ETL Concepts

Prepared By : Praneeth Komalla

TOPICS TO BE COVERED

- What is ETL
- Evolution of ETL Tools
- Advantages of ETL Tools
- Disadvantages of ETL Tools
- Phases of ETL Process
- Design Phase
- Development Phase
- Deployment Phase

What is ETL

❑ ETL is the automated and auditable data acquisition process from source system that involves one or more sub processes of

- Data extraction
- Data integration
- Data Cleansing
- Data transformation
- Data Loading

❑In simple words, ETL stands for Extract, Transform, and Load. That is, ETL programs periodically extract data from source systems, transform the data into a consistent format, and then load the data into the target data store.

What is not ETL

- ETL should not be confused with a data creation process. **It never creates new data**. If a list of hundred employees is being loaded, one more employee cannot be added to the list and make it hundred and one. Or if last name of customer is absent an we shouldn't substituting it with any value in last name.

- ETL **cannot change the meaning of data**. For example if Country is mentioned as 'India' and 'America' in source system. It can be loaded as 'IND' and 'USA' into the Data Warehouse respectively. This is OK because this does not change the business meaning of the data. It only has changed the representation of the data.

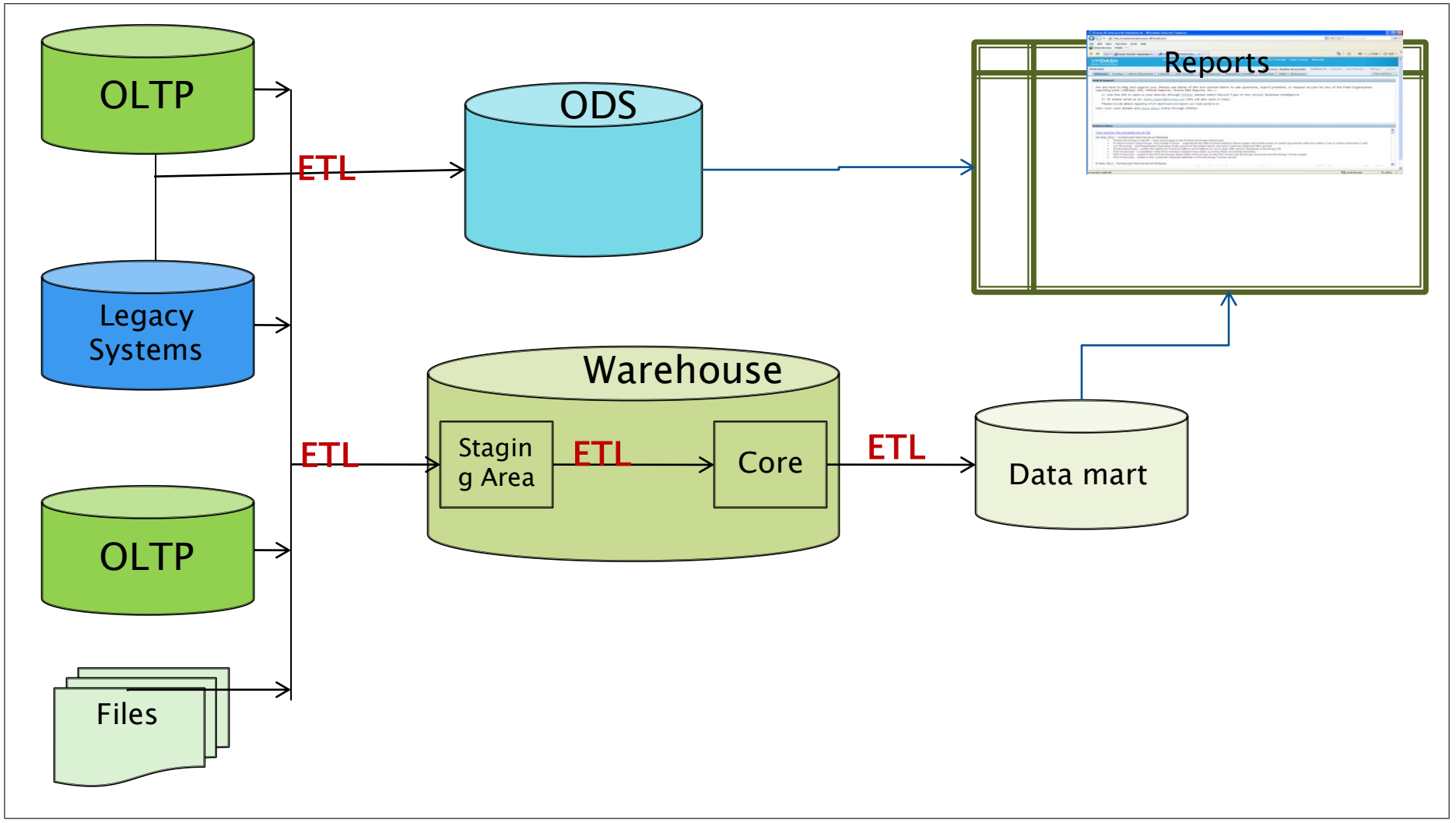
SOURCE SYSTEMS

❑Operational Systems (OLTP): An operational system is a term used in data warehousing to refer to a system that is used to process the day-to-day transactions of an organization. These systems are designed so processing of day-to-day transactions is performed efficiently and the integrity of the transactional data is preserved.

❑Legacy Systems : Legacy systems utilize outmoded programming languages, software and/or hardware that typically are no longer supported by the respective vendors.

❑Flat files/Excel files : These are manually prepared or auto generated files with data.

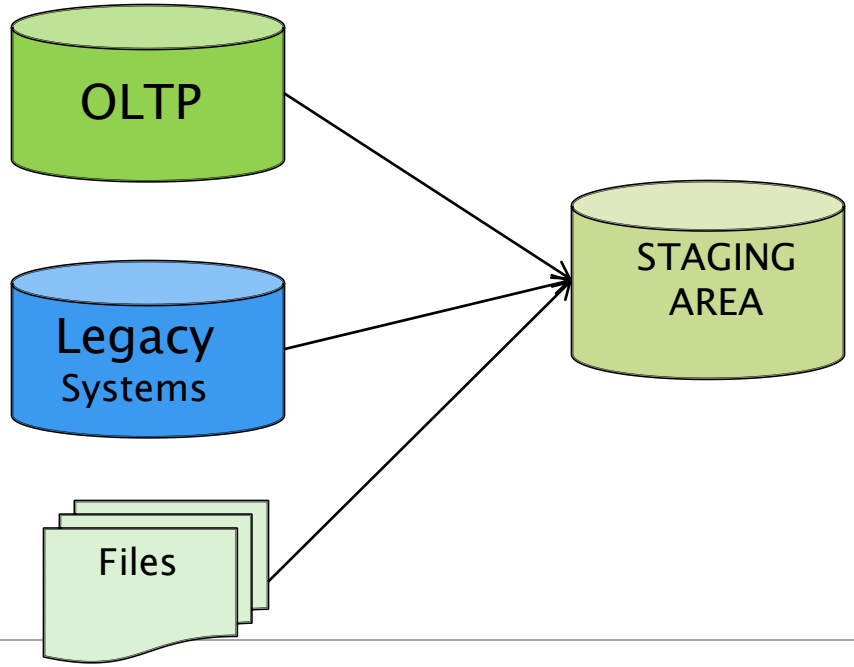
HIGH LEVEL DATA FLOW



StratApps
Imagination | Visualization | Innovation

Data Extraction

❑ Extraction is the operation of extracting data from a source system for further use in a data warehouse environment. This is the first step of the ETL process. After the extraction, this data can be transformed and loaded into the data warehouse.



Data Extraction

Designing and creating the extraction process is often one of the most time-consuming tasks in the ETL process.

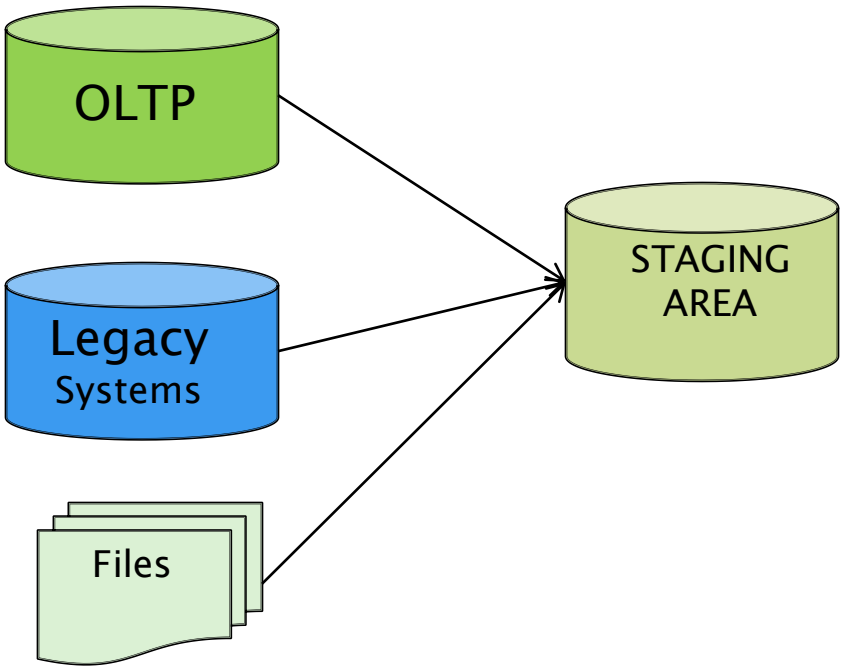
- The source systems might be very complex and poorly documented, and thus determining which data needs to be extracted can be difficult.

- The data has to be extracted normally not only once, but several times in a periodic manner to supply all changed data to the warehouse and keep it up-to-date.

- Moreover, the source system typically cannot be modified, nor can its performance or availability be adjusted, to accommodate the needs of the data warehouse extraction process.

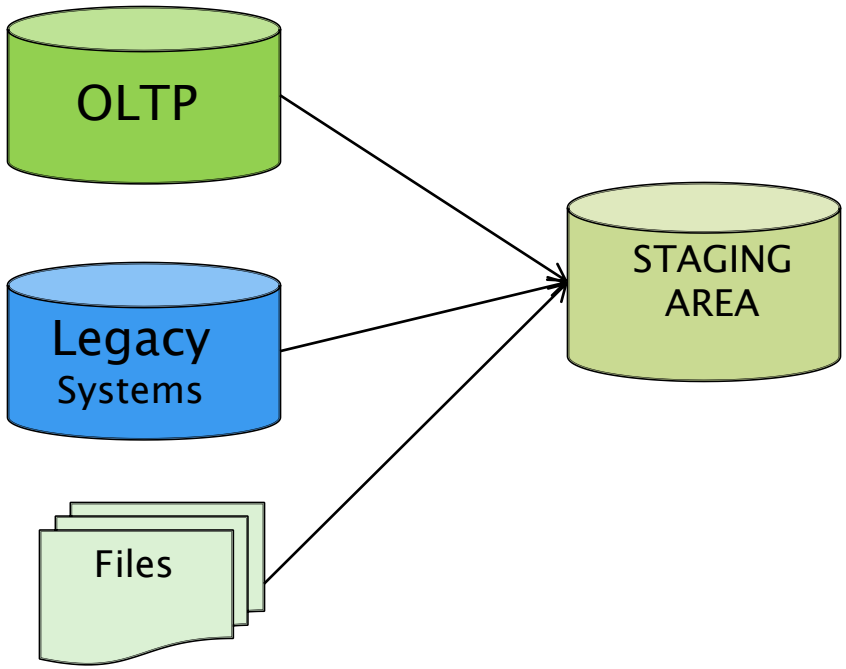
Data Integration

❑ Data integration system combines the data residing at different sources, and provides a unified, reconciled view of these data. It is often complex and time consuming as we are dealing with source systems.



Data Cleansing

❑ This process is to clean data coming from source systems as they are tend to be dirty. It is mostly done while loading staging area



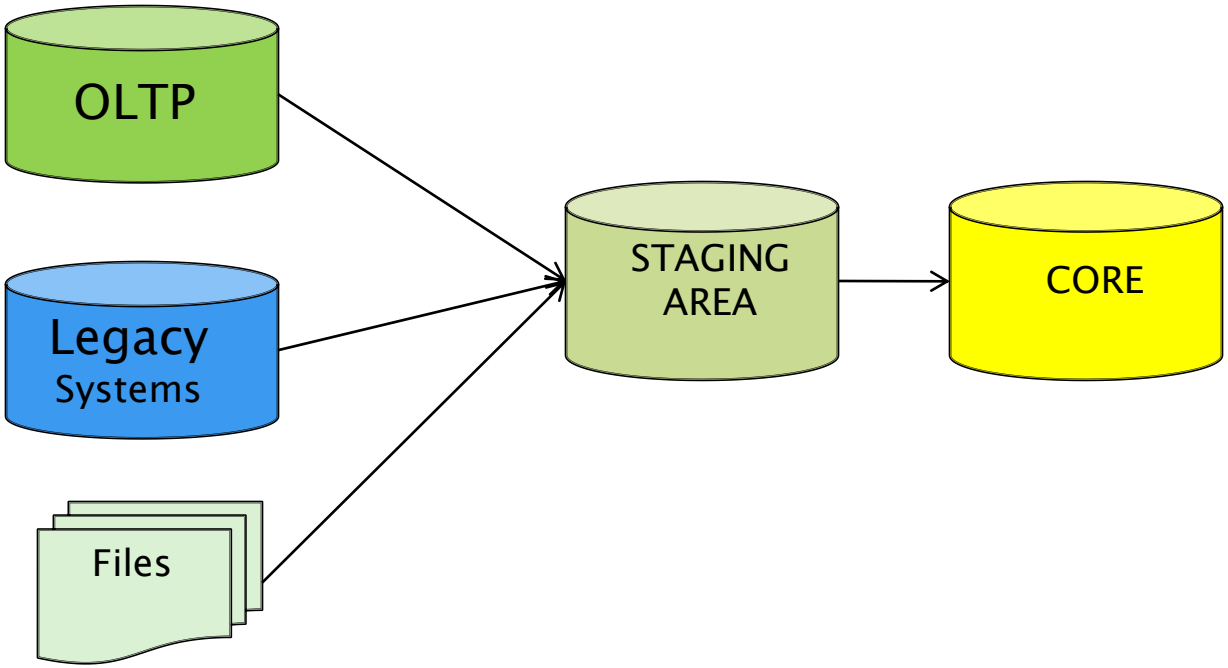
Data Cleansing

Reasons for Bad Data:

- ❑Dummy Values
- ❑Absence of Data
- ❑Multipurpose Fields
- ❑Cryptic Data
- ❑Contradicting Data
- ❑Inappropriate Use of Address Lines
- ❑Violation of Business Rules
- ❑Reused Primary Keys
- ❑Numeric values that fall outside of expected high and lows
- ❑Cols whose lengths are exceptionally short/long
- ❑Cols with certain values outside of discrete valid value sets

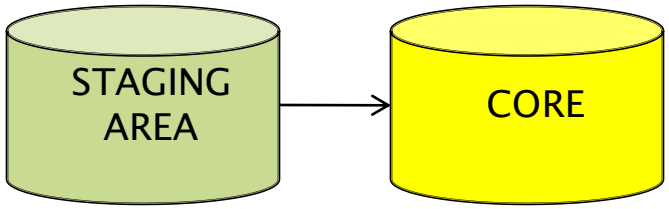
Data Transformation

❑ Data transformation is the process of converting information or data into the requirement of Warehouse tables. Now the would be consistent and correct



Data Loading

❑ After the data has been cleansed and transformed into a structure consistent with the data warehouse requirements, data is ready for loading into the data warehouse. This is considered to be a straightforward process. The key consideration in the Loading process is to achieve the speed of Loading.



Data Loading

Two types of load

❑ Initial load/Full Load – When Project go live to Production the first load is defined as Initial load. If we continue to load same as Initial load every time then the Load type is mentioned as Full Load.

- ETL for all data up till now
- Done when DW is started the first time
- Often problematic to get correct historical data
- Very heavy - large data volumes

❑ Incremental Load – After Initial load is complete, if we just extract latest changes and load into Warehouse then it is mentioned as Incremental load.

- Move only changes since last load
- Done periodically (../month/week/day/hour/...) after DW start
- Less heavy - smaller data volumes

StratApps
Imagination | Visualization | Innovation

Evolution of ETL Tools

Era	Title	Significance
Early 1990	Hand-coded ETL	Hand written custom codes
1993-1997	First generation ETL Tools	Code-based ETL tools
1999-2001	Second generation ETL tools	Engine-based ETL tools
2003-2006	ETL tools Today	Most efficient tools

Advantages of ETL Tools

❑Mappings, extract rules, cleansing rules, transformation rules, aggregation logic and loading rules are generally handled as separate objects in an ETL tool. This means that you can change one object in an ETL "string" without affecting the other objects. For example, you can change the loading logic for a particular target table (say, from direct insert to generating a flat loader file) without affecting the cleansing and/or transformation logic for that table. This compartmentalization eases maintenance, and reduces the need for retesting.

❑Objects in an ETL tool (e.g., transformation rules) can be reused.

❑ETL tools facilitate impact analysis when modifying or enhancing a data warehouse.

❑The most important characteristic of today's ETL tools is the type of parallelism they support.

Advantages of ETL Tools

- They have built in objects to handle recurring tasks such as aggregation, normalization, slowly changing dimensions so these do not need to be coded and recoded.

- The tools available today are providing the Debugger support with which developer can test the code for defect tracing

- The meta data trapped by an ETL tool graphically documents source and target database structures, mappings (a.k.a. "data genealogy"), cleansing rules (a.k.a. "business rules") and transformation rules. Such documentation is invaluable during impact analysis, or when bringing new team members up-to-speed on an on-going DW project.

- Ease of use. Because most ETL tools are GUI based and have repositories, they have increased ease of use and ease of modification.

Disadvantages of ETL Tools

- **Cost.** They are costly and have large technology and space requirements.

- **Complexity.** ETL tools may sometimes have difficulty with very complex transformation logic, as well as with complex staging requirements.

- **Performance.** Because they are generic, and many of them are interpretive, there can sometimes be performance issues over SQL, for transformations.

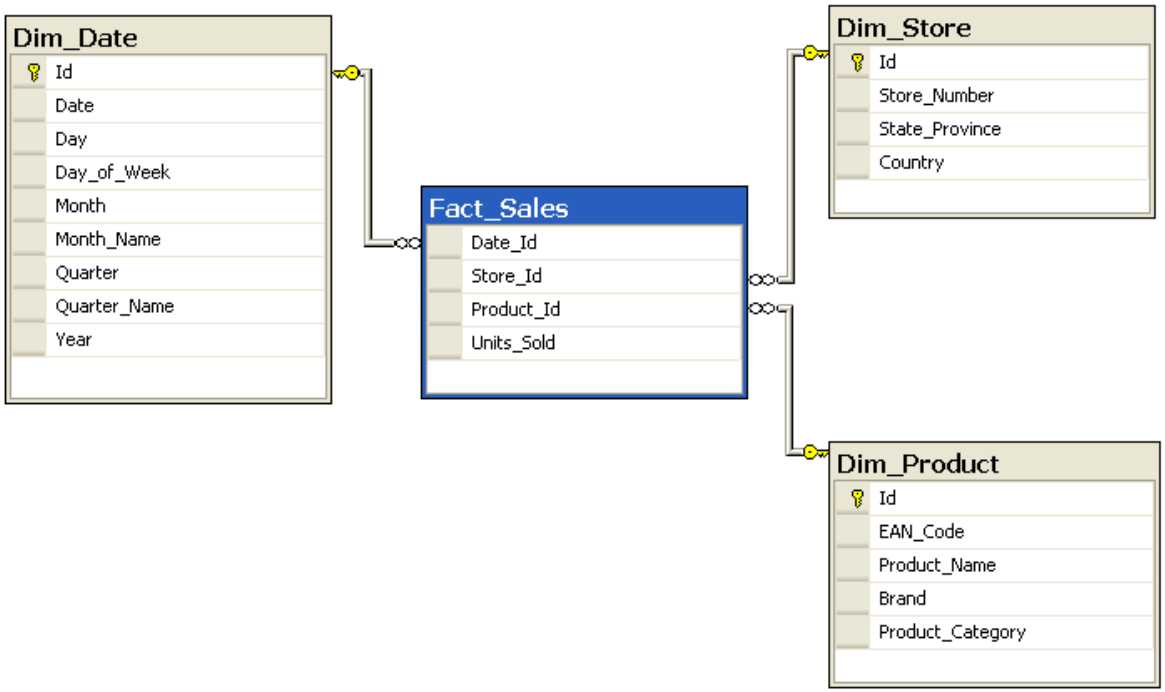
Phases of ETL Process

- Design Phase
- Development Phase
- Deployment Phase

Design Phase

- Understanding Source system data and Data Model.
- Identify Source tables/files which are required.
- Identify dependencies between tables in Data Model
- Prepare Logical Data Mapping
- Prepare Technical Design Document
- Identifying Dependencies of ETL code

Sample Data Model



StratApps
Imagination | Visualization | Innovation

Understanding Source Systems

- Identifying each table definition and dependencies
- Requirement of Data Cleansing
- During the initial load, capturing changes to data content in the source data is unimportant because you are most likely extracting the entire data source or a potion of it from a predetermined point in time.
- Ability to capture data changes in the source system instantly becomes priority

Logical Data Mapping

Target			Source			Transformation
Table Name	Column Name	Data Type	Table Name	Column Name	Data Type	

- The content of the logical data mapping document has been proven to be the critical element required to efficiently plan ETL processes
- The table type gives us our queue for the ordinal position of our data load processes—first dimensions, then facts.
- The primary purpose of this document is to provide the ETL developer with a clear-cut blueprint of exactly what is expected from the ETL process. This table must depict, without question, the course of action involved in the transformation process
- The transformation can contain anything from the absolute solution to nothing at all. Most often, the transformation can be expressed in SQL. The SQL may or may not be the complete statement

Technical Design Document

- This document explains the whole ETL process the team will be building as part of the project.
- Should contain Data flow which explains what databases are we using and of them which are as source and which are target and how the data is flowing from one database to another.
- List out all ETL objects like Informatica mappings or PL/SQL's which need to created.
- Should explain if the ETL load is Full load or Incremental and also how it is being executed.

Development Phase

- Understand Logical Data Mapping
- Perform data analysis. Specific to the Particular table. Like checking if there are duplicates
- Coding using ETL tool – Informatica, Datastage, PL/SQL to load data
- Handling Nulls. An unhandled NULL value can destroy any ETL process.
- Dates are very peculiar elements because they are the only logical elements that can come in various format. Most database systems support most of the various formats for display purposes but store them in a single standard format
- Perform Unit testing. Mainly to check Data Cleansing and also validating if Requirement in Logical data mapping is achieved or not.

Deployment Phase

- Consolidate all ETL code
- Perform Integration testing
- the formalization of a user-approved prototype for actual production use, including the development of documentation.

- Typically it involves at least two separate deployments

•Deployment into prototype like a production-test environment

•Deployment into Production Environment

Questions

- Email me at kpraneeth@stratapps.com