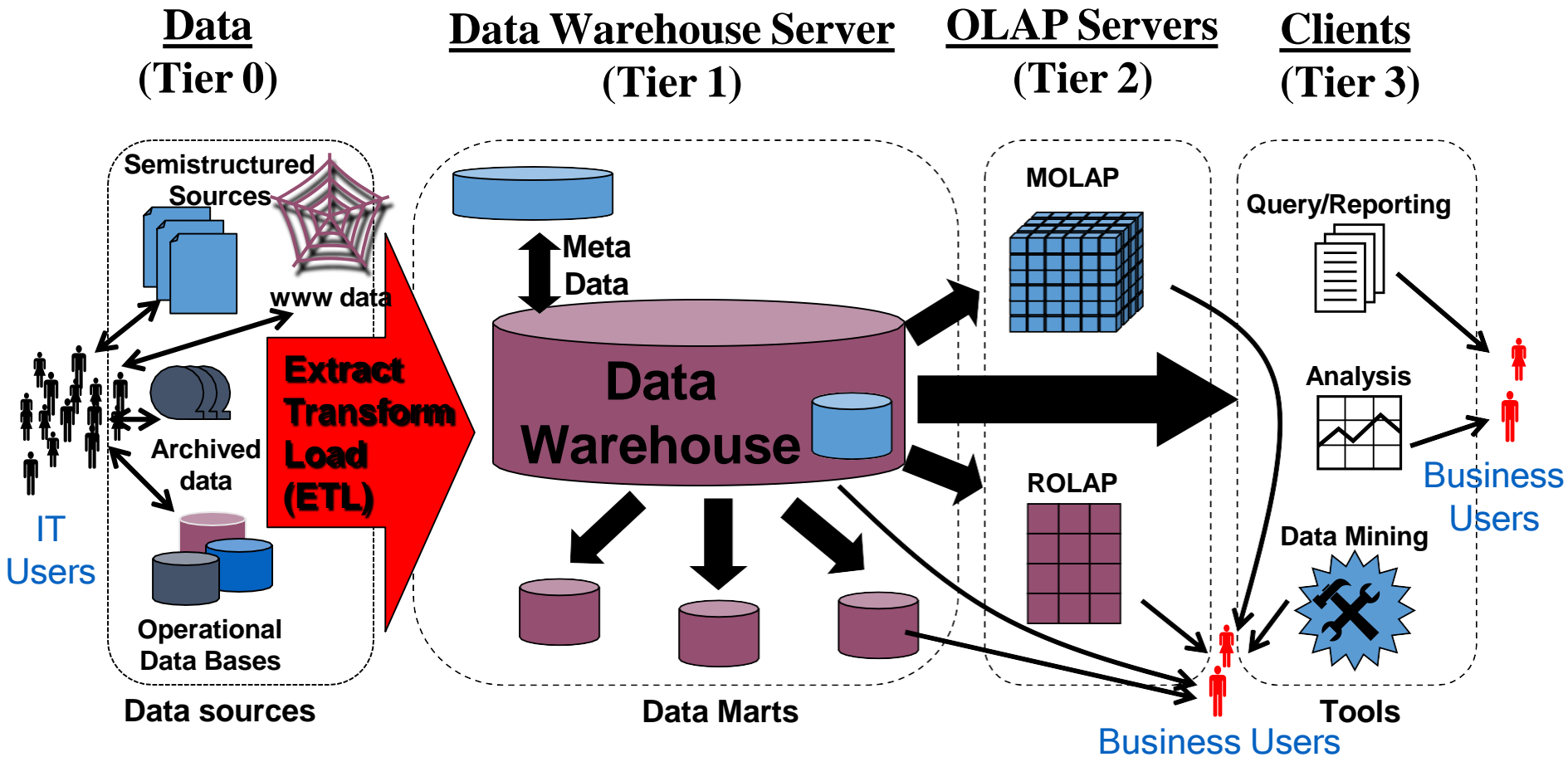


# **Data Warehousing and Business analytics**

## **Lecture-3**

### **Extract Transform Load (ETL)**

# Putting the pieces together

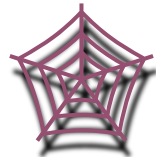


{Comment: All except ETL washed out look}

# The ETL Cycle

## EXTRACT

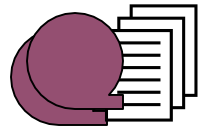
The process of reading data from different sources.



MIS Systems  
(Acct, HR)

**EXTRACT**

Archived data



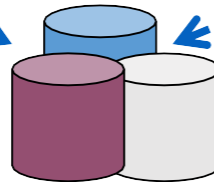
Other indigenous applications  
(COBOL, VB, C++, Java)

## TRANSFORM

The process of transforming the extracted data from its original state into a consistent state so that it can be placed into another database.

**TRANSFORM**

**CLEANSE**



Temporary Data storage

## LOAD

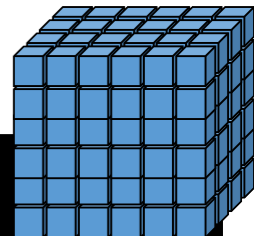
The process of writing the data into the target source.

Data Warehouse



**LOAD**

OLAP

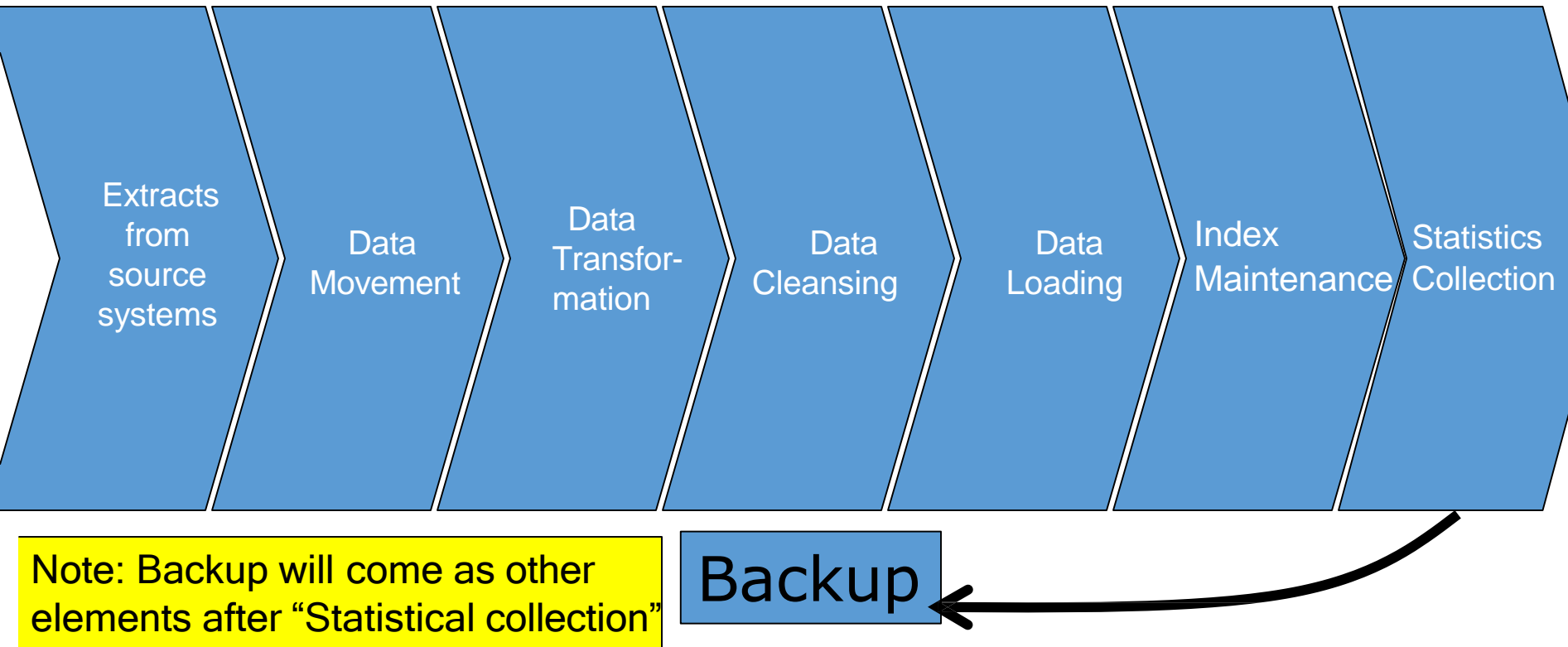


# ETL Processing

**ETL is independent yet interrelated steps.**

**It is important to look at the big picture.**

**Data acquisition time may include...**



**Back-up is a major task, its a DWH not a cube**

# Overview of Data Extraction

First step of ETL, followed by many.

Source system for extraction are typically OLTP systems.

A very complex task due to number of reasons:

- Very complex and poorly documented source system.
- Data has to be extracted not once, but number of times.
- 

The process design is dependent on:

- Which extraction method to choose?
- How to make available extracted data for further processing?

# Types of Data Extraction

- **Logical Extraction**
  - Full Extraction
  - Incremental Extraction
- **Physical Extraction**
  - Online Extraction
  - Offline Extraction
  - Legacy vs. OLTP

# Logical Data Extraction

- **Full Extraction**

- The data extracted completely from the source system.
- No need to keep track of changes.
- Source data made available as-is with any additional information.

- **Incremental Extraction**

- Data extracted after a well defined point/event in time.
- Mechanism used to reflect/record the temporal changes in data (column or table).
- Sometimes entire tables off-loaded from source system into the DWH.
- Can have significant performance impacts on the data warehouse server.

# Physical Data Extraction...

## ■ **Online Extraction**

- **Data extracted directly from the source system.**
- **May access source tables through an intermediate system.**
- **Intermediate system usually similar to the source system.**

## ■ **Offline Extraction**

- **Data NOT extracted directly from the source system, instead staged explicitly outside the original source system.**
- **Data is either already structured or was created by an extraction routine.**
- **Some of the prevalent structures are:**
  - **Flat files**
  - **Dump files**
  - **Redo and archive logs**
  - **Transportable table-spaces**



# Physical Data Extraction

- **Legacy vs. OLTP**
- Data moved from the source system
- Copy made of the source system data
- Staging area used for performance reasons

- **Basic tasks**

1. Selection
2. Splitting/Joining
3. Conversion
4. Summarization
5. Enrichment

# Aspects of Data Loading Strategies

- Need to look at:
  - Data freshness
  - System performance
  - Data volatility
- Data Freshness
  - Very fresh low update efficiency
  - Historical data, high update efficiency
  - Always trade-offs in the light of goals
- System performance
  - Availability of staging table space
  - Impact on query workload
- Data Volatility
  - Ratio of new to historical data
  - High percentages of data change (batch update)

# Three Loading Strategies

- Once we have transformed data, there are three primary loading strategies:
- [Full data refresh](#) with BLOCK INSERT or 'block slamming' into empty table.
- [Incremental data refresh](#) with BLOCK INSERT or 'block slamming' into existing (populated) tables.
- [Trickle/continuous feed](#) with constant data collection and loading using row level insert and update operations.

# ETL Issues

Data from different source systems will be different, poorly documented and dirty. Lot of analysis required.

Easy to collate addresses and names? Not really. No address or name standards.

Use software for standardization. Very expensive, as any “standards” vary from country to country, not large enough market.

# ETL Issues

Things would have been simpler in the presence of operational systems, but that is not always the case

Manual data collection and entry. Nothing wrong with that, but potential to introduces lots of problems.

Data is never perfect. The cost of perfection, extremely high vs. its value.

# “Some” Issues

- Usually, if not always underestimated
- Diversity in source systems and platforms
- Inconsistent data representations
- Complexity of transformations
- Rigidity and unavailability of legacy systems
- Volume of legacy data
- Web scrapping

# Complexity of problem/work underestimated

- Work seems to be deceptively simple.
- People start manually building the DWH.
- Programmers underestimate the task.
- Impressions could be deceiving.
- Traditional DBMS rules and concepts break down for very large heterogeneous historical databases.



# Diversity in source systems and platforms

Platform	OS	DBMS	MIS/ERP
Main Frame	VMS	Oracle	SAP
Mini Computer	Unix	Informix	PeopleSoft
Desktop	Win NT	Access	JD Edwards
	DOS	Text file	

Dozens of source systems across organizations

Numerous source systems within an organization

Need specialist for each

# Inconsistent data representations

## Same data, different representation

### Date value representations

Examples:

970314

03/14/1997

March 14 1997

1997-03-14

14-MAR-1997

2450521.5 (Julian date format)

### Gender value representations

Examples:

- Male/Female

- 0/1

- M/F

- PM/PF

# Multiple sources for same data element

Need to rank source systems on a per data element basis.

Take data element from source system with highest rank where element exists.

“Guessing” gender from name

Something is better than nothing?

Must sometimes establish “group ranking” rules to maintain data integrity.

First, middle and family name from two systems of different rank. People using middle name as first name.

# Complexity of required transformations

## **Simple one-to-one scalar transformations**

- 0/1 → M/F

## **One-to-many element transformations**

- 4 x 20 address field → House/Flat, Road/Street, Area/Sector, City.

## **Many-to-many element transformations**

- House-holding (who live together) and individualization (who are same) and same lands.

# Rigidity and unavailability of legacy systems

- Very difficult to add logic to or increase performance of legacy systems.
- Utilization of expensive legacy systems is optimized.
- Therefore, want to off-load transformation cycles to open systems environment.
- This often requires new skill sets.
- Need efficient and easy way to deal with incompatible mainframe data formats.

# Volume of legacy data

- Talking about not weekly data, but data spread over years.
- Historical data on tapes that are serial and very slow to mount etc.
- Need lots of processing and I/O to effectively handle large data volumes.
- Need efficient interconnect bandwidth to transfer large amounts of data from legacy sources to DWH.

# Web scrapping

- Lot of data in a web page, but is mixed with a lot of “junk”.
- Problems:
  - Limited query interfaces
    - Fill in forms
  - “Free text” fields
    - E.g. addresses
  - Inconsistent output
    - i.e., html tags which mark interesting fields might be different on different pages.
  - Rapid change without notice.

# Beware of data quality

- Data quality is always worse than expected.
- Will have a couple of lectures on data quality and its management.
- It is not a matter of few hundred rows.
- Data recorded for running operations is not usually good enough for decision support.
  - Correct totals don't guarantee data quality.
  - Not knowing gender does not hurt POS.
  - Centurion customers popping up.



# ETL vs. ELT

There are two fundamental approaches to data acquisition:

**ETL: Extract, Transform, Load** in which data transformation takes place on a separate transformation server.

**ELT: Extract, Load, Transform** in which data transformation takes place on the data warehouse server.

Combination of both is also possible

# Extracting Changed Data

## **Incremental data extraction**

Incremental data extraction i.e. what has changed, say during last 24 hrs if considering nightly extraction.

## **Efficient when changes can be identified**

This is efficient, when the small changed data can be identified efficiently.

## **Identification could be costly**

Unfortunately, for many source systems, identifying the recently modified data may be difficult or effect operation of the source system.

## **Very challenging**

Change Data Capture is therefore, typically the most challenging technical issue in data extraction.

## **Two CDC sources**

- Modern systems
- Legacy systems

# CDC in Modern Systems

- **Time Stamps**

- Works if timestamp column present
- If column not present, add column
- May not be possible to modify table, so add triggers

- **Triggers**

- Create trigger for each source table
- Following each DML operation trigger performs updates
- Record DML operations in a log

- **Partitioning**

- Table range partitioned, say along date key
- Easy to identify new data, say last week's data

# CDC in Legacy Systems

- **Changes recorded in tapes** Changes occurred in legacy transaction processing are recorded on the log or journal tapes.
- **Changes read and removed from tapes** Log or journal tape are read and the update/transaction changes are stripped off for movement into the data warehouse.
- **Problems with reading a log/journal tape are many:**
  - **Contains lot of extraneous data**
  - **Format is often arcane**
  - **Often contains addresses instead of data values and keys**
  - **Sequencing of data in the log tape often has deep and complex implications**
  - **Log tape varies widely from one DBMS to another.**

# CDC Advantages: Modern Systems

## Advantages

1. Immediate.
2. No loss of history
3. Flat files NOT required



Modern  
Systems

# CDC Advantages: Legacy Systems

## Advantages

1. No incremental on-line I/O required for log tape
2. The log tape captures all update processing
3. Log tape processing can be taken off-line.
4. No haste to make waste.

**Legacy  
Systems**

# Major Transformation Types

- Format revision
- Decoding of fields
- Calculated and derived values
- Splitting of single fields
- Merging of information
- Character set conversion
- Unit of measurement conversion
- Date/Time conversion
- Summarization
- Key restructuring
- Duplication



- Format revision

- Decoding of fields

Covered in De-Norm

- Calculated and derived values

Covered in issues

- Splitting of single fields

# Major Transformation Types

- **Merging of information**

Not really means combining columns to create one column.  
Info for product coming from different sources merging it into single entity.

- **Character set conversion**

For PC architecture converting legacy EBCDIC to ASCII

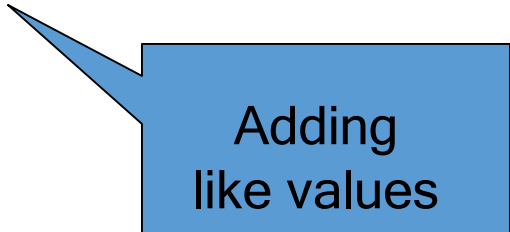
- **Unit of measurement conversion**

For companies with global branches Km vs. mile or lb vs Kg

- **Date/Time conversion**

November 14, 2005 as 11/14/2005 in US and 14/11/2005 in the British format.  
This date may be standardized to be written as 14 NOV 2005.

- **Aggregation & Summarization**
- **How they are different?**



Adding  
like values

Summarization with calculation across business dimension is aggregation. Example Monthly compensation = monthly sale + bonus

- **Why both are required?**
  - Grain mismatch (don't require, don't have space)
  - Data Marts requiring low detail
  - Detail losing its utility

# Major Transformation Types

- **Key restructuring** (inherent meaning at source)

92	42	4979	234
<b>Country_Code</b>	<b>City_Code</b>	<b>Post_Code</b>	<b>Product_Code</b>

- i.e. 92424979234 changed to 12345678

- **Removing duplication**

Incorrect or missing value

Inconsistent naming convention ONE vs 1

Incomplete information

Physically moved, but address not changed

Misspelling or falsification of names

- Domain value redundancy
  - Non-standard data formats
  - Non-atomic data values
  - Multipurpose data fields
  - Embedded meanings
  - Inconsistent data values
  - Data quality contamination

# Data content defects Examples

- Domain value redundancy
  - Unit of Measure
    - Dozen, Doz., Dz., 12
- Non-standard data formats
  - Phone Numbers
    - 1234567 or 123.456.7
- Non-atomic data fields
  - Name & Addresses
    - Dr. Nguyen Van A, PhD

- Embedded Meanings
  - RC, AP, RJ
    - received, approved, rejected

# Background

- **Other names:** Called as data scrubbing or cleaning.
- **More than data arranging:** DWH is NOT just about arranging data, but should be clean for overall health of organization. We drink clean water!
- **Big problem, big effect:** Enormous problem, as most data is dirty. GIGO
- **Dirty is relative:** Dirty means does not confirm to proper domain definition and vary from domain to domain.
- **Paradox:** Must involve domain expert, as detailed domain knowledge is required, so it becomes semi-automatic, but has to be automatic because of large data sets.
- **Data duplication:** Original problem was removing duplicates in one system, compounded by duplicates from many systems.



# Lighter Side of Dirty Data

- Year of birth 1995 current year 2005

{Comment: Show picture of baby}

- Born in 1986 hired in 1985
- Who would take it seriously? Computers while summarizing, aggregating, populating etc.
- Small discrepancies become irrelevant for large averages, but what about sums, medians, maximum, minimum etc.?

# Serious Side of dirty data

- **Decision making at the Government level on investment** based on rate of birth in terms of schools and then teachers. Wrong data resulting in over and under investment.
- **Direct mail marketing** sending letters to wrong addresses returned, or multiple letters to same address, loss of money and bad reputation and wrong identification of marketing region.

# 3 Classes of Anomalies...

## ■ Syntactically Dirty Data

- Lexical Errors
- Irregularities

## ■ Semantically Dirty Data

- Integrity Constraint Violation
- Business rule contradiction
- Duplication

## ■ Coverage Anomalies

- Missing Attributes
- Missing Records

# 3 Classes of Anomalies...

## ■ **Syntactically Dirty Data**

- Lexical Errors
- Discrepancies between the structure of the data items and the specified format of stored values
- e.g. number of columns used are unexpected for a tuple (mixed up number of attributes)
- Irregularities
- Non uniform use of units and values, such as only giving annual salary but without info i.e. in US\$ or PK Rs?

## ■ **Semantically Dirty Data**

- Integrity Constraint violation
- Contradiction
  - DoB > Hiring date etc.
- Duplication

- **Coverage Anomalies**

- Missing Attribute

- Result of omissions while collecting the data.
  - A constraint violation if we have null values for attributes where NOT NULL constraint exists.
  - Case more complicated where no such constraint exists.
  - Have to decide whether the value exists in the real world and has to be deduced here or not.

# Coverage Anomalies

- Equipment malfunction (bar code reader, keyboard etc.)
- Inconsistent with other recorded data and thus deleted.
- Data not entered due to misunderstanding/illegibility.
- Data not considered important at the time of entry (e.g. Y2K).

# Handling missing data

- Dropping records.
- “Manually” filling missing values.
- Using a global constant as filler.
- Using the attribute mean (or median) as filler.
- Using the most probable value as filler.

- Primary key problems
- Non-Primary key problems



# Primary key problems

- Same PK but different data.
- Same entity with different keys.
- PK in one system but not in other.
- Same PK but in different formats.

# Non primary key problems...

- Different encoding in different sources.
- Multiple ways to represent the same information.
- Sources might contain invalid data.
- Two fields with different data but same name.

# Non primary key problems

- Required fields left blank.
- Data erroneous or incomplete.
- Data contains null values.

1.Statistical

2.Pattern Based

3.Clustering

4.Association Rules