3-13
Loading XML Content Using C	3-14
Loading Large XML Files that Contain Small XML Documents	3-15
Loading Large XML Files Using SQL*Loader	3-16
Loading XML Documents into the Repository Using DBMS_XDB_REPOS	3-16
Loading Documents into the Repository Using Protocols	3-17
Querying XML Content Stored in Oracle XML DB	3-17
PurchaseOrder XML Document Used in Examples	3-18
Retrieving the Content of an XML Document Using Pseudocolumn OBJECT_VALUE	3-19
Accessing Fragments or Nodes of an XML Document Using XMLQUERY	3-20
Accessing Text Nodes and Attribute Values Using XMLCAST and XMLQUERY	3-21
Searching an XML Document Using XMLEXISTS, XMLCAST, and XMLQUERY	3-22
Performing SQL Operations on XMLType Fragments Using XMLTABLE	3-25
Updating XML Content Stored in Oracle XML DB	3-28
Generating XML Data from Relational Data	3-31
Generating XML Data from Relational Data Using SQL/XML Functions	3-31
Generating XML Data from Relational Data Using DBURITYPE	3-36

Character Sets of XML Documents	3-38
XML Encoding Declaration	3-38
Character-Set Determination When Loading XML Documents into the Database	3-39
Character-Set Determination When Retrieving XML Documents from the Database	3-40
Migrating XMLType Data to Transportable Binary XML (TBX)	3-41

Part II Manipulation of XML Data in Oracle XML DB

4 XQuery and Oracle XML DB

Overview of the XQuery Language	4-1
XPath Expressions Are XQuery Expressions	4-2
XQuery: A Functional Language Based on Sequences	4-3
XQuery Is About Sequences	4-4
XQuery Is Referentially Transparent	4-4
XQuery Update Has Side Effects on Your Data	4-5
XQuery Update Snapshots	4-5
XQuery Full Text Provides Full-Text Search	4-5
XQuery Expressions	4-6
FLWOR Expressions	4-8
Overview of XQuery in Oracle XML DB	4-9
When To Use XQuery	4-10
Predefined XQuery Namespaces and Prefixes	4-10
SQL/XML Functions XMLQUERY, XMLTABLE, XMLEXISTS, and XMLCAST	4-11
XMLQUERY SQL/XML Function in Oracle XML DB	4-12
XMLTABLE SQL/XML Function in Oracle XML DB	4-13
Chaining Calls to SQL/XML Function XMLTABLE	4-16
XMLEXISTS SQL/XML Function in Oracle XML DB	4-17
Using XMLEXISTS to Find a Node	4-18
XMLCAST SQL/XML Function in Oracle XML DB	4-19
Using XMLCAST to Extract the Scalar Value of an XML Fragment	4-20
URI Scheme oradb: Querying Table or View Data with XQuery	4-21
Oracle XQuery Extension Functions	4-22
ora:sqrt XQuery Function	4-23
ora:tokenize XQuery Function	4-23
Oracle XQuery Extension-Expression Pragmas	4-23
XQuery Static Type-Checking in Oracle XML DB	4-26
Oracle XML DB Support for XQuery	4-27
Support for XQuery and SQL	4-28
Implementation Choices Specified in the XQuery Standards	4-28
XQuery Features Not Supported by Oracle XML DB	4-29

XQuery Optional Features	4-29
Support for XQuery Functions and Operators	4-29
XQuery Functions fn:doc, fn:collection, and fn:doc-available	4-30
Support for XQuery Full Text	4-30
XQuery Full Text, XML Schema-Based Data, and Pragma ora:no_schema	4-31
Restrictions on Using XQuery Full Text with XMLExists	4-31
Supported XQuery Full Text FTSelection Operators	4-32
Supported XQuery Full Text Match Options	4-33
Unsupported XQuery Full Text Features	4-33
XQuery Full Text Errors	4-33

5 Query and Update of XML Data

Using XQuery with Oracle XML DB	5-1
XQuery Sequences Can Contain Data of Any XQuery Type	5-3
Querying XML Data in Oracle XML DB Repository Using XQuery	5-3
Querying Relational Data Using XQuery and URI Scheme oradb	5-5
Querying XMLType Data Using XQuery	5-10
Using Namespaces with XQuery	5-17
Querying XML Data Using SQL and PL/SQL	5-19
Using the SQL*Plus XQUERY Command	5-24
Using XQuery with PL/SQL, JDBC, and ODP.NET to Access Database Data	5-24
Updating XML Data	5-28
Updating an Entire XML Document	5-28
Replacing XML Nodes	5-29
Updating XML Data to NULL Values	5-35
Inserting Child XML Nodes	5-37
Deleting XML Nodes	5-40
Creating XML Views of Modified XML Data	5-41
Performance Tuning for XQuery	5-42
Rule-Based and Cost-Based XQuery Optimization	5-43
XQuery Optimization over Relational Data	5-44
XQuery Optimization over XML Schema-Based XMLType Data	5-45
Diagnosis of XQuery Optimization: XMLOptimizationCheck	5-47
Performance Improvement for fn:doc and fn:collection on Repository Data	5-48
Use EQUALS_PATH and UNDER_PATH Instead of fn:doc and fn:collection	5-49
Using Oracle XQuery Pragma ora:defaultTable	5-50

6 Indexes for XMLType Data

Oracle XML DB Tasks Involving Indexes	6-1
Overview of Indexing XMLType Data	6-2

XMLIndex	6-3
Advantages of XMLIndex	6-4
XMLIndex Components	6-5
XMLIndex Structured Component	6-6
Ignore the Index Content Tables; They Are Transparent	6-8
Data Type Considerations for XMLIndex Structured Component	6-8
Exchange Partitioning and XMLIndex	6-10
Creating, Dropping, Altering, and Examining an XMLIndex Index	6-12
Use of XMLIndex with a Structured Component	6-14
Using Namespaces and Storage Clauses with an XMLIndex Structured Component	6-15
Adding a Structured Component to an XMLIndex Index	6-16
Using Non-Blocking ALTER INDEX with an XMLIndex Structured Component	6-17
Modifying the Data Type of a Structured XMLIndex Component	6-19
Dropping an XMLIndex Structured Component	6-20
Indexing the Relational Tables of a Structured XMLIndex Component	6-20
How to Tell Whether XMLIndex is Used	6-20
Turning Off Use of XMLIndex	6-22
Guidelines for Using XMLIndex with a Structured Component	6-23
XMLIndex Partitioning and Parallelism	6-23
Collecting Statistics on XMLIndex Objects for the Cost-Based Optimizer	6-25
Data Dictionary Static Public Views Related to XMLIndex	6-25
PARAMETERS Clause for CREATE INDEX and ALTER INDEX	6-27
Using a Registered PARAMETERS Clause for XMLIndex	6-28
PARAMETERS Clause Syntax for CREATE INDEX and ALTER INDEX	6-28
Usage of XMLIndex_parameters_clause	6-35
Usage of XMLIndex_parameters	6-35
Usage of groups_clause and alter_index_group_clause	6-36
Usage of XMLIndex_xmltable_clause	6-36
Usage of column_clause	6-36
XML Search Index: Indexing for Full Text Search and Ad-hoc Queries	6-36
Creating and Using an XML Search Index	6-37
Maintenance of XML Search Indexes	6-39
Preference Defaults and Recommendations	6-39
Queries using an XML Search Index	6-41
Migrating to Use XML Search Index	6-44
Indexing XMLType Data Stored Object-Relationally	6-46
Indexing Non-Repeating Text Nodes or Attribute Values	6-46
Indexing Repeating (Collection) Elements	6-47

7 Transformation and Validation of XMLType Data

XSL Transformation and Oracle XML DB	7-1
--------------------------------------	-----

SQL Function XMLTRANSFORM and XMLType Method TRANSFORM()	7-3
XMLTRANSFORM and XMLType.transform(): Examples	7-4
XSL Transformation Using DBUri Servlet	7-9
Validation of XMLType Instances	7-11
Partial and Full XML Schema Validation	7-12
Partial Validation	7-13
Full Validation	7-13
Validating XML Data Stored as XMLType: Examples	7-15

Part III Relational Data To and From XML Data

8 Generation of XML Data from Relational Data

Overview of Generating XML Data	8-1
Generation of XML Data Using SQL Functions	8-2
XMLELEMENT and XMLATTRIBUTES SQL/XML Functions	8-3
Escape of Characters in Generated XML Data	8-6
Formatting of XML Dates and Timestamps	8-7
XMLElement Examples	8-7
XMLFOREST SQL/XML Function	8-11
XMLCONCAT SQL/XML Function	8-12
XMLAGG SQL/XML Function	8-13
XMLPI SQL/XML Function	8-16
XMLCOMMENT SQL/XML Function	8-17
XMLSERIALIZE SQL/XML Function	8-18
XMLPARSE SQL/XML Function	8-20
XMLCOLATTVAL Oracle SQL Function	8-21
XMLCDATA Oracle SQL Function	8-22
Generation of XML Data Using DBMS_XMLGEN	8-23
Using PL/SQL Package DBMS_XMLGEN	8-24
Functions and Procedures of Package DBMS_XMLGEN	8-26
DBMS_XMLGEN Examples	8-31
SYS_XMLAGG Oracle SQL Function	8-48
Ordering Query Results Before Aggregating, Using XMLAGG ORDER BY Clause	8-49
Returning a Rowset Using XMLTABLE	8-50

9 Relational Views over XML Data

Introduction to Creating and Using Relational Views over XML Data	9-1
Creating a Relational View over XML: One Row for Each XML Document	9-2
Creating a Relational View over XML: Mapping XML Nodes to Columns	9-3
Indexing Binary XML Data Exposed Using a Relational View	9-4

10 XMLType Views

What Are XMLType Views?	10-1
CREATE VIEW for XMLType Views: Syntax	10-2
Creating Non-Schema-Based XMLType Views	10-3
Creating XML Schema-Based XMLType Views	10-3
Creating XML Schema-Based XMLType Views Using SQL/XML Publishing Functions	10-4
Using Namespaces with SQL/XML Publishing Functions	10-7
Creating XML Schema-Based XMLType Views Using Object Types or Object Views	10-11
Creating XMLType Employee View, with Nested Department Information	10-12
Creating XMLType Department View, with Nested Employee Information	10-15
Creating XMLType Views from XMLType Tables	10-18
Referencing XMLType View Objects Using SQL Function REF	10-19
Using DML (Data Manipulation Language) on XMLType Views	10-19

Part IV XMLType APIs

11 PL/SQL APIs for XMLType

Overview of PL/SQL APIs for XMLType	11-1
PL/SQL APIs for XMLType: Features	11-2
Lazy Load of XML Data (Lazy Manifestation)	11-2
XMLType Data Type Supports XML Schema	11-2
XMLType Supports Data in Different Character Sets	11-2
PL/SQL APIs for XMLType: References	11-3
PL/SQL DOM API for XMLType (DBMS_XMLDOM)	11-5
Overview of the W3C Document Object Model (DOM) Recommendation	11-6
Oracle XML Developer's Kit Extensions to the W3C DOM Standard	11-7
Supported W3C DOM Recommendations	11-7
Difference Between DOM and SAX	11-8
PL/SQL DOM API for XMLType (DBMS_XMLDOM): Features	11-8
PL/SQL DOM API Support for XML Schema	11-9
Enhanced DOM Performance	11-9
Application Design Using Oracle XML Developer's Kit and Oracle XML DB	11-9
Preparing XML Data to Use the PL/SQL DOM API for XMLType	11-10
XML Schema Types Are Mapped to SQL Object Types	11-11
DOM Fidelity for XML Schema Mapping	11-12
Wrap Existing Data as XML with XMLType Views	11-12
DBMS_XMLDOM Methods Supported by Oracle XML DB	11-12
PL/SQL DOM API for XMLType: Node Types	11-13

PL/SQL Function NEWDOMDOCUMENT and DOMDOCUMENT Nodes	11-14
DOM NodeList and NamedNodeMap Objects	11-15
Overview of Using the PL/SQL DOM API for XMLType (DBMS_XMLDOM)	11-15
PL/SQL DOM API for XMLType – Examples	11-16
Large Node Handling Using DBMS_XMLDOM	11-18
Get-Push Model for Large Node Handling	11-20
Get-Pull Model for Large Node Handling	11-22
Set-Pull Model for Large Node Handling	11-23
Set-Push Model for Large Node Handling	11-24
Determining Binary Stream or Character Stream for Large Node Handling	11-25
PL/SQL Parser API for XMLType (DBMS_XMLPARSER)	11-26
PL/SQL XSLT Processor for XMLType (DBMS_XSLPROCESSOR)	11-27
PL/SQL XSLT Processor for XMLType: Features	11-28
Using the PL/SQL XSLT Processor API for XMLType (DBMS_XSLPROCESSOR)	11-29

12 PL/SQL Package DBMS_XMLSTORE

Using Package DBMS_XMLSTORE	12-1
Inserting an XML Document Using DBMS_XMLSTORE	12-2
Updating XML Data Using DBMS_XMLSTORE	12-4
Deleting XML Data Using DBMS_XMLSTORE	12-5

13 Java DOM API for XMLType

Overview of Java DOM API for XMLType	13-2
Access to XMLType Data Using JDBC	13-2
Using JDBC to Access XML Documents in Oracle XML DB	13-3
Manipulating XML Database Documents Using JDBC	13-5
Loading a Large XML Document into the Database Using JDBC	13-9
MS Windows Java Security Manager Permissions for Java DOM API with a Thick Connection	13-11
Creating XML Schema-Based Documents	13-11
XMLType Instance Representation in Java (JDBC or SQLJ)	13-12
Classes of Java DOM API for XMLType	13-12
Using the Java DOM API for XMLType	13-14
Large XML Node Handling with Java	13-14
Stream Extensions to Java DOM	13-15
Get-Pull Model	13-16
Get-Push Model	13-17
Set-Pull Model	13-17
Set-Push Model	13-18

14 C DOM API for XMLType

Overview of the C DOM API for XMLType	14-1
Access to XMLType Data Stored in the Database Using OCI	14-2
Creating XMLType Instances on the Client	14-3
XML Context Parameter for C DOM API Functions	14-3
OCIXmlDbInitXmlCtx() Syntax	14-3
OCIXmlDbFreeXmlCtx() Syntax	14-4
Initializing and Terminating an XML Context	14-4
Using the C API for XML with Binary XML	14-8
Using the Oracle XML Developer's Kit Pull Parser with Oracle XML DB	14-11
Common XMLType Operations in C	14-16

15 Oracle XML DB and Oracle Data Provider for .NET

Oracle XML DB and ODP.NET XML	15-1
Using XMLType Data with ODP.NET	15-1

Part V XML Schema and Object-Relational XMLType

16 Choice of XMLType Storage and Indexing

Introduction to Choosing an XMLType Storage Model and Indexing Approaches	16-1
XMLType Use Case Spectrum: Data-Centric to Document-Centric	16-3
Common Use Cases for XML Data Stored as XMLType	16-4
XMLType Use Case: No XML Fragment Updating or Querying	16-5
XMLType Use Case: Data Integration from Diverse Sources with Different XML Schemas	16-5
XMLType Use Case: Staged XML Data for ETL	16-6
XMLType Use Case: Semi-Structured XML Data	16-6
XMLType Use Case: Business Intelligence Queries	16-7
XMLType Use Case: XML Queries Involving Full-Text Search	16-7
XMLType Storage Model Considerations	16-8
XMLType Indexing Considerations	16-9
XMLType Storage Options: Relative Advantages	16-9

17 XML Schema Storage and Query: Basic

Overview of XML Schema	17-2
XML Schema for Schemas	17-3
XML Schema Features	17-3

XML Instance Documents	17-3
XML Namespaces and XML Schemas	17-3
Overview of Editing XML Schemas	17-3
Overview of Using XML Schema with Oracle XML DB	17-4
Why Use XML Schema with Oracle XML DB?	17-5
Overview of Annotating an XML Schema to Control Naming, Mapping, and Storage	17-6
DOM Fidelity	17-7
XMLType Methods Related to XML Schema	17-8
XML Schema Registration with Oracle XML DB	17-8
XML Schema Registration Actions	17-10
Registering an XML Schema with Oracle XML DB	17-10
SQL Types and Tables Created During XML Schema Registration	17-12
Default Tables for Global Elements	17-13
Database Objects That Depend on Registered XML Schemas	17-13
Local and Global XML Schemas	17-14
Local XML Schema	17-14
Global XML Schema	17-15
Fully Qualified XML Schema URLs	17-16
Deletion of an XML Schema	17-17
Listing All Registered XML Schemas	17-18
Creation of XMLType Tables and Columns Based on XML Schemas	17-19
Specification of XMLType Storage Options for XML Schema-Based Data	17-22
Binary XML Storage of XML Schema-Based Data	17-22
Object-Relational Storage of XML Schema-Based Data	17-25
Ways to Identify XML Schema Instance Documents	17-27
Attributes noNamespaceSchemaLocation and schemaLocation	17-28
XML Schema and Multiple Namespaces	17-28
XML Schema Data Types Are Mapped to Oracle XML DB Storage	17-29

18 XML Schema Storage and Query: Object-Relational Storage

Object-Relational Storage of XML Documents	18-2
How Collections Are Stored for Object-Relational XMLType Storage	18-3
SQL Types Created during XML Schema Registration for Object-Relational Storage	18-4
Default Tables Created during XML Schema Registration	18-5
Do Not Use Internal Constructs Generated during XML Schema Registration	18-6
Generated Names are Case Sensitive	18-6
SYS_XDBPD\$ and DOM Fidelity for Object-Relational Storage	18-6
Oracle XML Schema Annotations	18-7
Common Uses of XML Schema Annotations	18-8
XML Schema Annotation Example	18-9
Annotating an XML Schema Using DBMS_XMLSCHEMA_ANNOTATE	18-13

Available Oracle XML DB XML Schema Annotations	18-14
XML Schema Annotation Guidelines for Object-Relational Storage	18-16
Avoid Creation of Unnecessary Tables for Unused Top-Level Elements	18-17
Provide Your Own Names for Default Tables	18-18
Turn Off DOM Fidelity If Not Needed	18-18
Annotate Time-Related Elements with a Timestamp Data Type	18-19
Add Table and Column Properties	18-19
Store Large Collections Out of Line	18-19
Querying a Registered XML Schema to Obtain Annotations	18-20
You Can Apply Annotations from One XML Schema to Another	18-21
Use DBMS_XMLSCHEMA to Map XML Schema Data Types to SQL Data Types	18-21
Example of Mapping XML Schema Data Types to SQL	18-22
XML Schema Attribute Data Types Mapped to SQL	18-23
You Can Override the SQLType Value in an XML Schema When Declaring Attributes	18-24
XML Schema Element Data Types Mapped to SQL	18-24
Override of the SQLType Value in an XML Schema When Declaring Elements	18-25
How XML Schema simpleType Is Mapped to SQL	18-25
NCHAR, NVARCHAR2, and NCLOB SQLType Values Are Not Supported for SQLType	18-28
simpleType: How XML Strings Are Mapped to SQL VARCHAR2 Versus CLOB	18-29
How XML Schema Time Zones Are Mapped to SQL	18-29
How XML Schema complexType Is Mapped to SQL	18-30
Attribute Specification in a complexType XML Schema Declaration	18-30
complexType Extensions and Restrictions in Oracle XML DB	18-31
complexType Declarations in XML Schema: Handling Inheritance	18-31
How a complexType Based on simpleContent Is Mapped to an Object Type	18-34
How any and anyAttribute Declarations Are Mapped to Object Type Attributes	18-34
Creating XML Schema-Based XMLType Columns and Tables	18-35
Overview of Partitioning XMLType Tables and Columns Stored Object-Relationally	18-37
Examples of Partitioning XMLType Data Stored Object-Relationally	18-38
Partition Maintenance for XMLType Data Stored Object-Relationally	18-39
Specification of Relational Constraints on XMLType Tables and Columns	18-40
Adding Unique Constraints to the Parent Element of an Attribute	18-42
Out-Of-Line Storage of XMLType Data	18-43
Setting Annotation Attribute xdb:SQLInline to false for Out-Of-Line Storage	18-44
Storing Collections in Out-Of-Line Tables	18-47
Considerations for Working with Complex or Large XML Schemas	18-49
Circular and Cyclical Dependencies Among XML Schemas	18-50
For Circular XML Schema Dependencies Set Parameter GENTABLES to TRUE	18-51
complexType Declarations in XML Schema: Handling Cycles	18-51
Cyclical References Among XML Schemas	18-54
Support for Recursive Schemas	18-56

defaultTable Shared Among Common Out-Of-Line Elements	18-57
Query Rewrite when DOCID is Present	18-59
DOCID Column Creation Disabling	18-60
XML Fragments Can Be Mapped to Large Objects (LOBs)	18-60
ORA-01792 and ORA-04031: Issues with Large XML Schemas	18-61
Considerations for Loading and Retrieving Large Documents with Collections	18-62
Guidelines for Configuration Parameters xdbcore-loadableunit-size and xdbcore-xobmem-bound	18-64
Debugging XML Schema Registration for XML Data Stored Object-Relationally	18-64

19 XPath Rewrite for Object-Relational Storage

Overview of XPath Rewrite for Object-Relational Storage	19-1
Common XPath Expressions that Are Rewritten	19-3
XPath Rewrite for Out-Of-Line Tables	19-4
Guidelines for Using Execution Plans to Analyze and Optimize XPath Queries	19-5
Guideline: Look for underlying tables versus XML functions in execution plans	19-6
Guideline: Name the object-relational tables, so you recognize them in execution plans	19-6
Guideline: Create an index on a column targeted by a predicate	19-7
Guideline: Create indexes on ordered collection tables	19-9
Guideline: Use XMLOptimizationCheck to determine why a query is not rewritten	19-10

20 XML Schema Evolution

Overview of XML Schema Evolution	20-1
Copy-Based Schema Evolution	20-2
Scenario for Copy-Based Evolution	20-3
COPYEVOLVE Parameters and Errors	20-6
Limitations of Procedure COPYEVOLVE	20-8
Guidelines for Using Procedure COPYEVOLVE	20-8
Top-Level Element Name Changes	20-9
User-Created Virtual Columns of Tables Other Than Default Tables	20-9
Ensure That the XML Schema and Dependents Are Not Used by Concurrent Sessions	20-10
Rollback When Procedure DBMS_XMLSCHEMA.COPYEVOLVE Raises an Error	20-10
Failed Rollback From Insufficient Privileges	20-10
Privileges Needed for XML Schema Evolution	20-10
Update of Existing XML Instance Documents Using an XSLT Stylesheet	20-11
Examples of Using Procedure COPYEVOLVE	20-13
In-Place XML Schema Evolution	20-16
Restrictions for In-Place XML Schema Evolution	20-17
Backward-Compatibility Restrictions	20-17
Other Restrictions on In-Place Evolution	20-19

Supported Operations for In-Place XML Schema Evolution	20-19
Guidelines for Using In-Place XML Schema Evolution	20-21
inPlaceEvolve Parameters	20-22
The diffXML Parameter Document	20-23
diffXML Operations and Examples	20-24

Part VI Oracle XML DB Repository

21 Access to Oracle XML DB Repository Data

Overview of Oracle XML DB Repository	21-2
Oracle XML DB Provides Name-Level Locking	21-4
Two Ways to Access Oracle XML DB Repository Resources	21-5
Database Schema (User Account) XDB and Oracle XML DB Repository	21-6
Repository Terminology and Supplied Resources	21-6
Repository Terminology	21-7
Predefined Repository Files and Folders	21-8
Oracle XML DB Repository Resources	21-9
Where Is Repository Data Stored?	21-9
Names of Generated Tables	21-10
How Object-Relational Storage Is Defined for Repository Resources	21-10
Oracle ASM Virtual Folder	21-10
How Documents are Stored in Oracle XML DB Repository	21-11
Repository Data Access Control	21-12
Repository Path-Name Resolution	21-12
Link Types	21-13
Repository Links and Document Links	21-13
Hard Links and Weak Links	21-14
Creating a Weak Link with No Knowledge of Folder Hierarchy	21-15
How and When to Prevent Multiple Hard Links	21-16
Navigational or Path Access to Repository Resources	21-16
Access to Oracle XML DB Resources Using Internet Protocols	21-19
Where You Can Use Oracle XML DB Protocol Access	21-19
Overview of Protocol Access to Oracle XML DB	21-19
Retrieval of Oracle XML DB Resources	21-20
Storage of Oracle XML DB Resources	21-20
Internet Protocols and XMLType: XMLType Direct Stream Write	21-20
Access to Oracle ASM Files Using Protocols and Resource APIs – For DBAs	21-21
Query-Based Access to Repository Resources	21-23
Servlet Access to Repository Resources	21-24
Operations on Repository Resources	21-24

Accessing the Content of Repository Resources Using SQL	21-31
Access to the Content of XML Schema-Based Documents	21-32
Accessing Resource Content Using Element XMLRef in Joins	21-32
Update of the Content of Repository Documents	21-33
Update of Repository Content Using Internet Protocols	21-34
Update of Repository Content Using SQL	21-35
Updating a Document in the Repository by Updating Its Resource Document	21-35
Updating an XML Schema-Based Document in the Repository by Updating the Default Table	21-37
Querying Resources in RESOURCE_VIEW and PATH_VIEW	21-38
Oracle XML DB Hierarchical Repository Index	21-41

22 Configuration of Oracle XML DB Repository

Resource Configuration Files	22-2
Configuring a Resource	22-3
Common Configuration Parameters	22-4
Configuration Element ResConfig	22-4
Configuration Elements defaultChildConfig and configuration	22-5
Configuration Element applicationData	22-6

23 Use of XLink and XInclude with Oracle XML DB

Overview of XLink and XInclude	23-2
Link Types for XLink and XInclude	23-3
XLink and XInclude Links Model Document Relationships	23-3
XLink Link Types and XInclude Link Types	23-3
XInclude: Compound Documents	23-4
Oracle XML DB Support for XLink	23-5
Oracle XML DB Support for XInclude	23-6
Expanding Compound-Document Inclusions	23-7
Validation of Compound Documents	23-8
Update of a Compound Document	23-8
Compound Document Versioning, Locking, and Access Control	23-9
Use View DOCUMENT_LINKS to Examine XLink and XInclude Links	23-9
Querying DOCUMENT_LINKS for XLink Information	23-10
Querying DOCUMENT_LINKS for XInclude Information	23-11
Configuration of Repository Resources for XLink and XInclude	23-12
Configure the Treatment of Unresolved Links: Attribute UnresolvedLink	23-13
Configure the Type of Document Links to Create: Element LinkType	23-14
Configure the Path Format for Retrieval: Element PathFormat	23-14
Configure Conflict-Resolution for XInclude: Element ConflictRule	23-15

Configure the Decomposition of Documents Using XInclude: Element SectionConfig	23-15
XLink and XInclude Configuration Examples	23-16
Manage XLink and XInclude Links Using DBMS_XDB_REPOS.processLinks	23-18

24 Repository Access Using RESOURCE_VIEW and PATH_VIEW

Overview of Oracle XML DB RESOURCE_VIEW and PATH_VIEW	24-2
RESOURCE_VIEW Definition and Structure	24-4
PATH_VIEW Definition and Structure	24-5
The Difference Between RESOURCE_VIEW and PATH_VIEW	24-5
Operations You Can Perform Using UNDER_PATH and EQUALS_PATH	24-6
Oracle SQL Functions That Use RESOURCE_VIEW and PATH_VIEW	24-6
UNDER_PATH SQL Function	24-7
EQUALS_PATH SQL Function	24-8
PATH SQL Function	24-9
DEPTH SQL Function	24-9
Accessing Repository Data Paths, Resources and Links: Examples	24-10
Deleting Repository Resources: Examples	24-17
Deleting Nonempty Folder Resources	24-17
Updating Repository Resources: Examples	24-18
Working with Multiple Oracle XML DB Resources	24-21
Performance Guidelines for Oracle XML DB Repository Operations	24-22
Searching for Resources Using Oracle Text	24-23

25 Resource Versions

Overview of Oracle XML DB Repository Resource Versioning	25-1
Overview of PL/SQL Package DBMS_XDB_VERSION	25-3
Resource Versions and Resource IDs	25-4
Resource Versions and ACLs	25-5
Resource Versioning Examples	25-6

26 PL/SQL Access to Oracle XML DB Repository

DBMS_XDB_REPOS: Access and Manage Repository Resources	26-1
DBMS_XDB_REPOS: ACL-Based Security Management	26-3
DBMS_XDB_CONFIG: Configuration Management	26-7

27 Repository Access Control

Access Control Concepts	27-2
Authentication and Authorization	27-3

Principal: A User or Role	27-3
Database Roles and ACLs Map Privileges to Users	27-4
Principal DAV::owner	27-4
Privilege: A Permission	27-4
Access Control Entry (ACE)	27-5
Access Control List (ACL)	27-6
Database Privileges for Repository Operations	27-6
Privileges	27-7
Atomic Privileges	27-7
Aggregate Privileges	27-8
ACLs and ACEs	27-9
System ACLs	27-10
ACL and ACE Evaluation	27-11
ACL Validation	27-11
Element invert: Complement the Principals in an ACE	27-11
Overview of Working with Access Control Lists (ACLs)	27-12
Creating an ACL Using DBMS_XDB_REPOS.CREATERESOURCE	27-13
Retrieving an ACL Document, Given its Repository Path	27-14
Setting the ACL of a Resource	27-14
Deleting an ACL	27-15
Updating an ACL	27-15
Retrieving the ACL Document that Protects a Given Resource	27-17
Retrieving Privileges Granted to the Current User for a Particular Resource	27-17
Checking Whether the Current User Has Privileges on a Resource	27-18
Checking Whether a User Has Privileges Using the ACL and Resource Owner	27-19
Retrieving the Path of the ACL that Protects a Given Resource	27-19
Retrieving the Paths of All Resources Protected by a Given ACL	27-20
ACL Caching	27-21
Repository Resources and Database Table Security	27-21
Optimization: Do not enforce ACL-based security if you do not need it	27-22
Integration Of Oracle XML DB with LDAP	27-23

28 Repository Access Using Protocols

Overview of Oracle XML DB Protocol Server	28-1
Session Pooling	28-2
Oracle XML DB Protocol Server Configuration Management	28-3
Protocol Server Configuration Parameters	28-4
Configuring Secure HTTP (HTTPS)	28-8
Enabling the HTTP Listener to Use SSL	28-9
Enabling TCPS Dispatcher	28-10
Using Listener Status to Check Port Configuration	28-10

Configuring Protocol Port Parameters after Database Consolidation	28-11
Configuration and Management of Authentication Mechanisms for HTTP	28-11
Nonces for Digest Authentication	28-12
Oracle XML DB Repository and File-System Resources	28-13
Protocol Server Handles XML Schema-Based or Non-Schema-Based XML Documents	28-13
Event-Based Logging	28-13
Auditing of HTTP and FTP Protocols	28-13
FTP and the Oracle XML DB Protocol Server	28-14
Oracle XML DB Protocol Server: FTP Features	28-14
FTP Features That Are Not Supported	28-15
Supported FTP Client Methods	28-16
FTP Quote Methods	28-16
Uploading Content to Oracle XML DB Repository Using FTP	28-17
Using FTP with Oracle ASM Files	28-19
Using FTP on the Standard Port Instead of the Oracle XML DB Default Port	28-21
Using IPv6 IP Addresses with FTP	28-22
FTP Server Session Management	28-22
Handling Error 421. Modifying the Default Timeout Value of an FTP Session	28-22
FTP Client Failure in Passive Mode	28-23
HTTP(S) and Oracle XML DB Protocol Server	28-23
Oracle XML DB Protocol Server: HTTP(S) Features	28-23
Supported HTTP(S) Client Methods	28-25
Using HTTP(S) on a Standard Port Instead of an Oracle XML DB Default Port	28-25
Use of IPv6 IP Addresses with HTTP(S)	28-26
HTTPS: Support for Secure HTTP	28-26
Control of URL Expiration Time	28-27
Anonymous Access to Oracle XML DB Repository Using HTTP	28-27
Use of Java Servlets with HTTP(S)	28-28
Embedded PL/SQL Gateway	28-28
Transmission of Multibyte Data From a Client	28-29
Characters That Are Not ASCII in URLs	28-30
Character Sets for HTTP(S)	28-30
WebDAV and Oracle XML DB	28-31
Oracle XML DB WebDAV Features	28-31
WebDAV Features That Are Not Supported by Oracle XML DB	28-32
WebDAV Client Methods Supported by Oracle XML DB	28-32
WebDAV and Microsoft Windows	28-33
Creating a WebFolder in Microsoft Windows For Use With Oracle XML DB Repository	28-34
Use of WebDAV with Windows Explorer to Copy Files into Oracle XML DB Repository	28-34

29 User-Defined Repository Metadata

Overview of Metadata and XML	29-1
Kinds of Metadata – Uses of the Term	29-2
User-Defined Resource Metadata	29-3
Scenario: Metadata for a Photo Collection	29-3
Using XML Schemas to Define Resource Metadata	29-4
Addition, Modification, and Deletion of Resource Metadata	29-5
Adding Metadata Using APPENDRESOURCEMETADATA	29-6
Deleting Metadata Using DELETERESOURCEMETADATA	29-8
Adding Metadata Using SQL DML	29-8
Adding Metadata Using WebDAV PROPPATCH	29-9
Querying XML Schema-Based Resource Metadata	29-10
XML Image Metadata from Binary Image Metadata	29-12
Adding Non-Schema-Based Resource Metadata	29-12
PL/SQL Procedures Affecting Resource Metadata	29-14

30 Oracle XML DB Repository Events

Overview of Repository Events	30-2
Repository Events: Use Cases	30-2
Repository Events and Database Triggers	30-3
Repository Event Listeners and Event Handlers	30-3
Repository Event Configuration	30-3
Possible Repository Events	30-4
Repository Operations and Events	30-6
Repository Event Handler Considerations	30-8
Configuration of Repository Events	30-9
Configuration Element event-listeners	30-10
Configuration Element listener	30-11
Repository Events Configuration Examples	30-12

31 Guidelines for Oracle XML DB Applications in Java

Overview of Oracle XML DB Java Applications	31-2
HTTP(S): Access Java Servlets or Directly Access XMLType Resources	31-2
Use JDBC XMLType Support to Access Many XMLType Object Elements	31-3
Use Servlets to Manipulate and Write Out Data Quickly as XML	31-3
Oracle XML DB Java Servlet Support Restrictions	31-3
Configuration of Oracle XML DB Servlets	31-3
HTTP Request Processing for Oracle XML DB Servlets	31-7
Session Pool and Oracle XML DB Servlets	31-8

Native XML Stream Support	31-8
Oracle XML DB Servlet APIs	31-8
Oracle XML DB Servlet Example	31-9

32 Data Access Using URIs

Overview of Oracle XML DB URI Features	32-2
URIs and URLs	32-2
URIType and its Subtypes	32-3
Overview of DBUris and XDBUris	32-4
URIType PL/SQL Methods	32-5
HTTPURIType PL/SQL Method GETCONTENTTYPE()	32-6
DBURIType PL/SQL Method GETCONTENTTYPE()	32-7
DBURIType PL/SQL Method GETCLOB()	32-7
DBURIType PL/SQL Method GETBLOB()	32-8
Accessing Data Using URIType Instances	32-8
XDBUris: Pointers to Repository Resources	32-11
XDBUri URI Syntax	32-11
Using XDBUri: Examples	32-12
DBUris: Pointers to Database Data	32-14
View the Database as XML Data	32-15
DBUri URI Syntax	32-16
DBUris are Scoped to a Database and Session	32-18
Using DBUris —Examples	32-18
Targeting a Table Using a DBUri	32-19
Targeting a Row in a Table Using a DBUri	32-20
Targeting a Column Using a DBUri	32-20
Retrieving the Text Value of a Column Using a DBUri	32-22
Targeting a Collection Using a DBUri	32-22
Create New Subtypes of URIType Using Package URIFACTORY	32-23
Registering New URIType Subtypes with Package URIFACTORY	32-24
SYS_DBURIGEN SQL Function	32-26
Rules for Passing Columns or Object Attributes to SYS_DBURIGEN	32-27
Using SQL Function SYS_DBURIGEN: Examples	32-27
Inserting Database References Using SYS__DBURIGEN	32-28
Returning Partial Results Using SYS__DBURIGEN	32-28
Returning URLs to Inserted Objects Using SYS_DBURIGEN	32-29
DBUriServlet	32-30
Overriding the MIME Type Using a URL	32-32
Customizing DBUriServlet	32-32
Using Roles for DBUriServlet Security	32-33
Configuring Package URIFACTORY to Handle DBUris	32-34

33 Native Oracle XML DB Web Services

Overview of Native Oracle XML DB Web Services	33-1
Configuring and Enabling Web Services for Oracle XML DB	33-2
Configuring Web Services for Oracle XML DB	33-3
Enabling Web Services for a Specific User	33-4
Query Oracle XML DB Using a Web Service	33-4
Access to PL/SQL Stored Procedures Using a Web Service	33-6
Using a PL/SQL Function with a Web Service: Example	33-8

Part VII Oracle Tools that Support Oracle XML DB

34 Administration of Oracle XML DB

Upgrade or Downgrade of an Existing Oracle XML DB Installation	34-1
Authentication Considerations for Database Installation, Upgrade and Downgrade	34-2
Authentication Considerations for a Database Installation	34-3
Authentication Considerations for a Database Upgrade	34-3
Authentication Considerations for a Database Downgrade	34-3
Automatic Installation of Oracle XML DB	34-4
Validation of ACL Documents and Configuration File	34-4
Administration of Oracle XML DB Using Oracle Enterprise Manager	34-5
Configuration of Oracle XML DB Using xdbconfig.xml	34-5
Oracle XML DB Configuration File, xdbconfig.xml	34-6
Element xdbconfig (Top-Level)	34-7
Element sysconfig (Child of xdbconfig)	34-7
Element userconfig (Child of xdbconfig)	34-7
Element protocolconfig (Child of sysconfig)	34-8
Element httpconfig (Child of protocolconfig)	34-8
Element servlet (Descendant of httpconfig)	34-9
Oracle XML DB Configuration File Example	34-10
Oracle XML DB Configuration API	34-12
Configuration of Mappings from Default Namespace to Schema Location	34-13
Configuration of XML File Extensions	34-15
Oracle XML DB and Database Consolidation	34-15
Package DBMS_XDB_ADMIN	34-16

35	How to Load XML Data	
	Overview of Loading XMLType Data Into Oracle Database	35-1
	Load XMLType Data Using SQL*Loader	35-2
	Load XMLType LOB Data Using SQL*Loader	35-2
	Load LOB Data Using Predetermined Size Fields	35-3
	Load LOB Data Using Delimited Fields	35-3
	Load XML Columns Containing LOB Data from LOBFILES	35-3
	Specify LOBFILES	35-4
	Load XMLType Data Directly from a Control File Using SQL*Loader	35-4
	Loading Large XML Documents Using SQL*Loader	35-4
36	Export and Import of Oracle XML DB Data	
	Overview of Exporting and Importing XMLType Tables	36-1
	Export/Import Limitations for Oracle XML DB Repository	36-3
	Export/Import Syntax and Examples	36-3
	Performing a Table-Mode Export /Import	36-3
	Performing a Schema-Mode Export/Import	36-4
37	XML Data Exchange Using Oracle Database Advanced Queuing	
	XML and Oracle Database Advanced Queuing	37-1
	Oracle Database Advanced Queuing and XML Message Payloads	37-2
	Advantages of Using Oracle Database Advanced Queuing	37-3
	Oracle Database Advanced Queuing	37-3
	Message Queuing	37-4
	XMLType Attributes in Object Types	37-4
	Internet Data Access Presentation (iDAP): SOAP for AQ	37-4
	iDAP Architecture	37-5
	XMLType Queue Payloads	37-6
	Guidelines for Using XML and Oracle Database Advanced Queuing	37-8
	Store AQ XML Messages with Many PDFs as One Record	37-9
	Add New Recipients After Messages Are Enqueued	37-9
	Enqueue and Dequeue XML Messages	37-9
	Parse Messages with XML Content from AQ Queues	37-9
	Prevent the Listener from Stopping Until an XML Document Is Processed	37-10
	HTTPS with AQ	37-10
	Store XML in Oracle AQ Message Payloads	37-10
	iDAP and SOAP	37-10

A Oracle-Supplied XML Schemas and Examples

XDBResource.xsd: XML Schema for Oracle XML DB Resources	A-1
XDBResConfig.xsd: XML Schema for Resource Configuration	A-11
acl.xsd: XML Schema for ACLs	A-15
xdbconfig.xsd: XML Schema for Configuring Oracle XML DB	A-18
xdiff.xsd: XML Schema for Comparing Schemas for In-Place Evolution	A-32
Purchase-Order XML Schemas	A-34
XSLT Stylesheet Example, PurchaseOrder.xsl	A-44
Loading XML Data Using C (OCI)	A-50
Initializing and Terminating an XML Context (OCI)	A-54

B Oracle XML DB Restrictions

C XMLIndex Unstructured Component

Guidelines for Using XMLIndex with an Unstructured Component	C-5
Ignore the Path Table – It Is Transparent	C-7
Column VALUE of an XMLIndex Path Table	C-7
Secondary Indexes on Column VALUE	C-9
XPath Expressions That Are Not Indexed by an XMLIndex Unstructured Component	C-9
Using XMLIndex with an Unstructured Component	C-9
Creating Additional Secondary Indexes on an XMLIndex Path Table	C-11
Asynchronous (Deferred) Maintenance of XMLIndex Indexes	C-16
Syncing an XMLIndex Index in Case of Error ORA-08181	C-18
Advantages of Unstructured XMLIndex	C-18
XMLIndex Path Subsetting: Specifying the Paths You Want to Index	C-19
Examples of XMLIndex Path Subsetting	C-20
XMLIndex Path-Subsetting Rules	C-21
PARAMETERS Clause for CREATE INDEX and ALTER INDEX in Unstructured Index	C-21
Usage of PATHS Clause	C-21
Usage of create_index_paths_clause and alter_index_paths_clause	C-22
Usage of pikey_clause, path_id_clause, and order_key_clause	C-22
Usage of value_clause	C-22
Usage of async_clause	C-22

D Indexing XML Data for Full-Text Queries (pre-23ai)

Creating and Using an XML Search Index	D-2
--	-----

What To Do If an XML Search Index Is Not Picked Up	D-4
Pragma ora:no_schema: Using XML Schema-Based Data with XQuery Full Text	D-4
Pragma ora:use_xmltext_idx: Forcing the Use of an XML Search Index	D-5
Migrating from Using Oracle Text Index to XML Search Index	D-6

Index

List of Examples

3-1	Creating a Table with an XMLType Column	3-2
3-2	Creating a Table of XMLType	3-3
3-3	Creating a Sharded Table in a System-Sharded Environment	3-4
3-4	Creating a Duplicated Table in a Sharded Environment	3-4
3-5	Creating a Virtual Column for an XML Attribute in an XMLType Table	3-5
3-6	Creating a Virtual Column for an XML Attribute in an XMLType Column	3-5
3-7	Partitioning a Relational Table That Has an XMLType Column	3-6
3-8	Partitioning an XMLType Table	3-7
3-9	Error From Attempting to Insert an Incorrect XML Document	3-8
3-10	Constraining a Binary XML Table Using a Virtual Column	3-9
3-11	Constraining a Binary XML Column Using a Virtual Column: Uniqueness	3-10
3-12	Constraining a Binary XML Column Using a Virtual Column: Foreign Key	3-10
3-13	Enforcing Database Integrity When Loading XML Using FTP	3-11
3-14	Creating a Database Directory	3-13
3-15	Inserting XML Content into an XMLType Table	3-13
3-16	Inserting Content into an XMLType Table Using Java	3-14
3-17	Inserting Content into an XMLType Table Using C	3-14
3-18	Inserting XML Content into the Repository Using CREATERESOURCE	3-16
3-19	PurchaseOrder XML Instance Document	3-18
3-20	Retrieving an Entire XML Document Using OBJECT_VALUE	3-19
3-21	Accessing XML Fragments Using XMLQUERY	3-20
3-22	Accessing a Text Node Value Using XMLCAST and XMLQuery	3-21
3-23	Searching XML Content Using XMlexists, XMLCast, and XMLQuery	3-23
3-24	Joining Data from an XMLType Table and a Relational Table	3-25
3-25	Accessing Description Nodes Using XMLTABLE	3-26
3-26	Counting the Number of Elements in a Collection Using XMLTABLE	3-27
3-27	Counting the Number of Child Elements in an Element Using XMLTABLE	3-28
3-28	Updating a Text Node	3-28
3-29	Replacing an Entire Element Using XQuery Update	3-29
3-30	Changing Text Node Values Using XQuery Update	3-30
3-31	Generating XML Data Using SQL/XML Functions	3-32
3-32	Creating XMLType Views Over Conventional Relational Tables	3-34
3-33	Querying XMLType Views	3-34
3-34	Generating XML Data from a Relational Table Using DBURITYPE and getXML()	3-37
3-35	Restricting Rows Using an XPath Predicate	3-37
3-36	Restricting Rows and Columns Using an XPath Predicate	3-37

3-37	Creating a CSX Based Table p with the Following Specifications	3-41
3-38	Migrating Table p to TBX Based Table int_p Using Online Redefinition	3-42
4-1	Chaining XMLTable Calls	4-17
4-2	Finding a Node Using SQL/XML Function XMLEExists	4-18
4-3	Extracting the Scalar Value of an XML Fragment Using XMLCAST	4-20
4-4	Static Type-Checking of XQuery Expressions: oradb URI scheme	4-27
4-5	Static Type-Checking of XQuery Expressions: XML Schema-Based Data	4-27
5-1	Creating Resources for Examples	5-2
5-2	XMLQuery Applied to a Sequence of Items of Different Types	5-3
5-3	FLOWR Expression Using for, let, order by, where, and return	5-4
5-4	FLOWR Expression Using Built-In Functions	5-5
5-5	Querying Relational Data as XML Using XMLQuery	5-7
5-6	Querying Relational Data as XML Using a Nested FLWOR Expression	5-7
5-7	Querying Relational Data as XML Using XMLTable	5-9
5-8	Querying an XMLType Column Using XMLQuery PASSING Clause	5-12
5-9	Using XMLTABLE with XML Schema-Based Data	5-13
5-10	Using XMLQUERY with XML Schema-Based Data	5-14
5-11	Using XMLTABLE with PASSING and COLUMNS Clauses	5-14
5-12	Using XMLTABLE with RETURNING SEQUENCE BY REF	5-15
5-13	Using Chained XMLTABLE with Access by Reference	5-16
5-14	Using XMLTABLE to Decompose XML Collection Elements into Relational Data	5-16
5-15	Using XMLQUERY with a Namespace Declaration	5-17
5-16	Using XMLTABLE with the XMLNAMESPACES Clause	5-18
5-17	Querying XMLTYPE Data	5-19
5-18	Querying Transient XMLTYPE Data Using a PL/SQL Cursor	5-20
5-19	Extracting XML Data and Inserting It into a Relational Table Using SQL	5-21
5-20	Extracting XML Data and Inserting It into a Table Using PL/SQL	5-22
5-21	Searching XML Data Using SQL/XML Functions	5-23
5-22	Extracting Fragments Using XMLQUERY	5-23
5-23	Using the SQL*Plus XQUERY Command	5-24
5-24	Using XQuery with PL/SQL	5-25
5-25	Using XQuery with JDBC	5-26
5-26	Using XQuery with ODP.NET and C#	5-26
5-27	Updating XMLType Data Using SQL UPDATE	5-29
5-28	Updating XMLTYPE Data Using SQL UPDATE and XQuery Update	5-30
5-29	Updating Multiple Text Nodes and Attribute Nodes	5-31
5-30	Updating Selected Nodes within a Collection	5-32

5-31	Incorrectly Updating a Node That Occurs Multiple Times in a Collection	5-33
5-32	Correctly Updating a Node That Occurs Multiple Times in a Collection	5-34
5-33	NULL Updates – Element and Attribute	5-36
5-34	NULL Updates – Text Node	5-37
5-35	Inserting an Element into a Collection	5-38
5-36	Inserting an Element that Uses a Namespace	5-39
5-37	Inserting an Element Before an Element	5-39
5-38	Inserting an Element as the Last Child Element	5-40
5-39	Deleting an Element	5-40
5-40	Creating a View Using Updated XML Data	5-41
5-41	Optimization of XMLQuery over Relational Data	5-44
5-42	Optimization of XMLTable over Relational Data	5-44
5-43	Optimization of XMLQuery with Schema-Based XMLType Data	5-45
5-44	Optimization of XMLTable with Schema-Based XMLType Data	5-46
5-45	Unoptimized Repository Query Using fn:doc	5-49
5-46	Optimized Repository Query Using EQUALS_PATH	5-49
5-47	Repository Query Using Oracle XQuery Pragma ora:defaultTable	5-50
6-1	Making Query Data Compatible with Index Data – SQL Cast	6-10
6-2	Making Query Data Compatible with Index Data – XQuery Cast	6-10
6-3	Exchange-Partitioning Tables That Have an XMLIndex Structured Component	6-11
6-4	Exchange-Partitioning Reference-Partitioned Tables That Use XMLIndex	6-11
6-5	Data Used in Example of Exchange-Partitioning for Reference-Partitioned Tables	6-12
6-6	Creating an XMLIndex Index	6-13
6-7	Obtaining the Name of an XMLIndex Index on a Particular Table	6-13
6-8	Renaming and Dropping an XMLIndex Index	6-14
6-9	XMLIndex with a Structured Component, Using Namespaces and Storage Options	6-16
6-10	XMLIndex with a Structured Component, Specifying TABLESPACE at the Index Level	6-16
6-11	XMLIndex Index: Adding a Structured Component	6-17
6-12	Using DBMS_XMLINDEX.PROCESS_PENDING To Index XML Data	6-18
6-13	Dropping an XMLIndex Structured Component	6-20
6-14	Creating a B-tree Index on an XMLIndex Index Content Table	6-20
6-15	Using a Structured XMLIndex Component for a Query with Multilevel Chaining	6-21
6-16	Turning Off XMLIndex Using Optimizer Hints	6-23
6-17	Creating an XMLIndex Index in Parallel	6-24
6-18	Using Different PARALLEL Degrees for XMLIndex Internal Objects	6-24
6-19	Automatic Collection of Statistics on XMLIndex Objects	6-25
6-20	XQuery Full Text Query	6-42

6-21	Execution Plan for XQuery Full Text Query	6-42
6-22	A Range Search Query in an XMLEExists predicate answered by an XML Search Index	6-43
6-23	Execution Plan for a Range Search Query in an XMLEExists predicate answered by an XML Search Index	6-43
6-24	CREATE INDEX Using XMLCAST and XMLQUERY on a Singleton Element	6-47
6-25	CREATE INDEX Using EXTRACTVALUE on a Singleton Element	6-47
7-1	XSLT Stylesheet Example: PurchaseOrder.xsl	7-2
7-2	Registering an XML Schema and Inserting XML Data	7-4
7-3	Using SQL Function XMLTRANSFORM to Apply an XSL Stylesheet	7-6
7-4	Using XMLType Method TRANSFORM() with a Transient XSL Stylesheet	7-7
7-5	Using XMLTRANSFORM to Apply an XSL Stylesheet Retrieved Using XDBURITYPE	7-8
7-6	Error When Inserting Incorrect XML Document (Partial Validation)	7-13
7-7	Forcing Full XML Schema Validation Using a CHECK Constraint	7-14
7-8	Enforcing Full XML Schema Validation Using a BEFORE INSERT Trigger	7-14
7-9	Validating XML Using Method ISSCHEMAVALID() in SQL	7-15
7-10	Validating XML Using Method ISSCHEMAVALID() in PL/SQL	7-15
7-11	Validating XML Using Method SCHEMAVALIDATE() within Triggers	7-16
7-12	Checking XML Validity Using XMLISVALID Within CHECK Constraints	7-16
8-1	XMLELEMENT: Formatting a Date	8-8
8-2	XMLELEMENT: Generating an Element for Each Employee	8-8
8-3	XMLELEMENT: Generating Nested XML	8-8
8-4	XMLELEMENT: Generating Employee Elements with Attributes ID and Name	8-9
8-5	XMLELEMENT: Characters in Generated XML Data Are Not Escaped	8-9
8-6	Creating a Schema-Based XML Document Using XMLELEMENT with Namespaces	8-9
8-7	XMLELEMENT: Generating an Element from a User-Defined Data-Type Instance	8-10
8-8	XMLFOREST: Generating Elements with Attribute and Child Elements	8-11
8-9	XMLFOREST: Generating an Element from a User-Defined Data-Type Instance	8-12
8-10	XMLCONCAT: Concatenating XMLType Instances from a Sequence	8-13
8-11	XMLCONCAT: Concatenating XML Elements	8-13
8-12	XMLAGG: Generating a Department Element with Child Employee Elements	8-14
8-13	XMLAGG: Using GROUP BY to Generate Multiple Department Elements	8-15
8-14	XMLAGG: Generating Nested Elements	8-15
8-15	Using SQL/XML Function XMLPI	8-17
8-16	Using SQL/XML Function XMLCOMMENT	8-18
8-17	Using SQL/XML Function XMLSERIALIZE	8-19
8-18	Using SQL/XML Function XMLPARSE	8-20
8-19	XMLCOLATTVAL: Generating Elements with Attribute and Child Elements	8-21

8-20	Using Oracle SQL Function XMLCDATA	8-22
8-21	DBMS_XMLGEN: Generating Simple XML	8-33
8-22	DBMS_XMLGEN: Generating Simple XML with Pagination (Fetch)	8-33
8-23	DBMS_XMLGEN: Generating XML Using Object Types	8-35
8-24	DBMS_XMLGEN: Generating XML Using User-Defined Data-Type Instances	8-36
8-25	DBMS_XMLGEN: Generating an XML Purchase Order	8-38
8-26	DBMS_XMLGEN: Generating a New Context Handle from a REF Cursor	8-42
8-27	DBMS_XMLGEN: Specifying NULL Handling	8-43
8-28	DBMS_XMLGEN: Generating Recursive XML with a Hierarchical Query	8-44
8-29	DBMS_XMLGEN: Binding Query Variables Using SETBINDVALUE()	8-46
8-30	Using XMLAGG ORDER BY Clause	8-49
8-31	Returning a Rowset Using XMLTABLE	8-50
9-1	Creating a Relational View of XML Content	9-2
9-2	Accessing Individual Members of a Collection Using a View	9-4
9-3	XMLIndex Index that Matches Relational View Columns	9-5
9-4	XMLTable Expression Returned by PL/SQL Function getSIDXDefFromView	9-5
9-5	Querying Master Relational View of XML Data	9-6
9-6	Querying Master and Detail Relational Views of XML Data	9-6
9-7	Business-Intelligence Query of XML Data Using a View	9-6
10-1	Creating an XMLType View Using XMLELEMENT	10-3
10-2	Registering XML Schema emp_simple.xsd	10-5
10-3	Creating an XMLType View Using SQL/XML Publishing Functions	10-6
10-4	Querying an XMLType View	10-6
10-5	Using Namespace Prefixes with SQL/XML Publishing Functions	10-7
10-6	XML Schema with No Target Namespace	10-8
10-7	Creating a View for an XML Schema with No Target Namespace	10-8
10-8	Using SQL/XML Functions in XML Schema-Based XMLType Views	10-9
10-9	Creating Object Types for Schema-Based XMLType Views	10-12
10-10	Creating and Registering XML Schema emp_complex.xsd	10-13
10-11	Creating XMLType View emp_xml Using Object Type emp_t	10-14
10-12	Creating an Object View and an XMLType View Based on the Object View	10-15
10-13	Creating Object Types	10-15
10-14	Registering XML Schema dept_complex.xsd	10-16
10-15	Creating XMLType View dept_xml Using Object Type dept_t	10-17
10-16	Creating XMLType View dept_xml Using Relational Data Directly	10-18
10-17	Creating an XMLType View by Restricting Rows from an XMLType Table	10-18
10-18	Creating an XMLType View by Transforming an XMLType Table	10-19

10-19	Determining Whether an XMLType View Is Implicitly Updatable, and Updating It	10-19
11-1	Creating and Manipulating a DOM Document	11-17
11-2	Creating an Element Node and Obtaining Information About It	11-18
11-3	Creating a User-Defined Subtype of SYS.util_BinaryOutputStream()	11-21
11-4	Retrieving Node Value with a User-Defined Stream	11-21
11-5	Get-Pull of Binary Data	11-22
11-6	Get-Pull of Character Data	11-23
11-7	Set-Pull of Binary Data	11-24
11-8	Set-Push of Binary Data	11-25
11-9	Parsing an XML Document	11-27
11-10	Transforming an XML Document Using an XSL Stylesheet	11-30
12-1	Inserting Data with Specified Columns	12-2
12-2	Updating Data with Key Columns	12-4
12-3	DBMS_XMLSTORE.DELETEXML Example	12-5
13-1	Querying an XMLType Table Using JDBC	13-3
13-2	Selecting XMLType Data Using getString() and getCLOB()	13-3
13-3	Returning XMLType Data Using getSQLXML()	13-4
13-4	Returning XMLType Data Using an Output Parameter	13-4
13-5	Updating an XMLType Column Using SQL Constructor XMLType and Java String	13-6
13-6	Updating an XMLType Column Using SQLXML	13-6
13-7	Retrieving Metadata About an XMLType Column Using JDBC	13-6
13-8	Updating an XMLType Column Using JDBC	13-6
13-9	Updated Purchase-Order Document	13-8
13-10	Inserting an XMLType column using JDBC	13-10
13-11	Converting an XML String to an OracleClob Instance	13-10
13-12	Policy File Granting Permissions for Java DOM API	13-11
13-13	Creating a DOM Object with the Java DOM API	13-12
13-14	Using the Java DOM API with a Binary XML Column	13-20
14-1	Using OCIXMLDBINITXMLCTX() and OCIXMLDBFREEXMLCTX()	14-5
14-2	Using the C API for XML with Binary XML	14-9
14-3	Using the Oracle XML DB Pull Parser	14-11
14-4	Using the DOM to Count Ordered Parts	14-16
15-1	Retrieve XMLType Data to .NET	15-2
17-1	Registering an XML Schema Using DBMS_XMLSCHEMA.REGISTERSCHEMA	17-12
17-2	Objects Created During XML Schema Registration	17-12
17-3	Registering a Local XML Schema	17-15
17-4	Registering a Global XML Schema	17-16

17-5	Deleting an XML Schema with DBMS_XMLSCHEMA.DELETESCHEMA	17-18
17-6	Data Dictionary Table for Registered Schemas	17-18
17-7	Creating XML Schema-Based XMLType Tables and Columns	17-21
17-8	Creating an Object-Relational XMLType Table with Default Storage	17-26
17-9	Specifying Object-Relational Storage Options for XMLType Tables and Columns	17-26
17-10	Using STORE ALL VARRAYS AS	17-27
18-1	SQL Object Types for Storing XMLType Tables	18-4
18-2	Default Table for Global Element PurchaseOrder	18-5
18-3	Using Common Schema Annotations	18-10
18-4	Registering an Annotated XML Schema	18-11
18-5	Using DBMS_XMLSCHEMA_ANNOTATE	18-14
18-6	Querying View USER_XML_SCHEMAS for a Registered XML Schema	18-20
18-7	Querying Metadata from a Registered XML Schema	18-20
18-8	Mapping XML Schema Data Types to SQL Data Types Using Attribute SQLType	18-22
18-9	XML Schema Inheritance: complexContent as an Extension of complexTypes	18-32
18-10	Inheritance in XML Schema: Restrictions in complexTypes	18-33
18-11	XML Schema complexType: Mapping complexType to simpleContent	18-34
18-12	XML Schema: Mapping complexType to any/anyAttribute	18-35
18-13	Creating an XMLType Table that Conforms to an XML Schema	18-36
18-14	Creating an XMLType Table for Nested Collections	18-37
18-15	Using DESCRIBE with an XML Schema-Based XMLType Table	18-37
18-16	Specifying Partitioning Information During XML Schema Registration	18-39
18-17	Specifying Partitioning Information During Table Creation	18-39
18-18	Integrity Constraints and Triggers for an XMLType Table Stored Object-Relationally	18-41
18-19	Adding a Unique Constraint to the Parent Element of an Attribute	18-42
18-20	Setting SQLInline to False for Out-Of-Line Storage	18-45
18-21	Generated XMLType Tables and Types	18-45
18-22	Querying an Out-Of-Line Table	18-46
18-23	Storing a Collection Out of Line	18-47
18-24	Generated Out-Of-Line Collection Type	18-48
18-25	Renaming an Intermediate Table of REF Values	18-48
18-26	XPath Rewrite for an Out-Of-Line Collection	18-48
18-27	XPath Rewrite for an Out-Of-Line Collection, with Index on REFs	18-49
18-28	An XML Schema with Circular Dependency	18-50
18-29	XML Schema: Cycling Between complexTypes	18-53
18-30	XML Schema: Cycling Between complexTypes, Self-Reference	18-53
18-31	An XML Schema that Includes a Non-Existent XML Schema	18-55

18-32	Using the FORCE Option to Register XML Schema xm40.xsd	18-55
18-33	Trying to Create a Table Using a Cyclic XML Schema	18-55
18-34	Using the FORCE Option to Register XML Schema xm40a.xsd	18-56
18-35	Recursive XML Schema	18-57
18-36	Out-of-line Table	18-58
18-37	Invalid Default Table Sharing	18-58
18-38	Oracle XML DB XML Schema: Mapping complexType XML Fragments to LOBs	18-60
19-1	XPath Rewrite	19-2
19-2	XPath Rewrite for an Out-Of-Line Table	19-4
19-3	Using an Index with an Out-Of-Line Table	19-5
19-4	Execution Plan Generated When XPath Rewrite Does Not Occur	19-6
19-5	Analyzing an Execution Plan to Determine a Column to Index	19-8
19-6	Using DBMS_XMLSTORAGE_MANAGE.XPATH2TABCOLMAPPING	19-8
19-7	Creating an Index on a Column Targeted by a Predicate	19-8
19-8	Creating a Function-Based Index for a Column Targeted by a Predicate	19-8
19-9	Execution Plan Showing that Index Is Picked Up	19-9
19-10	Creating a Function-Based Index for a Column Targeted by a Predicate	19-9
19-11	Execution Plan for a Selection of Collection Elements	19-10
19-12	Creating an Index for Direct Access to an Ordered Collection Table	19-10
20-1	Revised Purchase-Order XML Schema	20-3
20-2	evolvePurchaseOrder.xsl: XSLT Stylesheet to Update Instance Documents	20-11
20-3	Loading Revised XML Schema and XSLT Stylesheet	20-15
20-4	Updating an XML Schema Using DBMS_XMLSCHEMA.COPYEVOLVE	20-15
20-5	Splitting a Complex Type into Two Complex Types	20-18
20-6	diffXML Parameter Document	20-25
21-1	Querying PATH_VIEW to Determine Link Type	21-15
21-2	Obtaining the OID Path of a Resource	21-15
21-3	Creating a Weak Link Using an OID Path	21-16
21-4	Accessing a Text Document in the Repository Using XDBURITYPE	21-31
21-5	Accessing Resource Content Using RESOURCE_VIEW	21-31
21-6	Accessing XML Documents Using Resource and Namespace Prefixes	21-32
21-7	Querying Repository Resource Data Using SQL Function REF and Element XMLRef	21-33
21-8	Selecting XML Document Fragments Based on Metadata, Path, and Content	21-33
21-9	Updating a Text Document Using UPDATE and XQuery Update on the Resource	21-35
21-10	Updating an XML Node Using UPDATE and XQuery Update on the Resource	21-36
21-11	Updating XML Schema-Based Documents in the Repository	21-37
21-12	Accessing Resources Using EQUALS_PATH and RESOURCE_VIEW	21-38

21-13	Determining the Path to XSLT Stylesheets Stored in the Repository	21-39
21-14	Counting Resources Under a Path	21-40
21-15	Listing the Folder Contents in a Path	21-40
21-16	Listing the Links Contained in a Folder	21-40
21-17	Finding Paths to Resources that Contain Purchase-Order XML Documents	21-41
21-18	Execution Plan Output for a Folder-Restricted Query	21-41
22-1	Resource Configuration File	22-5
22-2	applicationData Element	22-6
23-1	XInclude Used in a Book Document to Include Parts and Chapters	23-5
23-2	Expanding Document Inclusions Using XDBURITYPE	23-8
23-3	Querying Document Links Mapped From XLink Links	23-10
23-4	Querying Document Links Mapped From XInclude Links	23-11
23-5	Mapping XInclude Links to Hard Document Links, with OID Retrieval	23-17
23-6	Mapping XLink Links to Weak Links, with Named-Path Retrieval	23-17
23-7	Configuring XInclude Document Decomposition	23-17
23-8	Repository Document, Showing Generated xi:include Elements	23-18
24-1	Determining Paths Under a Path: Relative	24-10
24-2	Determining Paths Under a Path: Absolute	24-10
24-3	Determining Paths Not Under a Path	24-11
24-4	Determining Paths Using Multiple Correlations	24-11
24-5	Relative Path Names for Three Levels of Resources	24-12
24-6	Extracting Resource Metadata Using UNDER_PATH	24-12
24-7	Using Functions PATH and DEPTH with PATH_VIEW	24-13
24-8	Extracting Link and Resource Information from PATH_VIEW	24-13
24-9	All Repository Paths to a Certain Depth Under a Path	24-14
24-10	Locating a Repository Path Using EQUALS_PATH	24-15
24-11	Retrieve RESID of a Given Resource	24-15
24-12	Obtaining the Path Name of a Resource from its RESID	24-15
24-13	Folders Under a Given Path	24-16
24-14	Joining RESOURCE_VIEW with an XMLType Table	24-16
24-15	Deleting Resources	24-17
24-16	Deleting Links to Resources	24-17
24-17	Deleting a Nonempty Folder	24-18
24-18	Updating a Resource	24-19
24-19	Updating a Path in the PATH_VIEW	24-20
24-20	Updating Resources Based on Attributes	24-21
24-21	Finding Resources Inside a Folder	24-21

24-22	Copying Resources	24-22
24-23	Find All Resources Containing "Paper"	24-24
24-24	Find All Resources Containing "Paper" that are Under a Specified Path	24-24
25-1	Creating a Repository Resource	25-7
25-2	Creating a Version-Controlled Resource	25-7
25-3	Retrieving Resource Content by Referencing the Resource ID	25-8
25-4	Checking Out a Version-Controlled Resource	25-8
25-5	Updating Resource Content	25-8
25-6	Checking In a Version-Controlled Resource	25-9
25-7	Retrieving Resource Version Content Using XDBURITYPE and CREATEOIDPATH	25-9
25-8	Retrieving Resource Version Content Using GETCONTENTSCLOBBYRESID	25-9
25-9	Retrieving Resource Version Metadata Using GETRESOURCEBYRESID	25-10
25-10	Canceling a Check-Out Using UNCHECKOUT	25-11
26-1	Managing Resources Using DBMS_XDB_REPOS	26-3
26-2	Using DBMS_XDB_REPOS.GETACLDOCUMENT	26-4
26-3	Using DBMS_XDB_REPOS.SETACL	26-5
26-4	Using DBMS_XDB_REPOS.CHANGEPRIVILEGES	26-5
26-5	Using DBMS_XDB_REPOS.GETPRIVILEGES	26-6
26-6	Using DBMS_XDB_CONFIG.CFG_GET	26-8
26-7	Using DBMS_XDB_CONFIG.CFG_UPDATE	26-9
27-1	Simple Access Control Entry (ACE) that Grants a Privilege	27-5
27-2	Simple Access Control List (ACL) that Grants a Privilege	27-6
27-3	Complementing a Set of Principals with Element invert	27-12
27-4	Creating an ACL Using CREATERESOURCE	27-13
27-5	Retrieving an ACL Document, Given its Repository Path	27-14
27-6	Setting the ACL of a Resource	27-15
27-7	Deleting an ACL	27-15
27-8	Updating (Replacing) an Access Control List	27-16
27-9	Appending ACEs to an Access Control List	27-16
27-10	Deleting an ACE from an Access Control List	27-16
27-11	Retrieving the ACL Document for a Resource	27-17
27-12	Retrieving Privileges Granted to the Current User for a Particular Resource	27-17
27-13	Checking If a User Has a Certain Privileges on a Resource	27-18
27-14	Checking User Privileges Using ACLCheckPrivileges	27-19
27-15	Retrieving the Path of the ACL that Protects a Given Resource	27-20
27-16	Retrieving the Paths of All Resources Protected by a Given ACL	27-20
27-17	ACL Referencing an LDAP User	27-24

27-18	ACL Referencing an LDAP Group	27-24
28-1	Listener Status with FTP and HTTP(S) Protocol Support Enabled	28-10
28-2	Uploading Content to the Repository Using FTP	28-18
28-3	Navigating Oracle ASM Folders	28-20
28-4	Transferring Oracle ASM Files Between Databases with FTP proxy Method	28-20
28-5	FTP Connection Using IPv6	28-22
28-6	Modifying the Default Timeout Value of an FTP Session	28-23
29-1	Registering an XML Schema for Technical Photo Information	29-4
29-2	Registering an XML Schema for Photo Categorization	29-5
29-3	Add Metadata to a Resource – Technical Photo Information	29-7
29-4	Add Metadata to a Resource – Photo Content Categories	29-7
29-5	Delete Specific Metadata from a Resource	29-8
29-6	Adding Metadata to a Resource Using DML with RESOURCE_VIEW	29-8
29-7	Adding Metadata Using WebDAV PROPPATCH	29-10
29-8	Query XML Schema-Based Resource Metadata	29-11
29-9	Add Non-Schema-Based Metadata to a Resource	29-12
30-1	Resource Configuration File for Java Event Listeners with Preconditions	30-13
30-2	Resource Configuration File for PL/SQL Event Listeners with No Preconditions	30-14
30-3	PL/SQL Code Implementing Event Listeners	30-14
30-4	Java Code Implementing Event Listeners	30-16
30-5	Invoking Event Handlers	30-18
31-1	An Oracle XML DB Servlet	31-9
31-2	Registering and Mapping an Oracle XML DB Servlet	31-10
32-1	Using HTTPURITYPE PL/SQL Method GETCONTENTTYPE()	32-7
32-2	Creating and Querying a URI Column	32-9
32-3	Using Different Kinds of URI, Created in Different Ways	32-10
32-4	Access a Repository Resource by URI Using an XDBUri	32-12
32-5	Using PL/SQL Method GETXML with XMLCAST and XMLQUERY	32-14
32-6	Targeting a Complete Table Using a DBUri	32-19
32-7	Targeting a Particular Row in a Table Using a DBUri	32-20
32-8	Targeting a Specific Column Using a DBUri	32-21
32-9	Targeting an Object Column with Specific Attribute Values Using a DBUri	32-21
32-10	Retrieve Only the Text Value of a Node Using a DBUri	32-22
32-11	Targeting a Collection Using a DBUri	32-23
32-12	URIFACTORY: Registering the ECOM Protocol	32-25
32-13	SYS_DBURIGEN: Generating a DBUri that Targets a Column	32-26
32-14	Passing Columns with Single Arguments to SYS_DBURIGEN	32-27

32-15	Inserting Database References Using SYS_DBURIGEN	32-28
32-16	Creating the Travel Story Table	32-29
32-17	A Function that Returns the First 20 Characters	32-29
32-18	Creating a Travel View for Use with SYS_DBURIGEN	32-29
32-19	Retrieving a URL Using SYS_DBURIGEN in RETURNING Clause	32-30
32-20	Changing the Installation Location of DBUriServlet	32-33
32-21	Restricting Servlet Access to a Database Role	32-34
32-22	Registering a Handler for a DBUri Prefix	32-35
33-1	Adding a Web Services Configuration Servlet	33-3
33-2	Verifying Addition of Web Services Configuration Servlet	33-3
33-3	XML Schema for Database Queries To Be Processed by Web Service	33-5
33-4	Input XML Document for SQL Query Using Query Web Service	33-6
33-5	Output XML Document for SQL Query Using Query Web Service	33-6
33-6	Definition of PL/SQL Function Used for Web-Service Access	33-8
33-7	WSDL Document Corresponding to a Stored PL/SQL Function	33-8
33-8	Input XML Document for PL/SQL Query Using Web Service	33-10
33-9	Output XML Document for PL/SQL Query Using Web Service	33-10
34-1	Oracle XML DB Configuration File	34-10
34-2	Updating the Configuration File Using CFG_UPDATE and CFG_GET	34-12
35-1	Data File filelist.dat: List of XML Files to Load	35-5
35-2	Control File load_datra.ctf, for Loading Purchase-Order XML Documents	35-5
35-3	Loading XML Data Using Shell Command sqlldr	35-5
36-1	Exporting XMLType Data in TABLE Mode	36-4
36-2	Importing XMLType Data in TABLE Mode	36-4
36-3	Creating Table po2	36-5
36-4	Exporting XMLType Data in SCHEMA Mode	36-5
36-5	Importing XMLType Data in SCHEMA Mode	36-5
36-6	Importing XMLType Data in SCHEMA Mode, Remapping Schema	36-5
37-1	Creating a Queue Table and Queue	37-6
37-2	Creating a Transformation to Convert Message Data to XML	37-6
37-3	Applying a Transformation before Sending Messages Overseas	37-7
37-4	XMLType and AQ: Dequeueing Messages	37-7
A-1	Unannotated Purchase-Order XML Schema	A-35
A-2	Annotated Purchase-Order XML Schema	A-37
A-3	Revised Annotated Purchase-Order XML Schema	A-40
A-4	PurchaseOrder.xsl XSLT Stylesheet	A-44
A-5	Inserting XML Data into an XMLType Table Using C	A-50

A-6	Using OCIXmlDbInitXmlCtx() and OCIXmlDbFreeXmlCtx()	A-54
C-1	Path Table Contents for Two Purchase Orders	C-3
C-2	Naming the Path Table of an XMLIndex Index	C-10
C-3	Determining the System-Generated Name of an XMLIndex Path Table	C-10
C-4	Specifying Storage Options When Creating an XMLIndex Index	C-10
C-5	Dropping an XMLIndex Unstructured Component	C-11
C-6	Determining the Names of the Secondary Indexes of an XMLIndex Index	C-11
C-7	Obtaining the Name of an XMLIndex Index from Its Path-Table Name	C-13
C-8	Extracting Data from an XML Fragment Using XMLIndex	C-13
C-9	Creating a Function-Based Index on Path-Table Column VALUE	C-15
C-10	Trying to Create a Numeric Index on Path-Table Column VALUE Directly	C-15
C-11	Creating a Numeric Index on Column VALUE with Procedure createNumberIndex	C-15
C-12	Creating a Date Index on Column VALUE with Procedure createDateIndex	C-15
C-13	Creating an Oracle Text CONTEXT Index on Path-Table Column VALUE	C-15
C-14	Showing All Secondary Indexes on an XMLIndex Path Table	C-15
C-15	Specifying Deferred Synchronization for XMLIndex	C-17
C-16	Manually Synchronizing an XMLIndex Index Using SYNCINDEX	C-17
C-17	XMLIndex Path Subsetting with CREATE INDEX	C-20
C-18	XMLIndex Path Subsetting with ALTER INDEX	C-20
C-19	XMLIndex Path Subsetting Using a Namespace Prefix	C-21
D-1	Creating an XML Search Index	D-3
D-2	XQuery Full Text Query	D-3
D-3	Execution Plan for XQuery Full Text Query	D-3
D-4	XQuery Full Text Query with XML Schema-Based Data: Error ORA-18177	D-5
D-5	Using XQuery Pragma ora:no_schema with XML Schema-Based Data	D-5
D-6	Full-Text Query with XQuery Pragma ora:use_xmltext_idx	D-5

List of Figures

1-1	Oracle XML DB Benefits	1-3
1-2	Unifying Data and Content: Some Common XML Architectures	1-4
1-3	XMLType Storage	1-10
1-4	Oracle XML DB Repository Architecture	1-19
2-1	Oracle XML DB Storage Options for XML Data	2-4
3-1	Loading Content into the Repository Using Windows Explorer	3-17
4-1	XMLQUERY Syntax	4-12
4-2	XMLTABLE Syntax	4-13
4-3	XMLExists Syntax	4-17
4-4	XMLCast Syntax	4-19
7-1	XMLTRANSFORM Syntax	7-4
7-2	Using XMLTRANSFORM	7-4
7-3	Database XSL Transformation of a PurchaseOrder Using DBUri Servlet	7-10
7-4	Database XSL Transformation of Departments Table Using DBUri Servlet	7-11
8-1	XMLELEMENT Syntax	8-4
8-2	XMLAttributes Clause Syntax (XMLATTRIBUTES)	8-5
8-3	XMLFOREST Syntax	8-11
8-4	XMLCONCAT Syntax	8-13
8-5	XMLAGG Syntax	8-14
8-6	XMLPI Syntax	8-17
8-7	XMLComment Syntax	8-17
8-8	XMLSerialize Syntax	8-18
8-9	XMLParse Syntax	8-20
8-10	XMLCOLATTVAL Syntax	8-21
8-11	XMLCDATA Syntax	8-22
8-12	Using PL/SQL Package DBMS_XMLGEN	8-25
8-13	SYS_XMLAGG Syntax	8-48
10-1	Creating XMLType Views Clause: Syntax	10-3
11-1	Using the PL/SQL DOM API for XMLType	11-16
11-2	Using the PL/SQL Parser API for XMLType	11-26
11-3	Using the PL/SQL XSLT Processor for XMLType	11-29
13-1	Using the Java DOM API for XMLType	13-14
16-1	XML Use Cases and XMLType Storage Models	16-3
17-1	XMLSpy Graphical Representation of a Purchase-Order XML Schema	17-4
17-2	XMLSpy Support for Oracle XML DB Schema Annotations	17-7
17-3	Creating an XMLType Table – CREATE TABLE Syntax	17-20

17-4	Creating an XMLType Table – XMLType_table Syntax	17-20
17-5	Creating an XMLType Table – table_properties Syntax	17-20
17-6	Creating an XMLType Table – XMLType_virtual_columns Syntax	17-21
17-7	How Oracle XML DB Maps XML Schema-Based XMLType Tables	17-30
18-1	simpleType Mapping: XML Strings to SQL VARCHAR2 or CLOB	18-26
18-2	Mapping complexType to SQL for Out-Of-Line Storage	18-44
18-3	Cross Referencing Between Different complexTypes in the Same XML Schema	18-52
18-4	Self-Referencing Complex Type within an XML Schema	18-52
18-5	Cyclical References Between XML Schemas	18-54
18-6	Mapping complexType XML Fragments to CLOB Instances	18-60
21-1	A Folder Tree, Showing Hierarchical Structures in the Repository	21-3
21-2	Oracle XML DB Folders in Windows Explorer	21-17
21-3	Accessing Repository Data Using HTTP(S)/WebDAV and a Web Browser	21-18
21-4	Path-Based Access Using HTTP and a URL	21-18
21-5	Oracle ASM Virtual Folder Hierarchy	21-22
21-6	Updating and Editing Content Stored in Oracle XML DB Using Microsoft Word	21-34
24-1	Accessing Repository Resources Using RESOURCE_VIEW and PATH_VIEW	24-3
24-2	RESOURCE_VIEW and PATH_VIEW Structure	24-3
24-3	RESOURCE_VIEW and PATH_VIEW Explained	24-6
24-4	UNDER_PATH Syntax	24-7
24-5	EQUALS_PATH Syntax	24-9
24-6	PATH Syntax	24-9
28-1	Oracle XML DB Architecture: Protocol Server	28-3
28-2	Creating a WebFolder in Microsoft Windows	28-34
28-3	Copying Files into Oracle XML DB Repository	28-35
32-1	A DBUri Corresponds to an XML Visualization of Relational Data	32-15
32-2	SYS_DBURIGEN Syntax	32-26
37-1	Oracle Database Advanced Queuing and XML Message Payloads	37-3
37-2	iDAP Architecture for Performing AQ Operations Using HTTP(S)	37-5
C-1	XML Use Cases and XML Indexing	C-3

List of Tables

1-1	Static Data Dictionary Views Related to XML	1-16
3-1	SQL*Loader – Conventional and Direct-Path Load Modes	3-16
4-1	Common XPath Constructs	4-2
4-2	Predefined Namespaces and Prefixes	4-10
4-3	oradb Expressions: Column Types for Comparisons	4-22
6-1	Basic XML Indexing Tasks	6-2
6-2	Tasks Involving XMLIndex Indexes with a Structured Component	6-2
6-3	Tasks involving XML Search Index	6-2
6-4	XML Use Cases and XML Indexing	6-6
6-5	XML and SQL Data Type Correspondence for XMLIndex	6-9
6-6	XMLIndex Static Public Views	6-25
8-1	DBMS_XMLGEN Functions and Procedures	8-26
11-1	PL/SQL APIs Related to XML	11-3
11-2	XML and HTML DOM Node Types and Their Child Node Types	11-13
13-1	Java DOM API for XMLType: Classes	13-13
14-1	OCIXmlDbInitXMLCtx() Parameters	14-3
14-2	Common XMLType Operations in C	14-16
16-1	XMLType Storage Model Considerations	16-8
16-2	XMLType Indexing Considerations	16-9
16-3	XMLType Storage Models: Relative Advantages	16-10
17-1	XMLType Methods Related to XML Schema	17-8
17-2	CREATE TABLE Encoding Options for Binary XML	17-24
18-1	Annotations in Elements	18-15
18-2	Annotations in Elements Declaring Global complexType Elements	18-16
18-3	XML Schema String Data Types Mapped to SQL	18-26
18-4	XML Schema Binary Data Types (hexBinary/base64Binary) Mapped to SQL	18-26
18-5	Default Mapping of Numeric XML Schema Primitive Types to SQL	18-27
18-6	XML Schema Date and Time Data Types Mapped to SQL	18-27
18-7	Default Mapping of Other XML Schema Primitive and Derived Data Types to SQL	18-27
19-1	Sample of XPath Expressions that Are Rewritten to Underlying SQL Constructs	19-3
20-1	Parameters of Procedure DBMS_XMLSCHEMA.COPYEVOLVE	20-6
20-2	Errors Associated with Procedure DBMS_XMLSCHEMA.COPYEVOLVE	20-7
20-3	XML Schema Evolution: XMLType Table Temporary Table Columns	20-14
20-4	XML Schema Evolution: XMLType Column Temporary Table Columns	20-14
20-5	Procedure COPYEVOLVE Mapping Table	20-15
20-6	Parameters of Procedure DBMS_XMLSCHEMA.INPLACEEVOLVE	20-22

21-1	Synonyms for Oracle XML DB Repository Terms	21-8
21-2	Differences Between PATH_VIEW and RESOURCE_VIEW	21-23
21-3	Accessing Oracle XML DB Repository: API Options	21-25
24-1	Structure of RESOURCE_VIEW	24-5
24-2	Structure of PATH_VIEW	24-5
24-3	UNDER_PATH SQL Function Signature	24-7
25-1	Oracle XML DB Versioning Terms	25-2
25-2	PL/SQL Functions and Procedures in Package DBMS_XDB_VERSION	25-3
26-1	DBMS_XDB_REPOS Resource Access and Management Subprograms	26-2
26-2	DBMS_XDB_REPOS: Security Management Subprograms	26-4
26-3	DBMS_XDB_CONFIG: Configuration Management Subprograms	26-7
27-1	Database Privileges Needed for Operations on Oracle XML DB Resources	27-6
27-2	Atomic Privileges	27-8
27-3	Aggregate Privileges	27-8
28-1	Common Protocol Configuration Parameters	28-5
28-2	Configuration Parameters Specific to FTP	28-6
28-3	Configuration Parameters Specific to HTTP(S)/WebDAV (Except Servlet)	28-6
30-1	Predefined Repository Events	30-5
30-2	Oracle XML DB Repository Operations and Events	30-7
31-1	XML Elements Defined for Servlet Deployment Descriptors	31-3
31-2	Java Servlet 2.2 Methods that Are Not Implemented	31-8
32-1	URIType PL/SQL Methods	32-5
32-2	URIFACTORY PL/SQL Methods	32-23
32-3	DBUriServlet: Optional Arguments	32-31
33-1	Web Service Mapping Between XML and Oracle Database Data Types	33-7
34-1	DBMS_XDB_ADMIN Management Procedures	34-17
C-1	XMLIndex Path Table	C-2
C-2	Tasks Involving XMLIndex Indexes with an Unstructured Component	C-2
C-3	Index Synchronization	C-16
D-1	Migrating Oracle-Specific XML Queries to XQuery Full Text	D-6

Preface

This manual describes Oracle XML DB, and how you can use it to store, generate, manipulate, manage, and query XML data in the database.

After introducing you to the heart of Oracle XML DB, namely the `XMLType` framework and Oracle XML DB Repository, the manual provides a brief introduction to design criteria to consider when planning your Oracle XML DB application. It provides examples of how and where you can use Oracle XML DB.

The manual then describes ways you can store and retrieve XML data using Oracle XML DB, APIs for manipulating `XMLType` data, and ways you can view, generate, transform, and search on existing XML data. The remainder of the manual discusses how to use Oracle XML DB Repository, including versioning and security, how to access and manipulate repository resources using protocols, SQL, PL/SQL, or Java, and how to manage your Oracle XML DB application using Oracle Enterprise Manager. It also introduces you to XML messaging and Oracle Database Advanced Queuing `XMLType` support.

- [Audience](#)

Oracle XML DB Developer's Guide is intended for developers building XML Oracle Database applications.

- [Documentation Accessibility](#)

- [Diversity and Inclusion](#)

- [Related Documents](#)

- [Conventions](#)

- [Code Examples](#)

The code examples in this book are for illustration only. In many cases, however, you can copy and paste parts of examples and run them in your environment.

- [Syntax Descriptions](#)

Syntax descriptions are provided for various SQL, PL/SQL, or other command-line constructs in graphic form or Backus Naur Form (BNF).

Audience

Oracle XML DB Developer's Guide is intended for developers building XML Oracle Database applications.

An understanding of XML, XML Schema, XQuery, XPath, and XSL is helpful when using this manual.

Many examples provided here are in SQL, PL/SQL, Java, or C. A working knowledge of one of these languages is presumed.

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.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

For more information, see these Oracle resources:

- *Oracle Database XML Java API Reference*
- *Oracle XML Developer's Kit Programmer's Guide*
- *Oracle Database Error Messages*
- *Oracle Text Application Developer's Guide*
- *Oracle Text Reference*
- *Oracle Database Concepts*
- *Oracle Database Java Developer's Guide*
- *Oracle Database Development Guide*
- *Oracle Database Advanced Queuing User's Guide*
- *Oracle Database PL/SQL Packages and Types Reference*

Many of the examples in this book use the Oracle Database sample schemas. Refer to *Oracle Database Sample Schemas* for information about how these database schemas were created and how you can use them yourself.

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 OTN Registration

For additional information, see:

- Extensible Markup Language (XML) 1.0
- XML Schema and XML Schema resources
- XML Schema Part 0: Primer
- XML Schema Part 1: Structures
- XML Schema Part 2: Datatypes

- XML Schemas reference list
- XML and MIME Media-Types
- XML Pointer Language (XPointer)
- [XML Path Language \(XPath\) Version 1.0](#)
- XML Path Language (XPath) 2.0
- XPath Tutorial
- Unicode in XML and other Markup Languages, Unicode Technical Report #20
- Namespaces in XML 1.0
- XML Information Set
- Document Object Model (DOM)
- [XSL Transformations \(XSLT\) Version 1.0](#)
- Extensible Stylesheet Language (XSL) Version 1.1
- XSL references
- XSLT Tutorial
- Web Services Activity
- FTP Protocol Specification, IETF RFC959
- *ISO/IEC 13249-2:2000, Information technology - Database languages - SQL Multimedia and Application Packages - Part 2: Full-Text, International Organization For Standardization, 2000*

**Note:**

Throughout this manual, **XML Schema** refers to the XML Schema 1.0 recommendation.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	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.

Code Examples

The code examples in this book are for illustration only. In many cases, however, you can copy and paste parts of examples and run them in your environment.

- [Standard Database Schemas](#)
Many of the examples in this book use the standard database schemas that are included in your database. In particular, database schema `OE` contains XML purchase-order documents in XMLType table `purchaseorder`, and XML documents with warehouse information in XMLType column `warehouse_spec` of table `warehouses`.
- [Pretty Printing of XML Data](#)
To promote readability, especially of lengthy or complex XML data, output is sometimes shown pretty-printed (formatted) in code examples.
- [Execution Plans](#)
Some of the code examples in this book present execution plans. These are for illustration only. Running examples that are presented here in your environment is likely to result in different execution plans from those presented here.
- [Reminder About Case Sensitivity](#)
SQL is case-insensitive. XML is case-sensitive. Keep this in mind when you mix SQL and XML.

Standard Database Schemas

Many of the examples in this book use the standard database schemas that are included in your database. In particular, database schema `OE` contains XML purchase-order documents in XMLType table `purchaseorder`, and XML documents with warehouse information in XMLType column `warehouse_spec` of table `warehouses`.

The purchase-order documents are also contained in Oracle XML DB Repository, under the repository path `/home/OE/PurchaseOrders/2002/`. The XML schema that governs these documents is file `purchaseorder.xsd`, at repository location `/home/OE/purchaseorder.xsd`. An XSLT stylesheet that is used in some examples to transform purchase-order documents is file `purchaseorder.xsl`, at repository location `/home/OE/purchaseorder.xsl`. This XML schema and stylesheet can also be found in [Oracle-Supplied XML Schemas and Examples](#).



See Also:

- *Oracle Database Sample Schemas* for information about database schema `HR`
- *Oracle Database Sample Schemas* for information about database schema `OE`

Pretty Printing of XML Data

To promote readability, especially of lengthy or complex XML data, output is sometimes shown pretty-printed (formatted) in code examples.

Execution Plans

Some of the code examples in this book present execution plans. These are for illustration only. Running examples that are presented here in your environment is likely to result in different execution plans from those presented here.

Reminder About Case Sensitivity

SQL is case-insensitive. XML is case-sensitive. Keep this in mind when you mix SQL and XML.

When examining the examples in this book, keep in mind the following:

- SQL is case-insensitive, but names in SQL code are implicitly uppercase, unless you enclose them in double quotation marks (").
- XML is case-sensitive. You must refer to SQL names in XML code using the correct case: uppercase SQL names must be written as uppercase.

For example, if you create a table named `my_table` in SQL without using double quotation marks, then you must refer to it in XML code as `"MY_TABLE"`.

Syntax Descriptions

Syntax descriptions are provided for various SQL, PL/SQL, or other command-line constructs in graphic form or Backus Naur Form (BNF).

See *Oracle Database SQL Language Reference* for information about how to interpret these descriptions.