

Contents

Preface

| | |
|-----------------------------|-------|
| Audience | xlii |
| Documentation Accessibility | xlii |
| Diversity and Inclusion | xliii |
| Related Documentation | xliii |
| Syntax Diagrams | xliii |
| Conventions | xliii |

Part I Oracle Data Pump

1 Overview of Oracle Data Pump

| | | |
|-------|--|------|
| 1.1 | Oracle Data Pump Components | 1-2 |
| 1.2 | How Does Oracle Data Pump Move Data? | 1-4 |
| 1.2.1 | Using Data File Copying to Move Data | 1-4 |
| 1.2.2 | Using Direct Path to Move Data | 1-5 |
| 1.2.3 | Using External Tables to Move Data | 1-7 |
| 1.2.4 | Using Conventional Path to Move Data | 1-7 |
| 1.2.5 | Using Network Link Import to Move Data | 1-8 |
| 1.2.6 | Using a Parameter File (Parfile) with Oracle Data Pump | 1-9 |
| 1.3 | Using Oracle Data Pump With CDBs | 1-9 |
| 1.3.1 | About Using Oracle Data Pump in a Multitenant Environment | 1-10 |
| 1.3.2 | Using Oracle Data Pump to Move Data Into a CDB | 1-10 |
| 1.3.3 | Using Oracle Data Pump to Move PDBs Within or Between CDBs | 1-13 |
| 1.4 | Cloud Premigration Advisor Tool | 1-13 |
| 1.4.1 | What is the Cloud Premigration Advisor Tool (CPAT) | 1-13 |
| 1.5 | Required Roles for Oracle Data Pump Export and Import Operations | 1-14 |
| 1.6 | What Happens During the Processing of an Oracle Data Pump Job? | 1-15 |
| 1.6.1 | Coordination of an Oracle Data Pump Job | 1-16 |
| 1.6.2 | Tracking Progress Within an Oracle Data Pump Job | 1-16 |
| 1.6.3 | Filtering Data and Metadata During an Oracle Data Pump Job | 1-17 |
| 1.6.4 | Transforming Metadata During an Oracle Data Pump Job | 1-17 |
| 1.6.5 | Maximizing Job Performance of Oracle Data Pump | 1-17 |

| | | |
|----------|---|------|
| 1.6.6 | Loading and Unloading Data with Oracle Data Pump | 1-18 |
| 1.7 | How to Monitor Status of Oracle Data Pump Jobs | 1-19 |
| 1.8 | How to Monitor the Progress of Running Jobs with V\$SESSION_LONGOPS | 1-19 |
| 1.9 | File Allocation with Oracle Data Pump | 1-20 |
| 1.9.1 | Understanding File Allocation in Oracle Data Pump | 1-20 |
| 1.9.2 | Specifying Files and Adding Additional Dump Files | 1-20 |
| 1.9.3 | Default Locations for Dump, Log, and SQL Files | 1-21 |
| 1.9.3.1 | Understanding Dump, Log, and SQL File Default Locations | 1-21 |
| 1.9.3.2 | Understanding How to Use Oracle Data Pump with Oracle RAC | 1-23 |
| 1.9.3.3 | Using Directory Objects When Oracle Automatic Storage Management Is Enabled | 1-24 |
| 1.9.3.4 | The DATA_PUMP_DIR Directory Object and Pluggable Databases | 1-24 |
| 1.9.4 | Using Substitution Variables with Oracle Data Pump Exports | 1-25 |
| 1.10 | Exporting and Importing Between Different Oracle Database Releases | 1-25 |
| 1.11 | Exporting and Importing Blockchain Tables with Oracle Data Pump | 1-27 |
| 1.12 | Unload and Load Vectors Using Oracle Data Pump | 1-28 |
| 1.13 | Managing SecureFiles Large Object Exports with Oracle Data Pump | 1-29 |
| 1.14 | Oracle Data Pump Process Exit Codes | 1-30 |
| 1.15 | How Oracle Data Pump Manages Dump File Blocks | 1-30 |
| 1.15.1 | Dump Files for Exports | 1-31 |
| 1.15.2 | Trailer Block File Layout in Dump Files | 1-31 |
| 1.15.3 | Header Block File Layout in Dump Files | 1-33 |
| 1.15.4 | Types of Dump File Trailer Blocks | 1-33 |
| 1.16 | How to Monitor Oracle Data Pump Jobs with Unified Auditing | 1-34 |
| 1.17 | Encrypted Data Security Warnings for Oracle Data Pump Operations | 1-35 |
| 1.18 | How Does Oracle Data Pump Handle Timestamp Data? | 1-35 |
| 1.18.1 | TIMESTAMP WITH TIMEZONE Restrictions | 1-35 |
| 1.18.1.1 | Understanding TIMESTAMP WITH TIME ZONE Restrictions | 1-36 |
| 1.18.1.2 | Oracle Data Pump Support for TIMESTAMP WITH TIME ZONE Data | 1-36 |
| 1.18.1.3 | Time Zone File Versions on the Source and Target | 1-37 |
| 1.18.2 | TIMESTAMP WITH LOCAL TIME ZONE Restrictions | 1-38 |
| 1.19 | Character Set and Globalization Support Considerations | 1-38 |
| 1.19.1 | Data Definition Language (DDL) | 1-38 |
| 1.19.2 | Single-Byte Character Sets and Export and Import | 1-38 |
| 1.19.3 | Multibyte Character Sets and Export and Import | 1-38 |
| 1.20 | Oracle Data Pump Behavior with Data-Bound Collation | 1-39 |

2 Oracle Data Pump Export

| | | |
|-------|------------------------------------|-----|
| 2.1 | What Is Oracle Data Pump Export? | 2-1 |
| 2.2 | Starting Oracle Data Pump Export | 2-2 |
| 2.2.1 | Oracle Data Pump Export Interfaces | 2-2 |

| | | |
|---------|--|------|
| 2.2.2 | Oracle Data Pump Export Modes | 2-3 |
| 2.2.2.1 | Full Export Mode | 2-3 |
| 2.2.2.2 | Schema Mode | 2-5 |
| 2.2.2.3 | Table Mode | 2-5 |
| 2.2.2.4 | Tablespace Mode | 2-6 |
| 2.2.2.5 | Transportable Tablespace Mode | 2-6 |
| 2.2.3 | Network Considerations for Oracle Data Pump Export | 2-7 |
| 2.3 | Filtering During Export Operations | 2-8 |
| 2.3.1 | Oracle Data Pump Export Data Filters | 2-8 |
| 2.3.2 | Oracle Data Pump Metadata Filters | 2-9 |
| 2.4 | Parameters Available in Data Pump Export Command-Line Mode | 2-10 |
| 2.4.1 | About Oracle Data Pump Export Parameters | 2-14 |
| 2.4.2 | ABORT_STEP | 2-16 |
| 2.4.3 | ACCESS_METHOD | 2-16 |
| 2.4.4 | ATTACH | 2-17 |
| 2.4.5 | CHECKSUM | 2-18 |
| 2.4.6 | CHECKSUM_ALGORITHM | 2-19 |
| 2.4.7 | CLUSTER | 2-20 |
| 2.4.8 | COMPRESSION | 2-21 |
| 2.4.9 | COMPRESSION_ALGORITHM | 2-22 |
| 2.4.10 | CONTENT | 2-23 |
| 2.4.11 | CREDENTIAL | 2-24 |
| 2.4.12 | DATA_OPTIONS | 2-25 |
| 2.4.13 | DIRECTORY | 2-26 |
| 2.4.14 | DUMPFIL | 2-27 |
| 2.4.15 | ENABLE_SECURE_ROLES | 2-30 |
| 2.4.16 | ENCRYPTION | 2-30 |
| 2.4.17 | ENCRYPTION_ALGORITHM | 2-32 |
| 2.4.18 | ENCRYPTION_MODE | 2-33 |
| 2.4.19 | ENCRYPTION_PASSWORD | 2-34 |
| 2.4.20 | ENCRYPTION_PWD_PROMPT | 2-36 |
| 2.4.21 | ESTIMATE | 2-37 |
| 2.4.22 | ESTIMATE_ONLY | 2-38 |
| 2.4.23 | EXCLUDE | 2-38 |
| 2.4.24 | FILESIZE | 2-40 |
| 2.4.25 | FLASHBACK_SCN | 2-41 |
| 2.4.26 | FLASHBACK_TIME | 2-42 |
| 2.4.27 | FULL | 2-44 |
| 2.4.28 | HELP | 2-45 |
| 2.4.29 | INCLUDE | 2-45 |
| 2.4.30 | JOB_NAME | 2-48 |
| 2.4.31 | KEEP_MASTER | 2-48 |

| | | |
|--------|---|------|
| 2.4.32 | LOGFILE | 2-49 |
| 2.4.33 | LOGTIME | 2-50 |
| 2.4.34 | METRICS | 2-52 |
| 2.4.35 | NETWORK_LINK | 2-53 |
| 2.4.36 | NOLOGFILE | 2-54 |
| 2.4.37 | PARALLEL | 2-55 |
| 2.4.38 | PARALLEL_THRESHOLD | 2-56 |
| 2.4.39 | PARFILE | 2-58 |
| 2.4.40 | QUERY | 2-59 |
| 2.4.41 | REMAP_DATA | 2-61 |
| 2.4.42 | REUSE_DUMPFILES | 2-62 |
| 2.4.43 | SAMPLE | 2-63 |
| 2.4.44 | SCHEMAS | 2-64 |
| 2.4.45 | SERVICE_NAME | 2-64 |
| 2.4.46 | SOURCE_EDITION | 2-66 |
| 2.4.47 | STATUS | 2-66 |
| 2.4.48 | TABLES | 2-67 |
| 2.4.49 | TABLESPACES | 2-70 |
| 2.4.50 | TRANSPORT_DATAFILES_LOG | 2-70 |
| 2.4.51 | TRANSPORT_FULL_CHECK | 2-72 |
| 2.4.52 | TRANSPORT_TABLESPACES | 2-73 |
| 2.4.53 | TRANSPORTABLE | 2-74 |
| 2.4.54 | TTS_CLOSURE_CHECK | 2-76 |
| 2.4.55 | VERSION | 2-77 |
| 2.4.56 | VIEWS_AS_TABLES | 2-78 |
| 2.5 | Commands Available in Data Pump Export Interactive-Command Mode | 2-80 |
| 2.5.1 | About Oracle Data Pump Export Interactive Command Mode | 2-81 |
| 2.5.2 | ADD_FILE | 2-82 |
| 2.5.3 | CONTINUE_CLIENT | 2-83 |
| 2.5.4 | EXIT_CLIENT | 2-83 |
| 2.5.5 | FILESIZE | 2-83 |
| 2.5.6 | HELP | 2-84 |
| 2.5.7 | KILL_JOB | 2-84 |
| 2.5.8 | PARALLEL | 2-85 |
| 2.5.9 | START_JOB | 2-86 |
| 2.5.10 | STATUS | 2-86 |
| 2.5.11 | STOP_JOB | 2-87 |
| 2.6 | Examples of Using Oracle Data Pump Export | 2-87 |
| 2.6.1 | Performing a Table-Mode Export | 2-88 |
| 2.6.2 | Data-Only Unload of Selected Tables and Rows | 2-88 |
| 2.6.3 | Estimating Disk Space Needed in a Table-Mode Export | 2-88 |
| 2.6.4 | Performing a Schema-Mode Export | 2-89 |

| | | |
|-------|---|------|
| 2.6.5 | Performing a Parallel Full Database Export | 2-89 |
| 2.6.6 | Using Interactive Mode to Stop and Reattach to a Job | 2-89 |
| 2.6.7 | Continuing Table Loads when LOB Data Type Corruptions are Found | 2-90 |
| 2.7 | Syntax Diagrams for Oracle Data Pump Export | 2-92 |

3 Oracle Data Pump Import

| | | |
|---------|---|------|
| 3.1 | What Is Oracle Data Pump Import? | 3-1 |
| 3.2 | Starting Oracle Data Pump Import | 3-1 |
| 3.2.1 | Oracle Data Pump Import Interfaces | 3-2 |
| 3.2.2 | Oracle Data Pump Import Modes | 3-3 |
| 3.2.2.1 | About Oracle Data Pump Import Modes | 3-3 |
| 3.2.2.2 | Full Import Mode | 3-3 |
| 3.2.2.3 | Schema Mode | 3-5 |
| 3.2.2.4 | Table Mode | 3-5 |
| 3.2.2.5 | Tablespace Mode | 3-5 |
| 3.2.2.6 | Transportable Tablespace Mode | 3-6 |
| 3.2.3 | Network Considerations for Oracle Data Pump Import | 3-6 |
| 3.3 | Filtering During Import Operations | 3-7 |
| 3.3.1 | Oracle Data Pump Import Data Filters | 3-7 |
| 3.3.2 | Oracle Data Pump Import Metadata Filters | 3-8 |
| 3.4 | Parameters Available in Oracle Data Pump Import Command-Line Mode | 3-8 |
| 3.4.1 | About Import Command-Line Mode | 3-13 |
| 3.4.2 | ABORT_STEP | 3-15 |
| 3.4.3 | ACCESS_METHOD | 3-15 |
| 3.4.4 | ATTACH | 3-17 |
| 3.4.5 | CLUSTER | 3-18 |
| 3.4.6 | CONTENT | 3-19 |
| 3.4.7 | CREDENTIAL | 3-20 |
| 3.4.8 | DATA_OPTIONS | 3-22 |
| 3.4.9 | DIRECTORY | 3-24 |
| 3.4.10 | DUMPFIL | 3-25 |
| 3.4.11 | ENABLE_SECURE_ROLES | 3-28 |
| 3.4.12 | ENCRYPTION_PASSWORD | 3-28 |
| 3.4.13 | ENCRYPTION_PWD_PROMPT | 3-29 |
| 3.4.14 | ESTIMATE | 3-31 |
| 3.4.15 | EXCLUDE | 3-32 |
| 3.4.16 | FLASHBACK_SCN | 3-34 |
| 3.4.17 | FLASHBACK_TIME | 3-35 |
| 3.4.18 | FULL | 3-36 |
| 3.4.19 | HELP | 3-38 |
| 3.4.20 | INCLUDE | 3-38 |

| | | |
|--------|----------------------------------|------|
| 3.4.21 | INDEX_THRESHOLD | 3-40 |
| 3.4.22 | JOB_NAME | 3-41 |
| 3.4.23 | KEEP_MASTER | 3-42 |
| 3.4.24 | LOGFILE | 3-43 |
| 3.4.25 | LOGTIME | 3-44 |
| 3.4.26 | MASTER_ONLY | 3-45 |
| 3.4.27 | METRICS | 3-45 |
| 3.4.28 | NETWORK_LINK | 3-46 |
| 3.4.29 | NOLOGFILE | 3-48 |
| 3.4.30 | ONESTEP_INDEX | 3-48 |
| 3.4.31 | PARALLEL | 3-49 |
| 3.4.32 | PARALLEL_THRESHOLD | 3-52 |
| 3.4.33 | PARFILE | 3-53 |
| 3.4.34 | PARTITION_OPTIONS | 3-54 |
| 3.4.35 | QUERY | 3-56 |
| 3.4.36 | REMAP_DATA | 3-58 |
| 3.4.37 | REMAP_DATAFILE | 3-59 |
| 3.4.38 | REMAP_DIRECTORY | 3-60 |
| 3.4.39 | REMAP_SCHEMA | 3-61 |
| 3.4.40 | REMAP_TABLE | 3-63 |
| 3.4.41 | REMAP_TABLESPACE | 3-64 |
| 3.4.42 | SCHEMAS | 3-65 |
| 3.4.43 | SERVICE_NAME | 3-66 |
| 3.4.44 | SKIP_UNUSABLE_INDEXES | 3-67 |
| 3.4.45 | SOURCE_EDITION | 3-68 |
| 3.4.46 | SQLFILE | 3-69 |
| 3.4.47 | STATUS | 3-71 |
| 3.4.48 | STREAMS_CONFIGURATION | 3-71 |
| 3.4.49 | TABLE_EXISTS_ACTION | 3-72 |
| 3.4.50 | REUSE_DATAFILES | 3-74 |
| 3.4.51 | TABLES | 3-74 |
| 3.4.52 | TABLESPACES | 3-77 |
| 3.4.53 | TARGET_EDITION | 3-78 |
| 3.4.54 | TRANSFORM | 3-79 |
| 3.4.55 | TRANSPORT_DATAFILES | 3-87 |
| 3.4.56 | TRANSPORT_FULL_CHECK | 3-89 |
| 3.4.57 | TRANSPORT_TABLESPACES | 3-90 |
| 3.4.58 | TRANSPORTABLE | 3-92 |
| 3.4.59 | VERIFY_CHECKSUM | 3-94 |
| 3.4.60 | VERIFY_ONLY | 3-95 |
| 3.4.61 | VERSION | 3-96 |
| 3.4.62 | VIEWS_AS_TABLES (Network Import) | 3-97 |

| | | |
|-------|--|-------|
| 3.5 | Commands Available in Data Pump Export Interactive-Command Mode | 3-99 |
| 3.5.1 | Commands Available in Oracle Data Pump Import Interactive-Command Mode | 3-106 |
| 3.5.2 | CONTINUE_CLIENT | 3-108 |
| 3.5.3 | EXIT_CLIENT | 3-108 |
| 3.5.4 | HELP | 3-109 |
| 3.5.5 | KILL_JOB | 3-109 |
| 3.5.6 | PARALLEL | 3-110 |
| 3.5.7 | START_JOB | 3-110 |
| 3.5.8 | STATUS | 3-111 |
| 3.5.9 | STOP_JOB | 3-112 |
| 3.6 | Examples of Using Oracle Data Pump Import | 3-112 |
| 3.6.1 | Performing a Data-Only Table-Mode Import | 3-113 |
| 3.6.2 | Performing a Schema-Mode Import | 3-113 |
| 3.6.3 | Performing a Network-Mode Import | 3-113 |
| 3.6.4 | Using Wildcards in URL-Based Dumpfile Names | 3-114 |
| 3.7 | Syntax Diagrams for Oracle Data Pump Import | 3-114 |

4 Oracle Data Pump Legacy Mode

| | | |
|-------|--|------|
| 4.1 | Oracle Data Pump Legacy Mode Use Cases | 4-1 |
| 4.2 | Parameter Mappings | 4-1 |
| 4.2.1 | Using Original Export Parameters with Oracle Data Pump | 4-2 |
| 4.2.2 | Using Original Import Parameters with Oracle Data Pump | 4-5 |
| 4.3 | Management of File Locations in Oracle Data Pump Legacy Mode | 4-9 |
| 4.4 | Adjusting Existing Scripts for Oracle Data Pump Log Files and Errors | 4-11 |
| 4.4.1 | Log Files | 4-12 |
| 4.4.2 | Error Cases | 4-12 |
| 4.4.3 | Exit Status | 4-12 |

5 Oracle Data Pump Performance

| | | |
|-------|--|-----|
| 5.1 | Data Performance Improvements for Oracle Data Pump Export and Import | 5-1 |
| 5.2 | Tuning Performance | 5-2 |
| 5.2.1 | How To Manage Oracle Data Pump Resource Consumption | 5-2 |
| 5.2.2 | Effect of Compression and Encryption on Performance | 5-3 |
| 5.2.3 | Memory Considerations When Exporting and Importing Statistics | 5-3 |
| 5.3 | Initialization Parameters That Affect Oracle Data Pump Performance | 5-3 |
| 5.3.1 | Performance Guidelines for Oracle Data Pump Parameters | 5-4 |
| 5.3.2 | Setting the Size Of the Buffer Cache In a GoldenGate Replication Environment | 5-4 |
| 5.3.3 | Managing Resource Usage for Multiple User Oracle Data Pump Jobs | 5-4 |

6 Using the Oracle Data Pump API

| | | |
|-------|--|------|
| 6.1 | How Does the Oracle Data Pump Client Interface API Work? | 6-1 |
| 6.2 | DBMS_DATAPUMP Job States | 6-1 |
| 6.3 | What Are the Basic Steps in Using the Oracle Data Pump API? | 6-4 |
| 6.4 | Examples of Using the Oracle Data Pump API | 6-4 |
| 6.4.1 | Using the Oracle Data Pump API Examples with Your Database | 6-5 |
| 6.4.2 | Performing a Simple Schema Export with Oracle Data Pump | 6-5 |
| 6.4.3 | Performing a Table Mode Export to Object Store with Oracle Data Pump | 6-7 |
| 6.4.4 | Importing a Dump File and Remapping All Schema Objects | 6-11 |
| 6.4.5 | Importing a Table from an Object Store Using Oracle Data Pump | 6-13 |
| 6.4.6 | Using Exception Handling During a Simple Schema Export | 6-16 |
| 6.4.7 | Displaying Dump File Information for Oracle Data Pump Jobs | 6-19 |
| 6.4.8 | Network Mode and Schema Mode Import Over a Network Link | 6-22 |

Part II SQL*Loader

7 Understanding How to Use SQL*Loader

| | | |
|---------|--|------|
| 7.1 | SQL*Loader Features | 7-1 |
| 7.2 | SQL*Loader Parameters | 7-3 |
| 7.3 | SQL*Loader Control File | 7-4 |
| 7.4 | Input Data and Data Fields in SQL*Loader | 7-4 |
| 7.4.1 | How SQL*Loader Reads Input Data and Data Files | 7-5 |
| 7.4.2 | Fixed Record Format | 7-5 |
| 7.4.3 | Variable Record Format and SQL*Loader | 7-6 |
| 7.4.4 | Stream Record Format and SQL*Loader | 7-7 |
| 7.4.5 | Logical Records and SQL*Loader | 7-8 |
| 7.4.6 | Data Field Setting and SQL*Loader | 7-9 |
| 7.5 | LOBFILES and Secondary Data Files (SDFs) | 7-9 |
| 7.6 | Data Conversion and Data Type Specification | 7-10 |
| 7.7 | SQL*Loader Discarded and Rejected Records | 7-11 |
| 7.7.1 | The SQL*Loader Bad File | 7-11 |
| 7.7.1.1 | Records Rejected by SQL*Loader | 7-11 |
| 7.7.1.2 | Records Rejected by Oracle Database During a SQL*Loader Operation | 7-11 |
| 7.7.2 | The SQL*Loader Discard File | 7-11 |
| 7.8 | Log File and Logging Information | 7-12 |
| 7.9 | Conventional Path Loads, Direct Path Loads, and External Table Loads | 7-12 |
| 7.9.1 | Conventional Path Loads | 7-12 |
| 7.9.2 | Direct Path Loads | 7-13 |
| 7.9.3 | Parallel Direct Path | 7-13 |
| 7.9.4 | External Table Loads | 7-13 |

| | | |
|----------|---|------|
| 7.9.5 | Choosing External Tables Versus SQL*Loader | 7-14 |
| 7.9.6 | Behavior Differences Between SQL*Loader and External Tables | 7-15 |
| 7.9.6.1 | Multiple Primary Input Data Files | 7-15 |
| 7.9.6.2 | Syntax and Data Types | 7-15 |
| 7.9.6.3 | Byte-Order Marks | 7-16 |
| 7.9.6.4 | Default Character Sets, Date Masks, and Decimal Separator | 7-16 |
| 7.9.6.5 | Use of the Backslash Escape Character | 7-16 |
| 7.9.7 | Loading Tables Using Data Stored into Object Storage | 7-16 |
| 7.10 | Loading Objects, Collections, and LOBs with SQL*Loader | 7-18 |
| 7.10.1 | Supported Object Types | 7-18 |
| 7.10.1.1 | column objects | 7-19 |
| 7.10.1.2 | row objects | 7-19 |
| 7.10.2 | Supported Collection Types | 7-19 |
| 7.10.2.1 | Nested Tables | 7-19 |
| 7.10.2.2 | VARRAYs | 7-20 |
| 7.10.3 | SODA Collections and SQL*Loader | 7-20 |
| 7.10.4 | Supported LOB Data Types | 7-21 |
| 7.11 | Partitioned Object Support in SQL*Loader | 7-21 |
| 7.12 | Application Development: Direct Path Load API | 7-21 |
| 7.13 | SQL*Loader Case Studies | 7-22 |
| 7.13.1 | How to Access and Use the Oracle SQL*Loader Case Studies | 7-22 |
| 7.13.2 | Case Study Files | 7-23 |
| 7.13.3 | Running the Case Studies | 7-24 |
| 7.13.4 | Case Study Log Files | 7-24 |
| 7.13.5 | Checking the Results of a Case Study | 7-24 |

8 SQL*Loader Command-Line Reference

| | | |
|-------|---|------|
| 8.1 | Starting SQL*Loader | 8-1 |
| 8.1.1 | Specifying Parameters on the Command Line | 8-1 |
| 8.1.2 | Alternative Ways to Specify SQL*Loader Parameters | 8-2 |
| 8.1.3 | Using SQL*Loader to Load Data Across a Network | 8-3 |
| 8.2 | Command-Line Parameters for SQL*Loader | 8-3 |
| 8.2.1 | BAD | 8-7 |
| 8.2.2 | BINDSIZE | 8-8 |
| 8.2.3 | COLUMNARRAYROWS | 8-9 |
| 8.2.4 | COMPRESS_STREAM | 8-9 |
| 8.2.5 | CONTROL | 8-10 |
| 8.2.6 | CREDENTIAL | 8-11 |
| 8.2.7 | DATA | 8-13 |
| 8.2.8 | DATE_CACHE | 8-14 |
| 8.2.9 | DEFAULTS | 8-15 |

| | | |
|--------|---------------------------------------|------|
| 8.2.10 | DEGREE_OF_PARALLELISM | 8-17 |
| 8.2.11 | DIRECT | 8-18 |
| 8.2.12 | DIRECT_PATH_LOCK_WAIT | 8-18 |
| 8.2.13 | DISCARD | 8-19 |
| 8.2.14 | DISCARDMAX | 8-20 |
| 8.2.15 | DNFS_ENABLE | 8-21 |
| 8.2.16 | DNFS_READBUFFERS | 8-22 |
| 8.2.17 | EMPTY_LOBS_ARE_NULL | 8-22 |
| 8.2.18 | ERRORS | 8-23 |
| 8.2.19 | EXTERNAL_TABLE | 8-24 |
| 8.2.20 | FILE | 8-26 |
| 8.2.21 | GRANULE_SIZE | 8-27 |
| 8.2.22 | GSM_HOST | 8-27 |
| 8.2.23 | GSM_NAME | 8-28 |
| 8.2.24 | GSM_PORT | 8-29 |
| 8.2.25 | HELP | 8-29 |
| 8.2.26 | LOAD | 8-30 |
| 8.2.27 | LOAD_SHARDS | 8-30 |
| 8.2.28 | LOG | 8-31 |
| 8.2.29 | MULTITHREADING | 8-32 |
| 8.2.30 | NO_INDEX_ERRORS | 8-33 |
| 8.2.31 | OPTIMIZE PARALLEL | 8-33 |
| 8.2.32 | PARALLEL | 8-34 |
| 8.2.33 | PARFILE | 8-35 |
| 8.2.34 | PARTITION_MEMORY | 8-35 |
| 8.2.35 | READER_COUNT | 8-36 |
| 8.2.36 | READSIZE | 8-37 |
| 8.2.37 | RESUMABLE | 8-38 |
| 8.2.38 | RESUMABLE_NAME | 8-38 |
| 8.2.39 | RESUMABLE_TIMEOUT | 8-39 |
| 8.2.40 | ROWS | 8-40 |
| 8.2.41 | SDF_PREFIX | 8-41 |
| 8.2.42 | SILENT | 8-42 |
| 8.2.43 | SKIP | 8-43 |
| 8.2.44 | SKIP_INDEX_MAINTENANCE | 8-43 |
| 8.2.45 | SKIP_UNUSABLE_INDEXES | 8-44 |
| 8.2.46 | STREAMSIZE | 8-45 |
| 8.2.47 | TRIM | 8-46 |
| 8.2.48 | USERID | 8-47 |
| 8.3 | Exit Codes for Inspection and Display | 8-48 |

9 SQL*Loader Control File Reference

| | | |
|---------|--|------|
| 9.1 | Control File Contents | 9-2 |
| 9.2 | Comments in the Control File | 9-4 |
| 9.3 | Specifying Command-Line Parameters in the Control File | 9-4 |
| 9.3.1 | OPTIONS Clause for Schema Data | 9-5 |
| 9.3.2 | OPTIONS Clause for SODA Collections | 9-6 |
| 9.3.3 | Specifying the Number of Default Expressions to Be Evaluated At One Time | 9-7 |
| 9.4 | Specifying File Names and Object Names | 9-7 |
| 9.4.1 | File Names That Conflict with SQL and SQL*Loader Reserved Words | 9-7 |
| 9.4.2 | Specifying SQL Strings in the SQL*Loader Control File | 9-7 |
| 9.4.3 | Operating Systems and SQL Loader Control File Characters | 9-8 |
| 9.4.3.1 | Specifying a Complete Path | 9-8 |
| 9.4.3.2 | Backslash Escape Character | 9-8 |
| 9.4.3.3 | Nonportable Strings | 9-9 |
| 9.4.3.4 | Using the Backslash as an Escape Character | 9-9 |
| 9.4.3.5 | Escape Character Is Sometimes Disallowed | 9-9 |
| 9.5 | Identifying XMLType Tables | 9-9 |
| 9.6 | Specifying Field Order | 9-12 |
| 9.7 | Specifying Data Files | 9-12 |
| 9.7.1 | Understanding How to Specify Data Files | 9-13 |
| 9.7.2 | Examples of INFILE Syntax | 9-14 |
| 9.7.3 | Specifying Multiple Data Files | 9-15 |
| 9.8 | Specifying CSV Format Files | 9-15 |
| 9.9 | Loading VECTOR Columns from Character Data and fvec Format Files | 9-16 |
| 9.10 | Identifying Data in the Control File with BEGINDATA | 9-18 |
| 9.11 | Specifying Data File Format and Buffering | 9-18 |
| 9.12 | Specifying the Bad File | 9-19 |
| 9.12.1 | Understanding and Specifying the Bad File | 9-19 |
| 9.12.2 | Examples of Specifying a Bad File Name | 9-20 |
| 9.12.3 | How Bad Files Are Handled with LOBFILES and SDFs | 9-20 |
| 9.12.4 | Criteria for Rejected Records | 9-21 |
| 9.13 | Specifying the Discard File | 9-21 |
| 9.13.1 | Understanding and Specifying the Discard File | 9-22 |
| 9.13.2 | Specifying the Discard File in the Control File | 9-23 |
| 9.13.3 | Limiting the Number of Discard Records | 9-23 |
| 9.13.4 | Examples of Specifying a Discard File Name | 9-23 |
| 9.13.5 | Criteria for Discarded Records | 9-24 |
| 9.13.6 | How Discard Files Are Handled with LOBFILES and SDFs | 9-24 |
| 9.13.7 | Specifying the Discard File from the Command Line | 9-24 |
| 9.14 | Specifying a NULLIF Clause At the Table Level | 9-24 |
| 9.15 | Specifying Datetime Formats At the Table Level | 9-25 |

| | | |
|----------|---|------|
| 9.16 | Handling Different Character Encoding Schemes | 9-26 |
| 9.16.1 | Multibyte (Asian) Character Sets | 9-26 |
| 9.16.2 | Unicode Character Sets | 9-26 |
| 9.16.3 | Database Character Sets | 9-27 |
| 9.16.4 | Data File Character Sets | 9-28 |
| 9.16.5 | Input Character Conversion with SQL*Loader | 9-28 |
| 9.16.5.1 | Options for Converting Character Sets Using SQL*Loader | 9-28 |
| 9.16.5.2 | Considerations When Loading Data into VARRAYs or Primary-Key-Based REFs | 9-29 |
| 9.16.5.3 | CHARACTERSET Parameter | 9-30 |
| 9.16.5.4 | Control File Character Set | 9-31 |
| 9.16.5.5 | Character-Length Semantics | 9-31 |
| 9.16.6 | Shift-sensitive Character Data | 9-33 |
| 9.17 | Interrupted SQL*Loader Loads | 9-33 |
| 9.17.1 | Understanding Causes of Interrupted SQL*Loader Loads | 9-33 |
| 9.17.2 | Discontinued Conventional Path Loads | 9-34 |
| 9.17.3 | Discontinued Direct Path Loads | 9-34 |
| 9.17.3.1 | Load Discontinued Because of Space Errors | 9-34 |
| 9.17.3.2 | Load Discontinued Because Maximum Number of Errors Exceeded | 9-35 |
| 9.17.3.3 | Load Discontinued Because of Irrecoverable Errors | 9-35 |
| 9.17.3.4 | Load Discontinued Because a Ctrl+C Was Issued | 9-35 |
| 9.17.4 | Status of Tables and Indexes After an Interrupted Load | 9-35 |
| 9.17.5 | Using the Log File to Determine Load Status | 9-36 |
| 9.17.6 | Continuing Single-Table Loads | 9-36 |
| 9.18 | Assembling Logical Records from Physical Records | 9-36 |
| 9.18.1 | Using CONCATENATE to Assemble Logical Records | 9-37 |
| 9.18.2 | Using CONTINUEIF to Assemble Logical Records | 9-37 |
| 9.19 | Loading Logical Records into Tables | 9-40 |
| 9.19.1 | Specifying Table Names | 9-41 |
| 9.19.2 | INTO TABLE Clause | 9-41 |
| 9.19.3 | Table-Specific Loading Method | 9-42 |
| 9.19.4 | Loading Data into Empty Tables with INSERT | 9-42 |
| 9.19.5 | Loading Data into Nonempty Tables | 9-43 |
| 9.19.5.1 | Options for Loading Data Into Nonempty Tables | 9-43 |
| 9.19.5.2 | APPEND | 9-44 |
| 9.19.5.3 | APPEND_PARALLEL | 9-44 |
| 9.19.5.4 | REPLACE | 9-44 |
| 9.19.5.5 | Updating Existing Rows with REPLACE | 9-45 |
| 9.19.5.6 | TRUNCATE | 9-45 |
| 9.19.6 | Table-Specific OPTIONS Parameter | 9-45 |
| 9.19.7 | Loading Records Based on a Condition | 9-45 |
| 9.19.8 | Using the WHEN Clause with LOBFILES and SDFs | 9-46 |

| | | |
|-----------|---|------|
| 9.19.9 | Specifying Default Data Delimiters | 9-46 |
| 9.19.9.1 | fields_spec | 9-47 |
| 9.19.9.2 | termination_spec | 9-47 |
| 9.19.9.3 | enclosure_spec | 9-47 |
| 9.19.10 | Handling Records with Missing Specified Fields | 9-48 |
| 9.19.10.1 | SQL*Loader Management of Short Records with Missing Data | 9-48 |
| 9.19.10.2 | TRAILING NULLCOLS Clause | 9-49 |
| 9.20 | Index Options with SQL*Loader | 9-49 |
| 9.20.1 | Understanding the SORTED INDEXES Parameter | 9-49 |
| 9.20.2 | Understanding the SINGLEROW Parameter | 9-50 |
| 9.21 | Benefits of Using Multiple INTO TABLE Clauses | 9-50 |
| 9.21.1 | Understanding the SQL*Loader INTO TABLE Clause | 9-51 |
| 9.21.2 | Distinguishing Different Input Record Formats | 9-51 |
| 9.21.3 | Relative Positioning Based on the POSITION Parameter | 9-52 |
| 9.21.4 | Distinguishing Different Input Row Object Subtypes | 9-52 |
| 9.21.5 | Loading Data into Multiple Tables | 9-54 |
| 9.21.6 | Summary of Using Multiple INTO TABLE Clauses | 9-54 |
| 9.21.7 | Extracting Multiple Logical Records | 9-54 |
| 9.21.7.1 | Example of Extracting Multiple Logical Records From a Physical Record | 9-55 |
| 9.21.7.2 | Example of Relative Positioning Based on Delimiters | 9-55 |
| 9.22 | Bind Arrays and Conventional Path Loads | 9-56 |
| 9.22.1 | Differences Between Bind Arrays and Conventional Path Loads | 9-56 |
| 9.22.2 | Size Requirements for Bind Arrays | 9-56 |
| 9.22.3 | Performance Implications of Bind Arrays | 9-57 |
| 9.22.4 | Specifying Number of Rows Versus Size of Bind Array | 9-57 |
| 9.22.5 | Setting Up SQL*Loader Bind Arrays | 9-58 |
| 9.22.5.1 | Calculations to Determine Bind Array Size | 9-58 |
| 9.22.5.2 | Determining the Size of the Length Indicator | 9-59 |
| 9.22.5.3 | Calculating the Size of Field Buffers | 9-60 |
| 9.22.6 | Minimizing Memory Requirements for Bind Arrays | 9-62 |
| 9.22.7 | Calculating Bind Array Size for Multiple INTO TABLE Clauses | 9-62 |

10 SQL*Loader Field List Reference

| | | |
|--------|---|------|
| 10.1 | Field List Contents | 10-2 |
| 10.2 | Specifying the Position of a Data Field. | 10-3 |
| 10.2.1 | POSITION | 10-3 |
| 10.2.2 | Using POSITION with Data Containing Tabs | 10-4 |
| 10.2.3 | Using POSITION with Multiple Table Loads | 10-4 |
| 10.2.4 | Examples of Using POSITION in SQL*Loader Specifications | 10-5 |
| 10.3 | Specifying Columns and Fields | 10-5 |
| 10.3.1 | Options for Column and Field Specification | 10-5 |

| | | |
|-----------|--|-------|
| 10.3.2 | Specifying Filler Fields | 10-6 |
| 10.3.3 | Specifying the Data Type of a Data Field | 10-7 |
| 10.4 | SQL*Loader Data Types | 10-8 |
| 10.4.1 | Portable and Nonportable Data Type Differences | 10-8 |
| 10.4.2 | Nonportable Data Types | 10-9 |
| 10.4.2.1 | Categories of Nonportable Data Types | 10-10 |
| 10.4.2.2 | BINARY_DOUBLE | 10-11 |
| 10.4.2.3 | BINARY_FLOAT | 10-11 |
| 10.4.2.4 | BYTEINT | 10-12 |
| 10.4.2.5 | DECIMAL | 10-12 |
| 10.4.2.6 | DOUBLE | 10-13 |
| 10.4.2.7 | FLOAT | 10-13 |
| 10.4.2.8 | INTEGER(n) | 10-14 |
| 10.4.2.9 | LONG VARRAW | 10-14 |
| 10.4.2.10 | SMALLINT | 10-15 |
| 10.4.2.11 | VARGRAPHIC | 10-16 |
| 10.4.2.12 | VARCHAR | 10-17 |
| 10.4.2.13 | VARRAW | 10-18 |
| 10.4.2.14 | ZONED | 10-18 |
| 10.4.3 | Portable Data Types | 10-19 |
| 10.4.3.1 | Categories of Portable Data Types | 10-20 |
| 10.4.3.2 | CHAR | 10-21 |
| 10.4.3.3 | Datetime and Interval | 10-21 |
| 10.4.3.4 | GRAPHIC | 10-26 |
| 10.4.3.5 | GRAPHIC EXTERNAL | 10-27 |
| 10.4.3.6 | Numeric EXTERNAL | 10-27 |
| 10.4.3.7 | RAW | 10-28 |
| 10.4.3.8 | VARCHARC | 10-29 |
| 10.4.3.9 | VARRAWC | 10-30 |
| 10.4.3.10 | Conflicting Native Data Type Field Lengths | 10-31 |
| 10.4.3.11 | Field Lengths for Length-Value Data Types | 10-31 |
| 10.4.4 | SODA Collection Data Types | 10-31 |
| 10.4.4.1 | RAW(*) | 10-32 |
| 10.4.4.2 | CONTENTFILE(soda_filename) | 10-34 |
| 10.4.5 | Data Type Conversions | 10-36 |
| 10.4.6 | Data Type Conversions for Datetime and Interval Data Types | 10-37 |
| 10.4.7 | Specifying Delimiters | 10-38 |
| 10.4.7.1 | Syntax for Termination and Enclosure Specification | 10-39 |
| 10.4.7.2 | Delimiter Marks in the Data | 10-40 |
| 10.4.7.3 | Maximum Length of Delimited Data | 10-41 |
| 10.4.7.4 | Loading Trailing Blanks with Delimiters | 10-41 |
| 10.4.8 | How Delimited Data Is Processed | 10-41 |

| | | |
|-----------|--|-------|
| 10.4.8.1 | Fields Using Only TERMINATED BY | 10-42 |
| 10.4.8.2 | Fields Using ENCLOSED BY Without TERMINATED BY | 10-42 |
| 10.4.8.3 | Fields Using ENCLOSED BY With TERMINATED BY | 10-42 |
| 10.4.8.4 | Fields Using OPTIONALLY ENCLOSED BY With TERMINATED BY | 10-43 |
| 10.4.9 | Conflicting Field Lengths for Character Data Types | 10-44 |
| 10.4.9.1 | Predetermined Size Fields | 10-44 |
| 10.4.9.2 | Delimited Fields | 10-45 |
| 10.4.9.3 | Date Field Masks | 10-45 |
| 10.5 | Specifying Field Conditions | 10-46 |
| 10.5.1 | Comparing Fields to BLANKS | 10-46 |
| 10.5.2 | Comparing Fields to Literals | 10-46 |
| 10.6 | Using the WHEN, NULLIF, and DEFAULTIF Clauses | 10-46 |
| 10.7 | Examples of Using the WHEN, NULLIF, and DEFAULTIF Clauses | 10-48 |
| 10.8 | Loading Data Across Different Platforms | 10-50 |
| 10.9 | Understanding how SQL*Loader Manages Byte Ordering | 10-51 |
| 10.9.1 | Byte Order Syntax | 10-52 |
| 10.9.2 | Using Byte Order Marks (BOMs) | 10-53 |
| 10.9.2.1 | Suppressing Checks for BOMs | 10-54 |
| 10.10 | Loading All-Blank Fields | 10-55 |
| 10.11 | Trimming Whitespace | 10-55 |
| 10.11.1 | Data Types for Which Whitespace Can Be Trimmed | 10-57 |
| 10.11.2 | Specifying Field Length for Data Types for Which Whitespace Can Be Trimmed | 10-58 |
| 10.11.2.1 | Predetermined Size Fields | 10-58 |
| 10.11.2.2 | Delimited Fields | 10-58 |
| 10.11.3 | Relative Positioning of Fields | 10-59 |
| 10.11.3.1 | No Start Position Specified for a Field | 10-59 |
| 10.11.3.2 | Previous Field Terminated by a Delimiter | 10-59 |
| 10.11.3.3 | Previous Field Has Both Enclosure and Termination Delimiters | 10-59 |
| 10.11.4 | Leading Whitespace | 10-60 |
| 10.11.4.1 | Previous Field Terminated by Whitespace | 10-60 |
| 10.11.4.2 | Optional Enclosure Delimiters | 10-61 |
| 10.11.5 | Trimming Trailing Whitespace | 10-61 |
| 10.11.6 | Trimming Enclosed Fields | 10-61 |
| 10.12 | How the PRESERVE BLANKS Option Affects Whitespace Trimming | 10-61 |
| 10.13 | How [NO] PRESERVE BLANKS Works with Delimiter Clauses | 10-62 |
| 10.14 | Applying SQL Operators to Fields | 10-63 |
| 10.14.1 | Referencing Fields | 10-65 |
| 10.14.2 | Common Uses of SQL Operators in Field Specifications | 10-66 |
| 10.14.3 | Combinations of SQL Operators | 10-66 |
| 10.14.4 | Using SQL Strings with a Date Mask | 10-67 |
| 10.14.5 | Interpreting Formatted Fields | 10-67 |

| | | |
|---------|---|-------|
| 10.14.6 | Using SQL Strings to Load the ANYDATA Database Type | 10-67 |
| 10.15 | Using SQL*Loader to Generate Data for Input | 10-68 |
| 10.15.1 | Loading Data Without Files | 10-69 |
| 10.15.2 | CONSTANT Parameter | 10-69 |
| 10.15.3 | EXPRESSION Parameter | 10-70 |
| 10.15.4 | RECNUM Parameter | 10-70 |
| 10.15.5 | SYSDATE Parameter | 10-71 |
| 10.15.6 | SEQUENCE Parameter | 10-72 |
| 10.15.7 | Generating Sequence Numbers for Multiple Tables | 10-73 |

11 Loading Objects, LOBs, and Collections with SQL*Loader

| | | |
|----------|---|-------|
| 11.1 | Loading Column Objects | 11-1 |
| 11.1.1 | Understanding Column Object Attributes | 11-2 |
| 11.1.2 | Loading Column Objects in Stream Record Format | 11-2 |
| 11.1.3 | Loading Column Objects in Variable Record Format | 11-3 |
| 11.1.4 | Loading Nested Column Objects | 11-4 |
| 11.1.5 | Loading Column Objects with a Derived Subtype | 11-5 |
| 11.1.6 | Specifying Null Values for Objects | 11-6 |
| 11.1.6.1 | Specifying Attribute Nulls | 11-6 |
| 11.1.6.2 | Specifying Atomic Nulls | 11-6 |
| 11.1.7 | Loading Column Objects with User-Defined Constructors | 11-7 |
| 11.2 | Loading Object Tables with SQL*Loader | 11-10 |
| 11.2.1 | Examples of Loading Object Tables with SQL*Loader | 11-11 |
| 11.2.2 | Loading Object Tables with Subtypes | 11-12 |
| 11.3 | Loading REF Columns with SQL*Loader | 11-13 |
| 11.3.1 | Specifying Table Names in a REF Clause | 11-14 |
| 11.3.2 | System-Generated OID REF Columns | 11-15 |
| 11.3.3 | Primary Key REF Columns | 11-16 |
| 11.3.4 | Unscoped REF Columns That Allow Primary Keys | 11-16 |
| 11.4 | Loading LOBs with SQL*Loader | 11-18 |
| 11.4.1 | Overview of Loading LOBs with SQL*Loader | 11-18 |
| 11.4.2 | Options for Using SQL*Loader to Load LOBs | 11-19 |
| 11.4.3 | Loading LOB Data from a Primary Data File | 11-20 |
| 11.4.3.1 | LOB Data in Predetermined Size Fields | 11-21 |
| 11.4.3.2 | LOB Data in Delimited Fields | 11-22 |
| 11.4.3.3 | LOB Data in Length-Value Pair Fields | 11-23 |
| 11.4.4 | Loading LOB Data from LOBFILES | 11-24 |
| 11.4.4.1 | Overview of Loading LOB Data from LOBFILES | 11-24 |
| 11.4.4.2 | Dynamic Versus Static LOBFILE Specifications | 11-25 |
| 11.4.4.3 | Examples of Loading LOB Data from LOBFILES | 11-25 |
| 11.4.4.4 | Considerations When Loading LOBs from LOBFILES | 11-30 |

| | | |
|----------|---|-------|
| 11.4.5 | Loading Data Files that Contain LLS Fields | 11-31 |
| 11.5 | Loading BFILE Columns with SQL*Loader | 11-32 |
| 11.6 | Loading Collections (Nested Tables and VARRAYs) | 11-33 |
| 11.6.1 | Overview of Loading Collections (Nested Tables and VARRAYs) | 11-33 |
| 11.6.2 | Restrictions in Nested Tables and VARRAYs | 11-34 |
| 11.6.3 | Secondary Data Files (SDFs) | 11-36 |
| 11.7 | Choosing Dynamic or Static SDF Specifications | 11-37 |
| 11.8 | Loading a Parent Table Separately from Its Child Table | 11-37 |
| 11.8.1 | Memory Issues When Loading VARRAY Columns | 11-38 |
| 11.9 | Loading Modes and Options for SODA Collections | 11-39 |
| 11.9.1 | SQL*Loader and SODA_COLLECTION | 11-39 |
| 11.9.2 | Loading Empty SODA Collections Using INSERT | 11-41 |
| 11.9.3 | Loading Empty SODA Collections Using APPEND | 11-41 |
| 11.9.4 | Loading Empty SODA Collections Using REPLACE and TRUNCATE | 11-41 |
| 11.9.5 | Permitted SQL*Loader Command-Line Parameters for SODA Collections | 11-41 |
| 11.9.6 | Examples of Loading SODA Collections | 11-42 |
| 11.9.6.1 | Creating and Loading a Small SODA Collection | 11-42 |
| 11.10 | Load Character Vector Data Using SQL*Loader Example | 11-44 |
| 11.11 | Load Binary Vector Data Using SQL*Loader Example | 11-48 |

12 Conventional and Direct Path Loads

| | | |
|---------|---|-------|
| 12.1 | Data Loading Methods | 12-2 |
| 12.2 | Loading ROWID Columns | 12-2 |
| 12.3 | Conventional Path Loads | 12-2 |
| 12.3.1 | Conventional Path Load | 12-3 |
| 12.3.2 | When to Use a Conventional Path Load | 12-3 |
| 12.3.3 | Conventional Path Load of a Single Partition | 12-4 |
| 12.4 | Direct Path Loads | 12-4 |
| 12.4.1 | About SQL*Loader Direct Path Load | 12-5 |
| 12.4.2 | Loading into Synonyms | 12-5 |
| 12.4.3 | Field Defaults on the Direct Path | 12-5 |
| 12.4.4 | Integrity Constraints | 12-5 |
| 12.4.5 | When to Use a Direct Path Load | 12-6 |
| 12.4.6 | Restrictions on a Direct Path Load of a Single Partition | 12-6 |
| 12.4.7 | Restrictions on Using Direct Path Loads | 12-6 |
| 12.4.8 | Advantages of a Direct Path Load | 12-7 |
| 12.4.9 | Direct Path Load of a Single Partition or Subpartition | 12-8 |
| 12.4.10 | Direct Path Load of a Partitioned or Subpartitioned Table | 12-8 |
| 12.4.11 | Data Conversion During Direct Path Loads | 12-9 |
| 12.5 | Automatic Parallel Load of Table Data with SQL*Loader | 12-10 |
| 12.6 | Loading Modes and Options for Automatic Parallel Loads | 12-11 |

| | | |
|-----------|---|-------|
| 12.6.1 | Loading Modes for Automatic Parallel Loads | 12-12 |
| 12.6.2 | Non-Sharded Automatic Parallel Loading Modes for SQL*Loader | 12-12 |
| 12.6.3 | Sharded Automatic Parallel Loading Modes for SQL*Loader | 12-15 |
| 12.7 | Using Direct Path Load | 12-20 |
| 12.7.1 | Setting Up for Direct Path Loads | 12-21 |
| 12.7.2 | Specifying a Direct Path Load | 12-21 |
| 12.7.3 | Building Indexes | 12-21 |
| 12.7.3.1 | Improving Performance | 12-22 |
| 12.7.3.2 | Calculating Temporary Segment Storage Requirements | 12-22 |
| 12.7.4 | Indexes Left in an Unusable State | 12-23 |
| 12.7.5 | Preventing Data Loss with Data Saves | 12-23 |
| 12.7.5.1 | Using Data Saves to Protect Against Data Loss | 12-24 |
| 12.7.5.2 | Using the ROWS Parameter | 12-24 |
| 12.7.5.3 | Data Save Versus Commit | 12-24 |
| 12.7.6 | Data Recovery During Direct Path Loads | 12-25 |
| 12.7.6.1 | Media Recovery and Direct Path Loads | 12-25 |
| 12.7.6.2 | Instance Recovery and Direct Path Loads | 12-25 |
| 12.7.7 | Loading Long Data Fields | 12-26 |
| 12.7.8 | Loading Data As PIECED | 12-26 |
| 12.7.9 | Auditing SQL*Loader Operations That Use Direct Path Mode | 12-27 |
| 12.8 | Optimizing Performance of Manual Direct Path Loads | 12-27 |
| 12.8.1 | Minimizing Time and Space Required for Direct Path Loads | 12-27 |
| 12.8.2 | Preallocating Storage for Faster Loading | 12-28 |
| 12.8.3 | Presorting Data for Faster Indexing | 12-28 |
| 12.8.3.1 | Advantages of Presorting Data | 12-28 |
| 12.8.3.2 | SORTED INDEXES Clause | 12-29 |
| 12.8.3.3 | Unsorted Data | 12-29 |
| 12.8.3.4 | Multiple-Column Indexes | 12-29 |
| 12.8.3.5 | Choosing the Best Sort Order | 12-30 |
| 12.8.4 | Infrequent Data Saves | 12-30 |
| 12.8.5 | Minimizing Use of the Redo Log | 12-30 |
| 12.8.5.1 | Disabling Archiving | 12-31 |
| 12.8.5.2 | Specifying the SQL*Loader UNRECOVERABLE Clause | 12-31 |
| 12.8.5.3 | Setting the SQL NOLOGGING Parameter | 12-31 |
| 12.8.6 | Specifying the Number of Column Array Rows and Size of Stream Buffers | 12-32 |
| 12.8.7 | Specifying a Value for DATE_CACHE | 12-32 |
| 12.9 | Optimizing Direct Path Loads on Multiple-CPU Systems | 12-33 |
| 12.10 | Avoiding Index Maintenance | 12-34 |
| 12.11 | Direct Path Loads, Integrity Constraints, and Triggers | 12-35 |
| 12.11.1 | Integrity Constraints | 12-35 |
| 12.11.1.1 | Enabled Constraints | 12-35 |
| 12.11.1.2 | Disabled Constraints | 12-36 |

| | | |
|-----------|---|-------|
| 12.11.1.3 | Reenable Constraints | 12-36 |
| 12.11.2 | Database Insert Triggers | 12-37 |
| 12.11.2.1 | Replacing Insert Triggers with Integrity Constraints | 12-38 |
| 12.11.2.2 | When Automatic Constraints Cannot Be Used | 12-38 |
| 12.11.2.3 | Preparation of Database Triggers | 12-38 |
| 12.11.2.4 | Using an Update Trigger | 12-39 |
| 12.11.2.5 | Duplicating the Effects of Exception Conditions | 12-39 |
| 12.11.2.6 | Using a Stored Procedure | 12-39 |
| 12.11.3 | Permanently Disabled Triggers and Constraints | 12-40 |
| 12.11.4 | Increasing Performance with Concurrent Conventional Path Loads | 12-40 |
| 12.12 | Optimizing Performance of Direct Path Loads | 12-40 |
| 12.12.1 | Restrictions on Automatic and Manual Parallel Direct Path Loads | 12-41 |
| 12.12.2 | About SQL*Loader Parallel Data Loading Models | 12-42 |
| 12.12.3 | Concurrent Conventional Path Loads | 12-42 |
| 12.12.4 | Intersegment Concurrency with Direct Path | 12-42 |
| 12.12.5 | Intrasegment Concurrency with Direct Path | 12-43 |
| 12.12.6 | Restrictions on Manual Parallel Direct Path Loads | 12-43 |
| 12.12.7 | Initiating Multiple SQL*Loader Sessions Manually | 12-43 |
| 12.12.8 | Parameters for Manual Parallel Direct Path Loads | 12-44 |
| 12.12.8.1 | Using the FILE Parameter to Specify Temporary Segments | 12-45 |
| 12.12.9 | Enabling Constraints After a Parallel Direct Path Load | 12-46 |
| 12.12.10 | PRIMARY KEY and UNIQUE KEY Constraints | 12-46 |
| 12.13 | General Performance Improvement Hints | 12-46 |

13 SQL*Loader Express

| | | |
|---------|--|-------|
| 13.1 | What is SQL*Loader Express Mode? | 13-1 |
| 13.2 | Using SQL*Loader Express Mode | 13-1 |
| 13.2.1 | Starting SQL*Loader in Express Mode | 13-2 |
| 13.2.2 | Default Values Used by SQL*Loader Express Mode | 13-2 |
| 13.2.3 | How SQL*Loader Express Mode Handles Byte Order | 13-3 |
| 13.3 | SQL*Loader Express Mode Parameter Reference | 13-4 |
| 13.3.1 | BAD | 13-6 |
| 13.3.2 | CHARACTERSET | 13-7 |
| 13.3.3 | CSV | 13-8 |
| 13.3.4 | DATA | 13-9 |
| 13.3.5 | DATE_FORMAT | 13-10 |
| 13.3.6 | DEGREE_OF_PARALLELISM | 13-11 |
| 13.3.7 | DIRECT | 13-12 |
| 13.3.8 | DNFS_ENABLE | 13-13 |
| 13.3.9 | DNFS_READBUFFERS | 13-14 |
| 13.3.10 | ENCLOSED_BY | 13-14 |

| | | |
|---------|--|-------|
| 13.3.11 | EXTERNAL_TABLE | 13-15 |
| 13.3.12 | FIELD_NAMES | 13-16 |
| 13.3.13 | LOAD | 13-17 |
| 13.3.14 | NULLIF | 13-18 |
| 13.3.15 | OPTIONALLY_ENCLOSED_BY | 13-18 |
| 13.3.16 | PARFILE | 13-19 |
| 13.3.17 | SILENT | 13-20 |
| 13.3.18 | TABLE | 13-21 |
| 13.3.19 | TERMINATED_BY | 13-21 |
| 13.3.20 | TIMESTAMP_FORMAT | 13-22 |
| 13.3.21 | TRIM | 13-22 |
| 13.3.22 | USERID | 13-23 |
| 13.4 | SQL*Loader Express Mode Command-Line Parameters for SODA Collections | 13-24 |
| 13.5 | SQL*Loader Express Mode Syntax Diagrams | 13-25 |

Part III External Tables

14 External Tables Concepts

| | | |
|------|--|-------|
| 14.1 | How Are External Tables Created? | 14-1 |
| 14.2 | CREATE_EXTERNAL_PART_TABLE Procedure | 14-4 |
| 14.3 | CREATE_EXTERNAL_TABLE Procedure | 14-11 |
| 14.4 | Location of Data Files and Output Files | 14-13 |
| 14.5 | Access Parameters for External Tables | 14-14 |
| 14.6 | Data Type Conversion During External Table Use | 14-14 |
| 14.7 | Vectors in External Tables | 14-17 |

15 The ORACLE_LOADER Access Driver

| | | |
|---------|---------------------------------------|-------|
| 15.1 | About the ORACLE_LOADER Access Driver | 15-1 |
| 15.2 | access_parameters Clause | 15-2 |
| 15.3 | record_format_info Clause | 15-4 |
| 15.3.1 | Overview of record_format_info Clause | 15-6 |
| 15.3.2 | FIXED Length | 15-8 |
| 15.3.3 | VARIABLE size | 15-9 |
| 15.3.4 | DELIMITED BY | 15-9 |
| 15.3.5 | XMLTAG | 15-11 |
| 15.3.6 | CHARACTERSET | 15-13 |
| 15.3.7 | PREPROCESSOR | 15-13 |
| 15.3.8 | PREPROCESSOR_TIMEOUT | 15-18 |
| 15.3.9 | EXTERNAL VARIABLE DATA | 15-20 |
| 15.3.10 | LANGUAGE | 15-22 |

| | | |
|-----------|--|-------|
| 15.3.11 | TERRITORY | 15-22 |
| 15.3.12 | DATA IS...ENDIAN | 15-22 |
| 15.3.13 | BYTEORDERMARK [CHECK NOCHECK] | 15-23 |
| 15.3.14 | STRING SIZES ARE IN | 15-24 |
| 15.3.15 | LOAD WHEN | 15-24 |
| 15.3.16 | BADFILE NOBADFILE | 15-25 |
| 15.3.17 | DISCARDFILE NODISCARDFILE | 15-26 |
| 15.3.18 | LOGFILE NOLOGFILE | 15-26 |
| 15.3.19 | SKIP | 15-27 |
| 15.3.20 | FIELD NAMES | 15-27 |
| 15.3.21 | READSIZE | 15-30 |
| 15.3.22 | DATE_CACHE | 15-30 |
| 15.3.23 | string | 15-30 |
| 15.3.24 | condition_spec | 15-31 |
| 15.3.25 | [directory object name:] [filename] | 15-32 |
| 15.3.26 | condition | 15-33 |
| 15.3.26.1 | range start : range end | 15-33 |
| 15.3.27 | IO_OPTIONS clause | 15-34 |
| 15.3.28 | DNFS_DISABLE DNFS_ENABLE | 15-35 |
| 15.3.29 | DNFS_READBUFFERS | 15-35 |
| 15.4 | field_definitions Clause | 15-36 |
| 15.4.1 | Overview of field_definitions Clause | 15-36 |
| 15.4.2 | delim_spec | 15-41 |
| 15.4.2.1 | Example: External Table with Terminating Delimiters | 15-42 |
| 15.4.2.2 | Example: External Table with Enclosure and Terminator Delimiters | 15-43 |
| 15.4.2.3 | Example: External Table with Optional Enclosure Delimiters | 15-43 |
| 15.4.3 | trim_spec | 15-43 |
| 15.4.4 | MISSING FIELD VALUES ARE NULL | 15-45 |
| 15.4.5 | field_list | 15-45 |
| 15.4.6 | pos_spec Clause | 15-46 |
| 15.4.6.1 | pos_spec Clause Syntax | 15-47 |
| 15.4.6.2 | start | 15-47 |
| 15.4.6.3 | * | 15-47 |
| 15.4.6.4 | increment | 15-47 |
| 15.4.6.5 | end | 15-48 |
| 15.4.6.6 | length | 15-48 |
| 15.4.7 | datatype_spec Clause | 15-49 |
| 15.4.7.1 | datatype_spec Clause Syntax | 15-50 |
| 15.4.7.2 | [UNSIGNED] INTEGER [EXTERNAL] [(len)] | 15-51 |
| 15.4.7.3 | DECIMAL [EXTERNAL] and ZONED [EXTERNAL] | 15-51 |
| 15.4.7.4 | ORACLE_DATE | 15-51 |
| 15.4.7.5 | ORACLE_NUMBER | 15-52 |

| | | |
|-----------|---|-------|
| 15.4.7.6 | Floating-Point Numbers | 15-52 |
| 15.4.7.7 | DOUBLE | 15-52 |
| 15.4.7.8 | FLOAT [EXTERNAL] | 15-52 |
| 15.4.7.9 | BINARY_DOUBLE | 15-53 |
| 15.4.7.10 | BINARY_FLOAT | 15-53 |
| 15.4.7.11 | RAW | 15-53 |
| 15.4.7.12 | CHAR | 15-53 |
| 15.4.7.13 | date_format_spec | 15-54 |
| 15.4.7.14 | VARCHAR and VARRAW | 15-56 |
| 15.4.7.15 | VARCHARC and VARRAWC | 15-58 |
| 15.4.8 | init_spec Clause | 15-58 |
| 15.4.9 | LLS Clause | 15-59 |
| 15.5 | column_transforms Clause | 15-60 |
| 15.5.1 | transform | 15-60 |
| 15.5.1.1 | column_name FROM | 15-61 |
| 15.5.1.2 | NULL | 15-62 |
| 15.5.1.3 | CONSTANT | 15-62 |
| 15.5.1.4 | CONCAT | 15-62 |
| 15.5.1.5 | LOBFILE | 15-62 |
| 15.5.1.6 | lobfile_attr_list | 15-62 |
| 15.5.1.7 | STARTOF source_field (length) | 15-63 |
| 15.6 | Parallel Loading Considerations for the ORACLE_LOADER Access Driver | 15-64 |
| 15.7 | Performance Hints When Using the ORACLE_LOADER Access Driver | 15-64 |
| 15.8 | Restrictions When Using the ORACLE_LOADER Access Driver | 15-65 |
| 15.9 | Reserved Words for the ORACLE_LOADER Access Driver | 15-66 |

16 The ORACLE_DATAPUMP Access Driver

| | | |
|----------|---|-------|
| 16.1 | Using the ORACLE_DATAPUMP Access Driver | 16-1 |
| 16.2 | access_parameters Clause | 16-2 |
| 16.2.1 | Comments | 16-4 |
| 16.2.2 | ENCRYPTION | 16-4 |
| 16.2.3 | LOGFILE NOLOGFILE | 16-5 |
| 16.2.3.1 | Log File Naming in Parallel Loads | 16-5 |
| 16.2.4 | COMPRESSION | 16-6 |
| 16.2.5 | VERSION Clause | 16-7 |
| 16.2.6 | HADOOP_TRAILERS Clause | 16-8 |
| 16.2.7 | Effects of Using the SQL ENCRYPT Clause | 16-8 |
| 16.3 | Unloading and Loading Data with the ORACLE_DATAPUMP Access Driver | 16-9 |
| 16.3.1 | Parallel Loading and Unloading | 16-12 |
| 16.3.2 | Combining Dump Files | 16-13 |
| 16.4 | Supported Data Types | 16-14 |

| | | |
|--------|--|-------|
| 16.5 | Unsupported Data Types | 16-15 |
| 16.5.1 | Unloading and Loading BFILE Data Types | 16-16 |
| 16.5.2 | Unloading LONG and LONG RAW Data Types | 16-18 |
| 16.5.3 | Unloading and Loading Columns Containing Final Object Types | 16-19 |
| 16.5.4 | Tables of Final Object Types | 16-20 |
| 16.6 | Performance Hints When Using the ORACLE_DATAPUMP Access Driver | 16-21 |
| 16.7 | Restrictions When Using the ORACLE_DATAPUMP Access Driver | 16-21 |
| 16.8 | Reserved Words for the ORACLE_DATAPUMP Access Driver | 16-22 |

17 ORACLE_BIGDATA Access Driver

| | | |
|----------|---|-------|
| 17.1 | Accessing External Data Using the ORACLE_BIGDATA Driver | 17-1 |
| 17.2 | Enabling and Configuring your Database for Object Storage Access | 17-2 |
| 17.3 | Setting Up Credentials and Location Parameters for Object Stores | 17-2 |
| 17.3.1 | How to Create a Credential for Object Stores | 17-2 |
| 17.3.1.1 | Creating the Credential Object with DBMS_CREDENTIAL.CREATE_CREDENTIAL | 17-3 |
| 17.3.2 | How to Define the Location Clause for Object Storage | 17-4 |
| 17.4 | ORACLE_BIGDATA Accessing Files | 17-4 |
| 17.4.1 | Syntax Rules for Specifying Properties | 17-5 |
| 17.4.2 | ORACLE_BIGDATA Common Access Parameters | 17-6 |
| 17.4.3 | ORACLE_BIGDATA Specific Access Parameters | 17-8 |
| 17.4.3.1 | Avro-Specific Access Parameters | 17-8 |
| 17.4.3.2 | Examples of Creating External Tables with Avro Files | 17-9 |
| 17.4.3.3 | Parquet-Specific Access Parameters | 17-10 |
| 17.4.3.4 | Examples of Creating External Tables with Avro Files | 17-10 |
| 17.4.3.5 | Textfile and CSV-Specific Access Parameters | 17-12 |
| 17.4.3.6 | Examples of Creating External Tables | 17-17 |
| 17.5 | ORACLE_BIGDATA Accessing Apache Iceberg | 17-18 |
| 17.5.1 | Apache Iceberg Tables Overview | 17-19 |
| 17.5.2 | Supported Configurations for Apache Iceberg | 17-19 |
| 17.5.3 | Iceberg-Specific Access Parameters | 17-19 |
| 17.5.4 | Examples of Table Creation and Inline External Table SQL for Iceberg Tables | 17-20 |
| 17.5.4.1 | Creating a Table Pointing to the Manifest File | 17-20 |
| 17.5.4.2 | Inline External Table Query (Manifest File Reference) | 17-21 |
| 17.5.4.3 | Creating a Table Using DBMS_CLOUD | 17-21 |
| 17.5.4.4 | Creating a Table Using AWS Glue as a Catalog | 17-22 |
| 17.5.4.5 | Inline External Table Query Using AWS Glue Catalog | 17-22 |
| 17.5.4.6 | Creating an External Table Using DBMS_CLOUD with AWS Glue Catalog | 17-23 |
| 17.6 | ORACLE_BIGDATA Accessing Delta Sharing | 17-23 |
| 17.6.1 | Delta Sharing Protocol Overview | 17-24 |
| 17.6.2 | Supported Configurations for Delta Sharing Protocol | 17-24 |

| | | |
|----------|--|-------|
| 17.6.3 | Creating Credentials for Delta Sharing | 17-24 |
| 17.6.4 | Listing and Describing Delta Share Metadata | 17-25 |
| 17.6.5 | Delta Sharing Access Parameters | 17-27 |
| 17.6.6 | Examples of Creating External Tables for Delta Sharing | 17-27 |
| 17.7 | ORACLE_BIGDATA Accessing JSON Documents File Type | 17-29 |
| 17.7.1 | Overview of JSON Document Support | 17-29 |
| 17.7.2 | Access Parameters for JSON Document | 17-29 |
| 17.7.3 | Examples of JSONDOC Usage | 17-30 |
| 17.7.3.1 | Querying Line-Delimited JSON Documents | 17-30 |
| 17.7.3.2 | Querying JSON Arrays | 17-31 |
| 17.7.3.3 | Object wrapped JSON Arrays | 17-32 |
| 17.7.3.4 | Extended JSON (EJSON) Support | 17-33 |
| 17.7.3.5 | Single-JSON Document with Multiline Files | 17-33 |

18 External Tables Examples for Oracle Database

| | | |
|----------|---|-------|
| 18.1 | Using the ORACLE_LOADER Access Driver to Create Partitioned External Tables | 18-1 |
| 18.2 | Using the ORACLE_LOADER Access Driver to Create Partitioned Hybrid Tables | 18-4 |
| 18.3 | Using the ORACLE_DATAPUMP Access Driver to Create Partitioned External Tables | 18-5 |
| 18.4 | Using the ORACLE_BIGDATA Access Driver to Create Partitioned External Tables | 18-8 |
| 18.5 | Using the ORA_PARTITION_VALIDATION Function to Validate Partitioned External Tables | 18-10 |
| 18.6 | Using SQL*Loader for External Tables with Partition Values in File Paths | 18-11 |
| 18.7 | Loading LOBs with External Tables | 18-11 |
| 18.7.1 | Overview of LOBs and External Tables | 18-11 |
| 18.7.2 | Loading LOBs From External Tables with ORACLE_LOADER Access Driver | 18-13 |
| 18.7.2.1 | Loading LOBs from Primary Data Files | 18-13 |
| 18.7.2.2 | Loading LOBs from LOBFILE Files | 18-14 |
| 18.7.2.3 | Loading LOBs from LOB Location Specifiers | 18-17 |
| 18.7.3 | Loading LOBs with ORACLE_DATAPUMP Access Driver | 18-18 |
| 18.8 | Loading CSV Files From External Tables | 18-20 |
| 18.9 | Using Vector Data Types in External Tables | 18-27 |
| 18.9.1 | Understanding Vector Data Types in External Tables | 18-27 |
| 18.9.2 | Creating External Tables Using Oracle Loader Driver | 18-28 |
| 18.9.3 | Creating External Tables Using ORACLE_DATAPUMP Driver | 18-28 |
| 18.9.4 | Querying an Inline External Table | 18-29 |
| 18.9.5 | Performing a Semantic Similarity Search Using External Table | 18-30 |

Part IV Other Utilities

19 Cloud Premigration Advisor Tool

| | | |
|-----------|---|-------|
| 19.1 | What is the Cloud Premigration Advisor Tool | 19-2 |
| 19.2 | Prerequisites for Using the Cloud Premigration Advisor Tool | 19-3 |
| 19.3 | Downloading and Configuring Cloud Premigration Advisor Tool | 19-4 |
| 19.4 | Getting Started with the Cloud Premigration Advisor Tool (CPAT) | 19-5 |
| 19.5 | Connection Strings for Cloud Premigration Advisor Tool | 19-6 |
| 19.6 | Required Command-Line Strings for Cloud Premigration Advisor Tool | 19-8 |
| 19.7 | FULL Mode and SCHEMA Mode | 19-9 |
| 19.8 | Interpreting Cloud Premigration Advisor Tool (CPAT) Report Data | 19-9 |
| 19.9 | Command-Line Syntax and Properties | 19-11 |
| 19.9.1 | Premigration Advisor Tool Command-Line Syntax | 19-11 |
| 19.9.2 | Premigration Advisor Tool Command-Line Properties | 19-12 |
| 19.9.2.1 | analysisprops | 19-13 |
| 19.9.2.2 | connectstring | 19-14 |
| 19.9.2.3 | excludeschemas | 19-15 |
| 19.9.2.4 | full | 19-16 |
| 19.9.2.5 | gettargetprops | 19-16 |
| 19.9.2.6 | help | 19-17 |
| 19.9.2.7 | logginglevel | 19-17 |
| 19.9.2.8 | maxrelevantobjects | 19-18 |
| 19.9.2.9 | maxtextdatarows | 19-19 |
| 19.9.2.10 | migrationmethod | 19-19 |
| 19.9.2.11 | outdir | 19-20 |
| 19.9.2.12 | outfileprefix | 19-20 |
| 19.9.2.13 | pdbname | 19-21 |
| 19.9.2.14 | reportformat | 19-22 |
| 19.9.2.15 | schemas | 19-23 |
| 19.9.2.16 | sqltext | 19-24 |
| 19.9.2.17 | sysdba | 19-24 |
| 19.9.2.18 | targetcloud | 19-25 |
| 19.9.2.19 | username | 19-26 |
| 19.9.2.20 | version | 19-26 |
| 19.9.2.21 | updatecheck | 19-27 |
| 19.10 | List of Checks Performed By the Premigration Advisor Tool | 19-28 |
| 19.10.1 | dp_has_low_streams_pool_size | 19-34 |
| 19.10.2 | gg_enabled_replication | 19-35 |
| 19.10.3 | gg_force_logging | 19-36 |
| 19.10.4 | gg_has_low_streams_pool_size | 19-37 |
| 19.10.5 | gg_not_unique | 19-38 |
| 19.10.6 | gg_not_unique_bad_col_no | 19-39 |
| 19.10.7 | gg_not_unique_bad_col_yes | 19-40 |

| | | |
|----------|---|-------|
| 19.10.8 | gg_objects_not_supported | 19-41 |
| 19.10.9 | gg_supplemental_log_data_min | 19-41 |
| 19.10.10 | gg_tables_not_supported | 19-42 |
| 19.10.11 | gg_tables_not_supported | 19-43 |
| 19.10.12 | gg_user_objects_in_ggadmin_schemas | 19-44 |
| 19.10.13 | has_absent_default_tablespace | 19-44 |
| 19.10.14 | has_absent_temp_tablespace | 19-45 |
| 19.10.15 | has_active_data_guard_dedicated | 19-46 |
| 19.10.16 | has_active_data_guard_serverless | 19-47 |
| 19.10.17 | has_basic_file_lobs | 19-48 |
| 19.10.18 | has_clustered_tables | 19-48 |
| 19.10.19 | has_columns_of_rowid_type | 19-49 |
| 19.10.20 | has_columns_with_local_timezone | 19-50 |
| 19.10.21 | has_columns_with_media_data_types_adb | 19-51 |
| 19.10.22 | has_columns_with_media_data_types_default | 19-52 |
| 19.10.23 | has_columns_with_spatial_data_types | 19-53 |
| 19.10.24 | has_common_objects | 19-54 |
| 19.10.25 | has_compression_disabled_for_objects | 19-54 |
| 19.10.26 | has_csmig_schema | 19-55 |
| 19.10.27 | has_data_in_other_tablespaces_dedicated | 19-56 |
| 19.10.28 | has_data_in_other_tablespaces_serverless | 19-57 |
| 19.10.29 | has_db_link_synonyms | 19-58 |
| 19.10.30 | has_db_links | 19-59 |
| 19.10.31 | has_dbms_credentials | 19-59 |
| 19.10.32 | has_dbms_credentials | 19-60 |
| 19.10.33 | has_directories | 19-61 |
| 19.10.34 | has_enabled_scheduler_jobs | 19-62 |
| 19.10.35 | has_external_tables_dedicated | 19-63 |
| 19.10.36 | has_external_tables_default | 19-63 |
| 19.10.37 | has_external_tables_serverless | 19-64 |
| 19.10.38 | has_fmw_registry_in_system | 19-65 |
| 19.10.39 | has_illegal_characters_in_comments | 19-65 |
| 19.10.40 | has_ilm_ado_policies | 19-66 |
| 19.10.41 | has_incompatible_jobs | 19-67 |
| 19.10.42 | has_index_organized_tables | 19-68 |
| 19.10.43 | has_java_objects | 19-68 |
| 19.10.44 | has_java_source | 19-69 |
| 19.10.45 | has_libraries | 19-70 |
| 19.10.46 | has_logging_off_for_partitions | 19-70 |
| 19.10.47 | has_logging_off_for_subpartitions | 19-71 |
| 19.10.48 | has_logging_off_for_tables | 19-72 |
| 19.10.49 | has_low_streams_pool_size | 19-72 |

| | | |
|----------|--|--------|
| 19.10.50 | has_noexport_object_grants | 19-73 |
| 19.10.51 | has_oracle_streams | 19-74 |
| 19.10.52 | has_parallel_indexes_enabled | 19-75 |
| 19.10.53 | has_profile_not_default | 19-76 |
| 19.10.54 | has_public_synonyms | 19-76 |
| 19.10.55 | has_refs_to_restricted_packages_dedicated | 19-77 |
| 19.10.56 | has_refs_to_restricted_packages_serverless | 19-78 |
| 19.10.57 | has_refs_to_user_objects_in_sys | 19-78 |
| 19.10.58 | has_role_privileges | 19-79 |
| 19.10.59 | has_sqlt_objects_adb | 19-80 |
| 19.10.60 | has_sqlt_objects_default | 19-81 |
| 19.10.61 | has_sys_privileges | 19-82 |
| 19.10.62 | has_tables_that_fail_with_dblink | 19-82 |
| 19.10.63 | has_tables_with_long_raw_datatype | 19-83 |
| 19.10.64 | has_tables_with_xmltype_column | 19-84 |
| 19.10.65 | has_trusted_server_entries | 19-85 |
| 19.10.66 | has_unstructured_xml_indexes Check | 19-86 |
| 19.10.67 | has_user_defined_objects_in_sys | 19-86 |
| 19.10.68 | has_user_defined_objects_in_system | 19-87 |
| 19.10.69 | has_user_defined_objects_no_quota | 19-88 |
| 19.10.70 | has_user_defined_pvfs | 19-89 |
| 19.10.71 | has_users_with_10g_password_version | 19-89 |
| 19.10.72 | has_xmlschema_objects | 19-90 |
| 19.10.73 | has_xmltype_tables | 19-91 |
| 19.10.74 | modified_db_parameters_dedicated | 19-92 |
| 19.10.75 | modified_db_parameters_serverless | 19-92 |
| 19.10.76 | nls_character_set_conversion | 19-93 |
| 19.10.77 | nls_national_character_set | 19-94 |
| 19.10.78 | nls_nchar_ora_910 | 19-95 |
| 19.10.79 | options_in_use_not_available_dedicated | 19-96 |
| 19.10.80 | options_in_use_not_available_serverless | 19-97 |
| 19.10.81 | standard_traditional_audit_adb | 19-97 |
| 19.10.82 | standard_traditional_audit_default | 19-98 |
| 19.10.83 | timezone_table_compatibility_higher_dedicated | 19-99 |
| 19.10.84 | timezone_table_compatibility_higher_default | 19-100 |
| 19.10.85 | timezone_table_compatibility_higher_serverless | 19-100 |
| 19.10.86 | unified_and_standard_traditional_audit_adb | 19-101 |
| 19.10.87 | unified_and_standard_traditional_audit_default | 19-102 |
| 19.10.88 | xdb_resource_view_has_entries Check | 19-103 |
| 19.11 | Best Practices for Using the Premigration Advisor Tool | 19-103 |
| 19.11.1 | Generate Properties File on the Target Database Instance | 19-104 |
| 19.11.2 | Focus the CPAT Analysis | 19-104 |

| | | |
|---------|---|--------|
| 19.11.3 | Reduce the Amount of Data in Reports | 19-105 |
| 19.11.4 | Generate the JSON Report and Save Logs | 19-105 |
| 19.11.5 | Use Output Prefixes to Record Different Migration Scenarios | 19-106 |

20 DBMS_CLOUD Family of Packages

| | | |
|--------|---|-------|
| 20.1 | Using the DBMS_CLOUD Family of Packages | 20-1 |
| 20.2 | Installing DBMS_CLOUD | 20-2 |
| 20.3 | Create SSL Wallet with Certificates | 20-4 |
| 20.4 | Configure Your Environment to Use the New SSL Wallet | 20-5 |
| 20.5 | Configure the Database with ACES for DBMS_CLOUD | 20-5 |
| 20.6 | Verify Configuration of DBMS_CLOUD | 20-9 |
| 20.7 | Configuring Users or Roles to use DBMS_CLOUD | 20-10 |
| 20.7.1 | Grant Minimal Privileges to a User or Role for DBMS_CLOUD | 20-11 |
| 20.7.2 | Configure ACES for a User or Role to Use DBMS_CLOUD | 20-12 |
| 20.7.3 | Verify Setup of Users and Roles to Use DBMS_CLOUD | 20-15 |

21 Migrating From JSON To Duality

| | | |
|-------|---|--------|
| 21.1 | JSON Columns in Duality Views | 21-2 |
| 21.2 | About Migrations From JSON To Duality | 21-3 |
| 21.3 | JSON To Duality Migrator Components: Converter and Importer | 21-5 |
| 21.4 | JSON Configuration Fields Specifying Migrator Parameters | 21-8 |
| 21.5 | School Administration Example, Migrator Input Documents | 21-13 |
| 21.6 | Before Using the Converter (1): Create Database Document Sets | 21-18 |
| 21.7 | Before Using the Converter (2): Optionally Create Data-Guide JSON Schemas | 21-22 |
| 21.8 | JSON-To-Duality Converter: What It Does | 21-36 |
| 21.9 | Migrating To Duality, Simplified Recipe | 21-37 |
| 21.10 | Using the Converter, Default Behavior | 21-62 |
| 21.11 | Import After Default Conversion | 21-77 |
| 21.12 | Using the Converter with useFlexFields=false | 21-109 |
| 21.13 | Import After Conversion with useFlexFields=false | 21-114 |
| 21.14 | Errors That Migrator Configuration Alone Can't Fix | 21-133 |

22 Oracle SQL Access to Kafka

| | | |
|------|--|------|
| 22.1 | About Oracle SQL Access to Kafka Version 2 | 22-2 |
| 22.2 | Global Tables and Views for Oracle SQL Access to Kafka | 22-4 |
| 22.3 | Understanding how Oracle SQL Access to Kafka Queries are Performed | 22-4 |
| 22.4 | Streaming Kafka Data Into Oracle Database | 22-5 |
| 22.5 | Querying Kafka Data Records by Timestamp | 22-6 |
| 22.6 | About the Kafka Database Administrator Role | 22-7 |

| | | |
|----------|--|-------|
| 22.7 | Enable Kafka Database Access to Users | 22-7 |
| 22.8 | Data Formats Supported with Oracle SQL Access to Kafka | 22-8 |
| 22.8.1 | JSON Format and Oracle SQL Access to Kafka | 22-9 |
| 22.8.2 | Delimited Text Format and Oracle SQL Access to Kafka | 22-9 |
| 22.8.3 | Avro Formats and Oracle SQL Access to Kafka | 22-11 |
| 22.8.3.1 | About Using Avro Format with Oracle SQL Access to Kafka | 22-12 |
| 22.8.3.2 | Primitive Avro Types Supported with Oracle SQL Access to Kafka | 22-12 |
| 22.8.3.3 | Complex Avro Types Supported with Oracle SQL Access to Kafka | 22-13 |
| 22.8.3.4 | Avro Logical Types Supported with Oracle SQL Access to Kafka | 22-15 |
| 22.9 | Configuring Access to a Kafka Cluster | 22-18 |
| 22.9.1 | Create a Cluster Access Directory | 22-18 |
| 22.9.2 | The Kafka Configuration File (osakafka.properties) | 22-19 |
| 22.9.2.1 | About the Kafka Configuration File | 22-19 |
| 22.9.2.2 | Oracle SQL Access for Kafka Configuration File Properties | 22-20 |
| 22.9.2.3 | Creating the Kafka Access Directory | 22-23 |
| 22.9.3 | Kafka Configuration File Properties | 22-23 |
| 22.9.4 | Security Configuration Files Required for the Cluster Access Directory | 22-25 |
| 22.9.4.1 | SASL_SSL/GSSAPI | 22-26 |
| 22.9.4.2 | SASL_PLAINTEXT/GSSAPI | 22-27 |
| 22.9.4.3 | SASL_PLAINTEXT/SCRAM-SHA-256 | 22-27 |
| 22.9.4.4 | SASL_SSL/PLAIN | 22-28 |
| 22.9.4.5 | SSL with Client Authentication | 22-28 |
| 22.9.4.6 | SSL without Client Authentication | 22-29 |
| 22.10 | Creating Oracle SQL Access to Kafka Applications | 22-29 |
| 22.11 | Security for Kafka Cluster Connections | 22-31 |
| 22.12 | Configuring Access to Unsecured Kafka Clusters | 22-32 |
| 22.13 | Configuring Access to Secure Kafka Clusters | 22-33 |
| 22.14 | Administering Oracle SQL Access to Kafka Clusters | 22-35 |
| 22.14.1 | Updating Access to Kafka Clusters | 22-35 |
| 22.14.2 | Disabling or Deleting Access to Kafka Clusters | 22-35 |
| 22.15 | Guidelines for Using Kafka Data with Oracle SQL Access to Kafka | 22-36 |
| 22.15.1 | Kafka Temporary Tables and Applications | 22-36 |
| 22.15.2 | Sharing Kafka Data with Multiple Applications Using Streaming | 22-37 |
| 22.15.3 | Dropping and Recreating Kafka Tables | 22-37 |
| 22.16 | Choosing a Kafka Cluster Access Mode for Applications | 22-38 |
| 22.16.1 | Configuring Incremental Loads of Kafka Records Into an Oracle Database Table | 22-38 |
| 22.16.2 | Streaming Access to Kafka Records in Oracle SQL Queries | 22-39 |
| 22.16.3 | Seekable access to Kafka Records in Oracle SQL queries | 22-40 |
| 22.17 | Creating Oracle SQL Access to Kafka Applications | 22-40 |
| 22.17.1 | Creating Load Applications with Oracle SQL Access to Kafka | 22-41 |
| 22.17.2 | Creating Streaming Applications with Oracle SQL Access to Kafka | 22-42 |

| | | |
|---------|--|-------|
| 22.17.3 | Creating Seekable Applications with Oracle SQL Access to Kafka | 22-44 |
| 22.18 | Using Kafka Cluster Access for Applications | 22-46 |
| 22.18.1 | How to Diagnose Oracle SQL Access to Kafka Issues | 22-46 |
| 22.18.2 | Identifying and Resolving Oracle SQL Access to Kafka Issues | 22-48 |

23 ADRCI: ADR Command Interpreter

| | | |
|-----------|---|-------|
| 23.1 | About the ADR Command Interpreter (ADRCI) Utility | 23-2 |
| 23.2 | Definitions for Oracle Database ADRC | 23-2 |
| 23.3 | Starting ADRCI and Getting Help | 23-5 |
| 23.3.1 | Using ADRCI in Interactive Mode | 23-5 |
| 23.3.2 | Getting Help | 23-5 |
| 23.3.3 | Using ADRCI in Batch Mode | 23-6 |
| 23.4 | Setting the ADRCI Homepath Before Using ADRCI Commands | 23-7 |
| 23.5 | Viewing the Alert Log | 23-9 |
| 23.6 | Finding Trace Files | 23-10 |
| 23.7 | Viewing Incidents | 23-11 |
| 23.8 | Packaging Incidents | 23-12 |
| 23.8.1 | About Packaging Incidents | 23-12 |
| 23.8.2 | Creating Incident Packages | 23-13 |
| 23.8.2.1 | Creating a Logical Incident Package | 23-13 |
| 23.8.2.2 | Adding Diagnostic Information to a Logical Incident Package | 23-15 |
| 23.8.2.3 | Generating a Physical Incident Package | 23-16 |
| 23.9 | ADRCI Command Reference | 23-17 |
| 23.9.1 | CREATE REPORT | 23-19 |
| 23.9.2 | ECHO | 23-20 |
| 23.9.3 | EXIT | 23-20 |
| 23.9.4 | HOST | 23-20 |
| 23.9.5 | IPS | 23-21 |
| 23.9.5.1 | Using the <ADR_HOME> and <ADR_BASE> Variables in IPS Commands | 23-23 |
| 23.9.5.2 | IPS ADD | 23-23 |
| 23.9.5.3 | IPS ADD FILE | 23-25 |
| 23.9.5.4 | IPS ADD NEW INCIDENTS | 23-25 |
| 23.9.5.5 | IPS COPY IN FILE | 23-26 |
| 23.9.5.6 | IPS COPY OUT FILE | 23-26 |
| 23.9.5.7 | IPS CREATE PACKAGE | 23-27 |
| 23.9.5.8 | IPS DELETE PACKAGE | 23-29 |
| 23.9.5.9 | IPS FINALIZE | 23-30 |
| 23.9.5.10 | IPS GENERATE PACKAGE | 23-30 |
| 23.9.5.11 | IPS GET MANIFEST | 23-31 |
| 23.9.5.12 | IPS GET METADATA | 23-31 |

| | | |
|-----------|------------------------|-------|
| 23.9.5.13 | IPS PACK | 23-32 |
| 23.9.5.14 | IPS REMOVE | 23-34 |
| 23.9.5.15 | IPS REMOVE FILE | 23-35 |
| 23.9.5.16 | IPS SET CONFIGURATION | 23-36 |
| 23.9.5.17 | IPS SHOW CONFIGURATION | 23-36 |
| 23.9.5.18 | IPS SHOW FILES | 23-39 |
| 23.9.5.19 | IPS SHOW INCIDENTS | 23-40 |
| 23.9.5.20 | IPS SHOW PACKAGE | 23-41 |
| 23.9.5.21 | IPS UNPACK FILE | 23-41 |
| 23.9.6 | PURGE | 23-42 |
| 23.9.7 | QUIT | 23-43 |
| 23.9.8 | RUN | 23-44 |
| 23.9.9 | SELECT | 23-44 |
| 23.9.9.1 | AVG | 23-47 |
| 23.9.9.2 | CONCAT | 23-48 |
| 23.9.9.3 | COUNT | 23-48 |
| 23.9.9.4 | DECODE | 23-49 |
| 23.9.9.5 | LENGTH | 23-49 |
| 23.9.9.6 | MAX | 23-50 |
| 23.9.9.7 | MIN | 23-50 |
| 23.9.9.8 | NVL | 23-51 |
| 23.9.9.9 | REGEXP_LIKE | 23-51 |
| 23.9.9.10 | SUBSTR | 23-52 |
| 23.9.9.11 | SUM | 23-52 |
| 23.9.9.12 | TIMESTAMP_TO_CHAR | 23-53 |
| 23.9.9.13 | TOLOWER | 23-53 |
| 23.9.9.14 | TOUPPER | 23-54 |
| 23.9.10 | SET BASE | 23-54 |
| 23.9.11 | SET BROWSER | 23-55 |
| 23.9.12 | SET CONTROL | 23-55 |
| 23.9.13 | SET ECHO | 23-57 |
| 23.9.14 | SET EDITOR | 23-57 |
| 23.9.15 | SET HOMEPATH | 23-58 |
| 23.9.16 | SET TERMOUT | 23-58 |
| 23.9.17 | SHOW ALERT | 23-59 |
| 23.9.18 | SHOW BASE | 23-61 |
| 23.9.19 | SHOW CONTROL | 23-61 |
| 23.9.20 | SHOW HM_RUN | 23-64 |
| 23.9.21 | SHOW HOMEPATH | 23-65 |
| 23.9.22 | SHOW HOMES | 23-65 |
| 23.9.23 | SHOW INCDIR | 23-65 |
| 23.9.24 | SHOW INCIDENT | 23-67 |

| | | |
|---------|-----------------------|-------|
| 23.9.25 | SHOW LOG | 23-70 |
| 23.9.26 | SHOW PROBLEM | 23-71 |
| 23.9.27 | SHOW REPORT | 23-73 |
| 23.9.28 | SHOW TRACEFILE | 23-73 |
| 23.9.29 | SPOOL | 23-74 |
| 23.10 | Troubleshooting ADRCI | 23-75 |

24 DBVERIFY: Offline Database Verification Utility

| | | |
|--------|--|------|
| 24.1 | Using DBVERIFY to Validate Disk Blocks of a Single Data File | 24-1 |
| 24.1.1 | DBVERIFY Syntax When Validating Blocks of a Single File | 24-2 |
| 24.1.2 | DBVERIFY Parameters When Validating Blocks of a Single File | 24-2 |
| 24.1.3 | Example DBVERIFY Output For a Single Data File | 24-3 |
| 24.2 | Using DBVERIFY to Validate a Segment | 24-4 |
| 24.2.1 | DBVERIFY Syntax When Validating a Segment | 24-5 |
| 24.2.2 | DBVERIFY Parameters When Validating a Single Segment | 24-5 |
| 24.2.3 | Example DBVERIFY Output For a Validated Segment | 24-6 |

25 DBNEWID Utility

| | | |
|--------|---|-------|
| 25.1 | What Is the DBNEWID Utility? | 25-1 |
| 25.2 | Ramifications of Changing the DBID and DBNAME | 25-1 |
| 25.3 | Considerations for Global Database Names | 25-2 |
| 25.4 | Changing Both CDB and PDB DBIDs Using DBNEWID | 25-3 |
| 25.5 | Changing the DBID and DBNAME of a Database | 25-3 |
| 25.5.1 | Changing the DBID and Database Name | 25-4 |
| 25.5.2 | Changing Only the Database ID | 25-6 |
| 25.5.3 | Changing Only the Database Name | 25-7 |
| 25.5.4 | Troubleshooting DBNEWID | 25-8 |
| 25.6 | DBNEWID Syntax | 25-10 |
| 25.6.1 | DBNEWID Parameters | 25-10 |
| 25.6.2 | Restrictions and Usage Notes | 25-11 |
| 25.6.3 | Additional Restrictions for Releases Earlier Than Oracle Database 10g | 25-12 |

26 Using LogMiner to Analyze Redo Log Files

| | | |
|----------|---|------|
| 26.1 | LogMiner Benefits | 26-2 |
| 26.2 | Introduction to LogMiner | 26-3 |
| 26.2.1 | LogMiner Configuration | 26-3 |
| 26.2.1.1 | Objects in LogMiner Configuration Files | 26-3 |
| 26.2.1.2 | LogMiner Configuration Example | 26-4 |
| 26.2.1.3 | LogMiner Requirements | 26-4 |

| | | |
|----------|--|-------|
| 26.2.2 | Directing LogMiner Operations and Retrieving Data of Interest | 26-7 |
| 26.3 | Using LogMiner in a CDB | 26-7 |
| 26.3.1 | LogMiner V\$ Views and DBA Views in a CDB | 26-8 |
| 26.3.2 | The V\$LOGMNR_CONTENTS View in a CDB | 26-9 |
| 26.3.3 | Enabling Supplemental Logging in a CDB | 26-10 |
| 26.4 | How to Configure Supplemental Logging for Oracle GoldenGate | 26-11 |
| 26.4.1 | Oracle GoldenGate Integration with Oracle Database for Fine-Grained Supplemental Logging | 26-11 |
| 26.4.2 | Logical Replication of Tables with LogMiner and Oracle GoldenGate | 26-12 |
| 26.4.3 | Views that Show Tables Enabled for Oracle GoldenGate Automatic Capture | 26-13 |
| 26.5 | LogMiner Dictionary Files and Redo Log Files | 26-14 |
| 26.5.1 | LogMiner Dictionary Options | 26-14 |
| 26.5.1.1 | Using the Online Catalog | 26-15 |
| 26.5.1.2 | Extracting a LogMiner Dictionary to the Redo Log Files | 26-16 |
| 26.5.1.3 | Extracting the LogMiner Dictionary to a Flat File | 26-17 |
| 26.5.2 | Specifying Redo Log Files for Data Mining | 26-18 |
| 26.6 | Starting LogMiner | 26-19 |
| 26.7 | Querying V\$LOGMNR_CONTENTS for Redo Data of Interest | 26-19 |
| 26.7.1 | How to Use V\$LOGMNR_CONTENTS to Find Redo Data | 26-20 |
| 26.7.2 | How the V\$LOGMNR_CONTENTS View Is Populated | 26-22 |
| 26.7.3 | Querying V\$LOGMNR_CONTENTS Based on Column Values | 26-23 |
| 26.7.3.1 | Example of Querying V\$LOGMNR_CONTENTS Column Values | 26-23 |
| 26.7.3.2 | The Meaning of NULL Values Returned by the MINE_VALUE Function | 26-24 |
| 26.7.3.3 | Usage Rules for the MINE_VALUE and COLUMN_PRESENT Functions | 26-24 |
| 26.7.3.4 | Restrictions When Using the MINE_VALUE Function To Get an NCHAR Value | 26-24 |
| 26.7.4 | Querying V\$LOGMNR_CONTENTS Based on XMLType Columns and Tables | 26-24 |
| 26.7.4.1 | How V\$LOGMNR_CONTENTS Based on XMLType Columns and Tables are Queried | 26-25 |
| 26.7.4.2 | Restrictions When Using LogMiner With XMLType Data | 26-27 |
| 26.7.4.3 | Example of a PL/SQL Procedure for Assembling XMLType Data | 26-27 |
| 26.8 | Filtering and Formatting Data Returned to V\$LOGMNR_CONTENTS | 26-29 |
| 26.8.1 | Showing Only Committed Transactions | 26-30 |
| 26.8.2 | Skipping Redo Corruptions | 26-32 |
| 26.8.3 | Filtering Data by Time | 26-33 |
| 26.8.4 | Filtering Data by SCN | 26-34 |
| 26.8.5 | Formatting Reconstructed SQL Statements for Reprocessing | 26-34 |
| 26.8.6 | Formatting the Appearance of Returned Data for Readability | 26-35 |
| 26.9 | Reapplying DDL Statements Returned to V\$LOGMNR_CONTENTS | 26-36 |
| 26.10 | Calling DBMS_LOGMNR.START_LOGMNR Multiple Times | 26-36 |
| 26.11 | LogMiner and Supplemental Logging | 26-37 |
| 26.11.1 | Understanding Supplemental Logging and LogMiner | 26-38 |
| 26.11.2 | Database-Level Supplemental Logging | 26-39 |

| | | |
|-----------|---|-------|
| 26.11.2.1 | Minimal Supplemental Logging | 26-39 |
| 26.11.2.2 | Database-Level Identification Key Logging | 26-39 |
| 26.11.2.3 | Procedural Supplemental Logging | 26-41 |
| 26.11.3 | Disabling Database-Level Supplemental Logging | 26-41 |
| 26.11.4 | Table-Level Supplemental Logging | 26-42 |
| 26.11.4.1 | Table-Level Identification Key Logging | 26-42 |
| 26.11.4.2 | Table-Level User-Defined Supplemental Log Groups | 26-43 |
| 26.11.4.3 | Usage Notes for User-Defined Supplemental Log Groups | 26-44 |
| 26.11.5 | Tracking DDL Statements in the LogMiner Dictionary | 26-44 |
| 26.11.6 | DDL_DICT_TRACKING and Supplemental Logging Settings | 26-45 |
| 26.11.7 | DDL_DICT_TRACKING and Specified Time or SCN Ranges | 26-46 |
| 26.12 | Accessing LogMiner Operational Information in Views | 26-47 |
| 26.12.1 | Options for Viewing LogMiner Operational Information | 26-47 |
| 26.12.2 | Querying V\$LOGMNR_LOGS | 26-48 |
| 26.12.3 | Querying Views for Supplemental Logging Settings | 26-49 |
| 26.12.4 | Querying Individual PDBs Using LogMiner | 26-50 |
| 26.13 | Steps in a Typical LogMiner Session | 26-52 |
| 26.13.1 | Understanding How to Run LogMiner Sessions | 26-52 |
| 26.13.2 | Typical LogMiner Session Task 1: Enable Supplemental Logging | 26-54 |
| 26.13.3 | Typical LogMiner Session Task 2: Extract a LogMiner Dictionary | 26-54 |
| 26.13.4 | Typical LogMiner Session Task 3: Specify Redo Log Files for Analysis | 26-55 |
| 26.13.5 | Start LogMiner | 26-56 |
| 26.13.6 | Query V\$LOGMNR_CONTENTS | 26-57 |
| 26.13.7 | Typical LogMiner Session Task 6: End the LogMiner Session | 26-58 |
| 26.14 | Examples Using LogMiner | 26-58 |
| 26.14.1 | Examples of Mining by Explicitly Specifying the Redo Log Files of Interest | 26-59 |
| 26.14.1.1 | Example 1: Finding All Modifications in the Last Archived Redo Log File | 26-60 |
| 26.14.1.2 | Example 2: Grouping DML Statements into Committed Transactions | 26-62 |
| 26.14.1.3 | Example 3: Formatting the Reconstructed SQL | 26-64 |
| 26.14.1.4 | Example 4: Using the LogMiner Dictionary in the Redo Log Files | 26-67 |
| 26.14.1.5 | Example 5: Tracking DDL Statements in the Internal Dictionary | 26-75 |
| 26.14.1.6 | Example 6: Filtering Output by Time Range | 26-78 |
| 26.14.2 | LogMiner Use Case Scenarios | 26-80 |
| 26.14.2.1 | Using LogMiner to Track Changes Made by a Specific User | 26-80 |
| 26.14.2.2 | Using LogMiner to Calculate Table Access Statistics | 26-82 |
| 26.15 | Supported Data Types, Storage Attributes, and Database and Redo Log File Versions | 26-83 |
| 26.15.1 | Supported Data Types and Table Storage Attributes | 26-84 |
| 26.15.2 | Database Compatibility Requirements for LogMiner | 26-85 |
| 26.15.3 | Unsupported Data Types and Table Storage Attributes | 26-86 |
| 26.15.4 | Supported Databases and Redo Log File Versions | 26-86 |

27 Using the Metadata APIs

| | | |
|--------|---|-------|
| 27.1 | Why Use the DBMS_METADATA API? | 27-2 |
| 27.2 | Overview of the DBMS_METADATA API | 27-2 |
| 27.3 | Using the DBMS_METADATA API to Retrieve an Object's Metadata | 27-4 |
| 27.3.1 | How to Use the DBMS_METADATA API to Retrieve Object Metadata | 27-5 |
| 27.3.2 | Typical Steps Used for Basic Metadata Retrieval | 27-5 |
| 27.3.3 | Retrieving Multiple Objects | 27-7 |
| 27.3.4 | Placing Conditions on Transforms | 27-8 |
| 27.3.5 | Accessing Specific Metadata Attributes | 27-11 |
| 27.4 | Using the DBMS_METADATA API to Recreate a Retrieved Object | 27-13 |
| 27.5 | Using the DBMS_METADATA API to Retrieve Collections of Different Object Types | 27-16 |
| 27.6 | Filtering the Return of Heterogeneous Object Types | 27-17 |
| 27.7 | Using the DBMS_METADATA_DIFF API to Compare Object Metadata | 27-19 |
| 27.8 | Performance Tips for the Programmatic Interface of the DBMS_METADATA API | 27-27 |
| 27.9 | Example Usage of the DBMS_METADATA API | 27-27 |
| 27.9.1 | What Does the DBMS_METADATA Example Do? | 27-28 |
| 27.9.2 | Output Generated from the GET_PAYROLL_TABLES Procedure | 27-30 |
| 27.10 | Summary of DBMS_METADATA Procedures | 27-32 |
| 27.11 | Summary of DBMS_METADATA_DIFF Procedures | 27-33 |

28 Original Import

| | | |
|----------|---|------|
| 28.1 | What Is the Import Utility? | 28-2 |
| 28.2 | Table Objects: Order of Import | 28-3 |
| 28.3 | Before Using Import | 28-3 |
| 28.3.1 | Overview of Import Preparation | 28-4 |
| 28.3.2 | Running catexp.sql or catalog.sql | 28-4 |
| 28.3.3 | Verifying Access Privileges for Import Operations | 28-4 |
| 28.3.3.1 | Importing Objects Into Your Own Schema | 28-5 |
| 28.3.3.2 | Importing Grants | 28-5 |
| 28.3.3.3 | Importing Objects Into Other Schemas | 28-6 |
| 28.3.3.4 | Importing System Objects | 28-6 |
| 28.3.4 | Processing Restrictions | 28-7 |
| 28.4 | Importing into Existing Tables | 28-7 |
| 28.4.1 | Manually Creating Tables Before Importing Data | 28-7 |
| 28.4.2 | Disabling Referential Constraints | 28-7 |
| 28.4.3 | Manually Ordering the Import | 28-8 |
| 28.5 | Effect of Schema and Database Triggers on Import Operations | 28-8 |
| 28.6 | Invoking Import | 28-9 |

| | | |
|-----------|--|-------|
| 28.6.1 | Command-Line Entries | 28-9 |
| 28.6.2 | Parameter Files | 28-10 |
| 28.6.3 | Interactive Mode | 28-11 |
| 28.6.4 | Invoking Import As SYSDBA | 28-11 |
| 28.6.5 | Getting Online Help | 28-11 |
| 28.7 | Import Modes | 28-11 |
| 28.8 | Import Parameters | 28-14 |
| 28.8.1 | BUFFER | 28-17 |
| 28.8.2 | COMMIT | 28-17 |
| 28.8.3 | COMPILE | 28-18 |
| 28.8.4 | CONSTRAINTS | 28-18 |
| 28.8.5 | DATA_ONLY | 28-19 |
| 28.8.6 | DATAFILES | 28-19 |
| 28.8.7 | DESTROY | 28-19 |
| 28.8.8 | FEEDBACK | 28-20 |
| 28.8.9 | FILE | 28-20 |
| 28.8.10 | FILESIZE | 28-20 |
| 28.8.11 | FROMUSER | 28-21 |
| 28.8.12 | FULL | 28-22 |
| 28.8.12.1 | Points to Consider for Full Database Exports and Imports | 28-22 |
| 28.8.13 | GRANTS | 28-23 |
| 28.8.14 | HELP | 28-23 |
| 28.8.15 | IGNORE | 28-23 |
| 28.8.16 | INDEXES | 28-24 |
| 28.8.17 | INDEXFILE | 28-24 |
| 28.8.18 | LOG | 28-25 |
| 28.8.19 | PARFILE | 28-25 |
| 28.8.20 | RECORDLENGTH | 28-25 |
| 28.8.21 | RESUMABLE | 28-25 |
| 28.8.22 | RESUMABLE_NAME | 28-26 |
| 28.8.23 | RESUMABLE_TIMEOUT | 28-26 |
| 28.8.24 | ROWS | 28-26 |
| 28.8.25 | SHOW | 28-27 |
| 28.8.26 | SKIP_UNUSABLE_INDEXES | 28-27 |
| 28.8.27 | STATISTICS | 28-28 |
| 28.8.28 | STREAMS_CONFIGURATION | 28-28 |
| 28.8.29 | STREAMS_INSTANTIATION | 28-28 |
| 28.8.30 | TABLES | 28-29 |
| 28.8.30.1 | Table Name Restrictions | 28-30 |
| 28.8.31 | TABLESPACES | 28-31 |
| 28.8.32 | TOID_NOVALIDATE | 28-31 |
| 28.8.33 | TOUSER | 28-32 |

| | | |
|-----------|---|-------|
| 28.8.34 | TRANSPORT_TABLESPACE | 28-33 |
| 28.8.35 | TTS_OWNERS | 28-33 |
| 28.8.36 | USERID (username/password) | 28-33 |
| 28.8.37 | VOLSIZE | 28-33 |
| 28.9 | Example Import Sessions | 28-34 |
| 28.9.1 | Example Import of Selected Tables for a Specific User | 28-34 |
| 28.9.2 | Example Import of Tables Exported by Another User | 28-34 |
| 28.9.3 | Example Import of Tables from One User to Another | 28-35 |
| 28.9.4 | Example Import Session Using Partition-Level Import | 28-35 |
| 28.9.4.1 | Example 1: A Partition-Level Import | 28-36 |
| 28.9.4.2 | Example 2: A Partition-Level Import of a Composite Partitioned Table | 28-36 |
| 28.9.4.3 | Example 3: Repartitioning a Table on a Different Column | 28-37 |
| 28.9.5 | Example Import Using Pattern Matching to Import Various Tables | 28-39 |
| 28.10 | Exit Codes for Inspection and Display | 28-39 |
| 28.11 | Error Handling During an Import | 28-40 |
| 28.11.1 | Row Errors | 28-40 |
| 28.11.1.1 | Failed Integrity Constraints | 28-40 |
| 28.11.1.2 | Invalid Data | 28-41 |
| 28.11.2 | Errors Importing Database Objects | 28-41 |
| 28.11.2.1 | Object Already Exists | 28-41 |
| 28.11.2.2 | Sequences | 28-42 |
| 28.11.2.3 | Resource Errors | 28-42 |
| 28.11.2.4 | Domain Index Metadata | 28-42 |
| 28.12 | Table-Level and Partition-Level Import | 28-42 |
| 28.12.1 | Guidelines for Using Table-Level Import | 28-43 |
| 28.12.2 | Guidelines for Using Partition-Level Import | 28-43 |
| 28.12.3 | Migrating Data Across Partitions and Tables | 28-44 |
| 28.13 | Controlling Index Creation and Maintenance | 28-44 |
| 28.13.1 | Delaying Index Creation | 28-45 |
| 28.13.2 | Index Creation and Maintenance Controls | 28-45 |
| 28.13.2.1 | Example of Postponing Index Maintenance | 28-45 |
| 28.14 | Network Considerations for Using Oracle Net with Original Import | 28-46 |
| 28.15 | Character Set and Globalization Support Considerations | 28-46 |
| 28.15.1 | User Data | 28-47 |
| 28.15.1.1 | Effect of Character Set Sorting Order on Conversions | 28-47 |
| 28.15.2 | Data Definition Language (DDL) | 28-47 |
| 28.15.3 | Single-Byte Character Sets | 28-48 |
| 28.15.4 | Multibyte Character Sets | 28-48 |
| 28.16 | Using Instance Affinity | 28-48 |
| 28.17 | Considerations When Importing Database Objects | 28-49 |
| 28.17.1 | Importing Object Identifiers | 28-50 |
| 28.17.2 | Importing Existing Object Tables and Tables That Contain Object Types | 28-51 |

| | | |
|-----------|---|-------|
| 28.17.3 | Importing Nested Tables | 28-51 |
| 28.17.4 | Importing REF Data | 28-52 |
| 28.17.5 | Importing BFILE Columns and Directory Aliases | 28-52 |
| 28.17.6 | Importing Foreign Function Libraries | 28-53 |
| 28.17.7 | Importing Stored Procedures, Functions, and Packages | 28-53 |
| 28.17.8 | Importing Java Objects | 28-53 |
| 28.17.9 | Importing External Tables | 28-54 |
| 28.17.10 | Importing Advanced Queue (AQ) Tables | 28-54 |
| 28.17.11 | Importing LONG Columns | 28-54 |
| 28.17.12 | Importing LOB Columns When Triggers Are Present | 28-55 |
| 28.17.13 | Importing Views | 28-55 |
| 28.17.14 | Importing Partitioned Tables | 28-56 |
| 28.18 | Support for Fine-Grained Access Control | 28-56 |
| 28.19 | Snapshots and Snapshot Logs | 28-56 |
| 28.19.1 | Snapshot Log | 28-56 |
| 28.19.2 | Snapshots | 28-57 |
| 28.19.2.1 | Importing a Snapshot | 28-57 |
| 28.19.2.2 | Importing a Snapshot into a Different Schema | 28-57 |
| 28.20 | Transportable Tablespaces | 28-58 |
| 28.21 | Storage Parameters | 28-58 |
| 28.21.1 | The OPTIMAL Parameter | 28-59 |
| 28.21.2 | Storage Parameters for OID Indexes and LOB Columns | 28-59 |
| 28.21.3 | Overriding Storage Parameters | 28-59 |
| 28.22 | Read-Only Tablespaces | 28-59 |
| 28.23 | Dropping a Tablespace | 28-60 |
| 28.24 | Reorganizing Tablespaces | 28-60 |
| 28.25 | Importing Statistics | 28-60 |
| 28.26 | Using Export and Import to Partition a Database Migration | 28-61 |
| 28.26.1 | Advantages of Partitioning a Migration | 28-61 |
| 28.26.2 | Disadvantages of Partitioning a Migration | 28-62 |
| 28.26.3 | How to Use Export and Import to Partition a Database Migration | 28-62 |
| 28.27 | Tuning Considerations for Import Operations | 28-62 |
| 28.27.1 | Changing System-Level Options | 28-63 |
| 28.27.2 | Changing Initialization Parameters | 28-63 |
| 28.27.3 | Changing Import Options | 28-64 |
| 28.27.4 | Dealing with Large Amounts of LOB Data | 28-64 |
| 28.27.5 | Dealing with Large Amounts of LONG Data | 28-64 |
| 28.28 | Using Different Releases of Export and Import | 28-65 |
| 28.28.1 | Restrictions When Using Different Releases of Export and Import | 28-65 |
| 28.28.2 | Examples of Using Different Releases of Export and Import | 28-66 |

A Instant Client for SQL*Loader, Export, and Import

| | | |
|-------|---|-----|
| A.1 | What is the Tools Instant Client? | A-1 |
| A.2 | Choosing Which Instant Client to Install | A-2 |
| A.3 | Installing Instant Client Tools by Downloading from OTN | A-3 |
| A.3.1 | Installing Instant Client and Instant Client Tools RPM Packages for Linux | A-3 |
| A.3.2 | Installing Instant Client and Instant Client Tools from Unix or Windows Zip Files | A-4 |
| A.4 | Installing Tools Instant Client from the Client Release Media | A-4 |
| A.5 | List of Oracle Instant Client Tools Files | A-5 |
| A.6 | Configuring Tools Instant Client Package | A-6 |
| A.7 | Connecting to a Database with the Tools Instant Client Package | A-8 |
| A.8 | Uninstalling Tools Instant Client Package and Instant Client | A-9 |

B SQL*Loader Syntax Diagrams
