



Data Warehousing



Corporate Trainer's Profile

Corporate Trainers are having the experience of 4 to 12 years in development , working with TOP CMM level 5 companies (Project Leader /Project Manager) qualified from NIT/IIT/IIM and work exp in USA and UK.

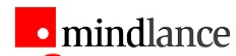


CMM (Capability Maturity Model) level Project Standard:-

The Capability Maturity Model (CMM) is a method for evaluating the maturity of organizations on a scale of 1 to 5.

Get the Opportunities to work on Client Projects Of US/UK, which follow the all standard of CMM level 5 Company.

Projects



SUMMARY: CURRENCY:
9i & 10g

After presenting entity relationship (ER) and dimensional modelling (DM) as competing alternatives, the presentation will focus on the use of DM techniques when developing and implementing a very large data warehouse. Using real-world business scenarios, SALES, for example, the instructor will coach the participants from logical through physical design of a data warehouse involving at least five dimensions and one-or-more star schemas.

Thus hands-on exercises include:

- Design and creation of dimensions
- Design and creation of star schemas
- Design and creation of hierarchies
- Design and creation of indexes
- Design and creation of integrity constraints
- Extract, Transform and Load (ETL) options
- Parallelism
- Design and create materialized views (summary tables)

Throughout the foregoing, best practices and performance issues are discussed.

PREREQUISITES: Completion of our course 'Data Warehouse Terms, Concepts & Architecture' (DWHTCA) or equivalent knowledge.

DURATION: 5 days

OBJECTIVES: Upon completion of this presentation, the participant should be able to monitor and tune large data warehouses in a BI (Business Intelligence) or DS (Decision Support) environment.

COURSE CONTENT:

1.ORACLE ARCHITECTURAL OVERVIEW

- Processes
- Memory
- Directory
- Dictionary and catalog
- Connectivity
- Replication
- Partitioning
- Database
- Real Application Cluster (RAC)

2.LOGICAL DESIGN CONCEPTS

- Why data modelling
- Requirements analysis
- Normalization vs. denormalization
- Entity relationship modelling
- Dimensional modelling
- OLAP vs. OLTP
- Star vs. snowflake schemas
- Metadata considerations
- Data marts
- Workshop

3.DIMENSIONAL MODELLING DESIGN (1) INITIAL STEPS

- How to establish business requirements
- How to choose a business process (e.g., sales)
- How to determine the business process grain (i.e., level of detail for fact table)
- How to choose dimensions (e.g., time)
- How to identify measurement (numeric facts) to populate the fact table

4.DIMENSIONAL MODELLING DESIGN (2) FACT TABLE

- Definition
- Granularity selection
- Measurements
- Additive vs. non-additive measures
- Foreign keys
- Joins with dimension tables
- Staging
- Workshop, case study

5.DIMENSIONAL MODELLING DESIGN (3) HIERARCHIES

- Definition
- Types
- Levels
- Level relationships
- Workshop, case study

6.DIMENSIONAL MODELLING DESIGN (4) INTEGRITY CONSTRAINTS

- Scope and purpose

Unique
NOT NULL
FOREIGN KEY
Enforced vs. not-enforced
Workshop, case study

7.DIMENSIONAL MODELLING DESIGN (5) SCHEMA DESIGN

Star or snowflake
Data warehouse or data mart
Naming conventions
Maintenance requirements
Workshop, case study

8.PHYSICAL DESIGN (1) LARGE DATA WAREHOUSE CONSIDERATIONS

The environment (e.g., machine configuration)
Disk layout and placement (e.g., RAID)
Table sizes (e.g., maximum size for materialized view)
Database partition (e.g., how many?)
Partition key considerations
Initialisation parameters
Buffer pools
Data warehouse loads (e.g., parallelism options)

9.PHYSICAL DESIGN (2) OBJECTS

Table spaces
Tables (partitioned vs. non-partitioned)
Index options
Integrity constraints
Materialized views (i.e., summary tables)
Creation of dimensions
Creation of hierarchies
I/O design considerations (e.g., striping and redundancy)
Best practices
Workshop, case study

10.PHYSICAL DESIGN (3) PARALLELISM

Definitions
When to consider (e.g., bulk loads, summaries)
How to enable parallelism

- Hardware requirements
- Query parallelism
- Partitioned and non-partitioned tables
- Data manipulation
- Types of parallelism (e.g., DML, DDL)
- How parallelism works
- Restrictions
- Best practices
- Workshop, case study

11.PHYSICAL DESIGN (4) PARTITIONING

- Definition
- Types
- When to consider
- Table compression
- Partition pruning
- Join techniques
- Range partitioning
- Index partitioning
- Best practices

12.PHYSICAL DESIGN (5) INDEXES

- Bitmap indexes
- B-tree indexes
- Compression
- Global vs. local indexes
- Best practices
- Workshop, case study

13.PHYSICAL DESIGN (6) INTEGRITY CONSTRAINTS

- Rationale
- Constraint states
- Unique constraints
- Foreign key constraints
- Enforced vs. not-enforced constraints
- Materialized views considerations
- Query rewrite considerations
- Best practices
- Workshop, case study

14.PHYSICAL DESIGN (7) CREATE DIMENSIONS

- Dimension hierarchical specification
- Integrity constraints
- Dimension validation
- Dimension maintenance (e.g., ALTER)
- Best practices
- Workshop, case study

15.PHYSICAL DESIGN (8) MATERIALIZED VIEW CREATION & MAINTENANCE

- Use cases
- Materialized view types
- How to create
- How to refresh
- How to partition
- How to tune
- Logs (e.g., staging options)
- Security considerations
- Query rewrite considerations

16.PHYSICAL DESIGN (9) ETL

- Options
- Extraction options
- Transformation options
- Loading options
- Change data capture and publishing

17.INTRODUCTION TO ORACLE DATA WAREHOUSING TOOLS

- Oracle Warehouse Builder
- Oracle Discoverer
- Oracle Reports
- OLAP and data mining

18.INTRODUCTION TO ORACLE SQL ADVISOR

- Use
- Tuning materialized views

19.DW PERFORMANCE CONSIDERATIONS

- Query rewrite
- Schema modelling
- Aggregation
- SQL modelling
- EXPLAIN
- I/O design
- Parallelism
- Initialisation parameters