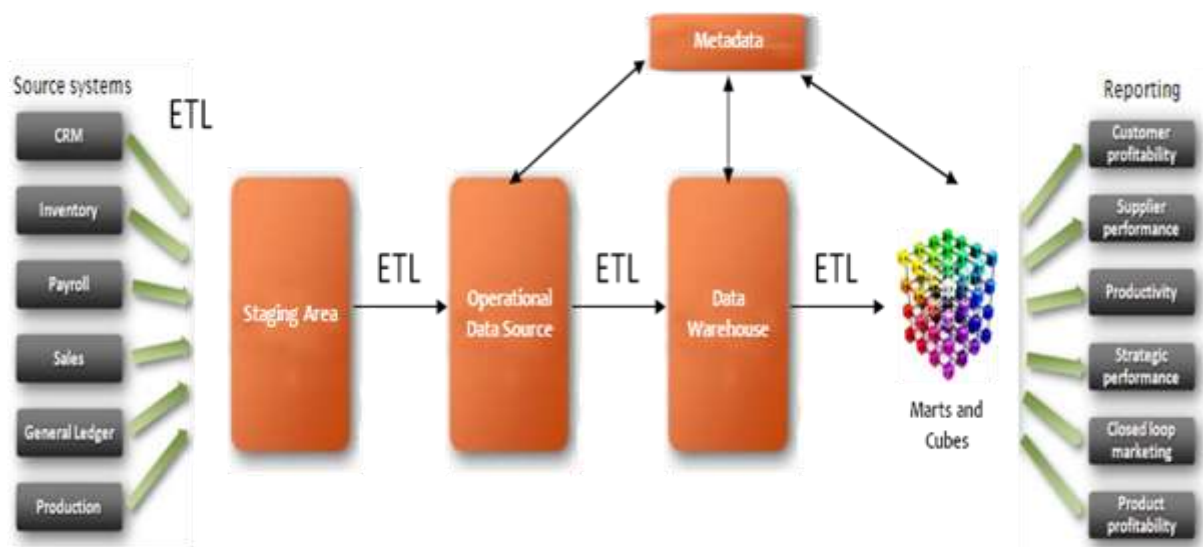


# Testing Data Warehouse Applications

## 1 Introduction

Today, people have realized the importance of data in a competitive market. For a business person, decision making is based on processed data from using reporting, analytical & forecasting tools. But in the background, these tools depend on a very complex & powerful entity known as a data warehouse, which is a conglomeration of meaningful data. A data warehouse enables

- quick and consistent access to data across the entire enterprise,
- informed decision making by consolidating data from multiple business units



The above picture depicts typical data warehouse architecture. The idea of this paper is to highlight, on a very high level, the testing approach and best practices in data warehouse testing.

## 2 Testing Approach

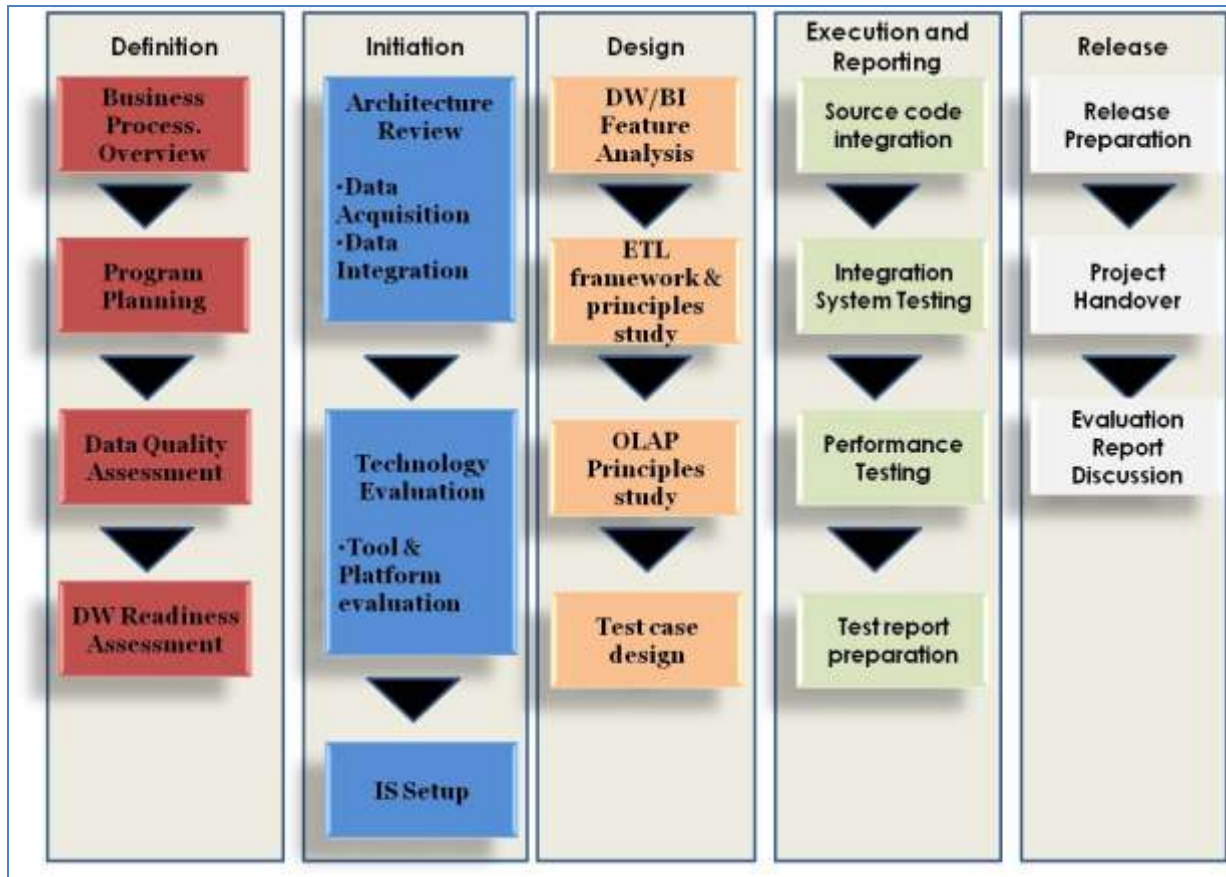
The challenges faced by businesses in implementing data warehouse are

- Understanding and building accurate data for reporting or analyzing tools
- Extracting the right data from the source systems
- Handing over data from source to transformational systems without affecting the operation of source systems
- Classification of data according to business rules
- Building classified and right data marts such as Financial, General Ledger for reporting
- Accessing real time data by building ODS(Operational Data store)
- Creating exact and meaningful data definitions in the tables
- Setting up databases to handle huge volume of data with multiple environments such as development, Testing and Production

## Testing Data Warehouse Applications

Most of the above mentioned challenges can be overcome by ensuring data integrity, appropriate classification and security. Thus data warehouse testing is business critical and contributes significantly towards reducing the total cost of ownership.

Below diagram explains various phases in the development of a data warehouse with DW Testing playing an integral part in the whole process.



The first step is to understand the ETL (Extract Transform and Load) process.

During the **Extract** process, one has to consider the number of source systems involved. The most common data source systems are relational databases and/or flat files. At this stage, source system data has to be verified and validated for the correctness of format it has to be extracted into. These extractions will be done in the staging tables. One has to focus on the recycling time of the staging tables. The frequency of extractions may be daily, weekly or even monthly. Load dates play a key role in the testing of the extraction process.

The **Transform** stage applies a series of rules or functions to the extracted data to derive it in the required form satisfying the business rules. Some data sources will require very little or even no manipulation of data. In other cases, one or more of the following transformations types may be required to meet the business needs of the end target.

- Selecting only certain columns to load (or selecting null columns not to load)
- Filtering conditions based on business rules
- Sorting data to validate against the source and target data comparison
- Consolidating data from multiple sources (e.g., lookup, merge, etc.)

## Testing Data Warehouse Applications

- Aggregation
- Splitting a column into multiple columns (e.g., putting a comma-separated list specified as a string in one column as individual values in different columns)
- Applying any form of simple or complex data validation.

The more complex the transformation type, more extensive would be the effort for defining the testing strategy.

The **Load** phase moves the data into the end target, usually being the data warehouse (DW). Depending on the requirements of the organization, this process ranges widely. It can be either a complete refresh of data (data will be wiped out and re-written) or just adding/modifying new or changed data. Test engineers can focus on testing the transformed data; it may be in the form reports, data reconciliation etc. Sometimes both sources and target data will be files. Spread sheet validations will help the test engineer to cross check the business rules without going through the complex data transformation business rules.

After successful loading, business reports will be generated from the data mart or the data will be pushed to sub systems. Data reflections on the subsystems or reports and format of the reports have to be tested. There are multiple tools used to generate business reports. Familiarity of the tool helps in the testing process.

Irrespective of ETL process, tools like Beyond Compare, WinMerge and MS Excel can help to compare data accuracy in between data movements. The challenge is in building the test cases where complex SQLs are involved. Once SQLs and test data are defined, creating automated scripts will reduce considerable time in regression testing.

The capability of a data warehouse is determined by the amount of data that can be processed in a given lead time. This is evaluated through performance testing where non-business critical data is used.

Below picture illustrates various types of testing in a Data Warehouse development lifecycle,



### 3 Best Practices

Following are some of the best practices in DW testing apart from common STLC processes

- Validate the signed off Data Model Diagrams ( Prerequisite before Testing )
- Build test cases based on the Design document. Design document forms the base for writing test cases. A more detailed design document will lead to detailed, complex text cases
- Drill down test cases to transformation rules in detail which may cover
  - small or broken SQL queries to validate any scenario.
  - appropriate data to validate ( sample or whole bunch of data )
  - rights/access to run jobs for testing in a particular environment dependant jobs to be run as a prerequisite.
- Write test cases which covers structure of database validation. This can be done with the help of Database Administrator.
- DW Report formats have different file types such as PDF or propriety tool file format (like Deski) or just an adhoc report in HTML. The test cases and the requirement documents should take into account the various reporting formats that would be available
- Cover end to end scenarios, for example, aggregated sum at source system level should match with output at the end reports .
- Include positive and negative scenarios, for example, exception handling, reloading process for a failed job etc.
- Review the test case with Development team
- Perform regression after bug fixing. If required automate the SQL scripts for regression testing.
- On a successful test execution, prepare a test summary report for User Acceptance Testing (UAT)
- On successful completion of UAT, SQA can sign off on "Go/ No Go" decision for the production move. This is part of Implementation Readiness review checklist.

These guidelines should be tailored based on the project context.

### 4 Conclusion

For effectiveness in data warehouse testing, the QA team should have the following skills

- Good SQL knowledge to be able to conduct mass data reconciliation testing.
- Experience in writing/understanding stored procedures
- Experience in Data warehousing with ETL testing.
- Experience in data manipulation tools
- Knowledge of automation framework and implementation will be an added advantage

## 5 Abbreviations/Acronyms

#	Acronym	Description
1	DW	Data warehouse
2	ETL	Extract Transform Load
3	QA	Quality Assurance
4	STLC	Software Testing Life Cycle

## 6 Definition

#	Name	Definition
1	Datawarehouse	a subject oriented, nonvolatile, integrated, time variant collection of data in support of management's decisions
2	Operational data store	a subject-oriented, integrated, volatile, current-valued, detailed-only collection of data in support of an organization's need for up-to-the-second, operational, integrated, collective information
3	Data mart	analytical data stores designed to focus on specific business functions for a specific community within an organization

## 7 Author Profile

Ramadurai Radhakrishnan, with more than 9 years of experience in the Software Industry has worked in various domains including Telecom Billing, Banking, Financials, and Tours and Travels. Expertise in Oracle (PL/SQL), Pro\*C/C++, UNIX (includes Scripting) and Manual Testing. Brain Bench certified in Java. His involvement ranges across Architecture, Requirement Analysis, Design, Development, Implementation, Testing, Documentation and Maintenance of Software Applications and Products. He has initiated and Led a Knowledge communities programs and Data warehouse Testing Competency. He implemented a successful Data Warehouse Testing projects for Banks in US. Also a very active consultant and trainer for Database and Data warehouse Testing