powersoft nineteen SQA
consider it done!

Big Data-Challenges and Opportunities

White paper

Table of Contents

Introduction

Every day 2.5 quintillion bytes of data are created (stated by ViaWest). This data comes from different sources i.e. digital pictures, videos, posts to social media sites, intelligent sensors, purchase/sale transaction records, and cell phone GPS signals etc. This is big data. There is no doubt that big data and especially what we can do with it, has the potential to become a very significant driving force for innovation and value creation. So, big data has become the next frontier for innovation, competition, and productivity.

Scope of this paper

This paper has been written to provide an insight about the following items:

- Big data and its challenges and opportunities
- Data Warehouse/ETL Testing and its general goals

What is big data?

What is big data? Here is a selected definition given by McKinsey Global Institute (MGI): 'big data refers to datasets whose size is beyond the ability of typical database software tools to capture, store, manage and analyze'.

Where do we find big data? Data in general, and increasingly big data, is an important production factor for all industries and business processes. MGI estimated that 7 Exabyte of new data were stored globally by enterprises in 2010. Interestingly, more than 50% of IP traffic is non-human, and Machine-to-Machine communication (M2M) will become increasingly important.

Big Data Facts

Facebook handles 300 million images/photos a day and about 105 terabytes of data every 30 minutes

Walmart handles 1m transaction per hour imported into databases containing 2.5 Petabytes of data

Google processes 25 Petabytes of data per day (=~ 25600 Terabytes)

AT&T transfers 30 Petabytes per day

eBay stores 6.5 Petabytes of data and processes 100 terabytes per month

Twitter processes 85 million tweets per day

The big data market will grow from $3.2 billion in 2010 to $32.4 billion in 2017 (as per Research Firm IDC).

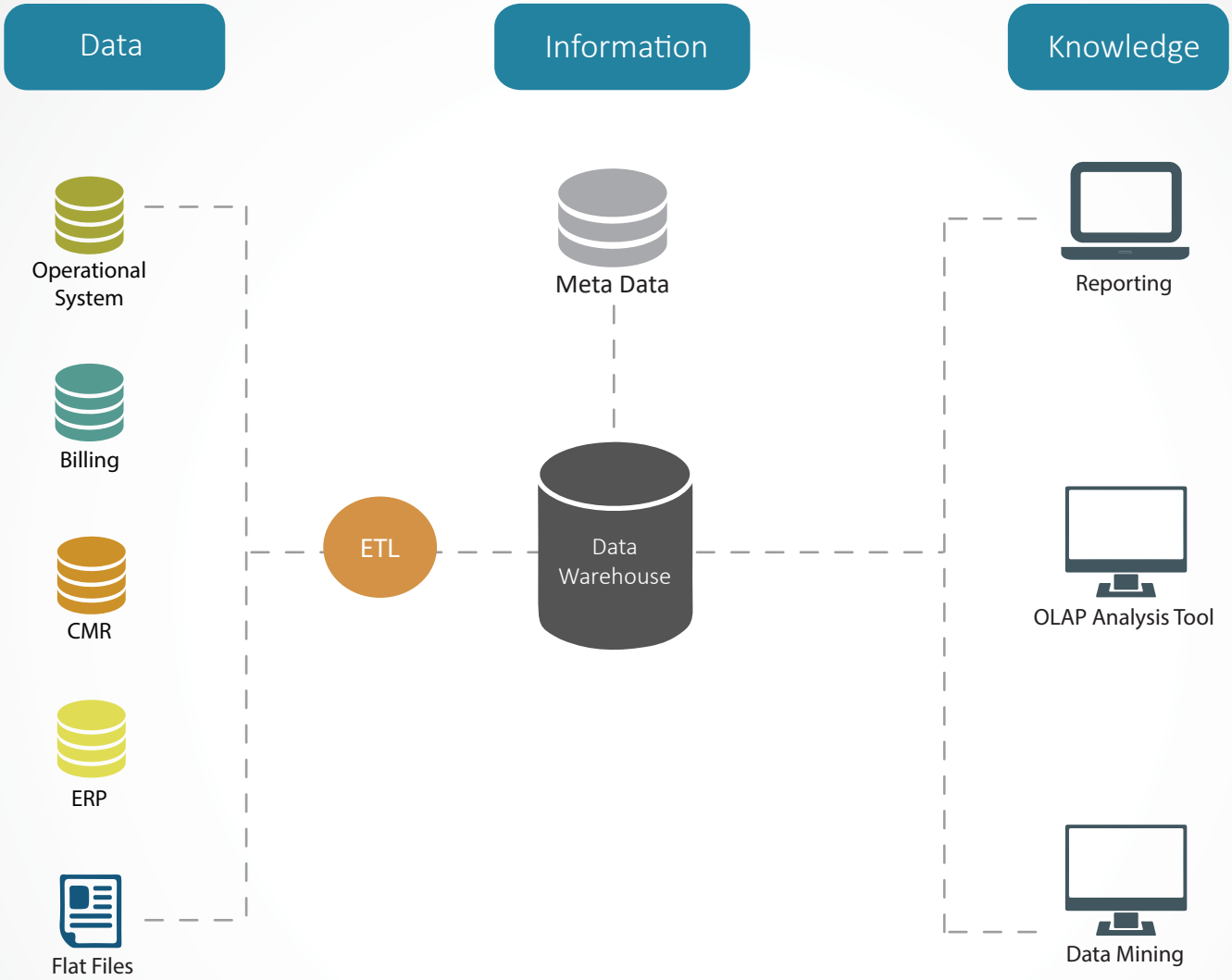
Big Data (Challenges and Opportunities)

The following table shows various opportunities and challenges associated with Big Data.

Opportunities	Challenges
Creating transparency	Dealing with 4 Vs (Volume, Velocity, Variety, Veracity) of big data
Discovering needs, exposing variability, improving business performance & operations	Data discovery
Segmenting customers (tailor products or services)	Data quality, relevance and comprehensive-ness
Replacing/supporting human decision making with automated algorithms –innovat-ing new business models, products, services etc.	Scalability
Improving decision quality on the basis of data patterns	Technology (innovative technology, tools and resources)
Mitigating risks associated with complex decisions by using data insight	Data privacy and security

Data Warehouse

A Data Warehouse is a compound and collaborative data model that captures the entire data of an organization. It brings together data from various sources into a single destination. It does not only collect data, but also stores it using ETL process (i.e. the data is Extracted, Transformed and Loaded). The data is then stored for querying and analytical purpose. Data Warehouse also brings ease in day to day OLAP operations.



Why Data Warehouse?

Below mentioned figure with description clarifies the need of Data Warehouse:

Decision Support System

The DSS (Decision Support System) monitors historical trends and provides suggestions after analyzing these trends. Any defect in this layer may result in poor decision making in organizations.

Compliance and Regularity Specifications

Safety, financial and health care institutions are required to comply with multiple standards for all data transactions including compliance in the historical data. Data transformation should be thoroughly tested as any kind of slippage or violation of compliance may result in financial loss and brand image distortion.

Data Centers Migrations

Data center migrations can be mandatory during organization mergers or acquisitions. It improves the infrastructure and failover mechanism.

Data Mart

Data Mart can be defined as specialized subset of a Data Warehouse i.e. Finance, Marketing, HR and Services, etc.

Improve Data Quality and Consistency

A Data Warehouse implementation includes the conversion of data from various sources into a common format. Since each data from the various departments is standardized, each department will produce results that are in line with all the other departments. So you can have more confidence in the accuracy of your data. And accurate data is the basis for strong business decisions.

Business Centric

Multiple industry giants are joining forces to provide integrated services in sectors like retail, business, social communications, banking, etc. Data transformation is the key to ensure seamless merger/acquisition in such business arena.

Data Warehouse Testing

This is clear that the organizations today need Data Warehouse testing more than ever before. Also, decisions are greatly dependent on the data contained in the Data Warehouse; that's why the data should be made reliable through diligent testing.

At high level, the scope of testing strategies and types must cover the following aspects and goals:

Creating Transparency	Ensures that all expected data is covered and loaded from source to target.
Data Transformation	Ensures that all data is transformed correctly according to business rules and/or design specifications.
Data Quality	Makes sure that the ETL software accurately rejects, substitutes default values, fixes or disregards, and reports incorrect data.
Performance and Scalability	Makes sure that data loads and queries are executed within anticipated time frames and that the technical design is scalable.
Audit, Logging and Error Handling Testing	Ensures that every ETL system logs all the processes run, their time, and duration. It also ensures the logging of exceptions or errors that are thrown by different processes.
Integration Testing	Ensures that the ETL process functions well with other upstream and downstream processes.
Deployment Testing	Ensures that the deployment process and documentation are complete and comprehensive enough when ETL processes are run out of data centers by production support operators.
User Acceptance Testing	Makes sure that the solution satisfies your current expectations and anticipates your future expectationst.
Regression Testing	Makes sure that current functionality stays intact whenever new code is released.

Data Completeness/Coverage

The primary test for data completeness is to ensure all the data is covered or loaded into the target DWH. Special case data (records) exist in every application. The coverage of all such special cases and boundary cases must be ensured. This includes, but is not limited to, validating that all records, all fields and the full contents of each field are loaded. Special checks for truncation, null records etc. should be performed.

Strategies to consider include:

- Comparing the record counts between source data, data loaded into the ODS and data loaded into the warehouse. You may also consider the rejected/suspense records.
- Comparing the count of distinct values of key fields between source data and data loaded to the warehouse. You could also do a comparison of the distinct values themselves.
- Utilizing a data profiling tool (Talend a free tool) that shows the type, range and distributions of data in a data set. This can be used to compare source and target data sets. The profilers can also identify defects from source systems that may be missed even when the data movement is correct.
- Verifying that each field is tested to its maximum length e.g. for a Varchar(10) field, make sure to test it with10 characters, 11 characters, blank, and null.
- Testing the boundaries of each field to find any database limitations i.e. For a decimal field with a precision of 3 include values of -99 and 999, and for date fields include the entire range of dates expected (including dates in multiple formats).

Data Transformation

Business requirements get translated into transformation logic. Once the data is transformed, thorough testing has to be executed to confirm that the underlying data complies with the expected transformation logic. Validating that data is transformed correctly based on business rules/logic can often be the most complex part of testing an ETL application. Multiple techniques can be adopted for this purpose. Sampling is one technique, in which sample records are selected and compared to validate data transformations. A combination of automated data profiling and data feed is another better long-term strategy. This ensures more test coverage.

A combination of automated data profiling and data feed is another strategy suitable in the long-term to ensure better test coverage.

Here are some simple data movement techniques:

- Create a set of scenarios of input data and expected outcomes and validate these with the customer. This is a good exercise to be performed during requirements and can also be used during testing. Also automate the procedure of populating data sets.
- Use the results of the data profiling tool to compare range and distribution of values in each field between source and target data.
- Validate processing of technical fields generated during ETL design such as flags and surrogate keys.
- Validate that data types in the warehouse as per data model.
- Validate the referential integrity between tables. This applies when you are testing behavior of child records that do not have a parent record and when checking many to many relationships.

Data Quality

Data quality as how the ETL process deals with data rejection, replacement, correction, and notification without changing any of the data.

Typically, data quality rules are defined during design e.g.

- If a state is implied wrongly, look up from the master list for correction.
- Substitute null if a certain decimal field has nonnumeric data.
- Validate the city and state based on ZIP code.
- Compare the product code to values in a lookup table. If there is no match, load anyway; however, report this to our clients etc.

Data quality rules applied to data will usually be invisible to the users once the application is in production; the users will only see what's loaded to the database. For this reason, it is important to ensure that the users are aware of how the bad data is being handled. The data quality reports often present patterns and trends that hint at underlying structural issues.

Performance and Scalability

Performance and Scalability testing ensures that loading of the initial data and subsequent queries on the same does not kill the system and are within acceptable performance limits. This also ensures that the system is scalable and can sustain further growth. As the volume of data in a Data Warehouse grows, ETL load times can be expected to increase and performance of queries can become a concern. This can be avoided by having in place a scalable architecture and good ETL design. The aim of the performance testing is to point out any potential weaknesses in the ETL design, such as reading a file multiple times or creating unnecessary intermediate files. The following strategies will help discover performance issues:

- Load the database with peak expected production volumes (this is often much higher than current production data) to ensure that this volume of data can be loaded by the ETL process within the agreed-upon window.
- Compare these ETL loading times to loads performed with a smaller amount of data to anticipate scalability concerns.
- The ETL processing time should also be tested segment by segment to understand potential bottlenecks.
- If parallelism is present in the ETL design, test it fully to ensure that there are no memory bottlenecks.
- Monitor the timing of the reject/suspended process and consider how large volumes of rejected data will be handled – often this leads to source tables being scanned fully twice and that deteriorates performance.
- Perform simple and complex outer join queries to validate query performance on large database volumes. Identify acceptable performance criteria for each query and benchmark against this.

Audit, Logging and Error handling Testing

This applies to validation of the technical processes of a Data Warehouse. Every ETL system must log what processes have run, when and for how long. Consider the following strategies:

- Consciously aborted processes are re-started from the point of failure.
- Sometime re-starting from the point of failure may not be possible; in that case we need to validate that the system can roll back to the original state and the ETL can be reprocessed and logged this state.
- Check the depth of error logging – does it log the error alone or does it also capture additional details including when and in which data row it occurs.

Integration Testing

A Data Warehouse implementation is a collection of many components; hence integration testing is essential. Integration testing shows how the application fits into the overall architecture. Individual components behaving correctly is no guarantee for the entire system to behave as expected. Integration brings with it many new issues like 'resource conflict' or 'deadlocks' etc.

Deployment Testing

Ultimately DWH/ETL processes are deployed on data centers by production support. The deployment process and documentation should be complete and comprehensive enough to run and verify all processes end to end.

Keep the following points in mind:

- Ensure that deployment testing is performed by a person who is new to the project and does not understand the nuances of the system you have designed.
- When things go fine, the data center personnel do not have much to do except monitoring; it is when things go wrong that they swing into action. Testing should cover instances where things go wrong (an ETL process hangs, a data file does not get loaded) and how the system effectively captures and presents these negative scenarios.

User Acceptance Testing

The ultimate goal of a DWH application is to make data available to end business users. Users have the best knowledge of their requirements, so their participation in the testing effort is a key component in the success of a Data Warehouse implementation.User-acceptance testing (UAT) in DWH typically focuses on data loaded to the Data Warehouse and views that have been created from data analysis. Consider the following strategies:

- Use data that is either from production or as close to production data as possible. Users typically find issues once they see the "real" data, sometimes leading to design changes.
- Plan for the system test team to support users during UAT. The users will likely have questions about the process of data population and ETL.
- Consider how the users would require the data loaded during UAT and negotiate how often the data will be refreshed.

Regression Testing

Regression testing is revalidation of an existing functionality with each new release. During test case building and data designing, there will likely be multiple executions with every new version release due to defect fixes, enhancement, or system changes. Automating the regression testing will make the testing process much more efficient. Test cases should be prioritized by risk in order to help determine which need to be rerun for each new release. An efficient strategy to retest a basic functionality is to store source data sets and results from successful runs of the code and compare the new test results with previous runs. When doing a regression test, it is much quicker to compare results to a previous execution than to do an entire data validation again.

Appendix

DWH – Data Warehouse
ETL – Extract Transform and Load
OLAP – Online Analytical Processing
DSS- Decision Support System

powersoft nineteen SQA
consider it done!

Contact Us

Explore ways to use our expertise in growing your business while establishing a valuable partnership with us.

Contact our consultants at:

E-mail: sqa@powersoft19.com
Website: www.powersoft19.com/sqa