Data Warehousing With Oracle

By Muhammad Ahmad Shahzad

Abstract

With the emergence of data warehousing, Decision Support Systems have evolved to its best. At the core of these warehousing systems lies a good database management system. Database server, used for data warehousing, is responsible to provide robust data management, scalability, high performance query processing and integration with other servers. Oracle being the initiator in warehousing servers, provides a wide range of features for facilitating data warehousing.

This paper is designed to review the features of data warehousing – conceptualizing the concept of data warehousing and lastly, features of Oracle servers for implementing a data warehouse.

Data Warehouse – A Conceptual Overview

Definition of Data Warehouse

W.H. Inmon, "father of data warehousing", defined data warehouse as: A data warehouse is a *Subject Oriented*, *Integrated*, *Non-volatile*, and *Time-variant* collection of data in support of management's decisions.

With the advancement in the computing technology, the fall in the computer hardware and change in the nature of business – the value of information have raised dramatically. The need of making decisions on the basis of large amount of data, which has the property of diversification along with the hugeness, have raised to a level not comparable to any phase throughout the history of Information Technology. Supplementing was the betterment of server operating systems and the explosion of Internets and Web based applications. The more organized Information database is – the better is the performance of the company. This indispensable requirement to store enormous amount of data lead to the Analytic Systems which in turn gave birth to the idea of Data Warehousing.

Data warehousing is about molding data into information, and storing this information based on the subject rather than application. As mentioned by W.H. Inmon, in one of his articles, the data warehouse environment is the foundation of DSS – Decision Support Systems.

Going back to the definition of data warehouse, the warehouse is a Subject Oriented, Integrated, Non-volatile, and Time-variant collection of data.

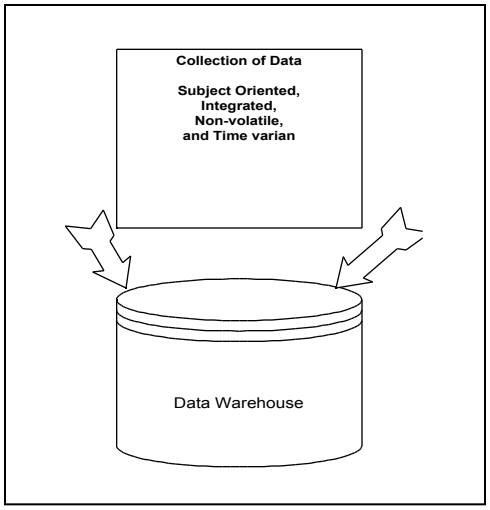


Figure 1

Subject-Oriented

In data warehousing the prime objective of storing data is to facilitate decision process of a company, and within any company data naturally concentrates around subject areas. This leads to the gathering of information around these subjects rather than around the applications or processes.

Integrated

Though the data in the data warehouses is scattered around different tables, databases or even servers but the data is integrated consistently in the values of variables, naming conventions and physical data definitions.

Nonvolatile

Being the snapshot of operational data on a given specific time, the data in the data warehouses should not be changed or updated – once its loaded from operational system. As the snapshot shows operational data at some moment of time and one expects data warehouse to reflect accurate values of that time frame. There exist only two operations – the time-based loading of data, accessing the loaded data.

Time-variant

The value of operational data changes on the basis of time. The time based archival of data from operational systems to data warehouse, makes the value of data, in the data warehouses, being function of time. As data warehouse gives accurate picture of operational data for some given time and the change in the data in warehouse is based on time based change in operational data, data in the data warehouse is called 'time-variant'.

From the operational systems to the requirement of DSS, to designing of data warehousing, to Implement to ongoing support, data warehousing does not use some alien concepts and is more or less based on the typical System Development Life Cycle (SDLC) concept.

Data warehouses possess a degree of multi-dimensioning in there nature. The advocates of Relational Modeling say that Multi-dimensioning of data is just another way of representation of data in two dimensional relational models. If we agree to the above rationale then the data warehousing comes in the umbrella of traditional RDBMS application development process. Yet indeed, there are some major differences when building a warehouse, including features like hugeness of data or accessibility or providing dynamic access etc. The most important difference is of course the way data is placed in data warehouses, its more like summarized, referenced, de-normalized representation. In short what ever or how ever we develop a data warehouse it should at least be capable of providing ad hoc complex, statistical, and analytical queries to facilitate decision making process.

Architecture of data warehouse

As repeatedly mentioned in this paper, the prime concern of providing a separate set of data – the data warehouse, is to facilitate Business Analysts in the process of Decision Making. Essentially data warehousing is the "warehousing" data outside operational systems and this has not significantly changed with the evolution of data warehousing systems. Prime reason of this separation is that the evaluation and analysis, done by analysts, require complex and analytic queries - the effect of which is the performance degradation of operational systems. Another important feature is the combination of data from more than one operational system to provide the ability of cross-referencing.

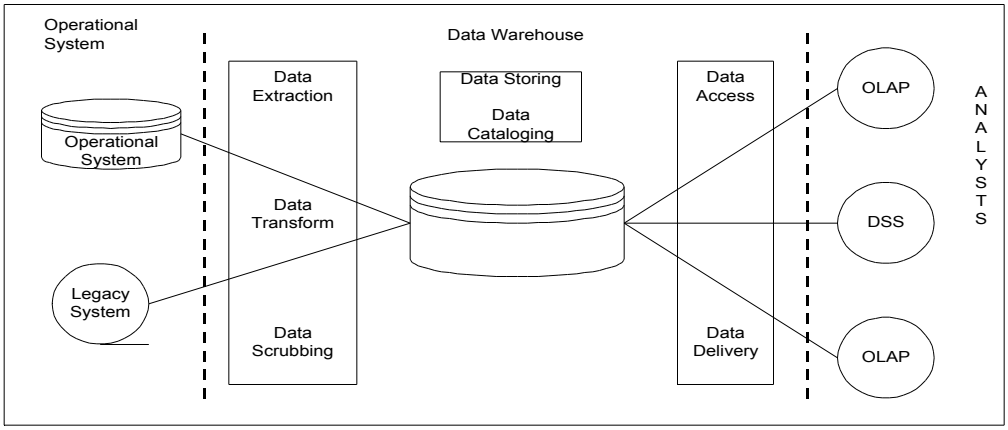


Figure 2

Most of the data warehousing done, posses three-tier architecture.

The base level from which data is extracted is operational system (OLTP) and the legacy systems, from which data is transformed and loaded into the warehouse database. So the middle level is the data warehouse and the top most level is the analytic system (OLAP) and Decision Support System (DSS). OLAP systems utilize the data warehouse to provide multi-dimensional view. Functionally a data warehouse can be divided into following:

- Data Extraction
- Transformation and Scrubbing
- Storing and Cataloging
- Data Access
- Data Delivery

All of the above functions are self-explanatory. The process starts with the extraction of data from operational system and legacy systems, then comes the transformation and cleaning of data during this process summarization and aggregation is also done. Data Storage represents the process of storing transformed and cleaned data in a relational database. Data access holds the query processing, multi-dimensional analysis and data mining. Lastly comes the function of data delivery to the end-users, which may be the part of data warehouse or can come under the umbrella of OLAPs.

A data warehouse, being unique in the class of applications, possesses a structure, which is different from other database applications. Being used for analytic purpose it is designed in a way so that it can facilitate complex queries. Mostly the business analysts focus on the summarized data, time variant data. So the data warehouses are designed to facilitate the above process. Data warehouses hold different levels of summarization and details. It also has two groups of detail data, the current detail data and older detail data.

Current detail data, reflecting the most current happening in the organization, is highly voluminous and is always stored on disk storage. It may reach space as much as gigabytes or even terabytes. The reason of being so sizable is that it asserts lowest level of granularity.

Old detail data, as the name shows, is the data, which is not that frequently, used. Due to the infrequent requirement this data is stored on some cheaper storage mediums like tape cartridge.

Then comes the level of summarization, the difference between lightly summarized and highly summarized is quite obvious. Lightly summarized data is the summary of detailed granulized data. Whereas highly summarized data is more compact than summarized and is based on lightly summarized figures. Both of these reside on disk media, as these are accessed very frequently.

Meta data, being very important data repository, resides on different dimension than other data classes. As it may be accessed by any of the other layers and work as a linkage warehouse and operational environment.

Warehouse Database Server – Its Role in Data Warehousing

Data in the data warehouse database is organized by subject rather than applications or processes, and this data is extracted and refreshed from operational system on a periodic basis. We have already discussed the three-tiered architecture in which first tier is the operational system, middle is the data warehouse database server and last one is front-ended client applications, including DSS and OLAP applications. In three-tiered architecture the warehouse database server works as the heart of warehouse application. Though a simpler form of data warehouse applications exist, in which the architecture is two tiered – tier1 includes the Operational System as well as Warehouse database and tier2 is the client front-end Decision Support applications.

One can't disagree to the fact that Database servers are at the core of every application that supports business decisions, specially data warehouses – providing robust data management and scalable, high-performance query processing.

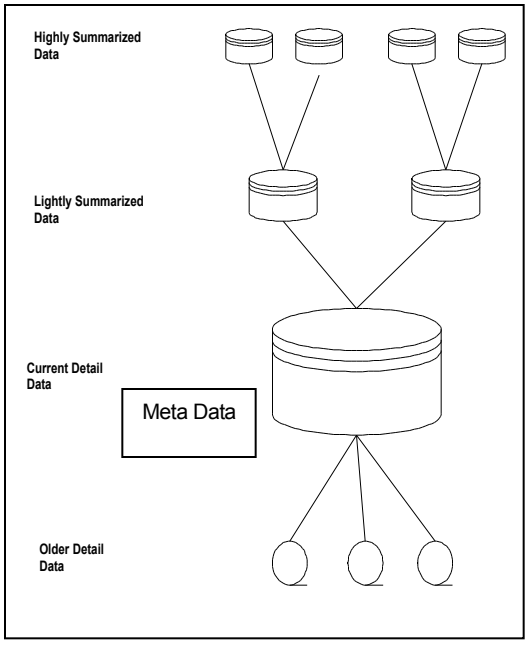


Figure 3

Warehouse servers are categorized in two types, RDBMS (Relational Database) and MDD (Multi-dimensional Database) – the choice is based on type of data stored in warehouse.

RDBMS is based on the concept of mathematical relation operation. The implementation of RDBMS is based on two-dimensional relationship of related data – called the tables. Whereas, MDD can be viewed as cube, where information is pilled on various axes of cube. Taking as an example, the case of Sales production of a company – Sales are related to salespersons, the geographical region, and some time frame, this result in three-dimensional view of data. The cross-section of these three can give the required data. However MDD just work with finite set of data and information which is highly related to each other.

Relational database technology has an edge on MDD, when we are considering huge data storage capacity or portability issue or security. RDBMS is an old and proven technology in data storage and recovery. MDD is popular for its Instance Response, Implementation ease, and integration with Meta-data. Either we choose MDD or RDBMS in both cases a database server has a very central role in the data warehouse architecture.

Data modeling – Star Schema as choice

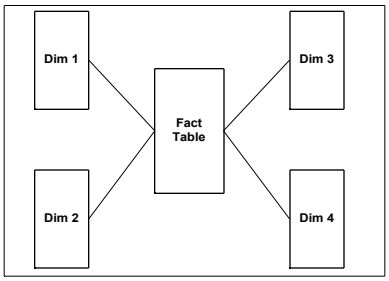
Data modeling, the process of making data models is not unique for warehousing; in fact we use this tool in the development of all kinds of database applications. The reason in data warehousing is pretty much the same.

- Defining the scope of data warehouse
- Viewing the complexity of the relationship between data
- Recognizing and controlling redundancy

Decision-makers during the analysis generally formulate complex queries, which are based on multiple dimensions. As data warehousing is done to facilitate this type of multi-dimensional queries for decision making, the modeling of data also tends to bear multiple dimensions. Data warehousing done using relational database technology generally holds modeling in star schema.

Star schema is the implementation of multiple dimensions in the relational modeling. This schema addresses data navigation difficulty and its dimensions are the categories by which analysts organize the information. Star schema at the lowest level is the relationship between tables, but with the expansion of scope of warehouse the model tends to become more and more complex. So it's a good practice to aggregate data into levels of hierarchy. The relationship among different objects is provided by introducing fact tables – tables having primary key compounded by primary keys of all dimensions.

The fact table is the central table in star architecture containing the data links or points to establish dimensions in different entities. Technically, this table is just intersection of entity primary keys.



Meta Data – the data about data

Meta data provides data repository. Providing both technical and business view of data stored in the data warehouse. It lays out the physical structures that includes:

- Data elements and their types
- Business definition for the data elements
- How to update data and on which frequency
- Different data elements having same meaning
- Valid values for each data elements

Meta data plays very important rule in the definition, building, management and maintenance of data warehouses. In a data warehouse Meta data are categorized into Business and Technical Meta data. Business Meta data describes what's in the warehouse, its meaning – in business terms. The business Meta data lies above technical Meta data, adding some more details to the extracted material. This type of Meta data is important as it facilitates business users and increases the accessibility. In contrast, technical Meta data describes the data elements as they exist in the warehouse. This type of Meta data is used for data modeling, initially, and once the warehouse is erected this Meta data is frequently used by warehouse administrator and software tools.

Using the Internet Technology

With the technology provided through Internets, the transfer of information has become very easy. On the other hand is the requirement of accessing data warehouse globally. Putting both together gives a very effective solution to give the access to data warehouses on the global scale. To provide global access to a data warehouse using web is like giving easy access to data on the whenever, whoever basis. However, there are some issues which has to be sorted before the effective utilization of internet technology, like the usage of web server along with the database server, security issues, and some issues like providing ways for query and report purpose.

Oracle – a choice for Implementing Data Warehouse

Since the conceptualization of data warehouse, many database venders have tried to mold their database systems for accommodating it. Amongst which was Oracle that systematically evolved to address specific needs of warehousing. When considering a data warehouse implemented in a RDBMS, there are some technology requirements like query processing, data storage, scalability, integration with other systems and lastly the security management.

Query Processing

Queries in a data warehouse generally involve very large amount of data. Also it's not rare to find complex operations like multi-table joins, sorting and aggregation in data warehouse queries. These operations are generally set-oriented; operating on some groups of records based on specified criteria. Most of the queries in decision making process are multi-dimensional in nature, based on star schema. Another important feature in query processing of data warehouse is that queries are not pre-defined and are based on the business-users runtime criteria.

Features like query optimization, access and joining methods and parallel execution of queries are very vital for performance of data warehouse.

Data Management and Scalability

This is the way data is loaded, organized, stored, accessed and maintained in a database. The database operations such as data loading, enforcing constraints, building indexes, collecting statistics on the data, reorganizing tables and indexes, building aggregates or summaries, and data purging are included in data management. Its not unusual to find very large databases when implementing a data warehouse, also the growth of a warehouse is in big data leaps. The database operations, listed above, are functions of database size.

To effectively meet the needs of data warehousing, the database server has to provide capacity to deal with large data volumes and data operations should also be tuned for the same reason.

While the scope of data warehouse is not at all limited, this feature leads to the scalability of both users and data. With the globalization of organizations – number of end-users, requiring to use warehouse, have increased dramatically. The supporting of this population of users is the responsibility of database server. This include supporting wide range of hardware, operating systems, and clients – trying to access data warehouse from widely apart physical positioning.

When considering scale of data, server has to support data volumes of gigabytes, terabytes or even beyond. The scalability doesn't merely mean the capacity to store immense data; it encompasses the ability to efficiently process queries, the capability to perform data management operations, and delivering business-critical availability, all at huge scale.

Integration with other systems

In the process of decision making, the analysts have to access data even beyond the boundaries of operational data and its not always wise to transfer each bit of data from systems like this to data warehouse. So database servers should provide provisions to link the warehouse application to systems – like SAP, BAAN or PeopleSoft.

Security Management

With the physical size of data warehouses and number of users requiring to access data warehouse in the process of decision making – the security of organization's critical data is at stake if database server is not able to manage security properly.

Oracle Server – Where does it stand

Oracle has been amongst the earlier database management systems extending its features to accommodate data warehouse related features. It was the era of Oracle7 when the concept of data warehouse came and Oracle-corporation right away recognized its importance. Oracle v7.3 provided features like parallel query execution, parallel data management, cost-based query optimization, efficient bitmap indexing and hash joining embedded in query execution. Then came Oracle8 – enhancing the features, already provided by Oracle v7.3. Linking the server with tools like Oracle Discoverer and Oracle express have made Oracle the must viable option for data warehousing. Below we are going to discuss features provided in Oracle to enhance the server capabilities for the implementation of data warehouses.

Query Processing

Oracle7 advanced its architecture to improve the Query Optimizer as well as the execution of query.

QUERY OPTIMIZATION: The main task of query optimizer is to choose the most efficient way to execute a SQL statement – the DML (Data Manipulation Language) are considered for optimization. Oracle produces an execution plan for the optimization purpose. Oracle Optimizer takes following steps for the selection of best execution plan:

Evaluation of query expression and modification as per required. The optimizer assesses expressions construct and whenever required introduces some modification to enhance the speed and reduce resource utilization. Some examples of which are given in Figure 5.

Figure 5
(Bold shows the optimized query)

```
{Where Clause}
  cl_name like 'xyz'
cl_name = 'xyz'
  cl_name in ('a', 'b', 'c')
cl_name='a' or cl_name='b' or
cl_name='c'
  cl_name > any (select amount
                 from payment
                 where place = 'xxx')
exists (select amount
        from payment
        where place = 'xxx'
        and cl_name > amount)
  cl_name > all (select amount
                 from payment
```

Transformation of complex and symbiotic queries into equivalent joins statements. In the process of transformation, optimizer modifies two types of queries; queries containing OR to UNION ALL and complex queries into join statements.

For queries having views – the optimizer merges the query statement with that of view. Examples of such optimization is given in Figure 6

Figure 6

```
create or replace view view1
as
select cl_1, cl_2
from   table1
where  cl1 > 10;

{when selecting from view1)
select cl2
from   view1
where  cl2 > 15;

{optimizer modifies the query into}
select cl2
from   table1
where  cl1 > 10
and    cl2 > 15;
```

Selection of Optimization approach from Rule-Based Optimization and Cost-Based Optimization. **Rule-based approach** chooses execution path based on heuristically ranked operations. When more than one execution paths exists, rule-based approach selects path with lower rank. **Cost-based approach** optimizes a query based on following steps:

- Firstly, all potential execution plans are predetermined by optimizer – plans are based on access paths.
- Then, optimizer estimates the cost of each execution plan based on the data distribution and storage characteristic statistics – the statistics are based on table structure, indexes and clusters, I/O and CPU time, the available memory .
- Lastly, optimizer compares the cost of execution plans and selects one with lowest cost.

The selections of appropriate access path when a query is based on more than one table. Generally there exists more than one access paths when the table-data is accessed. The optimizer chooses the most appropriate access path based on the Rule-based or Cost-based approach.

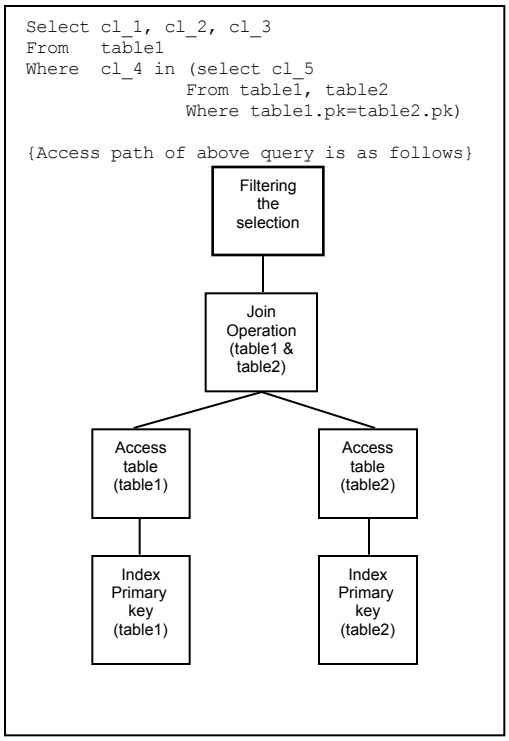


Figure 7

When joining more than two schemas, optimizer decides which pair to join first.

QUERY EXECUTION: Introducing the intra-query parallelism via Parallel Query option, Oracle7 provided parallel execution of complex queries having SQL operations like; SELECT, sub-queries in INSERT, DELETE OR UPDATE, CREATE TABLE based on sub-query, and CREATE INDEX commands. The parallel Query option improves the performance of data manipulation operations in very large databases, like warehouses. Best performance can be viewed on SMP (Symmetric Multiprocessor) and MPP (Massively Parallel Processing) machines. The query writers have to implicitly command the parallel query option and also declare degree of parallelism. Figure 8 explain how parallel query works in Oracle.

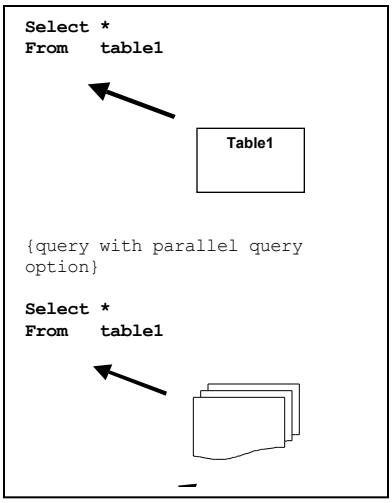


Figure 8
(Parallelism of degree 3)

In order to provide parallelism in query execution many initial parameters have to be configured. Once the system is configured to run queries with parallel query option, it is the task of *Query Coordinator Process* to initiate parallel query servers and coordinate between the results from these query servers. The number of query servers, running in parallel to complete one operation, is called *Degree of Parallelism*.

Data Management and Scalability

Oracle provides a data management architecture in which physical database structure is encapsulated by logical structure. By hiding the physical structure of database Oracle provides a level of manipulation through which one can introduce modification in the physical structure without hindering the usual workload on the DBMS. Logically database is based on *Table Spaces* – users access the table space to access data. Table Spaces are made of different *Data Files* which physically resides on Fixed Storage. To support very large space, specially for DSS systems, data base administrator can introduce very large table spaces – creating multiple data files for each table space.

Oracle7 introduced parallelism in data management operations like data loading, indexing and creation of summary tables. V7.3 also incorporated bit-mapped indexing as integrated server capability, adding to the already available indexing schemes like B-trees, clustered tables and hash clusters.

Oracle7 reputed itself in reliably managing wide range of users. Special features like replication of database and partitioning on distributed systems have been introduced in Oracle to support the scalability of users.

Database Security

With the provision of being multi-user database system, Oracle provides sound controls for the security of database. The controls include unauthorized access to database as well as individual schema objects, assessment of environment parameters such as disk usage and system resource usage. Oracle provides a set of privileges and a user is restricted by the grants given by these privileges. For effective management of privileges there exists *roles* – grouping privileges together and given some unique name. For most effective security management Oracle provides *Trusted Oracle*. Trusted Oracle provides a multi-level secure database management and mandatory access control (MAC). To monitor the user actions on database Oracle endure auditing of users.

Oracle8 – Object Relational Data Server

Most of the features described in the previous section are related to older versions of Oracle, with the parturition of Oracle8 – the Object-relational database, Oracle took a major step from the world of relational database technology to object-relational technology. In addition to other enhancements Oracle8 has advanced in some key areas related to data warehousing. Here we are just going to cling with the server enhancements for data warehousing.

Scalable Query Processing

Typically in data warehouse, queries are the most critical operations. No decision-maker can be expected to decide without the help of complex data retrieval. data warehouses, in deed, exist to support these decision-makers.

QUERY OPTIMIZATION: Based on the strong grounds provided by Oracle7, Oracle8 improved both - the query optimizer and query executor. The Oracle8 query optimizer mainly uses Cost-based approach; in addition it is also aware of parallelism and partitioning in the data. During query transformation, the optimizer rewrites the queries based on the techniques provided in Oracle7 optimizer – enhancing it by introduction of sophisticated cost-based query rewrites such as the Star Query Transformation and Anti-joins as well as Semi-joins.

Generally data warehouse design is based on star schema, which is characterized by one or more very large fact tables that contain the primary information of dimension tables, each of which contains information about the entries for a particular attribute in the fact table. The star query is based on the join between the fact table and dimension tables. These queries are based on star join, which is primary-key to foreign-key join of the dimension tables to a fact table. To have most effective performance the fact table is indexed using bit-mapped indexing. Oracle8, while executing star queries; firstly, retrieves filtered data from fact tables – fact tables are indexed using bit-mapped indexing, secondly, joins the fact table with dimension tables and retrieve the result. An example is shown below:

Figure 9

```
Select *
From fact, dim1, dim2, dim3
Where    fact.dim1pk = dim1.pk
And      fact.dim2pk = dim2.pk
And      fact.dim3pk = dim3.pk
And      dim1.key1 = 1000
And      dim2.key1 in ('A', 'B', 'C')
And      dim3.key1 = 'XXX'

{Star Query executor will run it as
following query}

select * from fact
where fact.dim1pk in (select pk
                      from dim1
                      where key1=1000)
and fact.dim2pk in (select pk
                      from dim2
                      where key1 in ('A',
                                'B', 'C'))
and fact.dim3pk in (select pk
                      from dim3
                      where key1='XXX')
```

QUERY EXECUTION: Oracle8 advancing in its 'best-of-both-worlds' parallel query architecture making use of the SMP as well as MPP architecture, introduced data partitioning. Oracle8's parallel query execution runs on two levels; parallel execution across partitions and parallel execution within partitions. The introduction of parallelism within data partitions have made Oracle8 a unique database management server as without this feature majority of database management servers were unable to utilize parallelism properly.

Data Management

Oracle8 improved the parallelism feature and indexing of tables. Oracle8 introduced partitioning of very large table as well as large indexes. In a non-partitioned index, a parallel index scan is done when a full table scan is required. Whereas in paralleled access on partitioned-indexes, one query slave is assigned to scan each partition of an index then the result of each of those are gathered together. It is the responsibility of DBA to set partitioned tables – the CREATE TABLE command have the provision for setting partitions based on partition key and a range for that key. Example of partitioning is given in Figure 11.

Oracle8 brought up support for bulk insert, update and delete operations in parallel. These data management operations in parallel provide efficient utilization of hardware resources.

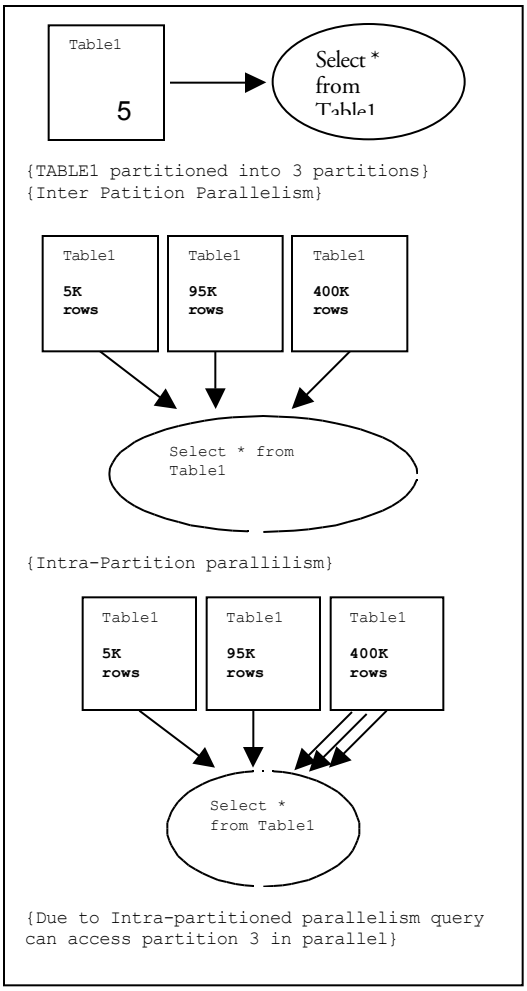


Figure 10

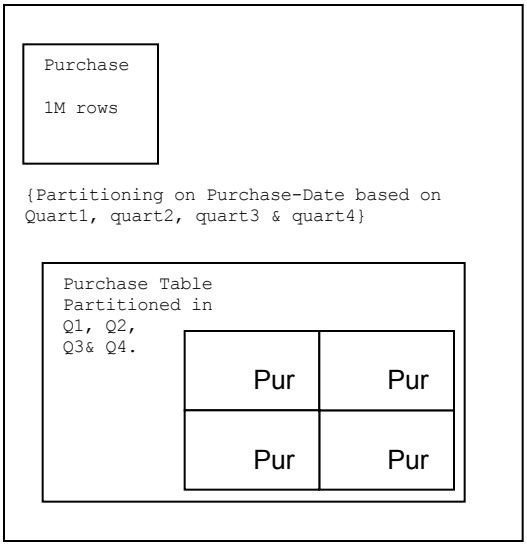


Figure 11

Oracle8, for supporting large objects, introduced enhanced functionality to support new data-types:

- BLOB for binary large objects
- CLOB for character large objects
- NCLOB for character large objects stored in the national character; analogous to NCHAR
- BFILE for binary files stored outside of the Oracle's universal data server

There exists functionality to store large objects within as well as outside database – BFILE is used as a locator within the database table pointing to OS file.

Scalability

Oracle8 provides database replication and partitioning along with management tools like *Oracle Enterprise Manager* and *Oracle Names* to support scalability. All the above features enhance Oracle's capability to manage very huge amount of clients and data. Database replication across distributed environment allows the provision of enormous amount of users accessing the database. The provision of partitioning ensures the database size to grow beyond terabyte boundary. Lastly Enterprise Manager and Oracle Names enhances the management and configuration of thousands of users.

To support network-computing Oracle8 database server has been re-designed. The key strengths of Oracle8 are:

- Ability to support tons of thousands of users – both entry level and business-critical users
- Ability to support huge amount of data
- Fastest data server in industry

New features have been introduced in server to enable data availability even in the cases of partial failures and maintenance. Oracle8 also managed to introduce improvements for Internet Users – Internet applications can now directly communicate with the server.

Database Security

With the support of tons of thousands of users – connecting the server as direct users, client/server users or Internet users, Oracle8 have to enhance its security measures to secure the integrity of data. The power of Oracle8's security measures are more prominent when considering highly networked environment, where most of the data is traveling through the network wires.

Oracle8 provides Discretionary Access Control (DAC), regulating all users access to database objects based on privileges. With the usage of Trusted Oracle B1-level data security can be guaranteed based on Mandatory Access Control (MAC).

Integration with other systems

To entertain the needs of linking the Oracle server (v7.3 or v8) Oracle has provided tools like Oracle Open Gateway and Oracle Transport Gateway.

Oracle8i – The Internet Database

With the advancement of Internet applications and support given by Internet to networked environment more and more organizations are planning to use this technology in the production of internet/intranet applications. Special consideration to this environment is given, when developing the data warehouses – providing access to thousands and

thousands of users on global scale. Oracle's newest server in marked – Oracle8i, is database-resident Java Virtual Machine for storing and executing Java code on the server, based on Oracle8 server architecture. This integration between Oracle8 features and Java helps in developing large scaled, internet-savvy applications like data warehouses. Oracle8i provides features to store and execute Java code within database, and to create stored procedures, database function and triggers in Java. With the use of Oracle8i the development, deployment and updation of internet applications have become very simplified.

Following are some of the features available in Oracle8i – for the management of data warehousing.

To enhance the data management of very large databases like data warehouses, the partitioning option have been enhanced in Oracle8i. Some of the advantages of partitioning would be:

- Recovery of individual partitions
- Only relevant partitions are considered during query execution
- Option of reorganization, addition and deletion of individual partitions – without affecting the data

Oracle8i introduced new types of partitioning – *Hash Partitioning* and *Composite Partitioning*. In hash partitioning Oracle8i uses a hash function to generate almost random numbers for partitioning key, this random number is latter used to access the partition in which the row is stored. The composite partitioning is the combination of key range partitioning and hash partitioning – first table is partitioned using key range partitioning, then another partition is created based on some different key using hash function.

The need of summarization and aggregation is very high while accessing a database for decision making. To facilitate this requirement Oracle8i uses *Materialized Views* – a view for which database server runs the view defining query and stores the results in database for future usage. Supplementing is the concept of Automatic Query Rewriter – using this rewriter query optimizer can decide when to use materialized views and when to use the tables directly. Now by using materialized views the database administrator can make summary tables – the most frequently used data type in data warehouses.

Another interesting feature introduced in Oracle8i is the *Transportation Tablespace*. This is used to transfer some data from one Oracle system to another Oracle system – from operational system to data warehouse. Oracle8i gives the facility to copy the transportation tablespace without any unloading or reloading of tablespace.

Oracle Warehouse Architecture

This architecture is designed using the RDBMS server and tools provided by Oracle. The Oracle warehouse can be developed using two-tiered or three-tiered architecture. The two-tiered architecture involves the database server at back-end and front-end decision support tools. A more complex warehouse involves separate tiers for data access from operational source, data storage and presentation of data for decision support.

Tier 1 – Accessing Source Data

Data can be accessed from multiple sources; including operational systems, Legacy systems and other Oracle applications. Utilities like *SQL*Loader*, *export/import* Oracle schemas, *SQL Stored Procedures* can be used for the data transfer. For transferring data from legacy systems and a wide range of other systems Oracle Transparent Gateways are used.

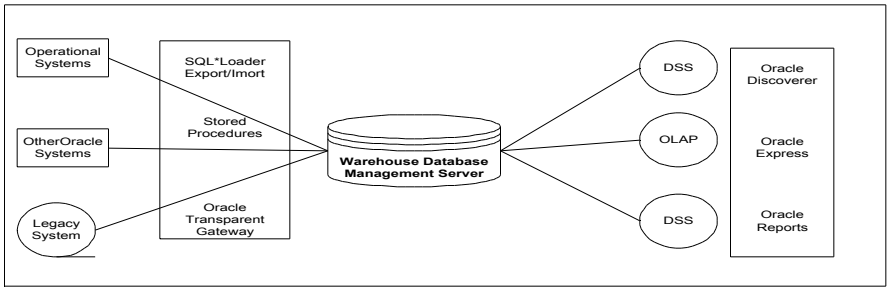
Tier 2 – The Server for Warehouse

The server for data warehousing can be of RDBMS or MDD type. Oracle provides solution for both options. If one decides to use Multi-Dimensional Database Architecture then there exist *Oracle Express Server*, and for Relation Database Architecture the option is *Oracle7*, *Oracle8* or *Oracle8i*. Warehouse can be designed by integrating the two.

Tier 3 – Decision Support System

To entertain Business Analysts, both DSS and OLAP tools can be provided in the data warehouse. With tools like *Oracle Reports*, *Oracle Discoverer* and *Oracle Express*, users can have access on data warehouse on the whenever and however basis.

Figure 13



Oracle Warehouse Toolkits

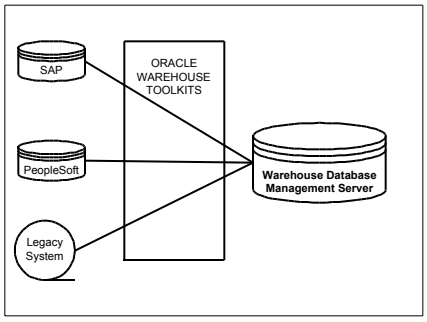
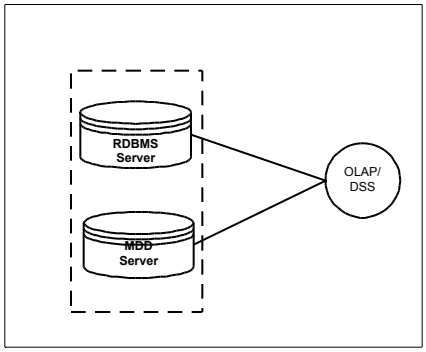


Figure 14

Considering the importance of data warehousing and reducing the effort required by data warehouse designers, Oracle provide set of toolkits. Using these tools it would be very easy to transfer operational data from other sources; including SAP, PeopleSoft and BAAN. Then just connect this warehouse to DSS or OLAP tools and the Decision support system is ready.. Oracle Warehouse Toolkits provides features like:

- Access to updated Operational data
- Easy-to-use, graphical analytical tools
- Fast and flexible analysis of information

Figure 15



In addition with the Ideal servers for warehouse data, the tools for DSS and OLAP, like Oracle Discoverer and Oracle Express, makes Oracle warehouse the ideal analytical environment. Oracle Discoverer, the GUI based reporting tool, gives an excellent interface for querying and reporting purposes. Typical queries like "what" and "how" can be entertained in Oracle Discoverer. Whereas OLAP tool – Oracle Express, provides graphical interface for answering "what-if" questions.

Generally, warehouse designers use CASE tool such as Oracle Designer/2000 for designing Oracle Warehouse model. Using this designer tool, RDBMS warehouse design can be implemented in Oracle8 or Oracle7, whereas MDD warehouse design in Oracle Express Server. The combination of RDBMS and MDD servers can also be designed to get the most optimum performance out of data warehouse.

References

W.H.Inmon – known as father of Data Warehousing

What is Data Warehouse?

Vivek R. Gupta – Senior Consultant, Services corporation, Chicago, Illinois.

An Introduction to Data Warehousing

David Heise – CIO Andrews University

Data Warehousing at Avondale College

Dr. James Goodnight – CEO SAS Institute Inc.

Data Warehousing: Understanding Its Role in a Business Management Architecture

Oracle Publications

Oracle7 – Server Concept Manual

Oracle Publications

Oracle7 – Server Tuning Manual

Oracle Publications

Oracle8 Concepts

Oracle White Paper, June 1997,

Oracle8 for Data Warehousing

Oracle White Paper, June 1997,

Oracle8 – Enabling Decisions in the New Business Era

Oracle White Paper, June 1997,

Oracle8 – The Database for Network Computing

Oracle White Paper, June 1997,

Star Queries in Oracle8

Oracle White Paper, June 1997,

Oracle8 – The Hub of Oracle Warehouse

Winter Corporation White Paper

Large Scale Data Warehousing with Oracle8i

Oracle White Paper, November 1998,

Oracle8i – The Database for Internet Computing

Oracle White Paper, March 1998,

Oracle Warehouse Toolkits