# Assignment Of

#  “Business intelligence”

# Topic:- difference b\w dashboard vs scoreboard

#

#  *Submitted To:- Submitted By:-*

# *Ms. Manisha Bhatnagar Gurjeet kaur*

# *(Asst.Prof.in Comp. Sci.) MCA 3rd (Sem-5th)*

#  *Roll No = 39*

# *S.S.D Women’s institute of technology (bathinda). (APPROVED BY AICTE , NEW DELHI. AFFILIATED TO PUNJABI UNIVERSITY, PATIALA)*

#

#  INDEX

|  |  |  |  |
| --- | --- | --- | --- |
| Sr.no | Particulars | Date | Remarks |
| 1. | Compare and Contrast b/w structured , Semi-structured and unstructured data | 10,aug,2017 |  |
| 2. | Comparison between OLAP amd OLTP and applications of OLAP and OLTP. | 24,Aug,2017 |  |
| 3. | Explain ETL Process. | 9,sep,2017 |  |
| 4 | Difference b\w dashboard and scoreboard | 23,nov,2017 |  |
|  |  |  |  |

# In computing, extract, transform, load (ETL) refers to a process in database usage and especially in data warehousing. The ETL process became a popular concept in the 1970s.Data extraction is where data is extracted from homogeneous or heterogeneous data sources; data transformation where the data is transformed for storing in the proper format or structure for the purposes of querying and analysis; data loading where the data is loaded into the final target database, more specifically, an operational data store, data mart, or data warehouse.

# ETL CONCEPT :- A Company data may be scattered in different locations and in different formats.

# Migrate the data into a data warehouse

# Convert the various formats and types to adhere to one consistent system.

# ETL REQUIREMENTS:-

# Any ETL Architecture must meet the following requirements:

# Business Requirement

# Compliance Requirement

# Data Profiling

# Data Security

# Data Integration

# Right Data at Right Time

# Archiving & Uneage

# Final End User Delivery Interface

# Available Skills

# Legacy License

# Alignment with overall Enterprise Architecture

# *ETL PROCESS:-*

# ETL is an abbreviation of the three words Extract, Transform and Load. It is an ETL process to extract data, mostly from different types of systems, transform it into a structure that’s more appropriate for reporting and analysis and finally load it into the database and or cube(s).

# *The three major steps in ETL:-*

|  |  |  |
| --- | --- | --- |
| 1. Extract data from source1. Extract from source | 2. Transform the data2. Transform the data | 3. Load the data3. Load the data |
| In this step we extract data from different internal and external sources, structured and/or unstructured. Plain queries are sent to the source systems, using native connections, message queuing, ODBC or OLE-DB middleware. The data will be put in a so-called Staging Area (SA), usually with the same structure as the source. In some cases we want only the data that is new or has been changed, the queries will only return the changes. Some ETL tools can do this automatically, providing a changed data capture (CDC) mechanism. | Once the data is available in the Staging Area, it is all on one platform and one database. So we can easily join and union tables, filter and sort the data using specific attributes, pivot to another structure and make business calculations. In this step of the ETL process, we can check on data quality and cleans the data if necessary. After having all the data prepared, we can choose to implement slowly changing dimensions. In that case we want to keep track in our analysis and reports when attributes changes over time. | Finally, data is loaded into a [data warehouse](https://www.passionned.com/data-integration/data-warehouse/), usually into fact and dimension tables. From there the data can be combined, aggregated and loaded into datamarts or cubes as is deemed necessary. The business user analysis and uses the transformed data with BI instruments like data visualization software, dashboards, OLAP tools and reporting tools. |

#  But, today, ETL is much more than that Most [ETL software](https://www.etltool.com/list-of-etl-tools/) also covers:

# data profiling and data quality control

# monitoring and cleansing of the data

# real-time and on-demand data integration

# extraction of Big Data using Hadoop

# master data management

# An ideal ETL architecture contains a data warehouse

# Below you’ll find the ideal ETL architecture supporting the three major steps in ETL.

# An overview of a data warehouse and ETL architecture displaying what is ETL.

# *Data profiling and data quality control:-*

# Profiling the data, wil give direct insight in the data quality of the source systems. It can display how many rows have missing or invalid values, or what the distribution is of the values in a specific column. Based on this knowledge, one can specify business rules in order to cleanse the data, or keep really bad data out of the data warehouse. Doing data profiling before designing your ETL process, you are better able to design a system that is robust and has a clear structure.

# *Meta data management & ETL:-*

# Information about all the data that is processed, from sources to targets by transformations, is often put into a metadata repository; a database containing all the metadata. The entire ETL process can be ‘managed’ with metadata management, for example one can query how a specific target attribute is built-up in the ETL process, called data lineage. Or, you want to know what the impact of a change will be, for example the size of the order identifier (id) is changed, and in which ETL steps this attribute plays a role.

#  *7 biggest benefits of using an ETL tool:-*

# We now generally recommend using an ETL tool, but a custom-built approach can still make sense, especially when it is model-driven. This publication summarizes the seven biggest benefits of ETL tools and offers guidance on making the right choice for your situation.

# *Visual flow:-*

# The single greatest advantage of an ETL tool is that it provides a visual flow of the system’s logic (if the tool is flow based). Each ETL tool presents these flows differently, but even the least-appealing of these ETL tools compare favorably to custom systems consisting of plain SQL, stored procedures and system scripts, and perhaps a handful of other technologies.

# *Structured system design:-*

# ETL tools are designed for the specific problem of data integration: [populating a data warehouse](https://www.passionned.com/data-integration/data-warehouse/) or integrating data from multiple sources, or even just moving the data. With maintainability and extensibility in mind, they provide in many cases a metadata-driven structure to the developers. This is particularly a big advantage for teams building their first data warehouse.

# *Operational resilience:-*

# Many of the home-grown data warehouses we have evaluated are rather fragile: they have many emergent operational problems. ETL tools provide functionality and standards for operating and monitoring the system in production. It is certainly possible to design and build a well instrumented hand-coded ETL application. Nonetheless, it’s easier for a data warehouse / business intelligence team to build on the features of an ETL tool to build a resilient ETL system.

# *Data-lineage and impact analysis:-*

# We would like to be able to right-click on a number in a report and see exactly how it was calculated, where the data was stored in the data warehouse, how it was transformed, when the data was most recently refreshed, and from what source system(s) the numbers were extracted. Impact analysis is the flip side of lineage: we’d like to look at a table or column in the source system and know which ETL procedures, tables, cubes, and user reports might be affected if a structural change is needed. In the absence of ETL standards that hand-coded systems could conform to, we must rely on [ETL vendors](https://www.etltool.com/etl-vendors/) to supply this functionality — though, unfortunately, just half of them have done so far (more results in [our survey](https://www.etltool.com/kb/etl-tools-data-integration-survey/)).

# *Advanced data profiling and cleansing:-*

# Most data warehouses are structurally complex, with many data sources and targets. At the same time, requirements for transformation are often fairly simple, consisting primarily of lookups and substitutions. If you have a complex transformation requirement, for example if you need to de-duplicate your customer list, you should buy on additional module on top of the ETL solution (data profiling / data cleansing). At the very least, ETL tools provide a richer set of cleansing functions than are available in SQL. Download the [ETL Tools& Data Integration Survey](https://www.etltool.com/kb/etl-tools-data-integration-survey/) to see how the ETL tools compare on this aspects.

# *Performance:-*

# You might be surprised that performance is listed as one of the last under the advantages of the ETL tools. It’s possible to build a high-performance data warehouse whether you use an ETL tool or not. It’s also possible to build an absolute dog of an data warehouse whether you use an ETL tool or not. We’ve never been able to test whether an excellent hand-coded data warehouse outperforms an excellent tool-based data warehouse; we believe the answer is that it’s situational. But the structure imposed by an ETL platform makes it easier for an (novice) ETL developer to build a high-quality system. Furthermore many ETL tools provide performance enhancing technologies, such as Massively Parallel Processing, Symmetric Multi-Processing and Cluster Awareness.

# *Big Data:-*

# A lot of ETL tools are now capable of combining structured data with unstructured data in one mapping. In addition they can handle very large amounts of data, that do not necessarily have to be stored in data warehouses. Now Hadoop-connectors or similar interfaces to big data sources are provided by almost 40% of the ETL tools nowadays. And the support for Big Data is growing continually.