

ODAT-7 - ORACLE DATABASE 23AI: PERFORMANCE MANAGEMENT AND TUNING

Categoria: **Database**

INFORMAZIONI SUL CORSO



Durata:
5 Giorni



Categoria:
Database



Qualifica Istruttore:
Oracle Certified
Professional



Dedicato a:
Professionista IT



Produttore:
Oracle

OBIETTIVI

PREREQUISITI

CONTENUTI

1: Overview

- Objectives 1-2
- What is performance management? 1-3
- Who manages performance? 1-4
- What does the DBA tune? 1-5
- Types of Tuning 1-6
- Tuning Methodology 1-7
- Effective Tuning Goals 1-9
- General Tuning Session 1-11
- Tuning a CDB 1-13
- Performance Tuning: Diagnostics 1-14
- Performance Tuning: Features and Tools 1-15
- Tuning Objectives 1-16
- Summary 1-17

2: Defining the Scope of Performance Issues

- Objectives 2-2
- Defining the Problem 2-3
- Limit the Scope 2-4
- Determining Tuning Priorities 2-5
- Common Tuning Problems 2-6
- Tuning Life Cycle Phases 2-8
- Tuning During the Life Cycle 2-9
- Application Design and Development 2-10

- Testing: Database Configuration 2-11
- Deployment 2-12
- Production 2-13
- Migration, Upgrade, and Environment Changes 2-14
- ADDM Tuning Session 2-15
- Performance Versus Business Requirements 2-16
- Monitoring and Tuning Tools: Overview 2-17
- Summary 2-19

3: Using the Time Model to Diagnose Performance Issues

- Objectives 3-2
- Time Model: Overview 3-3
- DB Time 3-4
- CPU and Wait Time Tuning Dimensions 3-5
- Time Model Statistics Hierarchy 3-6
- Time Model: Example 3-8
- Top Timed Events 3-9
- Summary 3-10

4: Using Statistics and Wait Events to Diagnose Performance Issues

- Objectives 4-2
- Dynamic Performance Views 4-3
- Dynamic Performance Views: Usage Examples 4-4
- Dynamic Performance Views: Considerations 4-5
- Statistic Levels 4-6
- Instance Activity and Wait Event Statistics 4-8
- System Statistic Classes 4-9
- Displaying Statistics 4-10
- Displaying SGA Statistics 4-11
- Wait Events 4-12
- Using the V\$EVENT_NAME View 4-13
- Wait Classes 4-14
- Displaying Wait Event Statistics 4-15
- Commonly Observed Wait Events 4-17
- Using the V\$SESSION_WAIT View 4-18
- Precision of System Statistics 4-19
- Summary 4-20

5: Using Log and Trace Files to Monitor Performance

- Objectives 5-2
- Viewing the Alert Log 5-3
- Using Alert Log Information as an Aid in Managing Performance 5-5
- Administering the DDL Log File 5-6
- Understanding the Debug Log File 5-7
- User Trace Files 5-8
- Background Processes Trace Files 5-9
- Summary 5-10

6: Using Enterprise Manager Cloud Control and SQL Developer to Monitor Performance

- Objectives 6-2
- Enterprise Manager: Overview 6-3
- Oracle Enterprise Manager Cloud Control Components 6-4
- Using Features of the Oracle Management Packs and Options 6-5
- Oracle SQL Developer 6-6
- SQL Developer Command Line (SQLcl) 6-7
- Summary 6-8

7: Using Statspack to View Performance Data

- Objectives 7-2
- Introduction to Statspack 7-3
- Statspack Scripts 7-4
- Installing Statspack 7-6
- Capturing Statspack Snapshots 7-7
- Configuring Snapshot Data Capture 7-8
- Statspack Snapshot Levels 7-9
- Statspack Baselines and Purging 7-11
- Reporting with Statspack 7-12
- Statspack Considerations 7-13
- Statspack Reports 7-14
- Reading a Statspack Report 7-15
- Statspack Report Drilldown Sections 7-16
- Report Drilldown Examples 7-18
- Load Profile Section 7-19
- Time Model Section 7-20
- Statspack and AWR 7-21
- Summary 7-22

8: Using Automatic Workload Repository

- Objectives 8-2
- Automatic Workload Repository: Overview 8-3
- Automatic Workload Repository Data 8-4
- Workload Repository 8-5
- AWR Administration 8-6
- AWR Snapshot Purging Policy 8-7
- Managing Snapshots with PL/SQL 8-8
- AWR Snapshot Settings 8-9
- Manual AWR Snapshots 8-10
- Managing AWR Data in a Multitenant Environment 8-11
- AWR Snapshots and ADDM in a Multitenant Architecture Database 8-12
- Generating AWR Reports 8-13
- Reading the AWR Report 8-14
- AWR Report: Multitenant Data 8-15
- Generating AWR Reports by Using SQL*Plus 8-16
- Statspack and AWR Reports 8-17
- Reading a Statspack or an AWR Report 8-18
- Compare Periods: Benefits 8-19

- Snapshots and Periods Comparisons 8-20
- Compare Periods: Results 8-21
- Compare Periods: Report 8-22
- Multitenant AWR Views 8-23
- Summary 8-24

9: Using Metrics and Alerts

- Objectives 9-2
- Metrics and Alerts 9-3
- Limitation of Base Statistics 9-4
- Typical Delta Tools 9-5
- Oracle Database Metrics 9-6
- Benefits of Metrics 9-7
- Viewing Metric History Information 9-8
- Viewing Metric Details 9-9
- Statistics Histograms 9-10
- Histogram Views 9-11
- Server-Generated Alerts 9-12
- Alert Usage Model 9-13
- Metric and Alert Views 9-14
- Summary 9-15

10: Using Baselines

- Objectives 10-2
- Comparative Performance Analysis with AWR Baselines 10-3
- Automatic Workload Repository Baselines 10-4
- AWR Baselines 10-5
- Types of Baselines 10-6
- Moving Window Baseline 10-7
- Baseline Templates 10-8
- Creating AWR Baselines 10-9
- Creating a Single AWR Baseline 10-10
- Creating a Repeating Baseline and Template 10-11
- Managing Baselines by Using the DBMS_WORKLOAD_REPOSITORY Package 10-12
- Generating a Baseline Template for a Single Time Period 10-13
- Creating a Repeating Baseline Template 10-14
- Baseline Views 10-15
- Performance Monitoring and Baselines 10-16
- Summary 10-17

11: Managing Automated Maintenance Tasks

- Objectives 11-2
- Automated Maintenance Tasks 11-3
- Maintenance Windows 11-5
- Default Maintenance Plan 11-6
- Automated Maintenance Task Priorities 11-7
- Configuring Automated Maintenance Tasks 11-8
- Summary 11-9

12: Using ADDM to Analyze Performance

- Objectives 12-2
- ADDM Performance Monitoring 12-3
- ADDM and Database Time 12-4
- DB Time-Graph and ADDM Methodology 12-5
- Top Performance Issues Detected 12-7
- ADDM Recommendations 12-8
- Creating a Manual ADDM Task 12-9
- ADDM Tasks in a Multitenant Architecture Database 12-10
- Changing ADDM Attributes 12-11
- Retrieving ADDM Reports by Using SQL 12-12
- Compare Period ADDM: Analysis 12-13
- Workload Compatibility 12-14
- Configuring Automatic ADDM Analysis at the PDB Level 12-15
- Using the DBMS_ADDM Package to Compare Periods 12-16
- Example: Using the DBMS_ADDM Package to Compare Periods 12-17
- Summary 12-18

13: Using Active Session History Data for First Fault System Analysis

- Objectives 13-2
- Active Session History: Overview 13-3
- Active Session History: Mechanics 13-4
- ASH Sampling: Example 13-5
- Accessing ASH Data 13-6
- Analyzing the ASH Data 13-7
- Using Enterprise Manager to View ASH Analysis 13-8
- Using Enterprise Manager to Generate ASH Reports 13-9
- Using the ASH Report Script to Generate a Report 13-10
- ASH Report Structure 13-11
- Determining the Source of Data 13-12
- Performing Skew Analysis 13-13
- Additional Automatic Workload Repository Views 13-14
- Summary 13-15

14: Using Emergency Monitoring and Real-Time ADDM to Analyze Performance Issues

- Objectives 14-2
- Emergency Monitoring: Challenges 14-3
- Emergency Monitoring: Goals 14-4
- Using Real-Time ADDM to Perform a Root Cause Analysis 14-5
- Using Real-Time ADDM 14-6
- Real-Time ADDM in the Database 14-8
- Using Real-Time ADDM 14-10
- Viewing Real-Time ADDM Results 14-11
- Summary 14-12

15: Overview of SQL Statement Processing

- Objectives 15-2

- SQL Statement Processing Phases 15-3
- Parsing 15-4
- SQL Cursor Storage 15-5
- Session Cursor Cache 15-6
- Cursor Usage and Parsing 15-7
- SQL Statement Processing Phases: Bind 15-8
- SQL Statement Processing Phases: Execute and Fetch 15-9
- Processing a DML Statement 15-10
- Commit Processing 15-12
- Identifying Poorly Performing SQL Statements 15-13
- Top SQL Reports 15-14
- SQL Monitoring 15-15
- Monitored SQL Execution Details 15-16
- Summary 15-17

16: Maintaining Indexes

- Objectives 16-2
- Creating Indexes 16-3
- Using Invisible and Unusable Indexes 16-4
- Dropping Indexes 16-5
- Reducing the Cost of SQL Operations 16-6
- Index Maintenance 16-7
- Using Advanced Index Compression 16-9
- Other Index Options 16-10
- SQL Access Advisor 16-11
- Automatic Indexing Task 16-12
- Automatic Index Task Workflow 16-14
- Automatic Indexing Task Reporting 16-16
- Automatic Indexing Views 16-18
- Summary 16-19

17: Maintaining Tables

- Objectives 17-2
- Reducing the Cost of SQL Operations 17-3
- Table Maintenance for Performance 17-4
- Table Reorganization Methods 17-5
- Space Management 17-6
- Extent Management 17-7
- Locally Managed Extents 17-8
- Large Extents: Considerations 17-9
- How Table Data Is Stored 17-11
- Anatomy of a Database Block 17-12
- Minimizing Block Visits 17-13
- Block Allocation 17-14
- Free Lists 17-15
- Block Space Management 17-16
- Block Space Management with Free Lists 17-17
- Automatic Segment Space Management 17-19

- Automatic Segment Space Management at Work 17-20
- Block Space Management with ASSM 17-21
- Creating an Automatic Segment Space Management Segment 17-22
- Migration and Chaining 17-23
- Guidelines for PCTFREE and PCTUSED 17-25
- Detecting Migration and Chaining 17-26
- Selecting Migrated Rows 17-27
- Eliminating Migrated Rows 17-28
- Shrinking Segments: Overview 17-30
- Shrinking Segments: Considerations 17-31
- Shrinking Segments by Using SQL 17-32
- Segment Shrink: Basic Execution 17-33
- Segment Shrink: Execution Considerations 17-34
- Data Compression 17-35
- Advanced Row Compression: Overview 17-37
- Advanced Row Compression: Concepts 17-38
- Using Advanced Row Compression 17-39
- Advanced Row Compression for DML Operations 17-40
- Advanced Index Compression 17-41
- How does Hybrid Columnar Compression work? 17-42
- Using the Compression Advisor 17-43
- Using the Compression Advisor for Indexes 17-44
- Viewing Table Compression Information 17-45
- Summary 17-46

18: Introduction to Query Optimizer

- Objectives 18-2
- Role of the Oracle Optimizer 18-3
- Functions of the Query Optimizer 18-5
- Selectivity 18-7
- Cardinality and Cost 18-8
- Changing Optimizer Behavior 18-9
- Setting and Viewing Optimizer Parameters 18-11
- Using Initialization Parameters to Control Optimizer Behavior 18-12
- Enabling Query Optimizer Features 18-14
- Influencing the Optimizer Approach 18-15
- Optimizing SQL Statements 18-16
- Access Paths 18-17
- Choosing an Access Path 18-18
- Summary 18-19

19: Understanding Execution Plans

- Objectives 19-2
- What is an execution plan? 19-3
- Methods for Viewing Execution Plans 19-4
- Uses of Execution Plans 19-5
- DBMS_XPLAN Package: Overview 19-6
- EXPLAIN PLAN Command 19-8

- EXPLAIN PLAN Command: Example 19-9
- EXPLAIN PLAN Command: Output 19-10
- Reading an Execution Plan 19-11
- Using the V\$SQL_PLAN View 19-12
- Querying V\$SQL_PLAN 19-13
- V\$SQL_PLAN_STATISTICS View 19-14
- Querying the AWR 19-15
- SQL*Plus AUTOTRACE 19-16
- Using SQL*Plus AUTOTRACE 19-17
- SQL*Plus AUTOTRACE: Statistics 19-18
- Adaptive Execution Plans 19-19
- Dynamic Plans 19-20
- Dynamic Plan: Adaptive Process 19-21
- Dynamic Plans: Example 19-22
- Continuous Adaptive Query Plans 19-23
- Automatic Re-Optimization 19-24
- Comparing Execution Plans 19-25
- Summary 19-26

20: Viewing Execution Plans by Using SQL Trace and TKPROF

- Objectives 20-2
- SQL Trace Facility 20-3
- How to Use the SQL Trace Facility 20-5
- Initialization Parameters 20-6
- Enabling SQL Trace 20-8
- Disabling SQL Trace 20-9
- Formatting Your Trace Files 20-10
- TKPROF Command Options 20-11
- Output of the TKPROF Command 20-13
- TKPROF Output with No Index: Example 20-18
- TKPROF Output with Index: Example 20-19
- Generating an Optimizer Trace 20-20
- Summary 20-21

21: Managing Optimizer Statistics

- Objectives 21-2
- Optimizer Statistics 21-3
- Types of Optimizer Statistics 21-4
- Optimizer Statistics Collection 21-5
- Dynamic Statistics 21-7
- Gathering Statistics and Setting Optimizer Statistics Preferences 21-8
- Setting Statistic Preferences 21-9
- Viewing and Managing Optimizer Statistics Preferences 21-11
- Extended Statistics 21-12
- Maintaining Optimizer Statistics 21-13
- Automated Maintenance Tasks 21-14
- Optimizer Statistics Advisor 21-15
- Optimizer Statistics Advisor Report 21-16

- Executing Optimizer Statistics Advisor Tasks 21-17
- Restoring Statistics 21-18
- Deferred Statistics Publishing: Overview 21-19
- Deferred Statistics Publishing: Example 21-21
- Managing Real-Time Statistics 21-22
- Configuring High-Frequency Automatic Optimizer Statistics Collection 21-23
- Summary 21-24

22: Using Automatic SQL Tuning

- Objectives 22-2
- Automatic SQL Tuning: Overview 22-3
- SQL Statement Profiling 22-4
- Plan Tuning Flow and SQL Profile Creation 22-5
- SQL Tuning Loop 22-6
- Using SQL Profiles 22-7
- Summary 22-8

23: Using the SQL Plan Management Feature

- Objectives 23-2
- SQL Plan Management: Overview 23-3
- SQL Plan Baseline: Architecture 23-4
- Loading SQL Plan Baselines 23-5
- Loading SQL Plan Baselines from AWR 23-6
- Evolving SQL Plan Baselines 23-7
- Important SQL Plan Baseline Attributes 23-8
- SQL Plan Selection 23-9
- Possible SQL Plan Manageability Scenarios 23-11
- SQL Performance Analyzer and SQL Plan Baseline Scenario 23-12
- Loading a SQL Plan Baseline Automatically 23-13
- Purging SQL Management Base Policy 23-14
- Enterprise Manager and SQL Plan Baselines 23-15
- Automatic SQL Plan Management 23-16
- SPM Evolve Advisor 23-19
- Summary 23-24

24: Overview of the SQL Advisors

- Objectives 24-2
- SQL Tuning Process 24-3
- SQL Tuning Advisor: Overview 24-4
- SQL Access Advisor: Overview 24-6
- SQL Performance Analyzer: Overview 24-7
- Summary 24-9

25: Using the SQL Tuning Advisor

- Objectives 25-2
- SQL Tuning Advisor: Overview 25-3
- SQL Tuning Advisor Architecture 25-5
- Automatic Tuning Optimizer 25-6

- Using the SQL Tuning Advisor 25-7
- SQL Tuning Advisor Options 25-8
- SQL Tuning Advisor Recommendations 25-9
- Alternative Execution Plans 25-10
- Summary 25-12

26: Using the SQL Access Advisor

- Objectives 26-2
- SQL Access Advisor: Overview 26-3
- Using the SQL Access Advisor 26-4
- Viewing Recommendations 26-5
- Viewing Recommendation Details 26-6
- Practice Overview 26-7
- Summary 26-8

27: Overview of Real Application Testing Components

- Objectives 27-2
- Real Application Testing: Overview 27-3
- Real Application Testing: Use Cases 27-5
- Summary 27-6

28: Using SQL Performance Analyzer to Determine the Impact of Changes

- Objectives 28-2
- SQL Performance Analyzer: Process 28-3
- Steps 6–7: Comparing/Analyzing Performance and Tuning Regressed SQL 28-5
- Capturing the SQL Workload 28-6
- Creating a SQL Performance Analyzer Task 28-7
- SQL Performance Analyzer Task Page 28-9
- SQL Performance Analyzer Comparison Report 28-10
- SQL Performance Analyzer: PL/SQL Example 28-11
- Tuning Regressed SQL Statements 28-13
- SQL Performance Analyzer: Data Dictionary Views 28-14
- Summary 28-15

29: Using Database Replay to Test System Performance

- Objectives 29-2
- Using Database Replay 29-3
- The Big Picture 29-4
- System Architecture: Capture 29-5
- System Architecture: Processing the Workload 29-7
- System Architecture: Replay 29-8
- Database Replay Workflow in Enterprise Manager 29-9
- Accessing Database Replay in Enterprise Manager 29-10
- Capture Considerations 29-11
- Replay Considerations: Preparation 29-13
- Replay Considerations 29-14
- Replay Customized Options 29-15
- Replay Analysis 29-16

- Database Replay Packages 29-18
- Data Dictionary Views: Database Replay 29-19
- Database Replay: PL/SQL Example 29-20
- Calibrating Replay Clients 29-22
- Capturing and Replaying in a CDB and PDBs 29-23
- Reporting 29-24
- Summary 29-26

30: Implementing Real-Time Database Operation Monitoring

- Objectives 30-2
- Overview 30-3
- Use Cases 30-4
- Defining a DB Operation 30-5
- Scope of a Composite DB Operation 30-6
- Database Operation Concepts 30-7
- Identifying a Database Operation 30-8
- Enabling Monitoring of Database Operations 30-9
- Identifying, Starting, and Completing a Database Operation 30-10
- Monitoring Database Operations in Sessions 30-11
- Monitoring the Progress of a Database Operation 30-12
- Monitoring SQL Operation Details 30-13
- Database Operation View: V\$SQL_MONITOR 30-14
- Database Operation Views 30-15
- Reporting Database Operations by Using Functions 30-16
- Database Operation Tuning 30-17
- Summary 30-18

31: Using Services to Monitor Applications

- Objectives 31-2
- What is a service? 31-3
- Service Attributes 31-4
- Service Types 31-5
- Creating Services 31-6
- Using the DBMS_SERVICE Package to Manage Services 31-7
- Where are services used? 31-8
- Using Services with Client Applications 31-9
- Using Services with the Resource Manager 31-10
- Using Enterprise Manager to Manage Consumer Group Mappings 31-11
- Services and the Resource Manager: Example 31-12
- Using Enterprise Manager to Create a Job Class 31-13
- Using Enterprise Manager to Create a Job 31-14
- Services and the Scheduler: Example 31-15
- Using Services with Metric Thresholds 31-16
- Using Enterprise Manager to Change Service Thresholds 31-17
- Services and Metric Thresholds: Example 31-18
- Service Aggregation and Tracing 31-19
- Services Statistics in Performance Home Page 31-20
- Top Services Statistics Using ASH Analytics Performance Page 31-21

- Service Aggregation Configuration 31-22
- Service Aggregation: Example 31-23
- Client Identifier Aggregation and Tracing 31-24
- Using the TRCSESS Utility 31-25
- Service Performance Views 31-26
- Summary 31-28

32: Overview of Memory Structures

- Objectives 32-2
- Managing Memory Caches and Structures 32-3
- Guidelines for Efficient Memory Usage 32-4
- Unified Memory 32-6
- Summary 32-9

33: Managing Shared Pool Performance

- Objectives 33-2
- Shared Pool Architecture 33-3
- Shared Pool Operation 33-4
- Library Cache 33-5
- Latch and Mutex 33-7
- Latch and Mutex: Views and Statistics 33-9
- Diagnostic Tools for Tuning the Shared Pool 33-11
- AWR/Statspack Indicators 33-13
- Top Timed Events 33-14
- Time Model 33-15
- Load Profile 33-16
- Instance Efficiencies 33-17
- Library Cache Activity 33-18
- Avoid Hard Parses 33-19
- Are cursors being shared? 33-20
- Candidate Cursors for Sharing 33-21
- Sharing Cursors 33-22
- Adaptive Cursor Sharing: Example 33-24
- Adaptive Cursor Sharing Views 33-26
- Interacting with Adaptive Cursor Sharing 33-27
- Reducing the Cost of Soft Parses 33-28
- Sizing the Shared Pool 33-29
- Shared Pool Advisory 33-30
- Shared Pool Advisory in an AWR Report 33-31
- Shared Pool Advisor 33-32
- Avoiding Fragmentation 33-33
- Large Memory Requirements 33-34
- Tuning the Shared Pool Reserved Pool 33-36
- Keeping Large Objects 33-38
- Data Dictionary Cache 33-40
- Dictionary Cache Misses 33-41
- SQL Query Result Cache: Overview 33-42
- Managing the SQL Query Result Cache 33-43

- Using the RESULT_CACHE Hint 33-45
- Using Table Annotation to Control Result Caching 33-46
- Using the DBMS_RESULT_CACHE Package 33-47
- Viewing SQL Result Cache Dictionary Information 33-48
- SQL Query Result Cache: Considerations 33-49
- Summary 33-50

34: Managing Buffer Cache Performance

- Objectives 34-2
- Buffer Cache: Highlights 34-3
- Database Buffers 34-4
- Buffer Hash Table for Lookups 34-6
- Working Sets 34-7
- Tuning Goals and Techniques 34-9
- Symptoms of a Buffer Cache Issue 34-11
- Cache Buffer Chains Latch Contention 34-12
- Finding Hot Segments 34-13
- Buffer Busy Waits 34-14
- Buffer Cache Hit Ratio 34-15
- Buffer Cache Hit Ratio is Not Everything 34-16
- Interpreting Buffer Cache Hit Ratio 34-17
- Read Waits 34-18
- Free Buffer Waits 34-19
- Solutions for Buffer Cache Issues 34-20
- Sizing the Buffer Cache 34-21
- Buffer Cache Size Parameters 34-22
- Dynamic Buffer Cache Advisory Parameter 34-23
- Buffer Cache Advisory View 34-24
- Using the V\$DB_CACHE_ADVICE View 34-25
- Using the Buffer Cache Advisor 34-26
- Caching Tables 34-27
- Automatic Big Table Caching 34-28
- Configuring Automatic Big Table Caching 34-29
- Using Automatic Big Table Caching 34-30
- Monitoring Automatic Big Table Caching 34-31
- Memoptimized Rowstore 34-32
- In-Memory Hash Index 34-33
- Multiple Buffer Pools 34-34
- Enabling Multiple Buffer Pools 34-35
- Calculating the Hit Ratio for Multiple Pools 34-36
- Multiple Block Sizes 34-37
- Multiple Database Writers 34-38
- Multiple I/O Slaves 34-39
- Using Multiple Writers and I/O Slaves 34-40
- Private Pool for I/O-Intensive Operations 34-41
- Automatically Tuned Multiblock Reads 34-42
- Database Smart Flash Cache Overview 34-43
- Using Database Smart Flash Cache 34-44

- Database Smart Flash Cache Architecture: Overview 34-45
- Configuring Database Smart Flash Cache 34-46
- Sizing Database Smart Flash Cache 34-47
- Enabling and Disabling Flash Devices 34-48
- Specifying Database Smart Flash Cache for a Table 34-49
- Full Database In-Memory Caching 34-50
- Setting Up Force Full Database Caching 34-51
- Monitoring Full Database In-Memory Caching 34-52
- Flushing the Buffer Cache (for Testing Only) 34-53
- Summary 34-54
- Practice Overview 34-55

35: Managing PGA and Temporary Space Performance

- Objectives 35-2
- SQL Memory Usage 35-3
- Performance Impact 35-4
- Automatic PGA Memory 35-5
- SQL Memory Manager 35-6
- Configuring Automatic PGA Memory 35-8
- Setting PGA_AGGREGATE_TARGET Initially 35-9
- Limiting the Size of the Program Global Area 35-10
- Managing the PGA for PDBs 35-11
- Monitoring SQL Memory Usage 35-12
- Monitoring SQL Memory Usage: Examples 35-14
- Tuning SQL Memory Usage 35-15
- PGA Target Advice Statistics 35-16
- PGA Target Advice Histograms 35-17
- Automatic PGA and Enterprise Manager 35-18
- Automatic PGA and AWR Reports 35-19
- Temporary Tablespace Management: Overview 35-20
- Temporary Tablespace: Locally Managed 35-21
- Configuring Temporary Tablespace 35-22
- Temporary Tablespace Group: Overview 35-24
- Temporary Tablespace Group: Benefits 35-25
- Creating Temporary Tablespace Groups 35-26
- Maintaining Temporary Tablespace Groups 35-27
- Viewing Tablespace Groups 35-28
- Monitoring Temporary Tablespace 35-29
- Shrinking a Temporary Tablespace 35-30
- Using the Tablespace Option When Creating a Temporary Table 35-31
- Summary 35-32

36: Configuring the Large Pool

- Objectives 36-2
- Large Pool Overview 36-3
- Tuning the Large Pool 36-4
- Summary 36-5

37: Using Automatic Shared Memory Management

- Objectives 37-2
- Oracle Database Architecture 37-3
- Granules 37-4
- Automatic Shared Memory Management: Overview 37-5
- SGA Sizing Parameters: Overview 37-6
- Dynamic SGA Transfer Modes 37-7
- Memory Broker Architecture 37-8
- Manually Resizing Dynamic SGA Parameters 37-9
- Behavior of Auto-Tuned SGA Parameters 37-10
- Behavior of Manually Tuned SGA Components 37-11
- Using the V\$SYSTEM_PARAMETER View 37-12
- Resizing SGA_TARGET 37-13
- Disabling Automatic Shared Memory Management 37-14
- Using the SGA Advisor 37-15
- Monitoring ASMM 37-16
- Managing SGA for PDBs 37-17
- Summary 37-18

38: Introduction to In-Memory Column Store

- Objectives 38-2
- Database In-Memory Feature Set 38-3
- Goals of In-Memory Column Store 38-5
- Benefits 38-7
- Overview 38-8
- Row Store Versus Column Store: 2D Vision 38-10
- In-Memory Column Unit 38-11
- Compare: In-Memory Column Store Cache and Buffer Cache 38-12
- Dual Format In-Memory 38-13
- Indexes Issues 38-14
- Process 38-15
- In-Memory Column Store: Dual Format of Segments in SGA 38-16
- Using OEM to Manage In-Memory Pool 38-17
- Summary 38-18

39: Configuring the In-Memory Column Store Feature

- Objectives 39-2
- Deploying IM Column Store 39-3
- Using OEM to Manage In-Memory Pool 39-4
- Deploying IM Column Store: Objects Setting 39-5
- Deploying IM Column Store: Columns Setting 39-6
- Defining IM Column Store Compression 39-7
- In-Memory Advisor 39-8
- IM Advisor or Compression Advisor? 39-9
- Computing Compression Ratio 39-10
- IM FastStart 39-11
- Automatic In-Memory: Overview 39-12
- AIM Action 39-13

- Configuring Automatic In-Memory 39-14
- Diagnostic Views 39-15
- Summary 39-16

40: Using the In-Memory Column Store Feature to Improve SQL Performance

- Objectives 40-2
- Query Benefits 40-3
- Testing and Comparing Query Performance 40-4
- Queries on In-Memory Tables: Simple Predicate 40-5
- MINMAX Pruning Statistics 40-6
- IM Column Store Statistics 40-7
- Execution Plan: TABLE ACCESS IN MEMORY FULL 40-8
- Queries on In-Memory Tables: Join 40-9
- Execution Plan: JOIN FILTER CREATE / USE 40-10
- Queries on In-Memory Tables: Join Groups 40-11
- Population of Expressions and Virtual Columns Results 40-12
- In-Memory Expression Unit (IMEU) 40-14
- Populating In-Memory Expression Results 40-15
- Populating In-Memory Expression Results Within a Window 40-16
- Waiting for In-Memory Segments to Be Populated 40-17
- Views 40-18
- Summary 40-19

41: Using In-Memory Column Store with Oracle Database Features

- Objectives 41-2
- Interaction with Other Products 41-3
- Optimizer 41-4
- IM Column Store and Real Application Clusters 41-6
- IM Column Store and Data Pump 41-7
- Data Pump TRANSFORM Names 41-8
- Automatic Data Optimization Interaction 41-9
- Managing Heat Map and Automatic Data Optimization Policies 41-10
- Creating ADO In-Memory Policies 41-11
- Summary 41-12

INFO

Materiale didattico: Materiale didattico ufficiale Oracle in formato digitale

Costo materiale didattico: incluso nel prezzo del corso a Calendario

Natura del corso: Operativo (previsti lab su PC)