Types of Data Warehouses

1. Host Based Data Warehouses
   - **Host Based (MVS) Data Warehouses**
     The data warehouses that reside on high-volume databases on MVS are the host-based type of data warehouses.
     Such data warehouses
     1. usually have very high volumes of data storage
     2. require support for both MVS and client-based report and query facilities.
     3. have very complex source systems
     4. require continuous maintenance since these must be used for mission-critical purposes.
     Steps to build such a data warehouse.
     - **Unload Phase** involves selecting and scrubbing the operational data.
     - **Transform Phase** for translating it into an appropriate form and defining the rules for accessing and storing it.
     - **Load phase** for moving the data directly into DB2 tables or a special file for moving it to another database or non-MVS warehouse.
   - **Host Based (Unix) Data Warehouses**
     Oracle and Informix RDBMSs provide the facilities for such data warehouses. Both of these databases can extract data from MVS-based databases as well as a larger number of other UNIX-based databases.

2. Host Based single-stage (LAN) Data Warehouses
   With a LAN-based warehouse, data delivery can be managed either centrally or from the workgroup environment so that business groups can meet and manage their own information needs without burdening centralized IT resources.
   Limitations/challenges:
   - LAN-based warehousing solutions are normally limited by both DBMS and hardware scalability factors.
   - Many LAN based enterprises have not implemented adequate job scheduling, recovery management, organized maintenance, and performance monitoring procedures to support robust warehousing solutions.
   - Often these warehouses are dependent on other platforms for source data. Building an environment that has data integrity, recoverability, and security needs careful design, planning and implementation. Otherwise, synchronisation of changes and loads from sources to server could cause innumerable problems.

3. LAN Based workgroup Data Warehouses
   In this warehouse, you extract data from a variety of sources (like Oracle, IMS, DB2) and provide multiple LAN-based warehouses.
   Designed for workgroup environment, it is ideal for any business organization that wishes to build a data warehouse, often called a data mart. Usually requires minimal initial investment and technical training. Its low startup cost and ease of use allow a workgroup to quickly build and easily manage its own custom data mart.
   Common Issues:
   - Lack of understanding how to distribute data and supporting intentional data redundancy for performance reasons.
   - Many organizations may not have adequate job scheduling, recovery management, and performance monitoring to support robust warehousing solutions.
   - Although providing +ve cost benefits, LAN-based warehousing solutions can be limited by both hardware and DBMS limitations.
For many large enterprises, similar skills in database design, maintenance, and recovery are not present in every workgroup environment.

4. Multistage Datawarehouses

This configuration is well suited to environments where endusers in different capacities require access to both summarized data for up-to-the-minute tactical decisions as well as summarized, cumulative data for long-term strategic decisions. Both ODS (Operation Data Store) and the data warehouse may reside on host-based on LAN-based databases, depending on volume and usage requirements. Typically the ODS stores only the most recent records. The data warehouse stores the historical evolution of the records.

5. Stationary Datawarehouses

In this type of a data warehouse, user are given direct access to the data, instead of moving from the sources. For many organizations, infrequent access, volume issues or corporate necessities dictate such an approach.

This is likely to impact performace since users will be competing with the production data stores.

Such a warehouse will require sophisticated middleware, possible with a single interface to the user. An integrated metadata repository becomes an absolute necessity under this environment.

6. Distributed Datawarehouses

There are at least two types of distributed data warehouses and their variations for the enterprise: local warehouses distributed throughout the enterprises and a global warehouse.

Useful when there are diverse businesses under the same enterprise umbrella. This approach may be necessary if a local warehouse already existed, prior to joining the enterprise.

Local data warehouses have the following common characteristics:

1. Activity occurs at local level
2. Majority of the operational processing is done at the local site.
3. Local site is autonomous
4. Each local data warehouse has its own unique structure and content of data.
5. The data is unique and of prime importance to that locality only.
6. Majority of the data is local and not replicated.
7. Any intersection of data between local data warehouses is coincidental.
8. Local site serves different geographic regions.
9. Local site serves different technical communities.

The primary motivation in implementing distributed data warehouses is that integration of the entire enterprise data does not make sense. It is reasonable to assume that an enterprise will have at least some natural intersections of data from one local site to another. If there is any intersection, then it is usually contained in a global data warehouse.

7. Virtual Datawarehouse

The data warehouse is a great idea, but it is complex to build and requires investment. Why not use a cheap and fast approach by eliminating the transformation steps of repositories for metadata and another database. This approach is termed the 'virtual data warehouse'.

To accomplish this there is need to define 4 kinds of information:

1. A data dictionary containing the definitions of the various databases.
2. A description of the relationship among the data elements.
3. The description of the way user will interface with the system.
4. The algorithms and business rules that define what to do and how to do it.

Disadvantages:

1. Since queries compete with production data transactions, performance can be degraded.
2. There is no metadata, no summary data or no individual DSS (Decision Support
System) integration or history. All queries must be repeated, causing additional burden on the system.

3. There is no refreshing process, causing the queries to be very complex.