Data Warehouses and Deployment

Ankita Dubey

This document contains the notes about data warehouses and lifecycle for data warehouse deployment project. This can be useful for students or working professionals to gain the basic knowledge about Data warehouses.
• **Data Warehouses**

A database consists of one or more files that need to be stored on a computer. In large organizations, databases are typically not stored on the individual computers of employees but in a central system. This central system typically consists of one or more computer servers. A server is a computer system that provides a service over a network. The server is often located in a room with controlled access, so only authorized personnel can get physical access to the server.

In a typical setting, the database files reside on the server, but they can be accessed from many different computers in the organization. As the number and complexity of databases grows, we start referring to them together as a data warehouse.

A data warehouse is a collection of databases that work together. A data warehouse makes it possible to integrate data from multiple databases, which can give new insights into the data. The ultimate goal of a database is not just to store data, but to help businesses make decisions based on that data. A data warehouse supports this goal by providing architecture and tools to systematically organize and understand data from multiple databases.

• **Data warehouse Deployment**

Lifecycle for data warehouse deployment project:

0. **Project Scoping and Planning**

Project Triangle – Scope, Time and Resource.

- Determine the scope of the project – what you would like to accomplish? This can be defined by questions to be answered. The number of logical star and number of the OLTP sources
- Time – What is the target date for the system to be available to the users
- Resource – What is our budget? What is the role and profile requirement of the resources needed to make this happen?

1. **Requirement**

- What are the business questions? How does the answers of these questions can change the business decision or trigger actions.
- What are the role of the users? How often do they use the system? Do they do any interactive reporting or just view the defined reports in guided navigation?
- How do you measure? What are the metrics?

2. **Front-End Design**
The front end design needs for both interactive analysis and the designed analytics workflow.
How does the user interact with the system?
What are their analysis processes?

3. Warehouse Schema Design

- Dimensional modelling – define the dimensions and fact and define the grain of each star schema.
- Define the physical schema – depending on the technology decision. If you use the relational technology, design the database tables.

4. OLTP to data warehouse mapping

- Logical mapping – table to table and column to column mapping. Also define the transformation rules.
- You may need to perform OLTP data profiling. How often the data changes? What is the data distribution?
- ETL Design – include data staging and the detail ETL process flow.

5. Implementation

- Create the warehouse and ETL staging schema
- Develop the ETL programs
- Create the logical to physical mapping in the repository
- Build the end user dashboard and reports

6. Deployment

- Install the Analytics reporting and the ETL tools.
- Specific Setup and Configuration for OLTP, ETL, and data warehouse.
- Sizing of the system and database
- Performance Tuning and Optimization

7. Management and Maintenance of the system

- Ongoing support of the end-users, including security, training, and enhancing the system.
- You need to monitor the growth of the data.

- Growth and maintenance of Data warehouse

Assume the following plausible scenario. All the user acceptance tests were successful. There were two pilots; one was completed to test the specialized end-user toolset and the other was an expandable seed pilot that led to the deployment. Your project team has successfully deployed the initial version of the data warehouse. The
users are ecstatic. The first week after deployment there were just a few teething problems. Almost all the initial users appear to be fully trained. With very little assistance from IT, the users seem to take care of themselves. The first set of OLAP cubes proved their worth and the analysts are already happy. Users are receiving reports over the Web. All the hard work has paid off. Now what? This is just the beginning. More data marts and more deployment versions have to follow. The team needs to ensure that it is well poised for growth. You need to make sure that the monitoring functions are all in place to constantly keep the team informed of the status. The training and support functions must be consolidated and streamlined. The team must confirm that all the administrative functions are ready and working. Database tuning must continue at a regular pace.

Immediately following the initial deployment, the project team must conduct review sessions. Here are the major review tasks:

- Review the testing process and suggest recommendations.
- Review the goals and accomplishments of the pilots.
- Survey the methods used in the initial training sessions. Document highlights of the development process.
- Verify the results of the initial deployment, matching these with user expectations.

The review sessions and their outcomes form the basis for improvement in the further releases of the data warehouse. As you expand and produce further releases, let the business needs, modelling considerations, and infrastructure factors remain as the guiding factors for growth. Follow each release close to the previous release. You can make use of the data modelling done in the earlier release. Build each release as a logical next step. Avoid disconnected releases. Build on the current infrastructure.

**MONITORING THE DATA WAREHOUSE**

When you implement an OLTP system, you do not stop with the deployment. The database administrator continues to inspect system performance. The project team continues to monitor how the new system matches up with the requirements and delivers the results. Monitoring the data warehouse is comparable to what happens in an OLTP system, except for one big difference. Monitoring an OLTP system dwindles in comparison with the monitoring activity in a data warehouse environment. As you can easily perceive, the scope of the monitoring activity in the data warehouse extends over many features and functions. Unless data warehouse monitoring takes place in a formalized manner, desired results cannot be achieved. The results of the monitoring gives you the data needed to plan for growth and to improve performance.

**Collection of Statistics**

The following is a random list that includes statistics for different uses. You will find most of these applicable to your environment. Here is the list:

- Physical disk storage space utilization
- Number of times the DBMS is looking for space in blocks or causes fragmentation
- Memory buffer activity
Using Statistics for Growth Planning

We indicate below the types of action that are prompted by the monitoring statistics:

- Allocate more disk space to existing database tables
- Plan for new disk space for additional tables
- Modify file block management parameters to minimize fragmentation
- Create more summary tables to handle large number of queries looking for summary information
- Reorganize the staging area files to handle more data volume
- Add more memory buffers and enhance buffer management
- Upgrade database servers
- Offload report generation to another middle tier
- Smooth out peak usage during the 24-hour cycle
- Partition tables to run loads in parallel and to manage backups

Using Statistics for Fine-Tuning

The next best use of statistics relates to performance. Below is the data warehouse functions that are normally improved based on the information derived from the statistics:

- Query performance
- Query formulation
- Incremental loads
- Frequency of OLAP loads
- OLAP system
- Data warehouse content browsing
- Report formatting
- Report generation

Publishing Trends for Users

This is a new concept not usually found in OLTP systems. In a data warehouse, the users must find their way into the system and retrieve the information by themselves.
They must know about the contents. Users must know about the currency of the data in the warehouse.

**USER TRAINING AND SUPPORT**

The transformation functions cover all the requirements. The staging area has been laid out well and it supports every function carried out there. Loading of the data warehouse takes place without a flaw. Your end-users have the most effective tools for information retrieval and the tools fit their requirements as closely as possible. Every component of the data warehouse works correctly and well. With everything in place and working, if the users do not have the right training and support, none of the team’s efforts matters. It could be one big failure. You cannot overstate the significance of user training and support, both initially and on an ongoing basis. True, when the project team selected the vendor tools, perhaps some of the users received initial training on the tools. This can never be a substitute for proper training. You have to set up a training program taking into consideration all of the areas where the users must be trained. In the initial period, and continuing after deployment of the first version of the data warehouse, the users need the support to carry on. Do not underestimate the establishment of a meaningful and useful support system. You know about the technical and application support function in OLTP system implementations. For a data warehouse, because the workings are different and new, proper support becomes even more essential.

**User Training Content**

While designing the content of user education, you have to make it broad and deep. Remember, the users to be trained in your organization come with different skills and knowledge levels. Generally, users preparing to use the data warehouse possess basic computer skills and know how computer systems work. But to almost all of the users, data warehousing must be novel and different. Among other things, three significant components must be present in the training program. First, the users must get a good grasp of what is available for them in the warehouse. They must clearly understand the data content and how to get to the data. Second, you must tell the users about the applications. What are the pre-constructed applications? Can they use the predefined queries and reports? If so, how? Next, you must train the users on the tools they need to employ to access the information.

**Preparing the Training Program**

The training program varies with the requirements of each organization. Here are a few general tips for putting together a solid user training program:

- A successful training program depends on the joint participation of user representatives and IT. The user representatives on the project team and the subject area experts in the user departments are suitable candidates to work with IT.
- Let both IT and users work together in preparing the course materials.
- Remember to include topics on data content, applications, and tool usage.
- Make a list of all the current users to be trained. Place these users into logical groups based on knowledge and skill levels. Determine what each group needs to
be trained on. By doing this exercise, you will be able to tailor the training program to exactly match the requirements of your organization.

- Determine how many different training courses would actually help the users. A good set of courses consists of an introductory course, an in-depth course, and a specialized course on tool usage.
- The introductory course usually runs for one day. Every user must go through this basic course.
- Have several tracks in the in-depth course. Each track caters to a specific user group and concentrates on one or two subject areas.
- The specialized course on tool usage also has a few variations, depending on the different tool sets. OLAP users must have their own course.
- Keep the course documentation simple and direct and include enough graphics. If the course covers dimensional modelling, a sample STAR schema helps the users to visualize the relationships. Do not conduct a training session without course materials.
- As you already know, hands-on sessions are more effective. The introductory course may just have a demo, but the other two types of courses go well with hands-on exercises.

**Delivering the Training Program**

Training programs must be ready before the deployment of the first version of the data warehouse. Schedule the training sessions for the first set of users closer to the deployment date. What the users learned at the training sessions will be fresh in their minds. How the first set of users perceive the usefulness of the data warehouse goes a long way to ensure a successful implementation, so pay special attention to the first group of users.

**User Support**

User support must commence the minute the first user clicks on his mouse to get into the data warehouse. This is not meant to be dramatic, but to emphasize the significance of proper support to the users. As you know, user frustration mounts in the absence of a good support system. Support structure must be in place before the deployment of the first version of the data warehouse. If you have a pilot planned or an early deliverable scheduled, make sure the users will have recourse to getting support.

**MANAGING THE DATA WAREHOUSE**

After the deployment of the initial version of the data warehouse, the management function switches gear. Until now, the emphasis remained on following through the steps of the data warehouse development life cycle. Design, construction, testing, user acceptance, and deployment were the watchwords. Now, at this point, data warehouse management is concerned with two principal functions. The first is maintenance management. The data warehouse administrative team must keep all the functions going in the best possible manner. The second is change management. As new versions of the warehouse are deployed, as new releases of the tools become available, as improvements and automation take place in the ETL functions, the administrative team’s focus includes enhancements and revisions. In this section, let
us consider a few important aspects of data warehouse management. We will point out the essential factors. Post deployment administration covers the following areas:

- Performance monitoring and fine-tuning
- Data growth management
- Storage management
- Network management
- ETL management
- Management of future data mart releases
- Enhancements to information delivery
- Security administration
- Backup and recovery management
- Web technology administration
- Platform upgrades
- Ongoing training
- User support

**Platform Upgrades**

Your data warehouse deployment platform includes the infrastructure, the data transport component, end-user information delivery, data storage, metadata, the database components, and the OLAP system components. More often, a data warehouse is a comprehensive cross-platform environment. The components follow a path of dependency, starting with computer hardware at the bottom, followed by the operating systems, communication systems, the databases, GUls, and then the application support software. As time goes on, upgrades to these components are announced by the vendors. After the initial rollout, have a proper plan for applying the new releases of the platform components. As you have probably experienced with OLTP systems, upgrades cause potentially serious interruption to the normal work unless they are properly managed. Good planning minimizes the disruption.

**Managing Data Growth**

Managing data growth deserves special attention. In a data warehouse, unless you are vigilant about data growth, it could get out of hand very soon and quite easily. Data warehouses already contain huge volumes of data. When you start with a large volume of data, even a small percentage increase can result in substantial additional data.

Here are just a few practical suggestions to manage data growth:
- Dispense with some detail levels of data and replace them with summary tables.
- Restrict unnecessary drill-down functions and eliminate the corresponding detail level data.
- Limit the volume of historical data. Archive old data promptly.
- Discourage analysts from holding unplanned summaries.
- Where genuinely needed, create additional summary tables.

**Storage Management**
As the volume of data increases, so does the utilization of storage. Because of the huge data volume in a data warehouse, storage costs rank very high as a percentage of the total cost. Here are a few tips on storage management to be used as guidelines:

- Additional rollouts of the data warehouse versions require more storage capacity.
- Plan for the increase.
- Ensure that the storage configuration is flexible and scalable. You must be able to add more storage with minimum interruption to the current users.
- Use modular storage systems. If not already in use, consider a switchover.
- If yours is a distributed environment with multiple servers having individual storage pools, consider connecting the servers to a single storage pool that can be intelligently accessed.
- As usage increases, plan to spread data over multiple volumes to minimize access bottlenecks.
- Ensure ability to shift data from bad storage sectors.
- Look for storage systems with diagnostics to prevent outages.

**ETL Management**

The following are useful suggestions on ETL (data extraction, transformation, loading) management:

- Run daily extraction jobs on schedule. If source systems are not available under extraneous circumstances, reschedule extraction jobs.
- If you employ data replication techniques, ensure that the result of the replication process checks out.
- Ensure that all reconciliation is complete between source system record counts and record counts in extracted files.
- Make sure all defined paths for data transformation and cleansing are traversed correctly.
- Resolve exceptions thrown out by the transformation and cleansing functions.
- Verify load image creation processes, including creation of the appropriate key values for the dimension and fact table rows.
- Check out the proper handling of slowly changing dimensions.
- Ensure completion of daily incremental loads on time.

**Data Model Revisions**

When you expand the data warehouse in future releases, the data model changes. If the next release consists of a new data mart on a new subject, then your model gets expanded to include the new fact table, dimension tables, and also any aggregate tables. The physical model changes. New storage allocations are made. Here is a partial list that may be expanded based on the conditions in your data warehouse environment:

- Revisions to metadata
- Changes to the physical design
- Additional storage allocations
- Revisions to ETL functions
- Additional predefined queries and preformatted reports
- Revisions to OLAP system
- Additions to security system
- Additions to backup and recovery system

**Information Delivery Enhancements**

- Ensure compatibility of the new tool set with all data warehouse components.
- If the new tool set is installed in addition to the existing one, switch your users over in stages.
- Ensure integration of end-user metadata.
- Schedule training on the new tool set.
- If there are any data-stores attached to the original tool set, plan for the migration of the data to the new tool set.

**Ongoing Fine-Tuning**

The techniques are very much the same except for one big difference: the data warehouse contains a lot more, in fact, many times more data than a typical OLTP system. The techniques will have to apply to an environment replete with mountains of data. Let us just go over a few practical suggestions:

- Have a regular schedule to review the usage of indexes. Drop the indexes that are no longer used.
- Monitor query performance daily. Investigate long-running queries. Work with the user groups that seem to be executing long-running queries. Create indexes if needed.
- Analyze the execution of all predefined queries on a regular basis. RDBMSs have query analyzers for this purpose.
- Review the load distribution at different times every day. Determine the reasons for large variations.
- Although you have instituted a regular schedule for ongoing fine-tuning, from time to time, you will come across some queries that suddenly cause grief. You will hear complaints from a specific group of users. Be prepared for such ad hoc fine-tuning needs. The data administration team must have staff set apart for dealing with these situations.