CHAPTER 3: LITERATURE REVIEW

3.1 INTRODUCTION

The information in an organization can be categorized in three ways according to need of managerial level i.e. strategic information used by top management e.g. information relating to long term planning, managerial information used by middle management e.g. sales analysis and operational information used by e.g. current stock levels etc. (EA99). EA99 suggested three different computer based systems to cater for the needs of each management level i.e. Decision support system (DSS), Management information system (MIS) and Data processing system (DPS) respectively. (ibid) As we know from general discussion in chapter one that data warehouse can cater for need of all three management levels in an efficient and effective manner.

All information system go through four stages namely system planning and selection, system analysis, system design and system implementation and operation. As our topic fall in the last stage of system development life cycle so we will discuss accordingly.

VGH01 listed seven steps for system implementation and operation, i.e. Coding, Testing, Installation, Documentation, Training, Support and Maintenance. These stages can further be grouped as, activities that lead to system going into operation coding, testing and installation, activities that are necessary for successful system operation documenting the system and training and supporting the users, activities that are ongoing and needed to keep the system working and up to date maintenance.
Coding is an intensive activity starts when design team is final with the physical design specification to turn the system into working computer code. Though testing is parallel to coding but it also needs proper planning to achieve the designed objectives of the system.

Installation of system is the replacing the existing system with the new one and includes conversion of existing data, software, documentation and work procedure according to new system. (VGH01) The outcomes of these activities are code, program documentation, test plan, test data, test results, user guides, user training plan, installation and conversion plan, hardware and software installation schedule, data conversion plan and site and facility remodeling plan.

The process of documentation is carried through out life cycle of the system but on this stage all the information about the system are properly and fully documented both for users and maintainers of the system. In corporations there may be specialized staff to provide training to maintenance workers and users and in small organization some users can be trained and rest can learn from them. The outcomes of these activates are system documentation, user documentation, user training classes, user training tutorials, computer based training aids, help desk, online help, bulletin boards and other support mechanisms.

The process of maintaining an information system is actually returning to the beginning of SDLC and repeating development steps. Four major activities occur within maintenance, obtaining maintenance requests, transforming request into changes, designing changes and implementing changes. The outcome of maintenance activity is new version of system along with update in all kind of documentation. In a system development life cycle this is the final stage and again leads to beginning of system
development so this is an critically important issue to deal with intensive care. As fundamental assumptions and properties of data, login or process module does not change in maintenance activity so wrongly identified maintenance will transform the system into a mess.

According to VGH01 maintenance is changes made to system to fix or enhance its functionality. They listed four types of maintenance namely corrective, adaptive, perfective and preventive. Corrective maintenance is changes made to a system to repair flaws in its design, coding or implementation. Adaptive maintenance is changes made to a system to evolve its functionality to changing business needs or technology. Perfective maintenance is changes made to a system to add new features or to improve performance and lastly preventive maintenance which is changes made to system to avoid possible future problems.

Now onward we will discuss maintenance issues relating to data warehouse to improve the performance of data warehouse through efficient maintenance.

**3.2 Data Warehouse Performance Management**

The process of data warehouse performance management is similar to that of the design of a data warehouse. It is similar in that like the design and analysis phases, the procedures utilized are very different from the processes adopted in a conventional OLTP type system life cycle. In a conventional system life cycle there exists usually numerous levels of analysis and planning. In the data warehouse environment the system builders are seldom given this luxury and are required to assemble a data warehouse in a rapid manner with little time for performance analysis and capacity planning. This makes the data
warehouse performance management process extremely difficult as the work loads very often cannot be predicted until finally the system is built for the first time and the data is in a production status. As a system goes into production for the first time only then may a system administrator discover there are performance problems.

If there are too many performance problems in the running of the data warehouse the viability of the project becomes marginal or questionable. It is important to remember the success of a data warehouse can only be measured once the data is loaded and users are able to make business level decisions by extracting data from the data warehouse.

The workload on a data warehouse hardly ever remains fixed. New users carry different kinds of demands, existing users change their focus and often the depth of their studies, the business cycle presents its own kinds of peaks and valleys, and in most cases the data warehouse expands as it stores data to cover longer periods of time. As the demand on a data warehouse changes, a lot of changes needed to be carried out to keep the performance graph in a positive direction. Some training courses needed to be introduced, some changed are needed for help desk, some indexes become obsolete and others need to be created, some aggregates are no longer referenced and others need to be evaluated, and the limits on parallel processing must be assessed and adjusted to fit the current demand. These and other tuning tasks should be carried out periodically to keep data warehouse performance smooth and constant.

3.3 Data Warehouse Maintenance

Data warehousing is becoming an increasingly important technology for information integration and data analysis. Given the dynamic nature of
modern distributed environments, both source data updates and schema changes are likely to occur autonomously and even concurrently in different data sources.

The data warehouse after its deployment needs to be treated as a production system, complete with service level agreements. Technical support for the data warehouse should constantly monitor the performance and system capacity trends and take measures to get maximum output from the system.

Six factors needed to be taken care of when dealing with ongoing data warehouse performance monitoring.

1. The data warehouse grows exponentially over time in terms of size and processing requirements.

2. Capacity management estimates, even based on the most precise calculations, are most likely to be still too conservative, requiring you to consider data warehouse expansion sooner than planned.

3. Data staging is a continuing challenge, especially if source systems are constantly in a state of change due to problems or changes. Some may be due for replacement under an ERP initiative, causing enormous changes to how data are sourced in the future.

4. Advances in technology in terms of network, hardware and software require more rapid release changes to be applied.

5. Ad hoc query access grows over time and must be carefully monitored as new and inexperienced users continue to run requests
against base tables rather than summary or aggregate tables to produce totals.

3.4 Software and Hardware Issues

client/server technology is less reliable, secure, and timely than its mainframe predecessor. Data access tools are just beginning to mature. Networks add new layers of complexity, and monitoring performance and tuning of servers is imperfect. The results are gaps in available technology and software, leaving users frustrated and their needs unmet. To overcome these problems warehouses needed to get their software and hardware updated in a timely manner to avoid any shortcomings in performance.

Updates to the data warehouse are inevitable; so too will be changes to package software, hardware servers, and the supporting network infrastructure. Three strategies are available to make changes to this technical layer depending upon the scope, timeframe and criticality of the data warehouse environment. These strategies include:

1. Installing new software releases, patches, hardware components or upgrades, and network connections (logical and physical) directly in the production environment.

2. Installing new software versions, hardware upgrades, and network improvement tasks in a temporary test environment and migrates or reconnects to production once certification testing has concluded.
3. Installing technical infrastructure changes into a permanent test or maintenance environment and migrate the production environment once certification testing has concluded.

### 3.4.1 Implementation Strategies

There are three implementation strategies to carry out these changes which are:

1. Peer to peer
2. Master to slave
3. Hybrid

**Trade Transition Approach (Peer to Peer)**

In this approach each environment acts as a distinct entity (RK00). This entails a large first data warehouse iteration, to be followed by smaller migrations of critical components over time because we are moving between distinct environment, this approach is the most time consuming and costly to adopt as a standard practice on an ongoing basis. This approach, however, safely isolates certification testing from the production environment.

**One Master Environment (Master to Slave)**

The master slave approach utilizes one environment for integrating changes and upgrades (RK00). The benefit of this approach is that new functionality and components are integrated and tested in one place as various new features and capabilities are introduced. After the certification testing is complete the master releases control to the slave.
3.5 A Hybrid Approach

In some cases, a combination, or hybrid, approach is adopted based on the type and nature of the certification requirement (RK00). For example, a traditional approach may be adequate for migrations of software releases but not so for upgrades to the network topology.

These three aspects of software and hardware installation are further complicated by the differing requirements between software, hardware, and networking components, since the features and functions of each product should align (e.g. a DBMS with strong parallelism features should be deployed on data servers with same properties). For implementation of new software, hardware, and network devices our considerations should include (RK00):

1. For software releases, the potential impact of changes to code and data structures should be taken into consideration. Even a simple unload/reload can become quite time consuming, especially if there is a VLDB implementation. In such cases, the users lose access to their data for a period of time unless some type of data mirroring is employed.

2. For installation of updated hardware such as servers and disks, one needs to be careful that the end users do not suffer. During certification and testing process users need to have access to their data all the time for enhanced performance. The options available largely depend upon the processor technology used since Cluster, AMP, MPP, and NUMA architectures have differing requirements for how the client and desktop technology are deployed.
3. The challenges for the network/communications environment are even more important especially if there is a distributed database operating across geographic regions and global boundaries. Enhancing the bandwidth and the ability of the network components and servers to handle large traffic volumes, as well as managing the differing types of protocols, often results in significant and risky challenges if not planned out well in advance.

3.5.1 The Certification Testing Process

Given below is a certification procedure which can be followed to effectively perform certification testing for software, hardware and the network (RK00). It describes overall management, training, procedure and script definition, and project control mechanisms that are required to provide due diligence to a testing process.

1. Initiate product certification: Includes collecting data warehouse product certification requirements and reviewing certification products requested for conformance to existing or future architecture. A core team member is selected who defines a certification strategy and leads it. The proposed certification budget is also presented in this phase.

2. Establish certification team environments: The working place and environment is defined here. The certification team obtains copies of products(s) for certification.
Certification procedures and techniques are outlined and developed along with the training of certification team.

3. Define certification test environment: A test environment is established for product certification. Data warehouse release stress
testing data and procedures for test verification are obtained. In the end certification test procedures, scripts and test data are developed/refined.

4. Undertake certification testing: During this phase the new products are installed or existing products are upgraded. Certification testing for all databases, decision support services, data staging, information access and supporting technology equipment is performed. Additional activities include product installation and testing, database testing, application components testing running in the same environment etc.

5. Conclude certification testing and prepare for production turnover: During this phase the capacity management is updated which after updating includes the new features, functionality and capabilities of the new environment. Update to the current data warehouse production environment is scheduled. Lists of certified products are also updated and certification results are prepared for review and approval.

6. Implement certified products: Once the approval for certification testing is received it is the time for implementing certified products. Current production components are backed up if required and new product releases are activated. The monitoring of the new product versions is handed over to the data warehouse support team. In the same way product problem management and support is also transferred to the data warehouse support group and the certification cycle is closed.

In undertaking a certification process, it is necessary to document what occurred and why and what was in or out of scope through the certification process (RK00). The generic deliverables described here discuss the types of information to be collected, which includes
1. Certification program
2. Certification business requirements
3. Certification test results (ibid)

3.5.2 Certification Program

A certification program is used to define the procedures, schedules, and facilities required to conduct certification testing (RK00). It describes the technical version or view of certification testing to the information services owner of the proposed data warehouse infrastructure improvement. To undertake the technical aspect of the certification program the following information is required.

1. A certification plan containing a list of steps, dates and resources presented as a GANTT chart or spreadsheet showing the certification testing scope and schedule or references to where this information can be found in electronic format.

2. A description of the test cycle(s) and each test cycle, test case, and the order in which they are exercised to verify the functionality of the hardware, software, or network improvement. Each test case consists of expected performance, inputs and outputs, before and after expected results, and a description of hardware, software or network product benchmark(s) to be met and a scoring method for testing.

3. A description of the automated and manual tools and techniques, if different from the development environment, for testing tools (software and hardware support facilities), test data generation method (generators, data entry, etc) and test benchmarking approach and related techniques.
4. A description of the proposed /available certification environment in terms of hardware, software, communications and user access profiles.

3.5.3 Certification Business Requirements

Certification business requirements describes the scope of certification testing in no technical terms and any conclusion or recommendations appropriate to the generation of any future enhancements to this or similar such efforts (RK00). It is also used to convey this understanding to the business stake holders and owners of the data warehouse. This deliverable contains the certification program definition explaining how the certification program was developed? Whether it was bought from some vendor or build in house? Is it customized from the current environment and describes the staffing model used to staff the certification program in terms of internal staff or external consulting?

A scope of testing explanation, which describes the scope in terms of duration, impact, risk, and cost of certification testing as detailed in the certification test program and includes information on test schedule, an end user introductory seminar schedule, list of facilities, allocated technology components like printers etc, and a list of technicians and business area end users for participation in the certification process. (RK00)

It also describes any risks associated with the certification program and certification program recommendations, which identify any final productivity improvements. (RK00)
3.5.4 Certification Test Results

As the name implies these are the results gathered after going through the certification testing process (RK00). Certification test results provide an evaluation on the state of readiness of the data warehouse infrastructure improvement, its associated interfaces, and all data or software conversion procedures. It identifies what was tested and what was not tested, what problem or issues remain outstanding, and any procedural impacts, work around, or risks.

Finally, it provides recommendations whether to proceed or not proceed to implementation with the hardware, software, or network improvement (RK00). This report should contain the following information:

1. A statement of purpose describing the intention behind certification testing and the approach used.

2. Certification program results describing the success achieved and an assessment of the state of readiness of the various hardware, software, and network components.

3. A summary of the certification program, which details the evaluation method. The evaluation methods may include interviewing, walkthroughs, kit review, feedback assessment and scoring results review.

4. Questions used to conduct certification testing and evaluation.

5. Any final recommendation to proceed or not proceed further based on possible impact on production process performance or capacity management limitations.
6. An appendix containing all detailed certification test program material compiled during the test process.

3.5.5. Major Issues in Data Warehousing

Data Sources The data sources provide the original data of the data warehouse. A data warehouse may integrate data from multiple autonomous and heterogeneous data sources, which could be either remote or local, and not under the control of the data warehouse users and administrators. In addition, the data sources may be structured (e.g., relational databases), semi structured (e.g., XML or RDF files) or flat files. Such arbitrary data sources pose several challenges to warehouse builder: to create a uniform repository integrating these data; to design easily understandable data warehouse schemas; and to express the transformations between the data source and data warehouse schemas.

Staging Area The staging area keeps whole copies of the data sources and brings them under the control of the data warehouse administrator. The data in the staging area may be heterogeneous and contain “dirty” (e.g. duplicate or inconsistent) data. No end user query services are available in this area, that is, the warehouse users cannot access the data in the staging area.

Data Warehouse The data warehouse contains the integrated data used to support the DSS processes. In contrast to the staging area, data in the data warehouse itself have a uniform schema and have been cleansed by removing dirty data. The processes of data cleansing and data transformation happen before loading data into the data warehouse.
The data warehouse typically consists of following components:

**Detailed Data:** The detailed data is the lowest level of source information necessary for supporting the DSS processes. It is normally stored in a single repository such as a relational or object-oriented database. The detailed data includes current detailed data and older detailed data. From the staging area to the detail data, the data needs to be transformed, cleansed, loaded and integrated. These processes compose a major part of building a data warehouse.

**Summarized Data:** The summarized data is derived from the detailed data, in order to allow faster processing of specific DSS functionality. For example, suppose the detailed data contains a relational table Sales(ProductID, LocationID, TimeID, SalesAmount). The summarized data may contain tables ProductSalesByLocation (ProductID, LocationID, SalesAmount) summarizing the total sales for products at locations; ProductSalesByTime(ProductID, TimeID, SalesAmount) summarising the total sales for products over time periods; LocationSalesByTime(LocationID, TimeID, SalesAmount) summarising the total sales for locations over time periods; TotalProductSales(ProductID, SalesAmount) summarising the total sales for products; TotalLocationSales(LocationID, Sales Amount) summarising the total sales for locations; Total TimeSales(ProductID, SalesAmount) summarising the total sales over time periods.

The summarized data are defined as views over the detailed data or over other summarizing views. Views in the data warehouse can be virtual or materialized. How to maintain these views, especially materialized ones, has been one of the key issues of data warehousing research.

**Metadata:** A data warehouse not only provides integrated data, but also provides information about the content and context of the data, i.e. metadata.
This metadata provides a directory of the structure of the warehouse contents. It provides information about the warehouse schema, and also about the mappings between the data in the data warehouse, such as from the data sources to the detailed data and from the detailed data to the summarized data.

3.5.6 End User Applications and Interfaces

The end user applications and interfaces provide a way for warehouse users to access warehouse data. In particular, data marts can be created over the data warehouse for different categories of DSS users. Data marts are defined from the warehouse data for specific DSS requirements of the enterprise. In contrast to the summarized data, data marts can have different data models and schemas from the ones of the detailed data of the warehouse. In practice, the same tools used to load the data warehouse database can be used to load the data marts.

The problem of query rewriting, also known as answering queries using views, has also received much attention in database research. Query rewriting aims to find efficient methods of answering a query using a set of previously materialized views over the database tables, rather than accessing the base tables themselves.