USN :										

CMR INSTITUTE OF TECHNOLOGY

SCHEME & SOLUTION

Internal Assessment Test 1 – SEP 2016

Sub:	Data warehousing & Data mining					Code:	10IS74
Date :	07/ 09/2016	Duration:	90 mins	Max Marks :	50	Sem:	VII	Branch:	ISE

Note: Answer any five questions:

1 (a) Explain the ETL operations in detail. [10]

The process of extracting data from source systems and bringing it into data warehouse is commonely called as ETL which stands for Extraction Transformation and Loading

1)INSTANCE IDENTITY PROBLEM

The same customer or client amy be represented in different in different data source system.there is thus a possibility of mismatching between the different system that need to be identified and corrected.

2)DATA ERRORS

diffrent type of errors rather than identify errors are possible

- data may have some missing attribute values
- mismatching codes in different data bases
- meaning of code values may not be known
- duplicate records
- wrong aggregation
- inconsistent use of null,spaces and empty values
- inappropriate address lines
- non unique identifier

3)RECORD LINKAGE PROBLEM

problem of linking different information from different data bases that relate to the same user.the problem is araised when a unique identifier is not available in all databases.

4) SEMANTIC INTEGRATION PROBLEM

this deals with integration of information found in hetrogenous oltp and legacy systems.some of the sources may be relationalsome may not be.

5)DATA INTEGRITY PROBLEM

this deals with issues like referential integrity,null values domain of values etc

DATA CLEANING HAVE THE FOLLOWING STEPS

1)PARSING:identify various components of the source data file and then establishes a relationship between those and the field in the target file

2)CORRECTING:

correcting based on the mathematical algorithm.it may be involved other information in enterprise

3)STANDERDIZING :

it involve the transformation of data into standerd form

4)MATCHING:

check whether the data extracted from different sources are matching or not

5)CONSOLIDATING:

all corrected standerdized and matched data can now be consolidated to build a single version of the enterprise data

2(A) List out the difference between the following

a) OLTP &Data warehouse, b) Data ware house & ODS

ODS	DW
Data of high quality and detailed level and assured availability	Data may not be perfect but sufficent for starategic analysis.data does not have to be highly available
Contain current and near current data	Contain historical data
Real time and near real time data loads	Normally batch data loads
Updated at field level	Data is appended not not uploaded
Detailed data only	Summarized and detailed data
Support rapid data update(3NF)	Typically multidimensional data mart to

	optimize query performance
Used for detailed decision making and operational reporting	Used for long term decision making and management reporting
Used in operational level	Used at the managerical level

PROPERTY	OLTP	DATA WAREHOUSE
Nature of the database	3NF	multidimensional
Indexes	few	many
Joins	many	some
Duplicated data	Normalized data	Denormalized data
Derived data and aggregations	rare	commonly
Queries	Mostly predefined	Mostly adhoc
Nature of queries	Mostly simple	Mostly complex
updates	All the time	Not allowed
Historical data	Often not available	essential

3. Explain the data warehouse architecture using Operational data stores[10]

logical area for a data warehouse.

While in the ODS, data can be scrubbed, resolved for redundancy and checked for compliance with the corresponding business rules. An ODS can be used for integrating disparate data from multiple sources so that business operations, analysis and reporting can be carried out while business operations are occurring. This is the place where most of the data used in current operation is housed before it's transferred to the data warehouse for longer term storage or archiving.

An ODS is designed for relatively simple queries on small amounts of data (such as finding the status of a customer order), rather than the complex queries on large amounts of data typical of the data warehouse. An ODS is similar to your short term memory in that it stores only very recent information; in comparison, the data warehouse is more like long term memory in that it stores relatively permanent information.

What Is a Data Warehouse?

A data warehouse is a database designed to enable business intelligence activities: it exists to help users understand and enhance their organization's performance. It is designed for query and analysis rather than for transaction processing, and usually contains historical data derived from transaction data, but can include data from other sources. Data warehouses separate analysis workload from transaction workload and enable an organization to consolidate data from several sources. This helps in:

- Maintaining historical records
- Analyzing the data to gain a better understanding of the business and to improve the business

In addition to a relational database, a data warehouse environment can include an extraction, transportation, transformation, and loading (ETL) solution, statistical analysis, reporting, data mining capabilities, client analysis tools, and other applications that manage the process of gathering data, transforming it into useful, actionable information, and delivering it to business users.

To achieve the goal of enhanced business intelligence, the data warehouse works with data collected from multiple sources. The source data may come from internally developed systems, purchased applications, third-party data syndicators and other sources. It may involve transactions, production, marketing, human resources and more. In today's world of big data, the data may be many billions of individual clicks on web sites or the massive data streams from sensors built into complex machinery.

Data warehouses are distinct from online transaction processing (OLTP) systems. With a data warehouse you separate analysis workload from transaction workload. Thus data warehouses are very much read-oriented systems. They have a far higher amount of data reading versus writing and updating. This enables far better analytical performance and avoids impacting your transaction systems. A data warehouse system can be optimized to consolidate data from many sources to achieve a key goal: it becomes your organization's "single source of truth". There is great value in having a consistent source of data that all users can look to; it prevents many disputes and enhances decision-making efficiency.

A data warehouse usually stores many months or years of data to support historical analysis. The data in a data warehouse is typically loaded through an extraction, transformation, and loading (ETL) process from multiple data sources. Modern data warehouses are moving toward an extract, load, transformation (ELT) architecture in which all or most data transformation is performed on the database that hosts the data warehouse. It is important to note that defining the ETL process is a very large part of the design effort of a data warehouse. Similarly, the speed and reliability of ETL operations are the foundation of the data warehouse once it is up and running.

Users of the data warehouse perform data analyses that are often time-related. Examples include consolidation of last year's sales figures, inventory analysis, and profit by product and by customer. But time-focused or not, users want to "slice and dice" their data however they see fit and a well-designed data warehouse will be flexible enough to meet those demands. Users will sometimes need highly aggregated data, and other times they will need to drill down to details. More sophisticated analyses include trend analyses and data mining, which use existing data to forecast trends or predict futures. The data warehouse acts as the underlying engine used by middleware business intelligence environments that serve reports, dashboards and other interfaces to end users.

Although the discussion above has focused on the term "data warehouse", there are two other important terms that need to be mentioned. These are the data mart and the operation data store (ODS).

Operational data stores exist to support daily operations. The ODS data is cleaned and validated, but it is not historically deep: it may be just the data for the current day. Rather than support the historically rich queries that a data warehouse can handle, the ODS gives data warehouses a place to get access to the most current data, which has not yet been loaded into the data warehouse. The ODS may also be used as a source to load the data warehouse. As data warehousing loading techniques have become more advanced, data warehouses may have less need for ODS as a source for loading data. Instead, constant trickle-feed systems can load the data warehouse in near real time.

A common way of introducing data warehousing is to refer to the characteristics of a data warehouse as set forth by William Inmon:

- Subject Oriented
- Integrated
- Nonvolatile
- Time Variant

Subject Oriented

Data warehouses are designed to help you analyze data. For example, to learn more about your company's sales data, you can build a data warehouse that concentrates on sales. Using this data warehouse, you can answer questions such as "Who was our best customer for this item last year?" or "Who is likely to be our best customer next year?" This ability to define a data warehouse by subject matter, sales in this case, makes the data warehouse subject oriented.

Integrated

Integration is closely related to subject orientation. Data warehouses must put data from disparate sources into a consistent format. They must resolve such problems as naming conflicts and inconsistencies among units of measure. When they achieve this, they are said to be integrated.

Nonvolatile

Nonvolatile means that, once entered into the data warehouse, data should not change. This is logical because the purpose of a data warehouse is to enable you to analyze what has occurred.

Time Variant

A data warehouse's focus on change over time is what is meant by the term time variant. In order to discover trends and identify hidden patterns and relationships in business, analysts need large amounts of data. This is very much in contrast to online transaction processing (OLTP) systems, where performance requirements demand that historical data be moved to an archive.

Key Characteristics of a Data Warehouse

The key characteristics of a data warehouse are as follows:

- Data is structured for simplicity of access and high-speed query performance.
- End users are time-sensitive and desire speed-of-thought response times.
- Large amounts of historical data are used.
- Queries often retrieve large amounts of data, perhaps many thousands of rows.
- Both predefined and ad hoc queries are common.
- The data load involves multiple sources and transformations.

In general, fast query performance with high data throughput is the key to a successful data warehouse.

Contrasting OLTP and Data Warehousing Environments

There are important differences between an OLTP system and a data warehouse. One major difference between the types of system is that data warehouses are not exclusively in third normal form (3NF), a type of data normalization common in OLTP environments.

Data warehouses and OLTP systems have very different requirements. Here are some examples of differences between typical data warehouses and OLTP systems:

- Workload

  Data warehouses are designed to accommodate ad hoc queries and data analysis. You might not know the workload of your data warehouse in advance, so a data warehouse should be optimized to perform well for a wide variety of possible query and analytical operations.

  OLTP systems support only predefined operations. Your applications might be specifically tuned or designed to support only these operations.

- Data modifications

A data warehouse is updated on a regular basis by the ETL process (run nightly or weekly) using bulk data modification techniques. The end users of a data warehouse do not directly update the data warehouse except when using analytical tools, such as data mining, to make predictions with associated probabilities, assign customers to market segments, and develop customer profiles.

In OLTP systems, end users routinely issue individual data modification statements to the database. The OLTP database is always up to date, and reflects the current state of each business transaction.

- Schema design

  Data warehouses often use partially denormalized schemas to optimize query and analytical performance.

  OLTP systems often use fully normalized schemas to optimize update/insert/delete performance, and to guarantee data consistency.

- Typical operations

  A typical data warehouse query scans thousands or millions of rows. For example, "Find the total sales for all customers last month."

  A typical OLTP operation accesses only a handful of records. For example, "Retrieve the current order for this customer."

- Historical data

  Data warehouses usually store many months or years of data. This is to support historical analysis and reporting.

  OLTP systems usually store data from only a few weeks or months. The OLTP system stores only historical data as needed to successfully meet the requirements of the current transaction.

Common Data Warehouse Tasks

As an Oracle data warehousing administrator or designer, you can expect to be involved in the following tasks:

- Configuring an Oracle database for use as a data warehouse
- Designing data warehouses
- Performing upgrades of the database and data warehousing software to new releases
- Managing schema objects, such as tables, indexes, and materialized views
- Managing users and security
- Developing routines used for the extraction, transformation, and loading (ETL) processes

- Creating reports based on the data in the data warehouse
- Backing up the data warehouse and performing recovery when necessary
- Monitoring the data warehouse's performance and taking preventive or corrective action as required

In a small-to-midsize data warehouse environment, you might be the sole person performing these tasks. In large, enterprise environments, the job is often divided among several DBAs and designers, each with their own specialty, such as database security or database tuning.

These tasks are illustrated in the following:

- For more information regarding partitioning, see Oracle Database VLDB and Partitioning Guide.
- For more information regarding database security, see Oracle Database Security Guide.
- For more information regarding database performance, see Oracle Database Performance Tuning Guide and Oracle Database SQL Tuning Guide.
- For more information regarding backup and recovery, see Oracle Database Backup and Recovery User's Guide.
- For more information regarding ODI, see Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator.

Data Warehouse Architectures

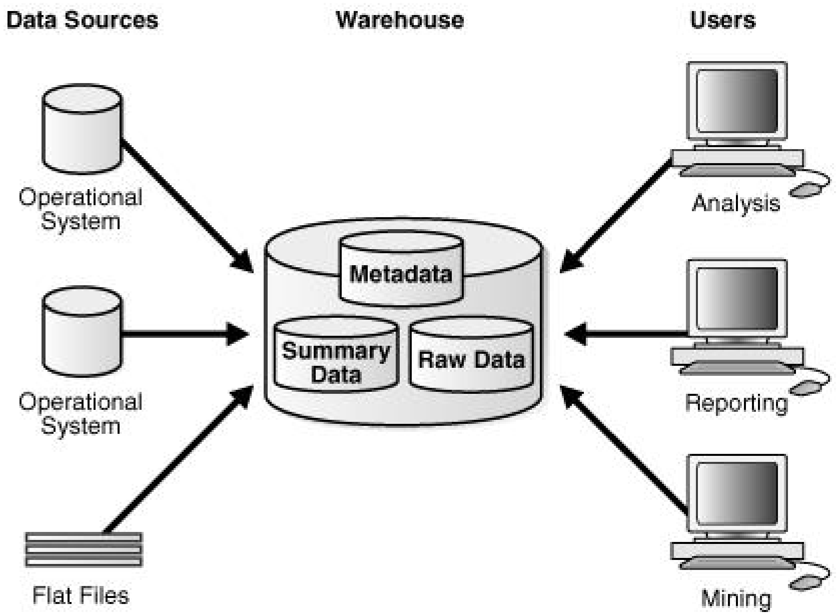
Data warehouses and their architectures vary depending upon the specifics of an organization's situation. Three common architectures are:

- Data Warehouse Architecture: Basic
- Data Warehouse Architecture: with a Staging Area
- Data Warehouse Architecture: with a Staging Area and Data Marts

Data Warehouse Architecture: Basic

Figure 1-1 shows a simple architecture for a data warehouse. End users directly access data derived from several source systems through the data warehouse.

Figure 1-1 Architecture of a Data Warehouse



Description of "Figure 1-1 Architecture of a Data Warehouse"

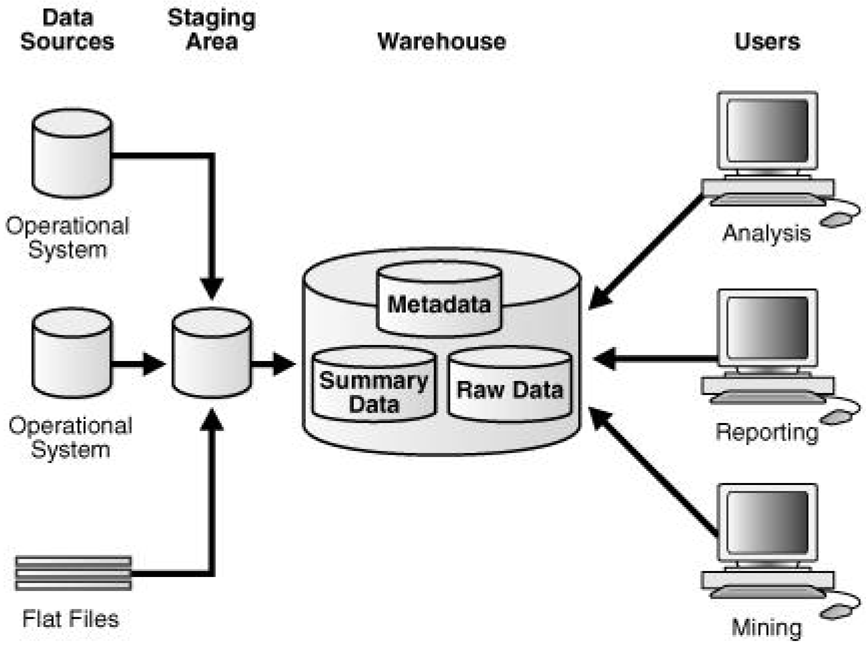
In Figure 1-1, the metadata and raw data of a traditional OLTP system is present, as is an additional type of data, summary data. Summaries are a mechanism to pre-compute common expensive, long-running operations for sub-second data retrieval. For example, a typical data warehouse query is to retrieve something such as August sales. A summary in an Oracle database is called a materialized view.

The consolidated storage of the raw data as the center of your data warehousing architecture is often referred to as an Enterprise Data Warehouse (EDW). An EDW provides a 360-degree view into the business of an organization by holding all relevant business information in the most detailed format.

Data Warehouse Architecture: with a Staging Area

You must clean and process your operational data before putting it into the warehouse, as shown in Figure 1-2. You can do this programmatically, although most data warehouses use a staging area instead. A staging area simplifies data cleansing and consolidation for operational data coming from multiple source systems, especially for enterprise data warehouses where all relevant information of an enterprise is consolidated. Figure 1-2 illustrates this typical architecture.

Figure 1-2 Architecture of a Data Warehouse with a Staging Area

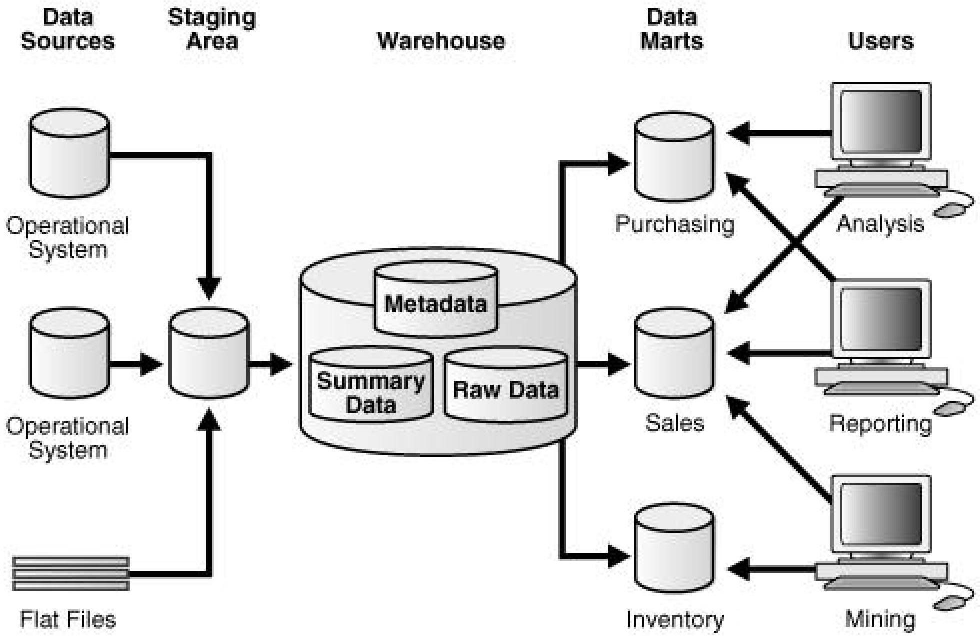


Description of "Figure 1-2 Architecture of a Data Warehouse with a Staging Area"

Data Warehouse Architecture: with a Staging Area and Data Marts

Although the architecture in Figure 1-2 is quite common, you may want to customize your warehouse's architecture for different groups within your organization. You can do this by adding data marts, which are systems designed for a particular line of business. Figure 1-3 illustrates an example where purchasing, sales, and inventories are separated. In this example, a financial analyst might want to analyze historical data for purchases and sales or mine historical data to make predictions about customer behavior.

Figure 1-3 Architecture of a Data Warehouse with a Staging Area and Data Marts



4 (a) Illustrate the construction of star schema,fact table,dimension table and snowflake schema using an example

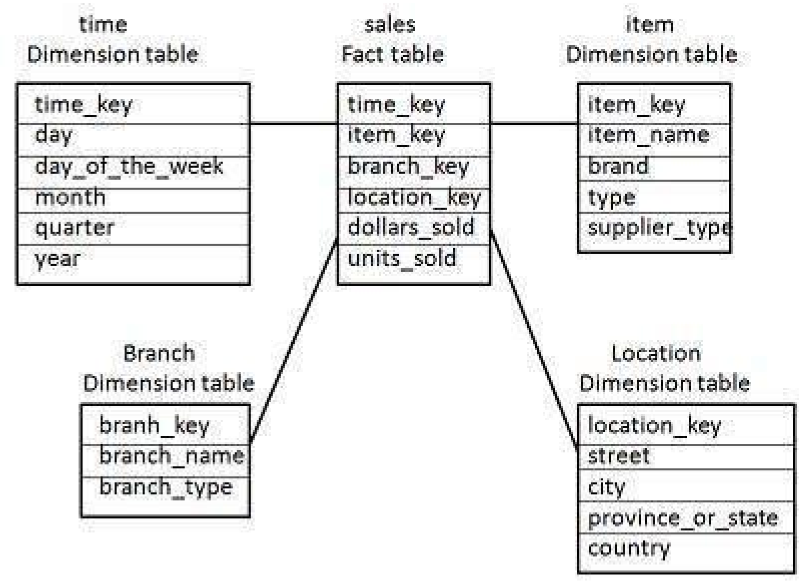
Schema is a logical description of the entire database. It includes the name and description of records of all record types including all associated data-items and aggregates. Much like a database, a data warehouse also requires to maintain a schema. A database uses relational model, while a data warehouse uses Star, Snowflake, and Fact Constellation schema. In this chapter, we will discuss the schemas used in a data warehouse.

Star Schema

- Each dimension in a star schema is represented with only one-dimension table.

- This dimension table contains the set of attributes.

- The following diagram shows the sales data of a company with respect to the four dimensions, namely time, item, branch, and location.



- There is a fact table at the center. It contains the keys to each of four dimensions.

- The fact table also contains the attributes, namely dollars sold and units sold.

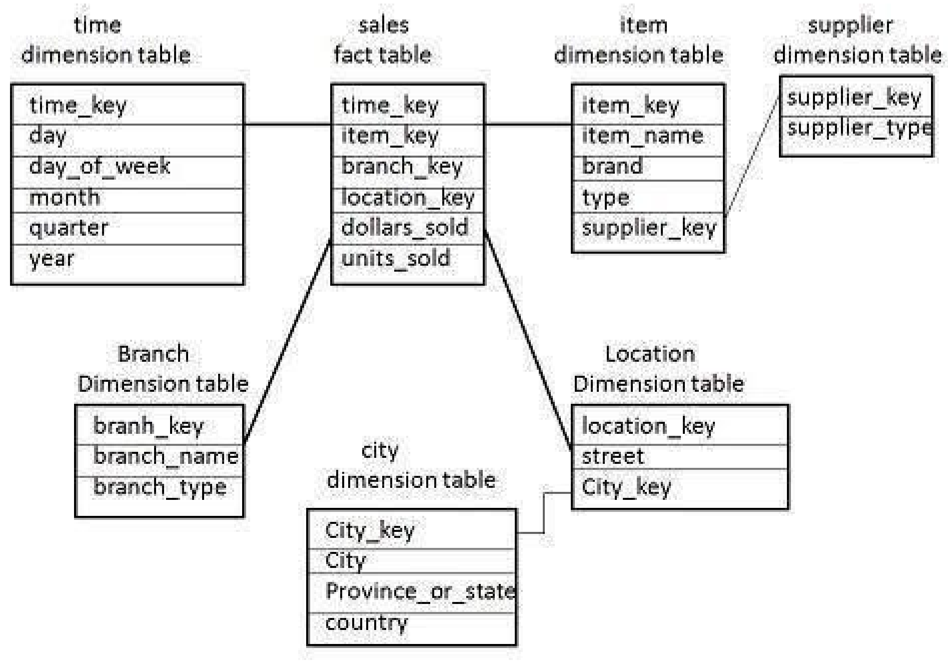
Note: Each dimension has only one dimension table and each table holds a set of attributes. For example, the location dimension table contains the attribute set {location_key, street, city, province_or_state,country}. This constraint may cause data redundancy. For example, "Vancouver" and "Victoria" both the cities are in the Canadian province of British Columbia. The entries for such cities may cause data redundancy along the attributes province_or_state and country.

Snowflake Schema

- Some dimension tables in the Snowflake schema are normalized.

- The normalization splits up the data into additional tables.

- Unlike Star schema, the dimensions table in a snowflake schema are normalized. For example, the item dimension table in star schema is normalized and split into two dimension tables, namely item and supplier table.



- Now the item dimension table contains the attributes item_key, item_name, type, brand, and supplier-key.

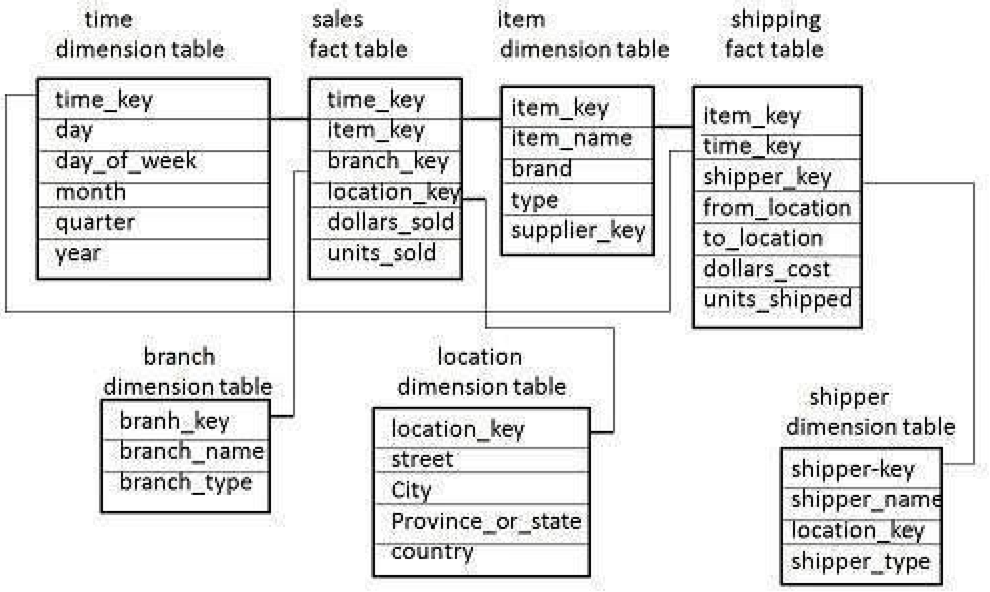
- The supplier key is linked to the supplier dimension table. The supplier dimension table contains the attributes supplier_key and supplier_type.

<b< style="box-sizing: border-box;">Note: Due to normalization in the Snowflake schema, the redundancy is reduced and therefore, it becomes easy to maintain and the save storage space.</b<>

Fact Constellation Schema

- A fact constellation has multiple fact tables. It is also known as galaxy schema.

- The following diagram shows two fact tables, namely sales and shipping.



- The sales fact table is same as that in the star schema.

- The shipping fact table has the five dimensions, namely item_key, time_key, shipper_key, from_location, to_location.

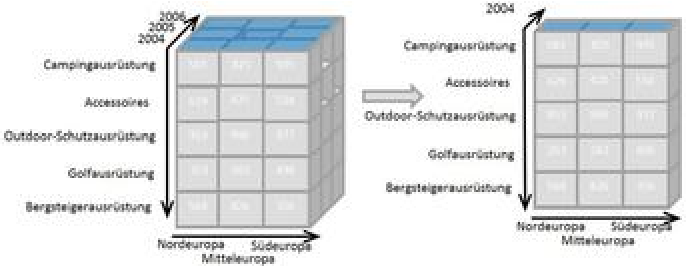
- The shipping fact table also contains two measures, namely dollars sold and units sold.

- It is also possible to share dimension tables between fact tables. For example, time, item, and location dimension tables are shared between the sales and shipping fact table.

5 A)Explain all OLAP operations with an example.

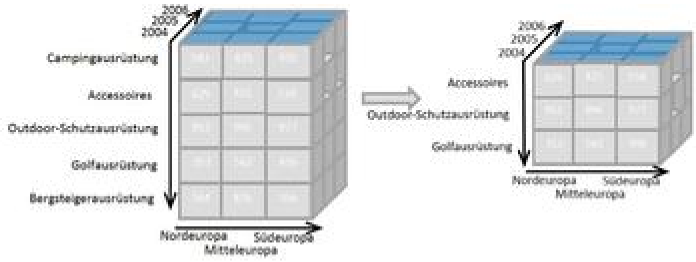
- OLAP data is typically stored in a star schema or snowflake schema in a relational data warehouse or in a special-purpose data management system. Measures are derived from the records in the fact table and dimensions are derived from the dimension tables.

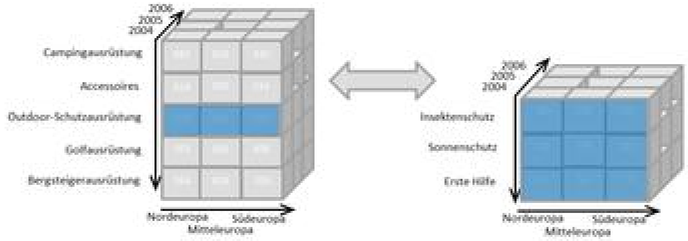
- Hierarchy
- The elements of a dimension can be organized as a hierarchy,[4] a set of parent-child relationships, typically where a parent member summarizes its children. Parent elements can further be aggregated as the children of another parent.[5]
- For example May 2005's parent is Second Quarter 2005 which is in turn the child of Year 2005. Similarly cities are the children of regions; products roll into product groups and individual expense items into types of expenditure.

- Operations
- Conceiving data as a cube with hierarchical dimensions leads to conceptually straightforward operations to facilitate analysis. Aligning the data content with a familiar visualization enhances analyst learning and productivity.[5] The user-initiated process of navigating by calling for page displays interactively, through the specification of slices via rotations and drill down/up is sometimes called "slice and dice". Common operations include slice and dice, drill down, roll up, and pivot.
- 

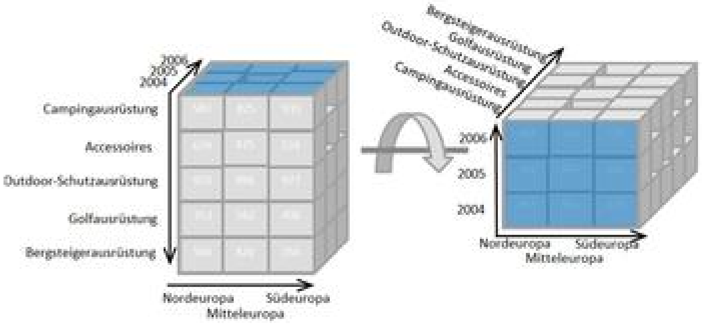
- OLAP slicing

- Slice is the act of picking a rectangular subset of a cube by choosing a single value for one of its dimensions, creating a new cube with one fewer dimension.[5] The picture shows a slicing operation: The sales figures of all sales regions and all product categories of the company in the year 2004 are "sliced" out of the data cube.



-
- OLAP dicing
- Dice: The dice operation produces a subcube by allowing the analyst to pick specific values of multiple dimensions.[6] The picture shows a dicing operation: The new cube shows the sales figures of a limited number of product categories, the time and region dimensions cover the same range as before.
-
- 

- OLAP Drill-up and drill-down
- Drill Down/Up allows the user to navigate among levels of data ranging from the most summarized (up) to the most detailed (down).[5] The picture shows a drill-down operation: The analyst moves from the summary category "Outdoor-Schutzausrüstung" to see the sales figures for the individual products.
- Roll-up: A roll-up involves summarizing the data along a dimension. The summarization rule might be computing totals along a hierarchy or applying a set of formulas such as "profit = sales - expenses".[5]
-

- 

- OLAP pivoting
- Pivot allows an analyst to rotate the cube in space to see its various faces. For example, cities could be arranged vertically and products horizontally while viewing data for a particular quarter. Pivoting could replace products with time periods to see data across time for a single product.[5][7]
- The picture shows a pivoting operation: The whole cube is rota

4. Explain a)OLAP – FASMI Characteristics

FASMI Characteristics

In the FASMI characteristics of OLAP systems, the name derived from the first letters of the characteristics are:

Fast: Asnotedearlier,mostOLAPqueriesshould beansweredveryquickly,perhaps within seconds. The performance of an OLAP system has to be like that of asearch engine. If the response takes more than say 20 seconds, the user is likely to moveaway to something else assuming there is a problem with the query. Achieving suchperformance is difficult.The data structuresmust be efficient.The hardware must bepowerful enough for the amount of data and the number of users. Full pre-computation of aggregates helps but is often not practical due to the large number of aggregates fail.

Analytic:An OLAP system mustprovide rich analytic functionalityand it isexpectedthatmostOLAPqueriescan beansweredwithoutanyprogramming.Thesystem should be able to cope with any relevant queries for the application and the user.Oftentheanalysiswillbe usingthevendor'sowntoolsalthoughOLAPsoftwarecapabilities differ widely between products in the market.

Shared: An OLAP system is shared resource although it is unlikely to beshared by hundreds of users. An OLAP system is likely to be accessed only by a selectgroup of managers and may be used merely by dozens of users. Being a shared system,an OLAPsystem should be provide adequate security for confidentiality as well as integrity.

Multidimensional:This isthebasicrequirement.WhateverOLAPsoftware isbeing used, it must provide a multidimensional conceptual view of the data. It is becauseof themultidimensionalview of datathat weoftenreferto thedataas acube. Adimensionoftenhashierarchiesthatshowparent/ child relationshipsbetweenthemembers of a dimension. The multidimensional structure should allow such hierarchies.

Information: OLAP systems usually obtain information from a data warehouse.The system should be able to handle a large amount of input data. The capacity of anOLAP system to handle information and its integration with the data warehouse may becritical.

(b) Explain Guidelines for data warehouse implementation[5]

1. Build incrementally: Data warehouses must be built incrementally. Generally itis recommended that a data mart may first be built with one particular project inmind and once it is implemented a number of other sections of the enterprise mayalso wish to implement similar systems. An enterprise data warehouse can thenbe implemented in an iterative manner allowing all data marts to extractinformation from the data warehouse. Data warehouse modelling itselfis an iterative methodology as users become familiar with the technology and arethen able to understand and express their requirements more clearly.

2. Need a champion: A data warehouse project must have a champion who iswilling to carry out considerable research into expected costs and benefits of theproject. Data warehousing projects require inputs from many units in amenterprise and therefore need to be driven by someone who is capable ofinteraction with people in the enterprise and can actively persuade colleagues.Without the cooperation of other units, the data model for the warehouse and thedata required to populate the warehouse may be more complicated than they needto be. Studies have shown that having a champion can help adoption and successof data warehousing projects.

3. Senior management support: A data warehouse project must be fully supportedby the senior management. Given the resource intensive nature of such projectsand the time they can take to implement, a warehouse project calls for a sustainedcommitment from senior management. This can sometimes be difficult since it may be hard to quantify the benefits of data warehouse technology and themanagers may consider it a cost without any explicit return on investment. Datawarehousing project studies show that top management support is essential forthe success of a data warehousing project.

4. Ensure quality: Only data that has been cleaned and is of a quality that isunderstood by the organization should be loaded in the data warehouse. The dataquality in the source systems is not always high and often little effort is made toimprove data quality in the source systems. Improved data quality, whenrecognized by senior managers and stakeholders, is likely to lead to improvedsupport for a data warehouse project.

5. Corporate strategy: A data warehouse project must fit with corporate strategyand business objectives. The objectives of the project must be clearly definedbefore the start of the project. Given the importance of senior managementsupport for a data warehousing project, the fitness of the project with thecorporate strategy is essential.

6. Business plan: The financial costs (hardware, software, peopleware), expectedbenefits and a project plan (including an ETL plan) for a data warehouse projectmust be clearly outlined and understood by all stakeholders. Without suchunderstanding, rumours about expenditure and benefits can become the onlysource of information, undermining the project.

7. Training: A data warehouse project must not overlook data warehouse trainingrequirements. For a data warehouse project to be successful, the users must betrained to use the warehouse and to understand its capabilities. Training of usersand professional development of the project team may also be required since datawarehousing is a complex task and the skills of the project team are critical to thesuccess of the project.

8. Adaptability: The project should build in adaptability so that changes may bemade to the data warehouse if and when required. Like any system, a datawarehouse will need to change, as needs of an enterprise change. Furthermore,once the data warehouse is operational, new applications using the data warehouse are almost certain to be proposed. The system should be able tosupport such new applications.

9. Joint management: The project must be managed by both IT and businessprofessionals in the enterprise. To ensure good communication with thestakeholders and that the project is focused on assisting the enterprise’s business,business professionals must be involved in the project along with technical professionals.