

Material Snippet of Course Part 1

Class1:

1 General Setup:

2 Few important handysql:

Class2:

1 HISTOGRAM

1 Cardinality Algorithm

Endpoint numbers and values:

Popular and Non-Popular values:

Bucket compression

2 TYPES OF HISTOGRAMS

FREQUENCY HISTOGRAM:

TOP FREQUENCY HISTOGRAM: (12C)

Height balance histogram:

Hybrid Histogram: (12c)

3 How we calculate rows to get desired percentage:

Class3:

1 JOIN METHODS

1 HASH JOIN

2 NESTED LOOP JOIN

3 SORT-MERGE

Class4

1 About B-Tree Index Access

How Index Storage Affects Index Scans

Unique and Nonunique Indexes

B-Tree Indexes and Nulls

2 Full Table Scans

1 When the Optimizer Considers a Full Table Scan

2 How a Full Table Scan Works

3 Full Table Scan: Example

3 Table Access by Index Rowid

1 When the Optimizer Chooses Table Access by Rowid

2 How Table Access by Rowid Works

3 Table Access by Rowid: Example

4 Sample Table Scans

1 When the Optimizer Chooses a Sample Table Scan

54 TABLE ACCESS BY INDEX ROWID BATCHED

65 INDEX UNIQUE SCAN

76 INDEX RANGE SCAN

1 In general, the process is as follows:

87 FULL INDEX SCAN

98 FAST FULL INDEX SCAN

109 INDEX SKIP SCAN

When the Optimizer Considers Index Skip Scans

How Index Skip Scans Work

Example Composite Index Entries

1110 INDEX JOIN SCAN

When the Optimizer Considers Index Join Scans

HowIndexJoinScansWork

Class5

1BitmapIndexAccessPaths

1AboutBitmapIndexAccess

PurposeofBitmapIndexes

BitmapsandRowids

BitmapJoinIndexes

BitmapStorage

BitmapConversiontoRowid

WhentheOptimizerChoosesBitmapConversiontoRowid

BitmapConversiontoRowid:Example

2BitmapIndexSingleValue

WhentheOptimizerConsidersBitmapIndexSingleValue

HowBitmapIndexSingleValueWorks

BitmapIndexSingleValue:Example

3BitmapIndexRangeScans

WhentheOptimizerConsidersBitmapIndexRangeScans

HowBitmapIndexRangeScansWork

BitmapIndexRangeScans:Example

4BitmapMerge

WhentheOptimizerConsidersBitmapMerge

BitmapMerge:Example

Class

1TableClusterAccessPaths

1ClusterScans

WhentheOptimizerConsidersClusterScans

HowClusterScansWork

ClusterScans:Example

2HashScans

WhentheOptimizerConsidersaHashScan

HowaClusterScanWorks

ClusterScan:Example

2WhatisTableClusters

1OverviewofIndexedClusters

IndexClusterCreation

IndexClusterWorks

2OverviewofHashClusters

HashClusterCreation

HashClusterQueries

HashClusterVariations

HashClusterStorage

Class7

Tablepartitioning

1BasicsofPartitioning

2PartitioningKey

3PartitionedTables

4WhentoPartitionaTable

5WhentoPartitionanIndex

6PartitionedIndex-OrganizedTables

7SystemPartitioning

8PartitioningforInformationLifecycleManagement

9PartitioningandLOBData

10 Collections in XML Type and Object Data

2 Benefits of Partitioning

1 Partitioning for Performance

Partition Pruning

Partition-Wise Joins

2 Partitioning for Manageability

3 Partitioning for Availability

3 Partitioning Strategies

1 Single-Level Partitioning

Range Partitioning

Hash Partitioning

List Partitioning

2 Composite Partitioning

Composite Range-Range Partitioning

Composite Range-Hash Partitioning

Composite Range-List Partitioning

Composite List-Range Partitioning

Composite List-Hash Partitioning

Composite List-List Partitioning

3 Partitioning Extensions

Manageability Extensions

Interval Partitioning

Partition Advisor

4 Partitioning Key Extensions

Reference Partitioning

Virtual Column-Based Partitioning

4 Overview of Partitioned Indexes

Deciding on the Type of Partitioned Index to Use

1 Local Partitioned Indexes

Global Range Partitioned Indexes

Global Hash Partitioned Indexes

Maintenance of Global Partitioned Indexes

2 Global Nonpartitioned Indexes

3 Miscellaneous Information about Creating Indexes on Partitioned Tables

4 Partitioned Indexes on Composite Partitions

Class 8

1 Managing Index-Organized Tables

1 What Are Index-Organized Tables?

2 Creating Index-Organized Tables

Example: Creating an Index-Organized Table

Restrictions for Index-Organized Tables

Creating Index-Organized Tables that Contain Object Types

Choosing and Monitoring a Threshold Value

Using the INCLUDING Clause

Parallelizing Index-Organized Table Creation

Using Key Compression

3 Maintaining Index-Organized Tables

Altering Index-Organized Tables

Moving (Rebuilding) Index-Organized Tables

4 Creating Secondary Indexes on Index-Organized Tables

Syntax for Creating the Secondary Index

Maintaining Physical Guesses in Logical Rowids

- BitmapIndexes
- 5AnalyzingIndex-OrganizedTables
 - CollectingOptimizerStatisticsforIndex-OrganizedTables
 - ValidatingtheStructureofIndex-OrganizedTables
- 6UsingtheORDERBYClausewithIndex-OrganizedTables
- 7ConvertingIndex-OrganizedTablestoRegularTables
- 2OptimizerStatisticsConcepts
 - 1IntroductiontoOptimizerStatistics
 - 2TableStatistics
 - 3ColumnStatistics
 - 4IndexStatistics
 - 5IndexOptimizerHint
 - 6OptimizersHint
 - TypeofHints
 - SyntaxofHint
 - SpecifyingaFullSetofHints
 - SpecifyingaQueryBlockinaHint
 - SpecifyingGlobalTableHints
 - 7UsingOptimizerHints
 - HintsforOptimizationApproachesandGoals
 - ALL_ROWS
 - FIRST_ROWS(n)
 - RULE
 - 8HintsforAccessPaths
 - FULL
 - CLUSTER
 - HASH
 - INDEX
 - NO_INDEX
 - INDEX_ASC
 - INDEX_COMBINE
 - INDEX_JOIN
 - INDEX_DESC
 - INDEX_FFS
 - NO_INDEX_FFS
 - INDEX_SS
 - INDEX_SS_ASC
 - INDEX_SS_DESC
 - NO_INDEX_SS
 - 9HintsforQueryTransformations
 - NO_QUERY_TRANSFORMATION
 - USE_CONCAT
 - NO_EXPAND
 - REWRITE
 - NO_REWRITE
 - MERGE
 - NO_MERGE
 - STAR_TRANSFORMATION
 - NO_STAR_TRANSFORMATION
 - FACT
 - NO_FACT
 - UNNEST

NO_UNNEST
10HintsforJoinOrders
LEADING
ORDERED
11HintsforJoinOperations
USE_NL
NO_USE_NL
USE_NL_WITH_INDEX
USE_MERGE
NO_USE_MERGE
USE_HASH
NO_USE_HASH
12HintsforParallelExecution
PARALLEL
NO_PARALLEL
PQ_DISTRIBUTE

Class9

1DatabaseOptimizerParameters

Class10

1AWR

2ADDM

3ALERTLOG

Class11

1WhatIsDatabasesizing

2DatabasesizingrespecttoRAM,CPU,PGAandSGA