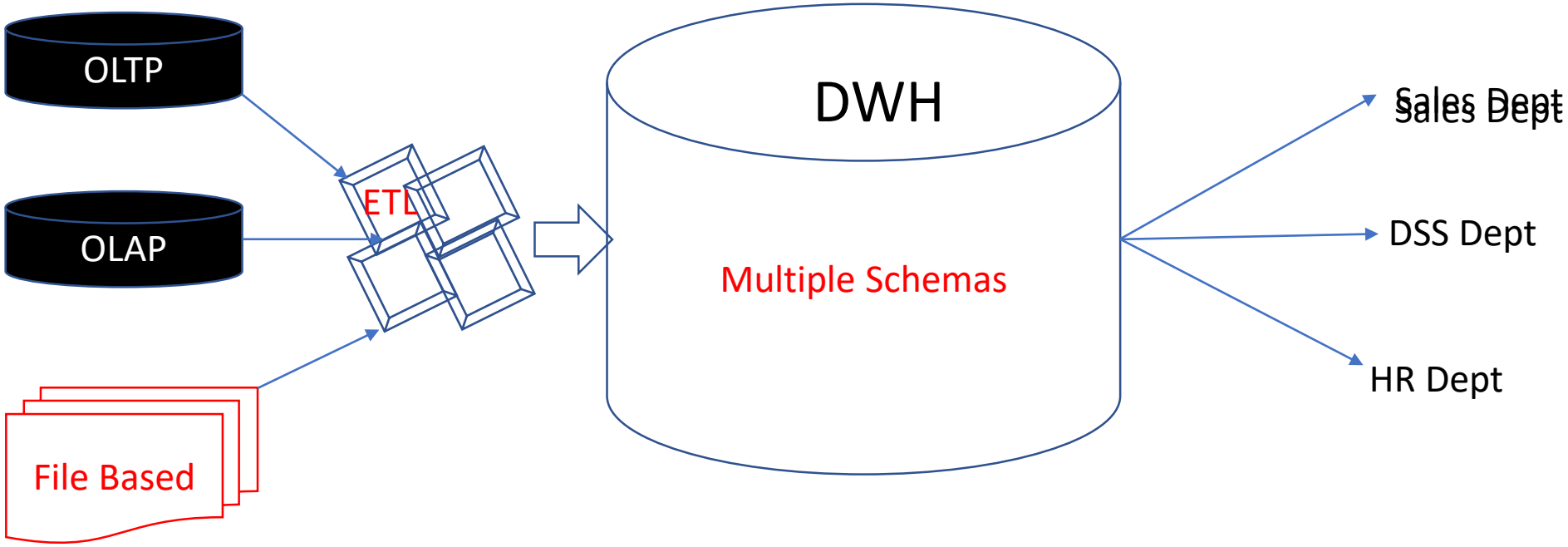
Data Warehouse

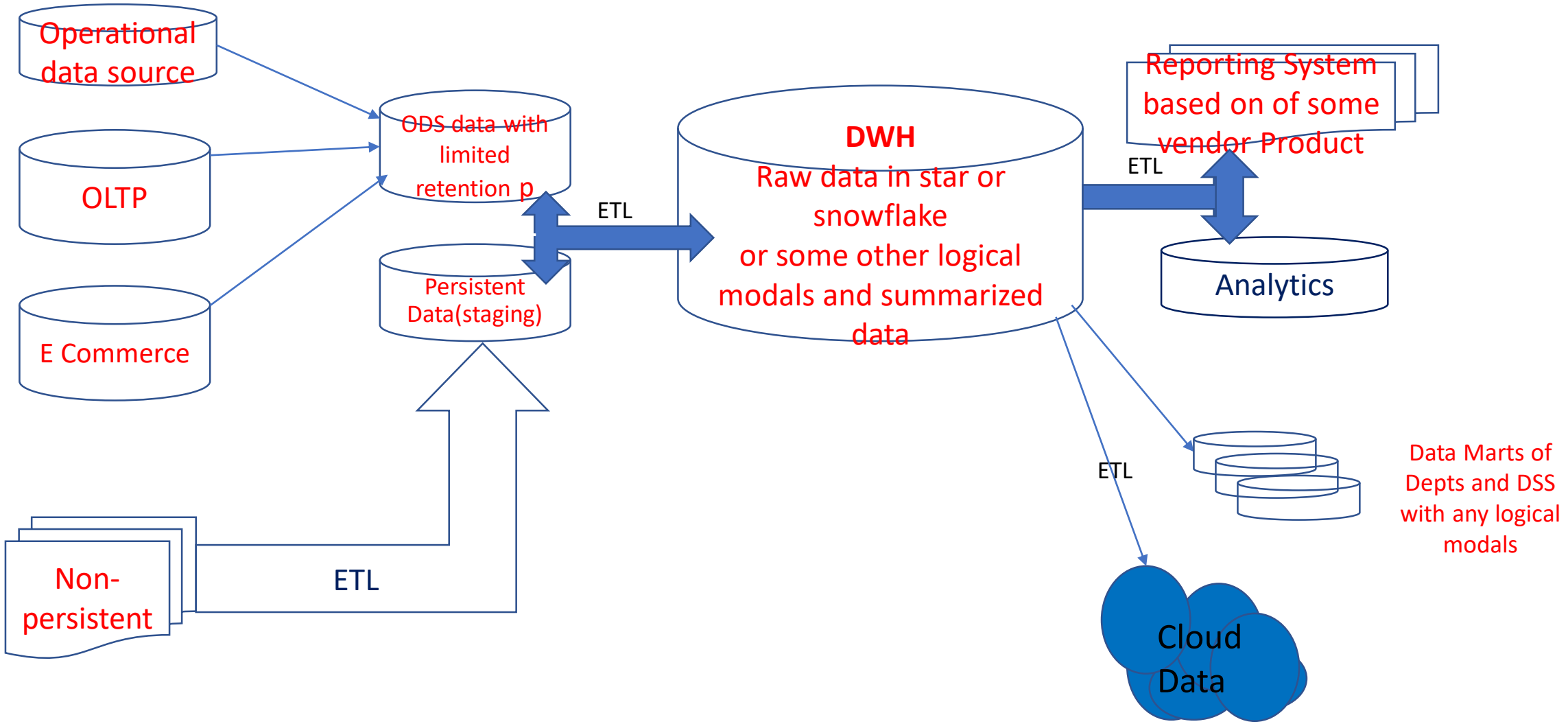
Srinivas Maddali

Data Warehouse Architectures

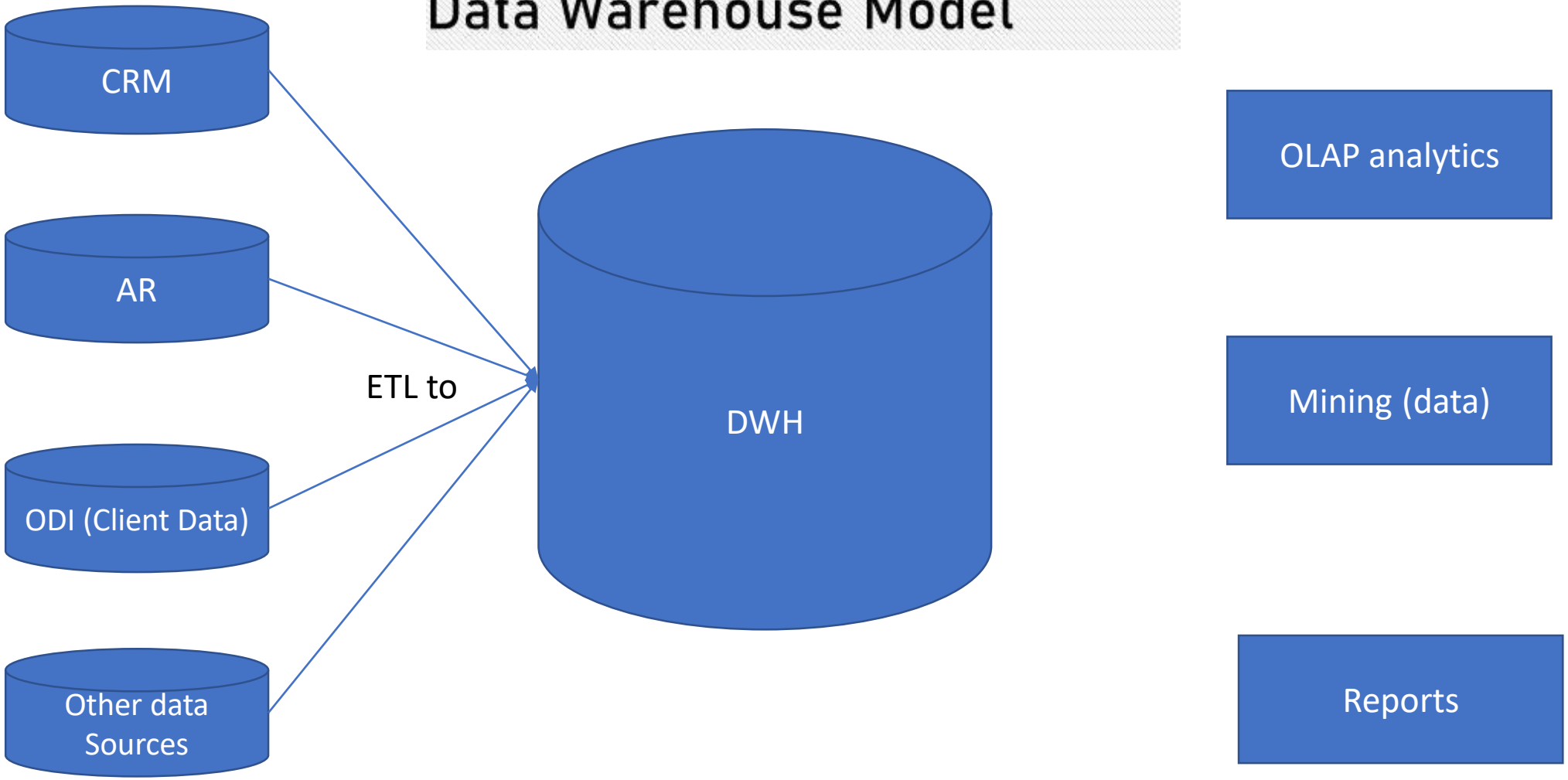
Traditional Architecture



Non-Traditional DWH Design

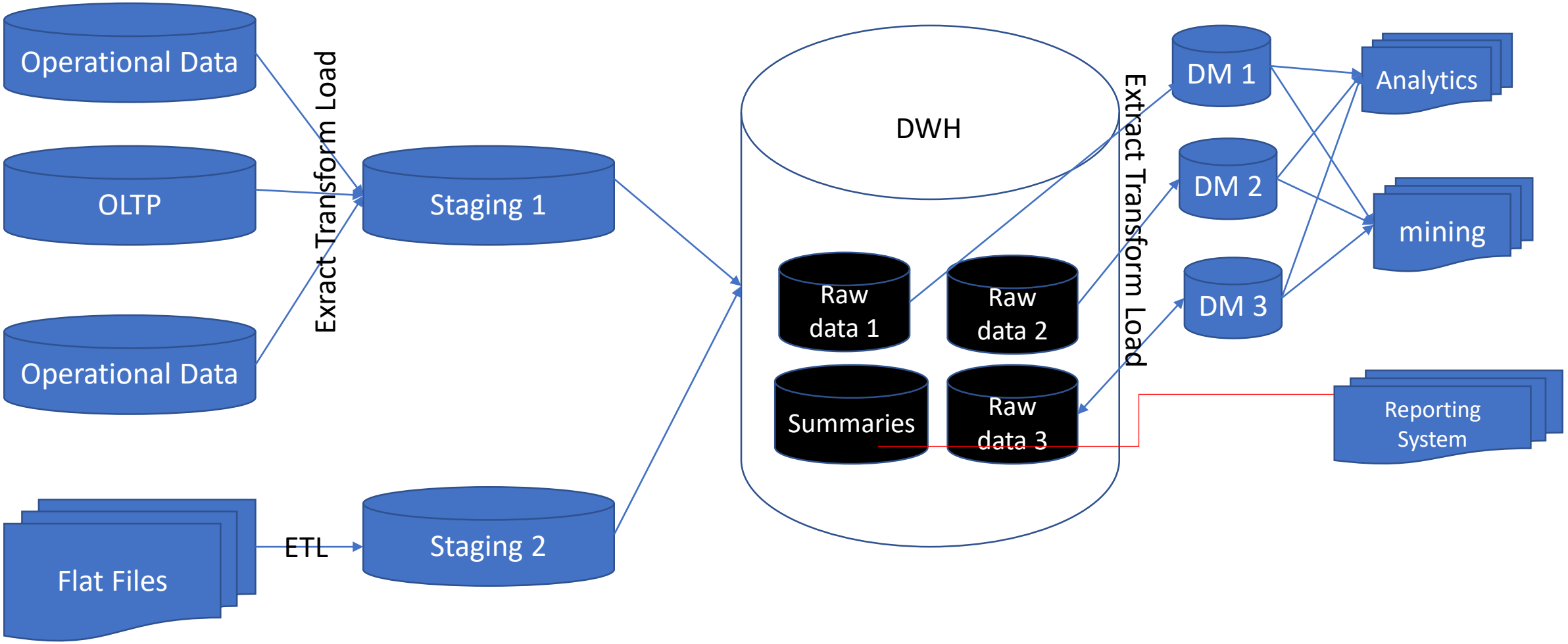


Extract Transform and Load Data Warehouse Model

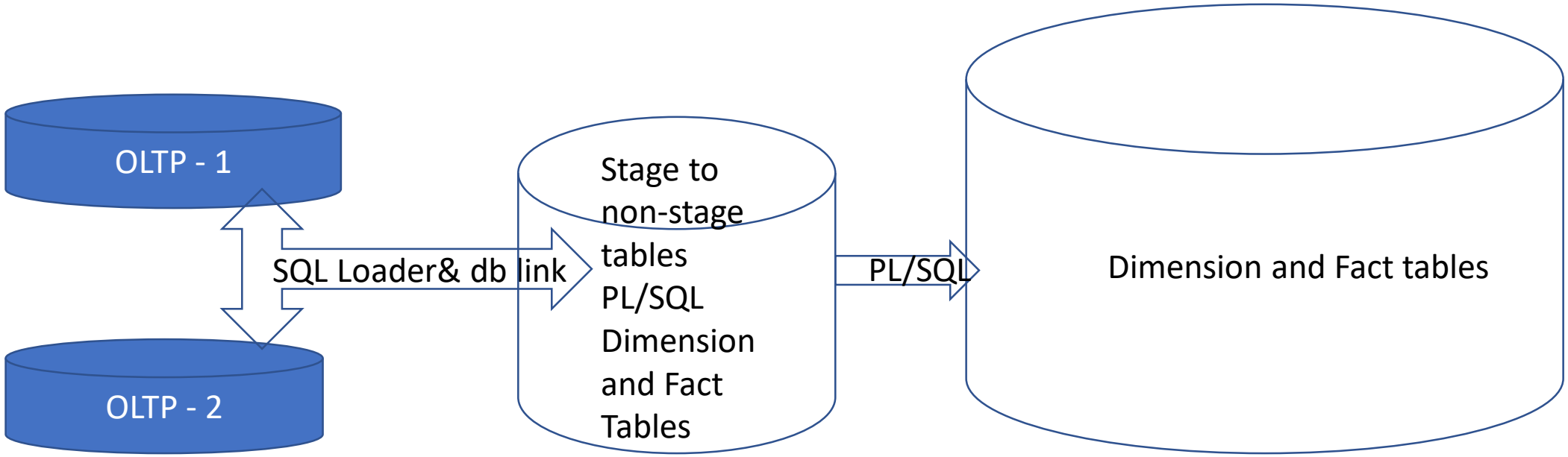


Data Warehouse With ETL, Staging, DWH, Marts, ETL, and End Users For Analytics and Mining

Srinivas Maddali



Reporting and Historical Analytics
(Oracle – In one project the following design is used)

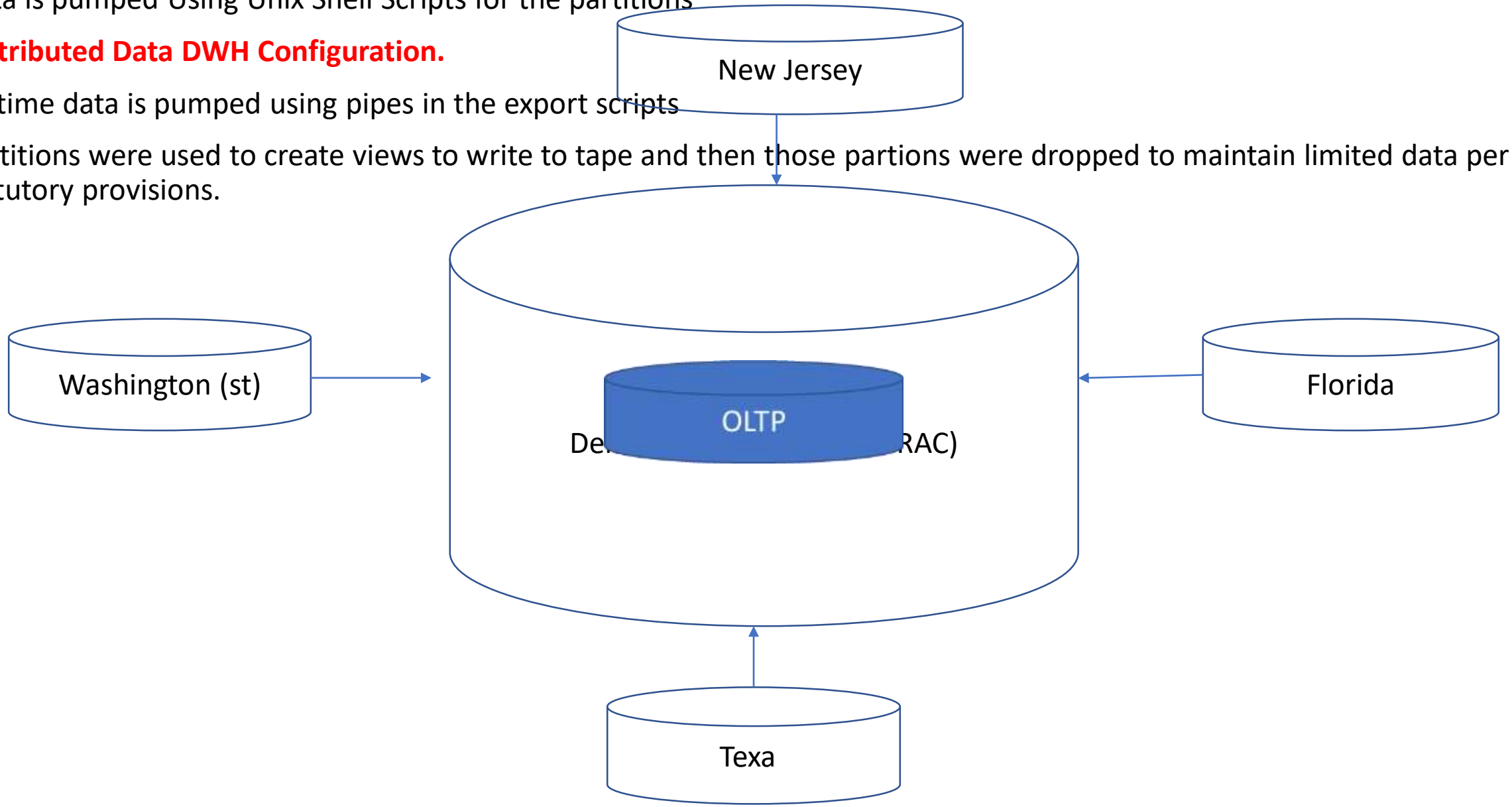


Data is pumped Using Unix Shell Scripts for the partitions

Distributed Data DWH Configuration.

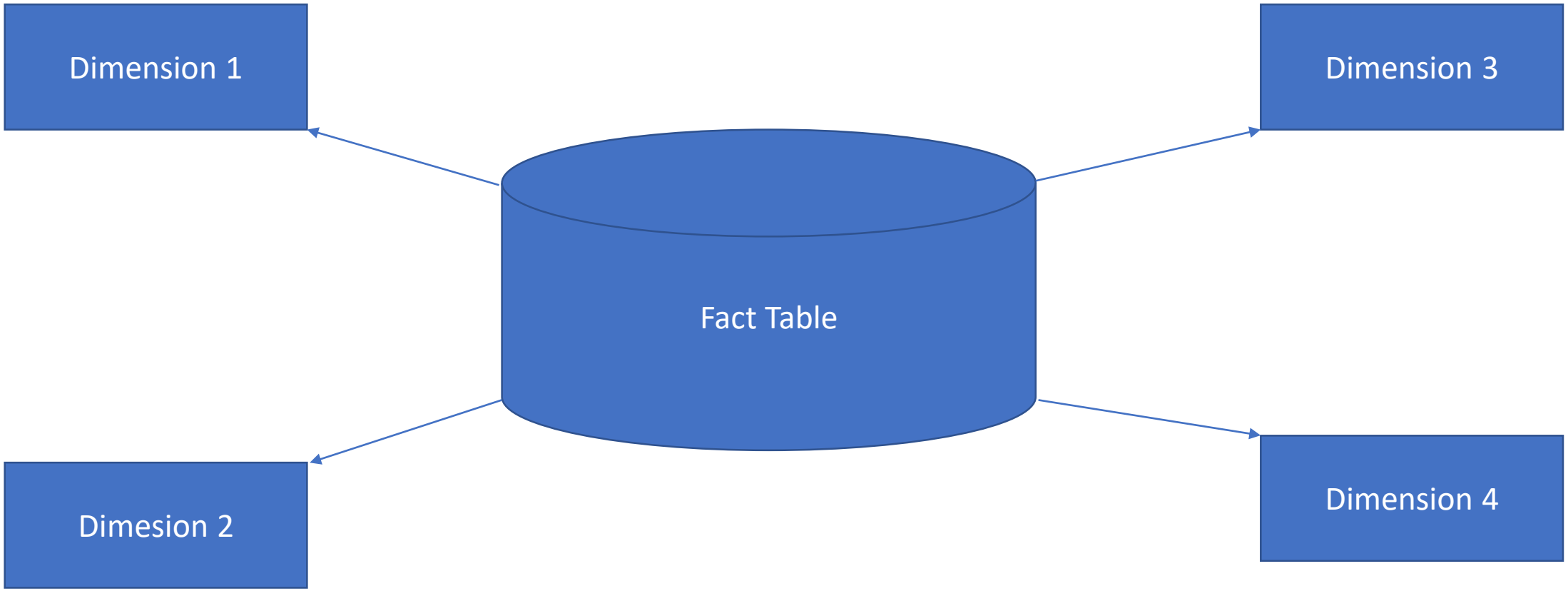
1st time data is pumped using pipes in the export scripts

Partitions were used to create views to write to tape and then those partions were dropped to maintain limited data per statutory provisions.

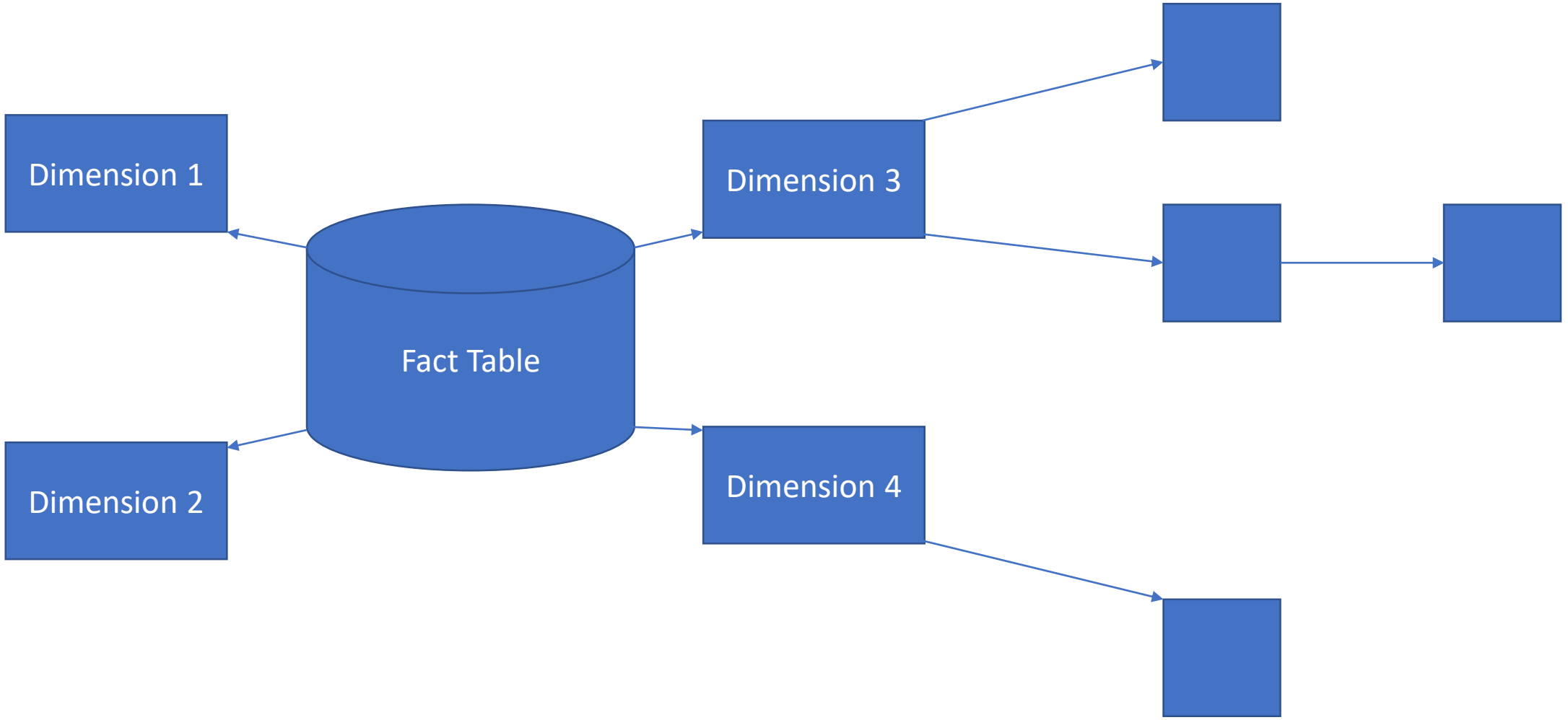


Schema In Data Warehouses

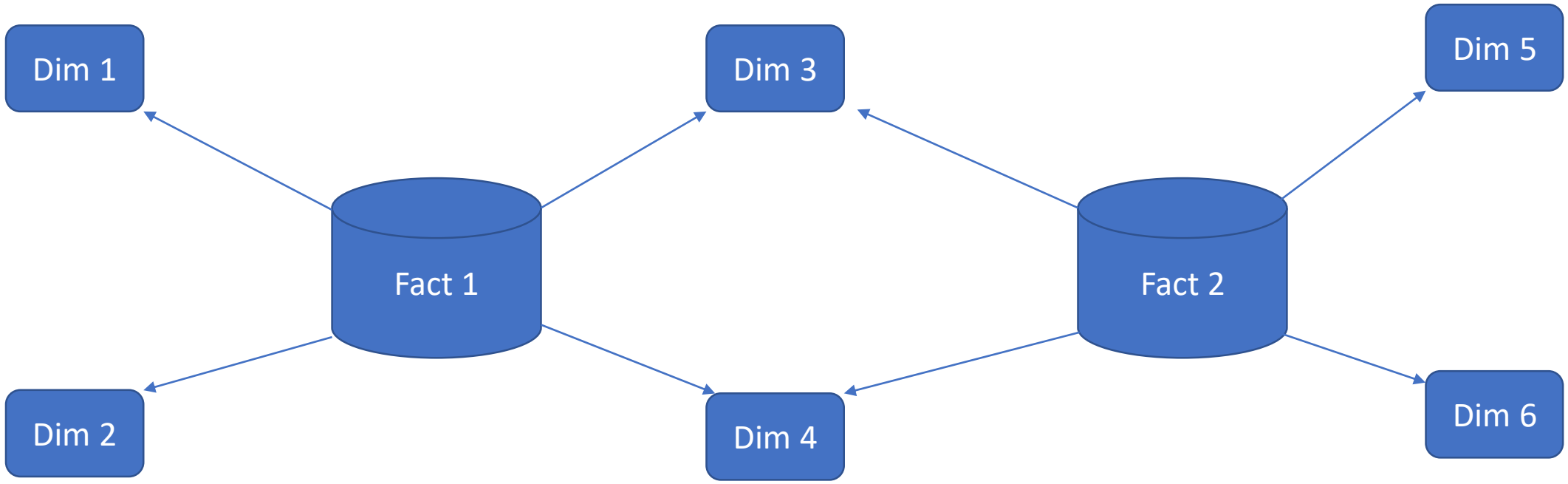
Star Schema



Snowflake Schema



Galaxy Schema



Tom Fact 1 and Fact 2 Dim 3 and Dim 4 dimensions are common. 2 Fact tables can have same Time dimension and likewise another dimension (ex: product). Etc,.

Important Concepts Of Data Warehouse Dimension Tables

In Snowflake Schema -- The snowflake schema is an extension of the star schema, where each point of the star explodes into more points. In a star schema, each dimension is represented by a single dimensional table, whereas in a snowflake schema, that dimensional table is normalized into multiple lookup tables, each representing a level in the dimensional hierarchy. The dimension can be sub-divided. Example – Time dimension can be <=7 (year, quarter, month, week, day and date). Likewise, supplier dimension can be >4 (supplier, zone, product group, product, Etc.,). Likewise, Sales can be >3 (product, store, location, Etc.,.)

Star Schema - In the star schema design, a single object (the fact table) sits in the middle and is radically connected to other surrounding objects (dimension lookup tables) like a star. Each dimension is represented as a single table. The primary key in each dimension table is related to a foreign key in the fact table.

Dimensions can be slow changing.

Type 1: The new record replaces the original record. No trace of the old record exists.

Example: Latest Record is only maintained. Historical Information of the record is not available. Business can not track the history. If you do not want the historical information, use this model.

Type 2: A new record is added.

Historical Information is available. Storage usage is a concern.

Type 3: The original record is modified to reflect the change. The data is updated every time the data is changed with date effective. The data can be viewed parent – child relations.

Designing Data Warehouse

Consider the following:

a. Gather business requirements.
b. Setup a test environment before modelling data
c. Model the data
    i. logical model
    ii. Physical model
d. Choose the ETL and/or ELT solution or method (vendor or home-grown)
e. OLAP – Marts (reporting database and summaries with rollup capabilities). Sometimes this data can be transformed data from the fact table. Fact table historical table and the transformed data can also be historical. Storage is the concern. Facts which can be represented with multi dimension can form into a cube or
    i. MOLAP – Multidimensional OLAP – Attributes, levels, hierarchies
    ii. ROLAP -- Relational OLAP – uses a relational database. Supports larger user groups and greater amounts of data.
    iii. HOLAP – Hybrid OLAP. - HOLAP technologies attempt to combine the advantages of MOLAP and ROLAP. For summary-type information, HOLAP leverages cube technology for faster performance. When detail information is needed, HOLAP can "drill through" from the cube into the underlying relational data. Crystal Reports, Informatica, Cognos and other tools does this work. Homegrown Front End tools can also generate the reports.
    Data retention is limited is limited with historical information because of the nature of the database.
f. OLAP vs DWH (multiple marts and summaries with historical data) and dimensions usable by multiple marts – Fact Tables Galaxy Schema.
g. Partitioning od the data for performance
h. Front End
    i. For business executives (private IP)
    ii. For clients to the business (Public IP) sitting on the edge network zone.
i. Optimization of SQL queries
j. Ad-hoc queries

Designing Data Warehouse

Some Natures of Fact Tables In DWH:

1. Fact tables can be factless. Fact table can be with no “measures”. The Fact table can be an intersection of “dimension”
2. There are some junk dimensions. Like “yes” or ‘no” columns. They may be needed in the source table for the OLTP application functionality. But they are NOT dimensions in DWH.
3. Some dimensions with the same meaning that appear in more than one fact table but with different PK. The meaning of the dimension in different fact tables have different attributes in source table/s. Example: week definition. Quarter identification by different Enterprises

DWH design after rolled into Production:

1. Once QA gives a thumb’s up
2. users are properly trained
3. Documentation is distributed
4. Documentation source is published to all stake holders
5. General Maintenance Schedules are announced and processis defined
6. Emergency Maintenance Process is determind
7. RCA process is defined
8. Change Management Process is defined
9. DWH application (frontend) and DWH are to be viewed differently
10. Enhancements (incremental) qualification, integration and deployment process is to be defined
11. DWH growth trends are observed and Retention policy is to be defined
12. DWH data archival Process is to be defined