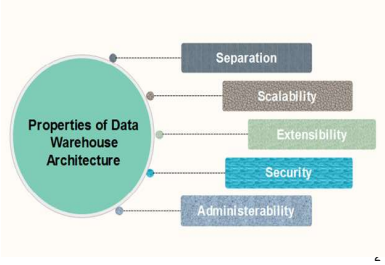
Data Warehouse Architectures

1

Properties of Data Warehouse Architectures

- The following architecture properties are necessary for a data warehouse system:



2

Properties of DW Architectures

- **1. Separation:** Analytical and transactional processing should be keep apart as much as possible.
- **2. Scalability:** Hardware and software architectures should be simple to upgrade the data volume, which has to be managed and processed, and the number of user's requirements, which have to be met, progressively increase.

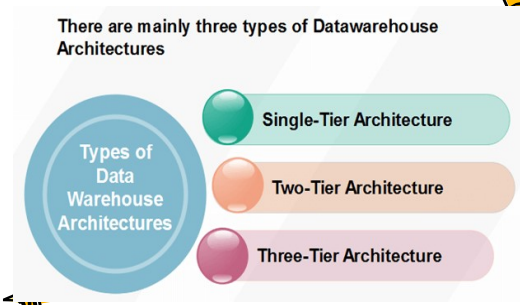
3

Properties of DW Architectures

- **3. Extensibility:** The architecture should be able to perform new operations and technologies without redesigning the whole system.
- **4. Security:** Monitoring accesses are necessary because of the strategic data stored in the data warehouses.
- **5. Administerability:** Data Warehouse management should not be complicated.

4

Types of Data Warehouse Architectures



5

Single-Tier Architecture

- Single-Tier architecture is not periodically used in practice.
- Its purpose is to minimize the amount of data stored to reach this goal; it removes data redundancies.
- The figure shows the only layer physically available is the source layer.
- In this method, data warehouses are virtual.

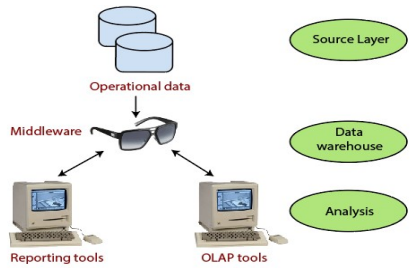
6

Single-Tier Architecture

- This means that the data warehouse is implemented as a multidimensional view of operational data created by specific middleware, or an intermediate processing layer.

7

Single-Tier Architecture



Single-Tier Data Warehouse Architecture

8

Single-Tier Architecture

- The vulnerability of this architecture lies in its failure to meet the requirement for separation between analytical and transactional processing.
- Analysis queries are agreed to operational data after the middleware interprets them.
- In this way, queries affect transactional workloads.

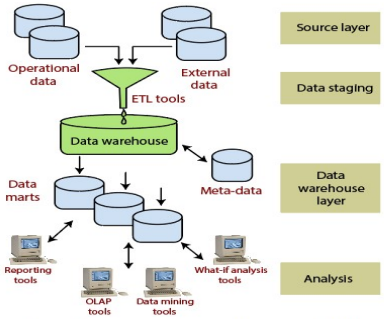
9

Two-Tier Architecture

- The requirement for separation plays an essential role in defining the two-tier architecture for a data warehouse system, as shown in fig:
-

10

Two-Tier Architecture



Two-Tier Data Warehouse Architecture

11

Two-Tier Architecture

- Although it is typically called two-layer architecture to highlight a separation between physically available sources and data warehouses, in fact, consists of four subsequent data flow stages:

12

Two-Tier Architecture

- **Source layer:** A data warehouse system uses a heterogeneous source of data.
- That data is stored initially to corporate relational databases or legacy databases, or it may come from an information system outside the corporate walls.

13

Two-Tier Architecture

- **Data Staging:** The data stored to the source should be extracted, cleansed to remove inconsistencies and fill gaps, and integrated to merge heterogeneous sources.
- The **Extraction**, **Transformation**, and **Loading Tools (ETL)** can combine heterogeneous schemata, extract, transform, cleanse, validate, filter, and load source data into a data warehouse.

14

Two-Tier Architecture

- **Data Warehouse layer:** Information is saved to one logically centralized individual repository: a data warehouse.
- The data warehouses can be directly accessed, but it can also be used as a source for creating data marts, which partially replicate data warehouse contents and are designed for specific enterprise departments.

15

Two-Tier Architecture

- **Analysis:** In this layer, integrated data is efficiently, and flexible accessed to issue reports, dynamically analyze information, and simulate hypothetical business scenarios.
- It should feature aggregate information navigators, complex query optimizers, and customer-friendly GUIs.

16

Three-Tier Architecture

- The three-tier architecture consists of the source layer (containing multiple source system), the reconciled layer and the data warehouse layer (containing both data warehouses and data marts).
- The reconciled layer sits between the source data and data warehouse.

17

Three-Tier Architecture

- The main advantage of the **reconciled layer** is that it creates a standard reference data model for a whole enterprise.
- At the same time, it separates the problems of source data extraction and integration from those of data warehouse population.

18

Three-Tier Architecture

- In some cases, the **reconciled layer** is also directly used to accomplish better some operational tasks, such as producing daily reports that cannot be satisfactorily prepared using the corporate applications or generating data flows to feed external processes periodically to benefit from cleaning and integration.

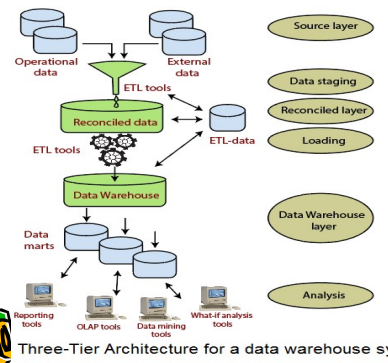
19

Three-Tier Architecture

- This architecture is especially useful for the extensive, enterprise-wide systems.
- A disadvantage of this structure is the extra file storage space used through the extra redundant reconciled layer.
- It also makes the analytical tools a little further away from being real-time.

20

Three-Tier Architecture



Three-Tier Architecture for a data warehouse system

21

Three-Tier Architecture

- Data Warehouses usually have a three-level (tier) architecture that includes:
  - Bottom Tier (Data Warehouse Server)
  - Middle Tier (OLAP Server)
  - Top Tier (Front end Tools).

22

A bottom-tier

- It consists of the **Data Warehouse server**, which is almost always an RDBMS.
- It may include several specialized data marts and a metadata repository.
- Data from operational databases and external sources (such as user profile data provided by external consultants) are extracted using application program interfaces called a gateway.

23

A bottom-tier

- A gateway is provided by the underlying DBMS and allows customer programs to generate SQL code to be executed at a server.
- **Examples** of gateways contain **ODBC** (Open Database Connection) and **OLE-DB** (Open-Linking and Embedding for Databases), by **Microsoft**, and **JDBC** (Java Database Connection).

24

A middle-tier

- A **middle-tier** which consists of an **OLAP server** for fast querying of the data warehouse.
- The OLAP server is implemented using either
  - A **Relational OLAP (ROLAP) model**, i.e., an extended relational DBMS that maps functions on multidimensional data to standard relational operations.

25

A middle-tier

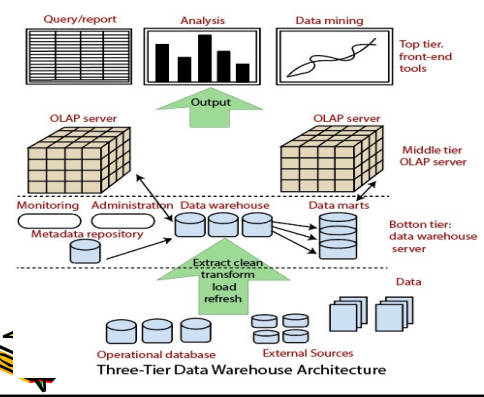
- A **Multidimensional OLAP (MOLAP) model**, i.e., a particular purpose server that directly implements multidimensional information and operations.

26

A top-tier

- A **top-tier** that contains **front-end tools** for displaying results provided by OLAP, as well as additional tools for data mining of the OLAP-generated data.
- The overall Data Warehouse Architecture is shown in fig:

27



Three-Tier Data Warehouse Architecture

28

The metadata repository

- The **metadata repository** stores information that defines DW objects.
- It includes the following parameters and information for the middle and the top-tier applications:
  - A description of the DW structure, including the warehouse schema, dimension, hierarchies, data mart locations, and contents, etc.

29

The metadata repository

- Operational metadata, which usually describes the currency level of the stored data, i.e., active, archived or purged, and warehouse monitoring information, i.e., usage statistics, error reports, audit, etc.
- System performance data, which includes indices, used to improve data access and retrieval performance.

30

The **metadata repository**

- Information about the mapping from operational databases, which provides source **RDBMSs** and their contents, cleaning and transformation rules, etc.
- Summarization algorithms, predefined queries, and reports business data, which include business terms and definitions, ownership information, etc.

31

Questions