UNIT I

DATA WAREHOUSING

CONTENTS

1.1.Data warehouse: Basic Concepts

A data warehousing is defined as a technique for collecting and managing data from varied sources to provide meaningful business insights. It is a blend of technologies and components which aids the strategic use of data.

It is electronic storage of a large amount of information by a business which is designed for query and analysis instead of transaction processing. It is a process of transforming data into information and making it available to users in a timely manner to make a difference

1.1.1Characteristics of Data warehouse

A data warehouse has following characteristics:

- Subject-Oriented
- Integrated
- Time-variant
- Non-volatile

Subject-Oriented

A data warehouse is subject oriented as it offers information regarding a theme instead of companies' ongoing operations. These subjects can be sales, marketing, distributions, etc.

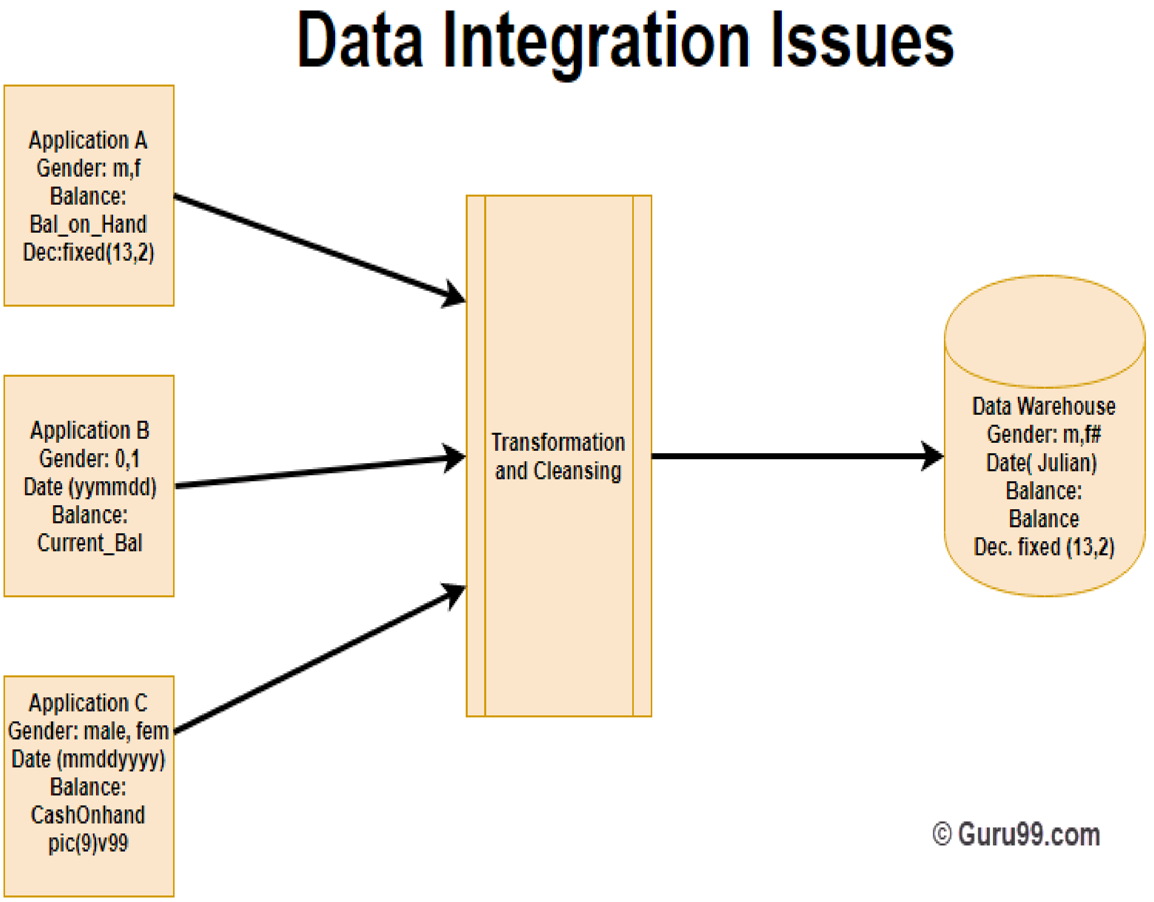
A data warehouse never focuses on the ongoing operations. Instead, it put emphasis on modeling and analysis of data for **decision making**. It also provides a simple and concise view around the specific subject by excluding data which not helpful to support the decision process.

Integrated

In Data Warehouse, integration means the establishment of a common unit of measure for all similar data from the dissimilar database. The data also needs to be stored in the Datawarehouse in common and universally acceptable manner.

A data warehouse is developed by integrating data from varied sources like a mainframe, relational databases, flat files, etc. Moreover, it must keep consistent naming conventions, format, and coding.

This integration helps in effective analysis of data. Consistency in naming conventions, attribute measures, encoding structure etc. have to be ensured. Consider the following example:



In the above example, there are three different application labeled A, B and C. Information stored in these applications are Gender, Date, and Balance. However, each application's data is stored different way.

- In Application A gender field store logical values like M or F
- In Application B gender field is a numerical value,
- In Application C application, gender field stored in the form of a character value.
- Same is the case with Date and balance

However, after transformation and cleaning process all this data is stored in common format in the Data Warehouse.

Time-Variant

The time horizon for data warehouse is quite extensive compared with operational systems. The data collected in a data warehouse is recognized with a particular period and offers information from the historical point of view. It contains an element of time, explicitly or implicitly.

One such place where Datawarehouse data display time variance is in in the structure of the record key. Every primary key contained with the DW should have either implicitly or explicitly an element of time. Like the day, week month, etc.

Another aspect of time variance is that once data is inserted in the warehouse, it can't be updated or changed.

Non-volatile

Data warehouse is also non-volatile means the previous data is not erased when new data is entered in it.

Data is read-only and periodically refreshed. This also helps to analyze historical data and understand what & when happened. It does not require transaction process, recovery and concurrency control mechanisms.

Activities like delete, update, and insert which are performed in an operational application environment are omitted in Data warehouse environment. Only two types of data operations performed in the Data Warehousing are

1. Data loading
2. Data access

1.1.2.Here, are some major differences between Application and Data Warehouse

Operational Application	**Data Warehouse**
Complex program must be coded to make sure that data upgrade processes maintain high integrity of the final product.	This kind of issues does not happen because data update is not performed.
Data is placed in a normalized form to ensure minimal redundancy.	Data is not stored in normalized form.
Technology needed to support issues of transactions, data recovery, rollback, and	It offers relative simplicity in technology.

resolution as its deadlock is quite complex.

1.1.3.Data Warehouse Architectures

There are mainly three types of Datawarehouse Architectures: -

Single-tier architecture

The objective of a single layer is to minimize the amount of data stored. This goal is to remove data redundancy. This architecture is not frequently used in practice.

Two-tier architecture

Two-layer architecture separates physically available sources and data warehouse. This architecture is not expandable and also not supporting a large number of end-users. It also has connectivity problems because of network limitations.

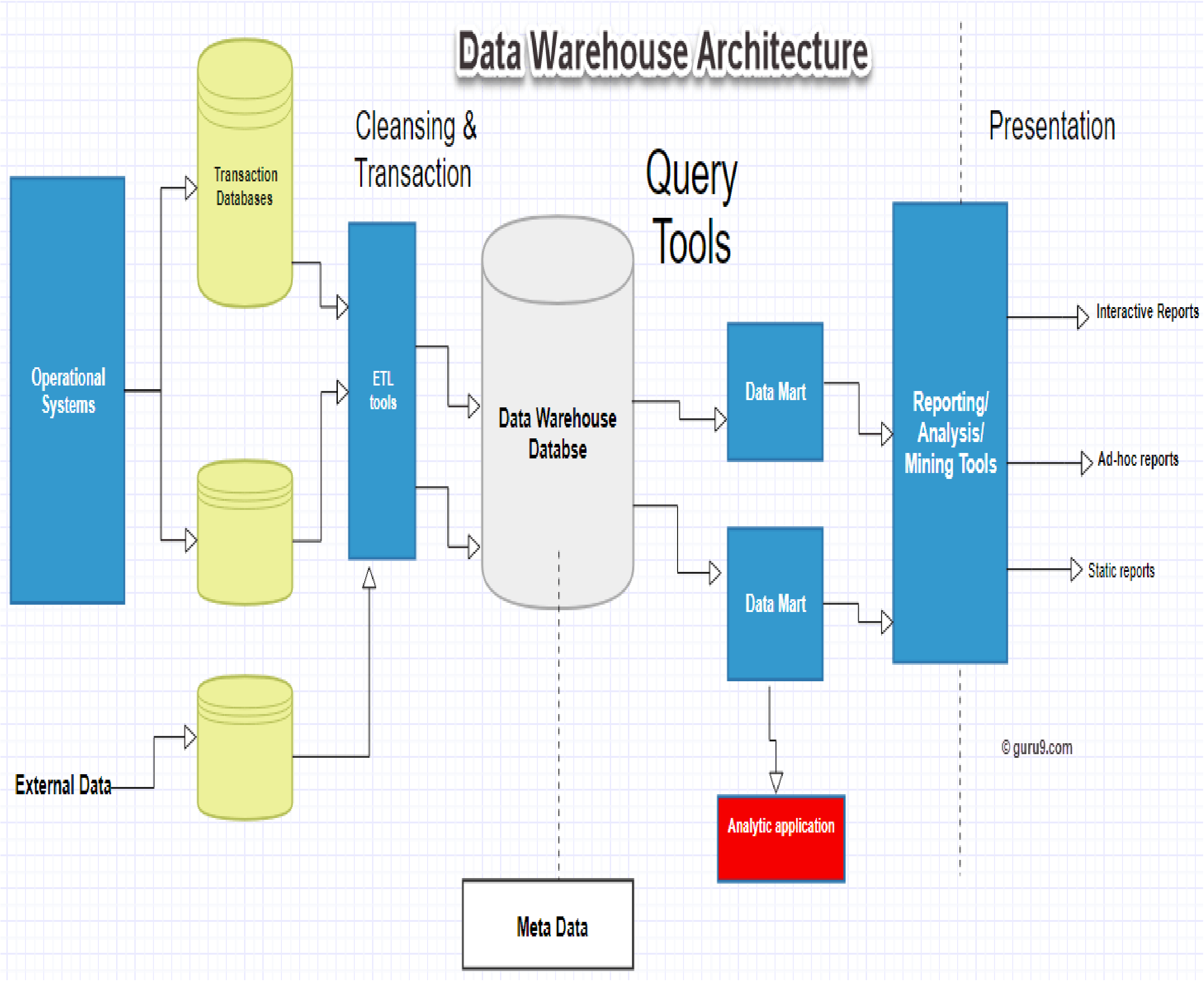
Three-tier architecture

This is the most widely used architecture.

It consists of the Top, Middle and Bottom Tier.

1. **Bottom Tier:** The database of the Datawarehouse servers as the bottom tier. It is usually a relational database system. Data is cleansed, transformed, and loaded into this layer using back-end tools.
2. **Middle Tier:** The middle tier in Data warehouse is an OLAP server which is implemented using either ROLAP or MOLAP model. For a user, this application tier presents an abstracted view of the database. This layer also acts as a mediator between the end-user and the database.
3. **Top-Tier:** The top tier is a front-end client layer. Top tier is the tools and API that you connect and get data out from the data warehouse. It could be Query tools, reporting tools, managed query tools, Analysis tools and Data mining tools.

1.1.4.Datawarehouse Components



The data warehouse is based on an RDBMS server which is a central information repository that is surrounded by some key components to make the entire environment functional, manageable and accessible

There are mainly five components of Data Warehouse:

Data Warehouse Database

The central database is the foundation of the data warehousing environment. This database is implemented on the RDBMS technology. Although, this kind of implementation is constrained by the fact that traditional RDBMS system is optimized for transactional database processing and not for data warehousing. For instance, ad-hoc query, multi-table joins, aggregates are resource intensive and slow down performance.

Hence, alternative approaches to Database are used as listed below-

- In a datawarehouse, relational databases are deployed in parallel to allow for scalability. Parallel relational databases also allow shared memory or shared nothing model on various multiprocessor configurations or massively parallel processors.
- New index structures are used to bypass relational table scan and improve speed.
- Use of multidimensional database (MDDBs) to overcome any limitations which are placed because of the relational data model. Example: Essbase from Oracle.

Sourcing, Acquisition, Clean-up and Transformation Tools (ETL)

The data sourcing, transformation, and migration tools are used for performing all the conversions, summarizations, and all the changes needed to transform data into a unified format in the datawarehouse. They are also called Extract, Transform and Load (ETL) Tools.

Their functionality includes:

- Anonymize data as per regulatory stipulations.
- Eliminating unwanted data in operational databases from loading into Data warehouse.
- Search and replace common names and definitions for data arriving from different sources.
- Calculating summaries and derived data
- In case of missing data, populate them with defaults.
- De-duplicated repeated data arriving from multiple datasources.

These Extract, Transform, and Load tools may generate cron jobs, background jobs, Cobol programs, shell scripts, etc. that regularly update data in datawarehouse. These tools are also helpful to maintain the Metadata.

These ETL Tools have to deal with challenges of Database & Data heterogeneity.

Metadata

The name Meta Data suggests some high- level technological concept. However, it is quite simple. Metadata is data about data which defines the data warehouse. It is used for building, maintaining and managing the data warehouse.

In the Data Warehouse Architecture, meta-data plays an important role as it specifies the source, usage, values, and features of data warehouse data. It also defines how data can be changed and processed. It is closely connected to the data warehouse.

For example, a line in sales database may contain:

```
4030 KJ732 299.90
```

This is a meaningless data until we consult the Meta that tell us it was

- Model number: 4030
- Sales Agent ID: KJ732
- Total sales amount of $299.90

Therefore, Meta Data are essential ingredients in the transformation of data into knowledge.

Metadata helps to answer the following questions

- What tables, attributes, and keys does the Data Warehouse contain?
- Where did the data come from?
- How many times do data get reloaded?
- What transformations were applied with cleansing?

Metadata can be classified into following categories:

1. **Technical Meta Data**: This kind of Metadata contains information about warehouse which is used by Data warehouse designers and administrators.
2. **Business Meta Data:** This kind of Metadata contains detail that gives end-users a way easy to understand information stored in the data warehouse.

1.1.5 Query Tools

One of the primary objects of data warehousing is to provide information to businesses to make strategic decisions. Query tools allow users to interact with the data warehouse system.

These tools fall into four different categories:

1. Query and reporting tools
2. Application Development tools
3. Data mining tools
4. OLAP tools

1. Query and reporting tools:

Query and reporting tools can be further divided into

- Reporting tools

- Managed query tools

Reporting tools: Reporting tools can be further divided into production reporting tools and desktop report writer.

1. Report writers: This kind of reporting tool are tools designed for end-users for their analysis.
2. Production reporting: This kind of tools allows organizations to generate regular operational reports. It also supports high volume batch jobs like printing and calculating. Some popular reporting tools are Brio, Business Objects, Oracle, PowerSoft, SAS Institute.

Managed query tools:

This kind of access tools helps end users to resolve snags in database and SQL and database structure by inserting meta-layer between users and database.

2. Application development tools:

Sometimes built-in graphical and analytical tools do not satisfy the analytical needs of an organization. In such cases, custom reports are developed using Application development tools.

3. Data mining tools:

Data mining is a process of discovering meaningful new correlation, pattens, and trends by mining large amount data. Data mining tools are used to make this process automatic.

4. OLAP tools:

These tools are based on concepts of a multidimensional database. It allows users to analyse the data using elaborate and complex multidimensional views.

1.1.6Data warehouse Bus Architecture

Data warehouse Bus determines the flow of data in your warehouse. The data flow in a data warehouse can be categorized as Inflow, Upflow, Downflow, Outflow and Meta flow.

While designing a Data Bus, one needs to consider the shared dimensions, facts across data marts.

Data Marts

A data mart is an access layer which is used to get data out to the users. It is presented as an option for large size data warehouse as it takes less time and money to build. However, there is no standard definition of a data mart is differing from person to person.

In a simple word Data mart is a subsidiary of a data warehouse. The data mart is used for partition of data which is created for the specific group of users.

Data marts could be created in the same database as the Datawarehouse or a physically separate Database.

Data warehouse Architecture Best Practices

To design Data Warehouse Architecture, you need to follow below given best practices:

- Use a data model which is optimized for information retrieval which can be the dimensional mode, denormalized or hybrid approach.
- Need to assure that Data is processed quickly and accurately. At the same time, you should take an approach which consolidates data into a single version of the truth.
- Carefully design the data acquisition and cleansing process for Data warehouse.
- Design a MetaData architecture which allows sharing of metadata between components of Data Warehouse
- Consider implementing an ODS model when information retrieval need is near the bottom of the data abstraction pyramid or when there are multiple operational sources required to be accessed.
- One should make sure that the data model is integrated and not just consolidated. In that case, you should consider 3NF data model. It is also ideal for acquiring ETL and Data cleansing tools

1.2Modeling

1.3Design and usage

1.4Implementation : Data cube Computation Methods

The best way to address the business risk associated with a Datawarehouse implementation is to employ a three-prong strategy as below

1. **Enterprise strategy**: Here we identify technical including current architecture and tools. We also identify facts, dimensions, and attributes. Data mapping and transformation is also passed.
2. **Phased delivery**: Datawarehouse implementation should be phased based on subject areas. Related business entities like booking and billing should be first implemented and then integrated with each other.
3. **Iterative Prototyping**: Rather than a big bang approach to implementation, the Datawarehouse should be developed and tested iteratively.

Here, are key steps in Datawarehouse implementation along with its deliverables.

Step	**Tasks**	**Deliverables**
1	Need to define project scope	Scope Definition
2	Need to determine business needs	Logical Data Model
3	Define Operational Datastore requirements	Operational Data Store Model
4	Acquire or develop Extraction tools	Extract tools and Software
5	Define Data Warehouse Data requirements	Transition Data Model
6	Document missing data	To Do Project List
7	Maps Operational Data Store to Data Warehouse	D/W Data Integration Map
8	Develop Data Warehouse Database design	D/W Database Design
9	Extract Data from Operational Data Store	Integrated D/W Data Extracts
10	Load Data Warehouse	Initial Data Load
11	Maintain Data Warehouse	On-going Data Access and Subsequent Loads

1.5Data Generalization by Attribute

Oriented Induction approach.

Conceptually, the data cube can be viewed as a kind of multidimensional data generalization. In general, *data generalization* summarizes data by replacing relatively low-level values (e.g., numeric values for an attribute *age*) with higher-level concepts (e.g., *young*, *middle-aged*, and *senior*), or by reducing the number of dimensions to summarize data in concept space involving fewer dimensions (e.g., removing*birth_date*and *telephone number* when summarizing the behavior of a group of students). Given the large amount of data stored in databases, it is useful to be able to describe concepts in concise and succinct terms at generalized (rather than low) levels of abstraction.

1.6. QUESTION BANK

PART-A

1. Define data warehouse?

2. What are operational databases?

3. Define OLTP? (Nov/Dec 2009)

4. Define OLAP?

5. How a database design is represented in OLTP systems?

6. How a database design is represented in OLAP systems?

7. Write short notes on multidimensional data model? (Nov/Dec 2012)

8. Define data cube?

9. What are facts?

10. What are dimensions?

11. Define dimension table?

12. Define fact table?

13. What are lattice of cuboids?

14. What is apex cuboid?

15. List out the components of star schema? (Nov/Dec 2010)

16. What is snowflake schema?

17. List out the components of fact constellation schema? (Nov/Dec 2009)

18. Point out the major difference between the star schema and the snowflake schema?

19. Which is popular in the data warehouse design, star schema model (or) snowflake schema model?

20. Define concept hierarchy?

21. Define total order?

22. Define partial order?

23. Define schema hierarchy?

24. List out the OLAP operations in multidimensional data model? (Nov/Dec 2011)

25. What is roll-up operation?

26. What is drill-down operation?

27. What is slice operation?

28. What is dice operation?

29. What is pivot operation?

30. List out the views in the design of a data warehouse? (Nov/Dec 2010)

31. What are the methods for developing large software systems?

32. How the operation is performed in waterfall method?

33. How the operation is performed in spiral method?

34. List out the steps of the data warehouse design process?

35. Define ROLAP? 36. Define MOLAP? (Nov/Dec 2010)

37. Define HOLAP? 38. What is enterprise warehouse?

39. What is data mart?

40. What are dependent and independent data marts?

41. What is virtual warehouse?

42. Define indexing?

43. What are the types of indexing? (Nov/Dec 2011)

44. Define metadata? (Nov/Dec 2012)

45. Define VLDB?

PART-B (16 MARKS)

1. Discuss the components of data warehouse. (Nov/Dec 2010)
2. List out the differences between OLTP and OLAP.
3. Discuss the various schematic representations in multidimensional model.
4. Explain the OLAP operations I multidimensional model. (Nov/Dec 2012)
5. Explain the design and construction of a data warehouse.
6. Explain the three-tier data warehouse architecture. (April/May 2011)
7. Explain indexing.
8. Write notes on metadata repository.
9. Write short notes on VLDB.

1.7. REFERENCES

en.wikipedia.org/wiki/data warehousing