

Relational Database Design

CS 537- Big Data Analytics

Dr. Faisal Kamiran

What is OLAP vs OLTP?

Online Analytical Processing (OLAP)

Databases optimized for these workloads allow for **complex analytical and ad hoc queries**. These types of databases are optimized for reads.

Online Transactional Processing (OLTP)

Databases optimized for these workloads allow for **less complex queries in large volume**. The types of queries for these databases are read, insert, update and delete.

Example

- **OLAP queries**
 - “How many shoes were sold in Lahore in a specific month.”
- **OLTP queries**
 - “The price of the shoe.”

OLTP queries will perform very little aggregations while
OLAP is designed to have heavy aggregations

Structuring Your Database

- **Normalization**
 - To reduce data redundancy and increase data integrity.
- **Denormalization**
 - Combine multiple tables in order to facilitate faster queries
 - Must be done in read heavy workloads to increase performance

Normalization

The process of structuring a relational database in accordance with a series of **normal forms** in order to **reduce data redundancy and increase data integrity**

Data Redundancy: Goal is to remove duplicate data


Data Integrity: Make sure that you get the correct answer from the database
(update data at one place and that becomes the truth)

Normalization

Normalization

Album ID	Album Name	Artist Name	Year	List of Songs
1	Burning Sun	Keating	1970	[Burning Sun, Feet, Moon is jealous]
2	Soul	Harvey	1960	[Hey Ma, Jenifer, Life is good]

Does it follow the normalization rules?



Do you think the table is in Normal Form?

First Normal Form

- Atomic Values: Each cell contains **unique** and **single** values
- There should not be any tuples or any list of values in a single cell

Album ID	Album Name	List of Songs
1	Burning Sun	[Burning Sun, Feet, Moon is jealous]
2	Soul	[Hey Ma, Jenifer, Life is good]

Album ID	Album Name	Song
1	Burning Sun	Burning Sun
1	Burning Sun	Feet
1	Burning Sun	Moon is jealous
2	Soul	Hey Ma
2	Soul	Jenifer
2	Soul	Life is good

First Normal Form

How to Reach 1st Normal Form

- Separate different relations into different tables
- We do not want a single giant table

Customer and Sales table could have been merged. We could have a single table with all possible information

Customer table

Name	Email	ID	City
Amanda	jdoe@xyz.com	abc	NYC
Toby	n/a	def	NYC

Sales table

Name	Amount
Amanda	100.00
Toby	50.00

First Normal Form

How to Reach 1st Normal Form

- Keep relationships between tables together with **foreign keys**

There should be a way to link these tables together. The tables are linked together with foreign keys.

Customer table

Name	Email	ID	City
Amanda	jdoe@xyz.com	abc	NYC
Toby	n/a	def	NYC

Sales table

Name	Amount
Amanda	100.00
Toby	50.00

First Normal Form

How to Reach 1st Normal Form

- Atomic values: each cell contains unique and single values
- Keep relationships between tables together with **foreign keys**

Customer table

Name	Email	ID	City
Amanda	jdoe@xyz.com	abc	NYC
Toby	n/a	def	NYC

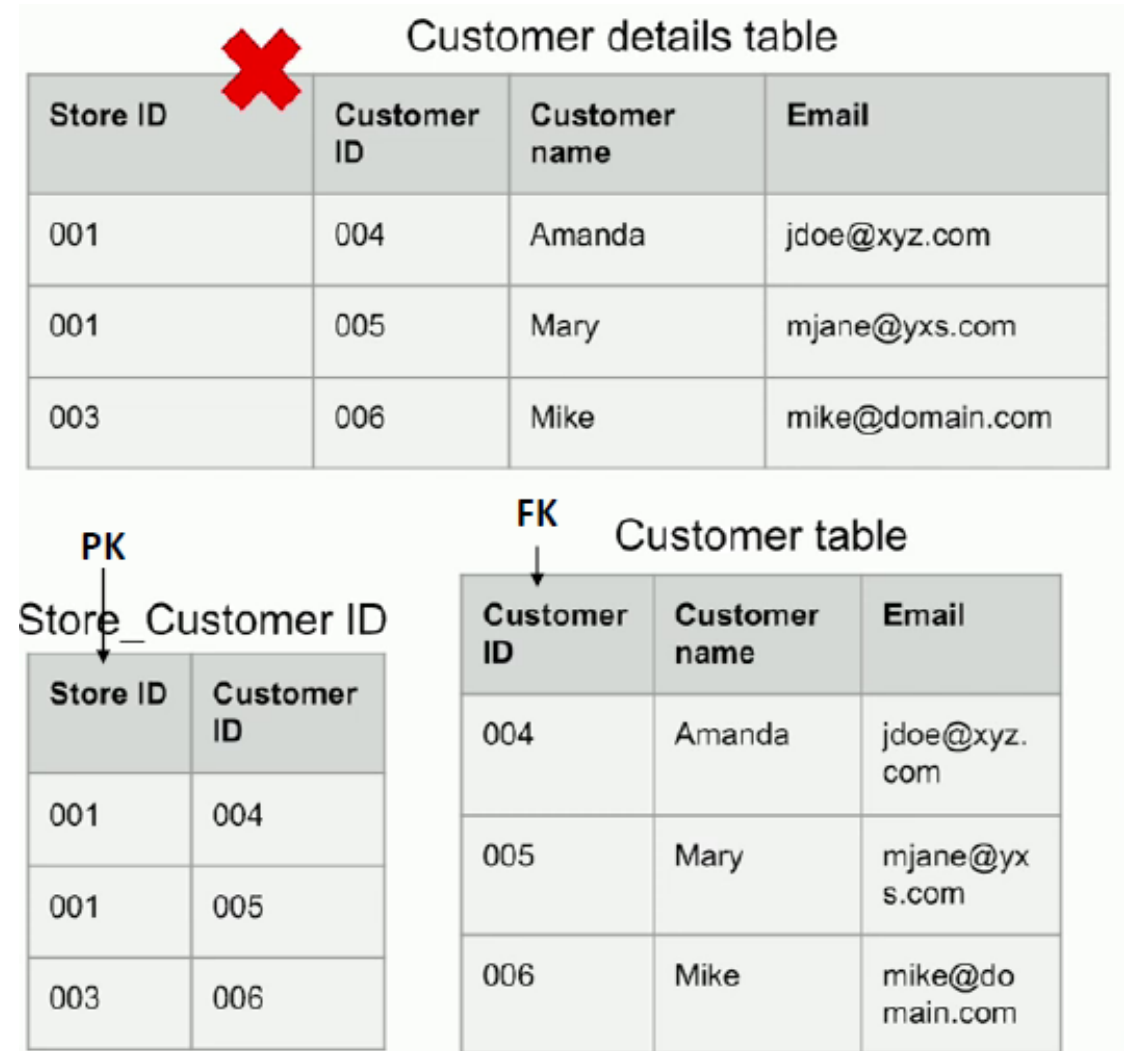
Sales table

Name	Amount
Amanda	100.00
Toby	50.00

Second Normal Form

How to Reach 2nd Normal Form

- Have reached 1NF
- All tables in the table must rely on the Primary Key



Third Normal Form

How to Reach 2nd Normal Form

- Have reached 2NF
- No transitive dependencies

Lead singer is related to the name of the band.
Changing the band will change the lead singer

 Awards table

Music Award	Year	Winner Record of Year	Lead Singer
Grammy	1965	The Beatles	John Lennon
CMA	2000	Faith Hill	Faith Hill
Grammy	1970	The Beatles	John Lennon
VMA	2001	U2	Bono

Diagram showing a transitive dependency: Winner Record of Year → Lead Singer. A double-headed arrow connects 'The Beatles' in the Winner Record of Year column to 'John Lennon' in the Lead Singer column for the first and third rows.

Awards Table

Music Award	Year	Winner Record of Year
Grammy	1965	The Beatles
CMA	2000	Faith Hill
Grammy	1970	The Beatles
VMA	2001	U2

Lead Singer

Band Name	Lead Singer
The Beatles	John Lennon
Faith Hill	Faith Hill
U2	Bono

Consequences of Normalization

- Data redundancy is reduced or eliminated.
- Relations are broken into smaller, related tables.
- Using all the attributes from the original relation requires joining these smaller tables.

Denormalization

Deliberately reintroducing some redundancy, so that we can access data faster.



Denormalization

Objective: To improve the read performance of a database at the expense of losing some write performance by adding redundant copies of data.

- JOINS allow outstanding flexibility but are extremely slow.
- Denormalization is preferred for databases with heavy reads
- Denormalization is done **after** normalization
- Denormalization utilizes more space as multiple copies of the data are stored

Denormalization

- Denormalization is all about performance.
- You do not need heavy joins to answer queries.
- We have separate tables with duplicate copies of data to increase performance.
- We first perform normalization and then denormalization.

How much a customer spent?

Customer		
Name	City	Amount
Amanda	NYC	100.00
Toby	NYC	30.00

Shipping		
Name	City	Item
Amanda	NYC	Shirt
Toby	NYC	Pants

The type of items we need to ship?

When should denormalization be done?

We want a logical design change

- We want to model our data differently
- Reads will be faster (select)
- Writes will be slower (insert, update, delete)

Data Consistency

- There are multiple copies of data so each copy should be updated or deleted at the same time
- City and Name should be inserted or updated in both tables

Customer		
Name	City	Amount
Amanda	NYC	100.00
Toby	NYC	30.00

Shipping		
Name	City	Item
Amanda	NYC	Shirt
Toby	NYC	Pants

Normalization & Denormalization

Normalization	Denormalization
Redundancy and inconsistency is reduced	Redundancy is added for quick execution of queries
Number of tables increases	Number of tables decreases
Data integrity is maintained	Does not maintain data integrity
Optimizes memory usage	Does not optimize memory usage

Data Warehouse

A Business Perspective

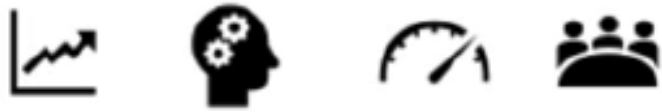
Operational vs Analytical Business Processes



Operational Processes

Make it work!

- Find goods & make orders (for customers)
- Stock and find goods (for inventory staff)
- Pick up & deliver goods (for delivery staff)

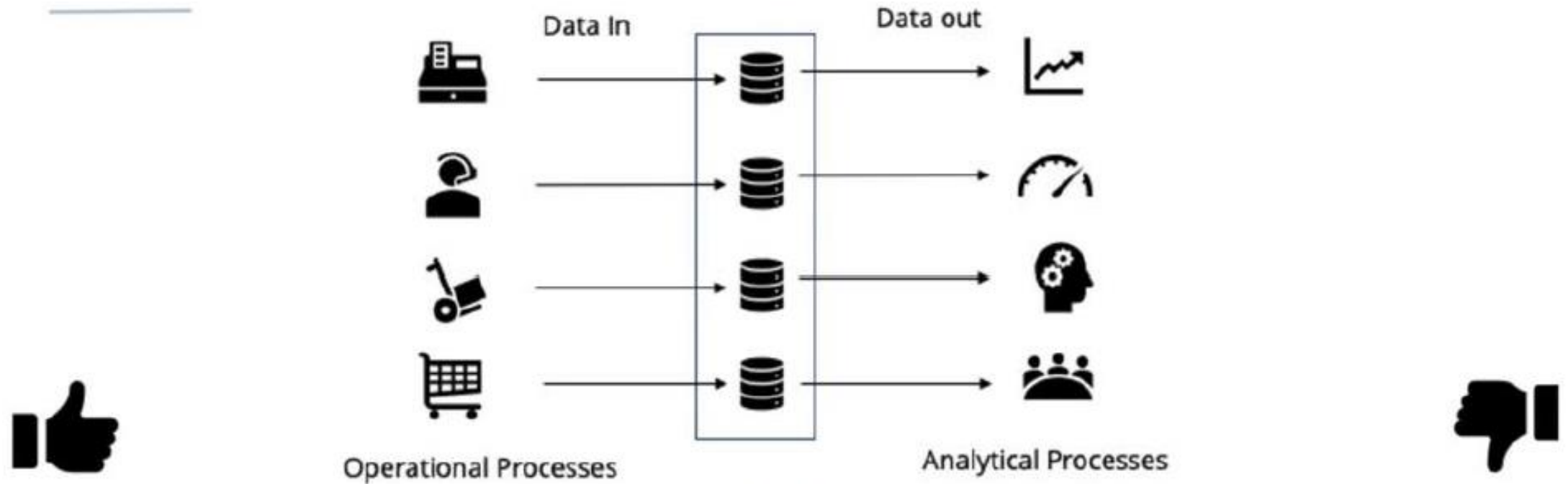


Analytical Processes

What is going on?

- Assess the performance of sales staff (for HR)
- See the effect of different sales channels (for marketing)
- Monitor sales growth (for management)

Same data source for operational & analytical processes?



Operational Databases

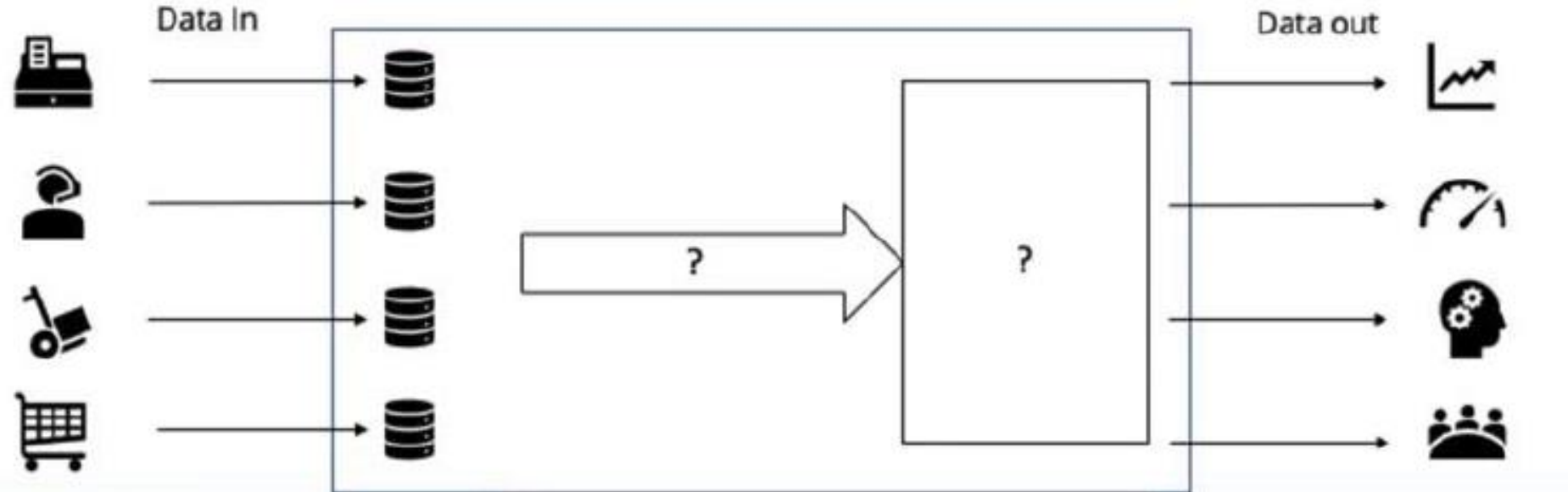
- Excellent for operations
- No redundancy, high integrity

Operational Databases

- Too slow for analytics, too many joins
- Too hard to understand

Solution: Create two processing modes

Create a system for them to co-exist



OLTP

Online **transactional** processing

OLAP

Online **analytical** processing

Data Warehouse is a system (including processes, technologies & data representations) that enables us to support analytical processes

What is a Data Warehouse?

Tech Perspective: DWH Definition 1

A data warehouse is a copy of transaction data specifically structured for query and analysis.

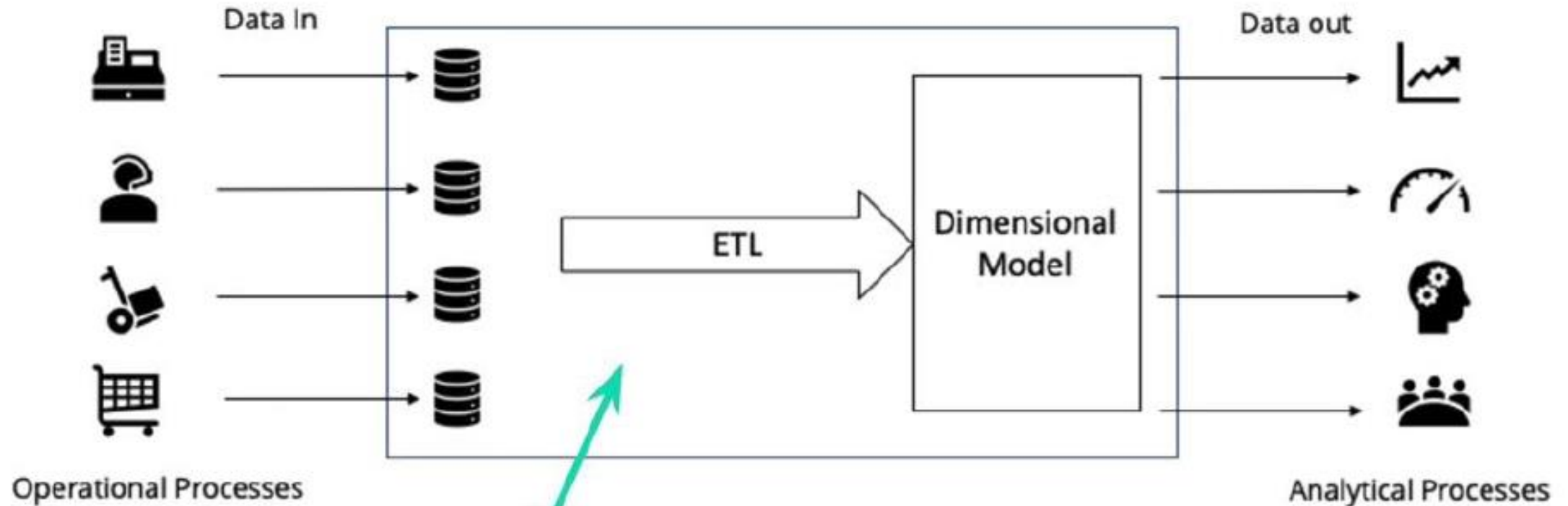
REF: KIMBALL

Tech Perspective: DWH Definition 2

A data warehouse is a system that **retrieves** and **consolidates** data **periodically** from the source systems into a **dimensional or normalized** data store. It usually **keeps years of history** and is **queried for business intelligence** or other **analytical activities**. It is typically **updated in batches**, not every time a transaction happens in the source system.

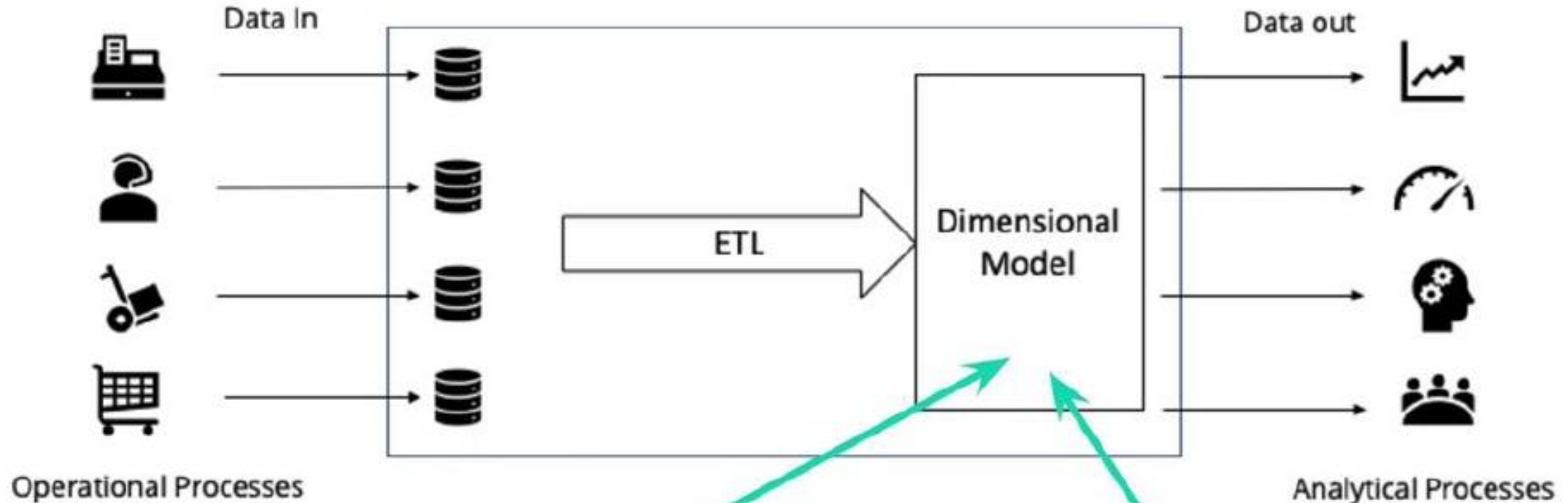
REF: INMON

DWH: Tech Perspective



Extract the data and from the source systems used for operations, **Transform** the data and **Load** it into a dimensional model

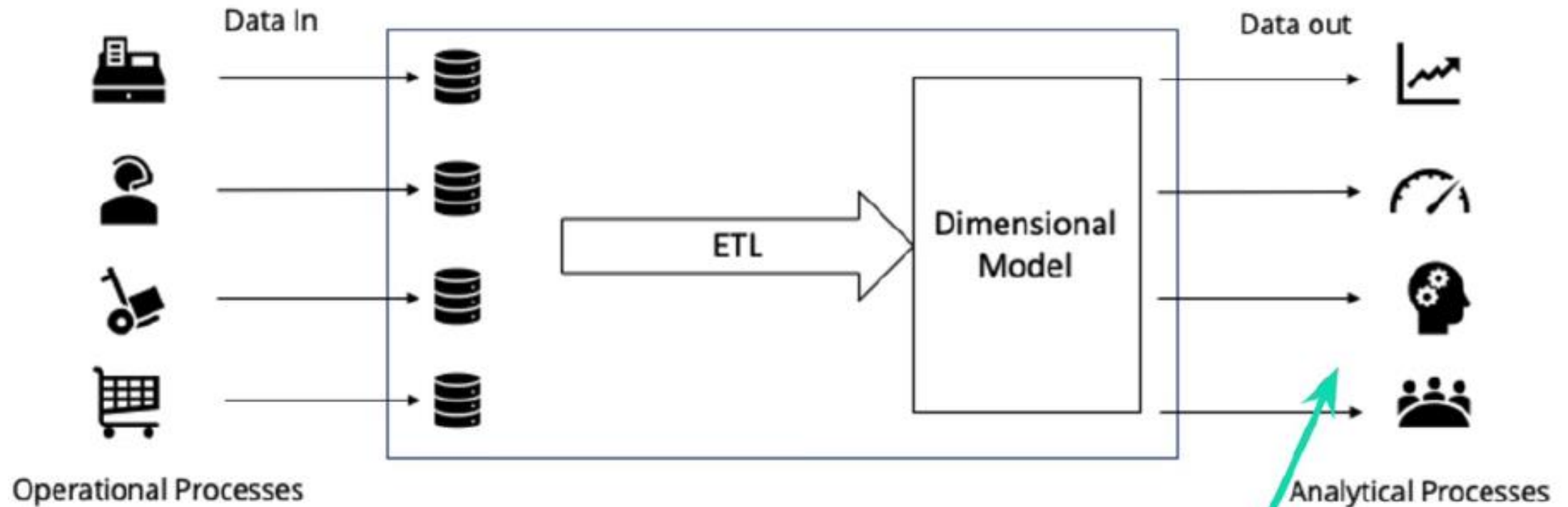
DWH: Tech Perspective



The **dimensional model** is designed to a) make it **easy** for business users to work with the data, b) improve analytical **queries performance**

The **technologies** used for storing dimensional models are **different** than traditional technologies

DWH: Tech Perspective



*Business-user-facing application are needed, with clear visuals, aka **Business Intelligence (BI) apps***

Data Warehouse Goals

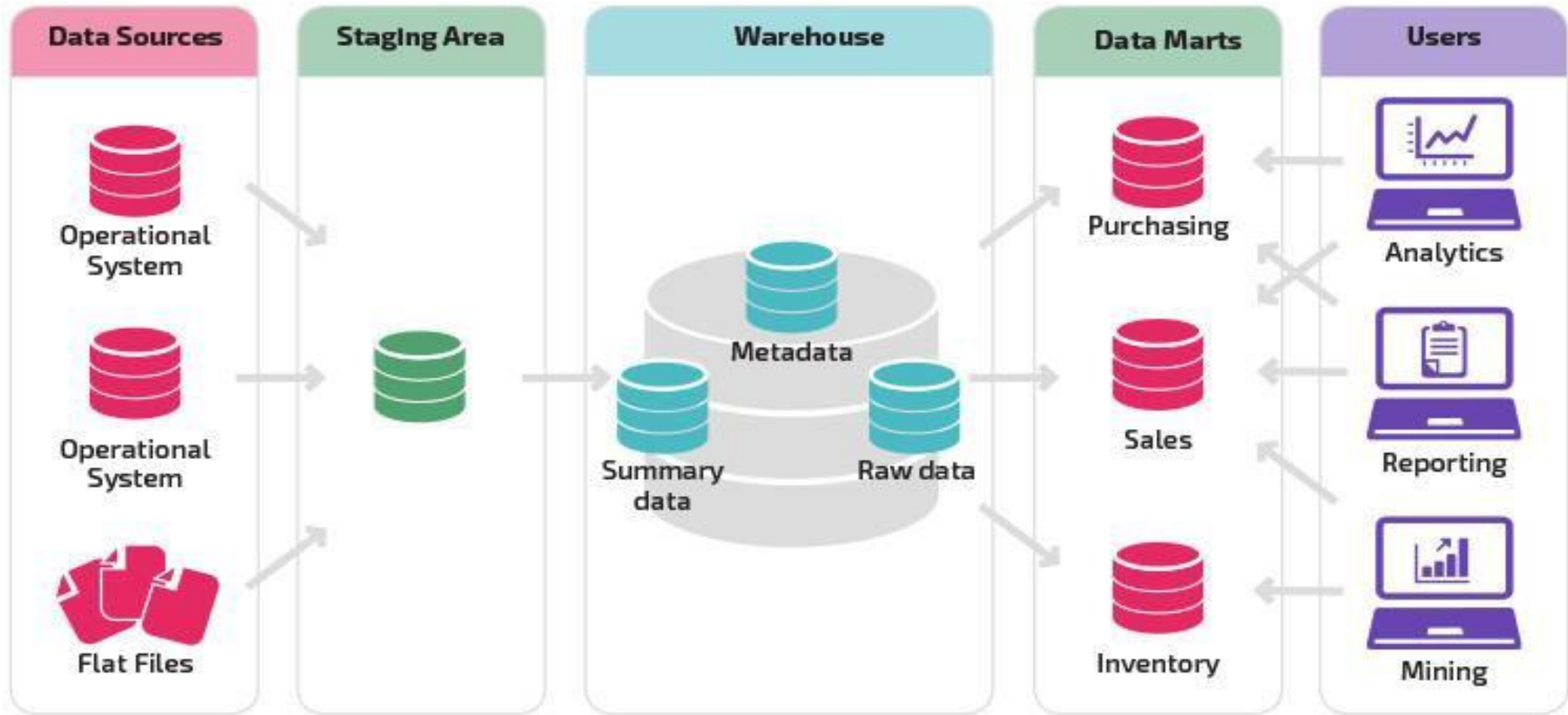
- Simple to understand
- Enterprise level information Consolidation
- Adaptive and resilient to change
- Handles new questions well
- Secure
- Improved business decision making



Data Marts VS Data Warehouse

- **Data Warehouse** is a large centralized repository of data that contains information from many sources within an organization. The collated data is used to guide business decisions through analysis, reporting, and data mining tools.
- **Data Mart** is a subset of a data warehouse oriented to a specific business line. Data marts contain repositories of summarized data collected for analysis on a specific section or unit within an organization, for example, the sales department.

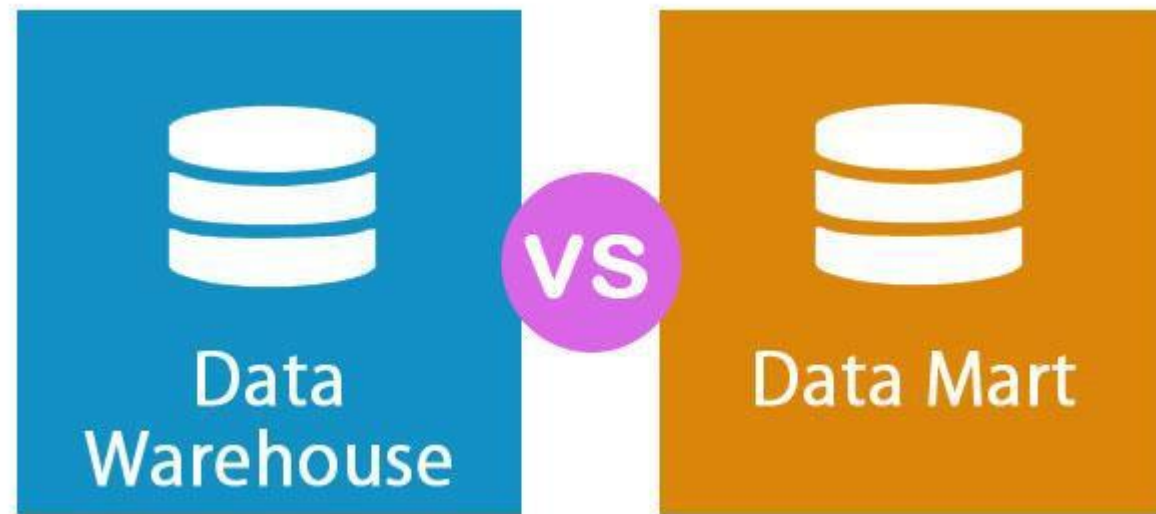
Data Marts VS Data Warehouse



<https://panoply.io/data-warehouse-guide/data-mart-vs-data-warehouse/>

Inmon vs. Kimball (Data Warehouse Structures)

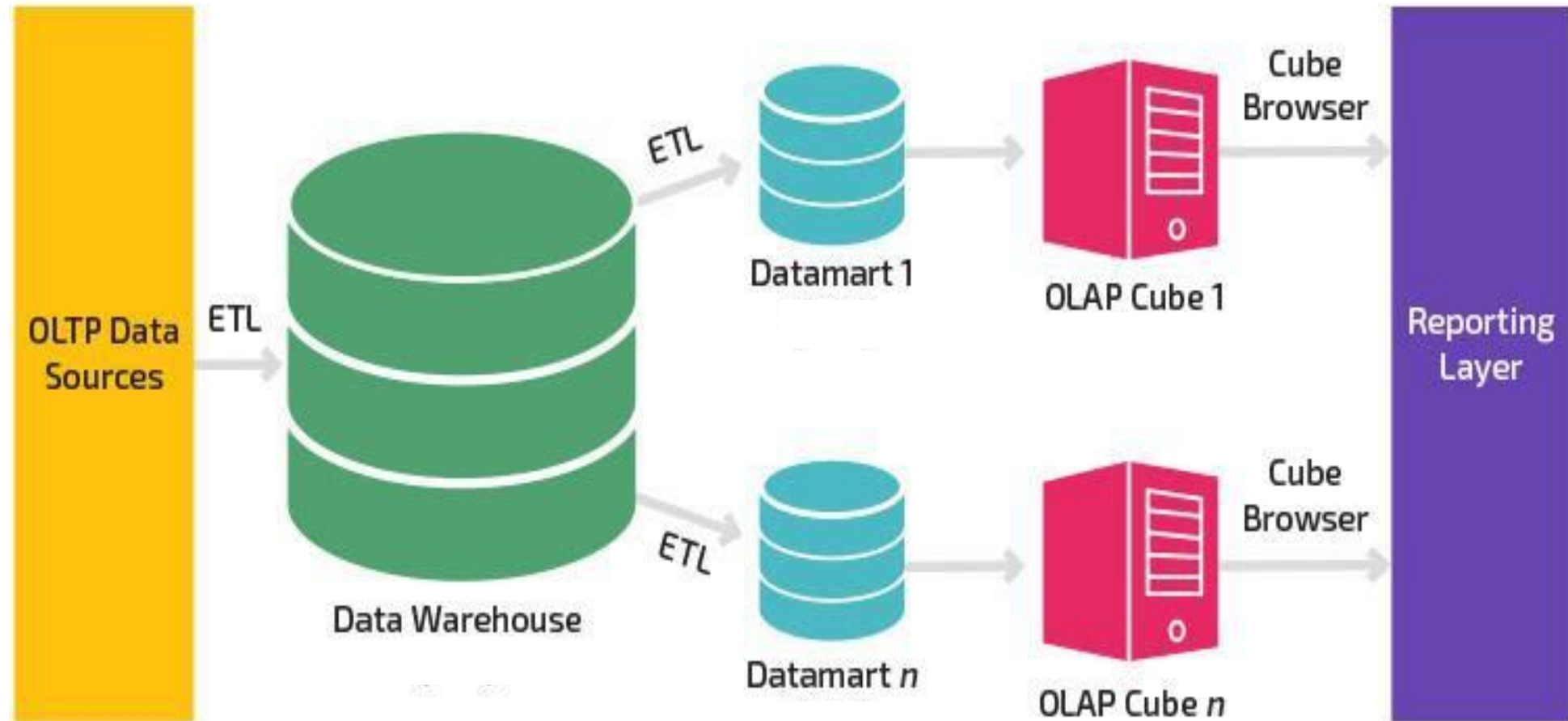
- Two data warehouse pioneers, Bill Inmon and Ralph Kimball differ in their views on how data warehouses should be designed from the organization's perspective.



Bill Inmon's approach

- Favors a top-down design in which the data warehouse is the centralized data repository and the most important component of an organization's data systems.
- The Inmon approach first builds the centralized corporate data model, and the data warehouse is seen as the physical representation of this model.
- Dimensional data marts related to specific business lines can be created from the data warehouse when they are needed.

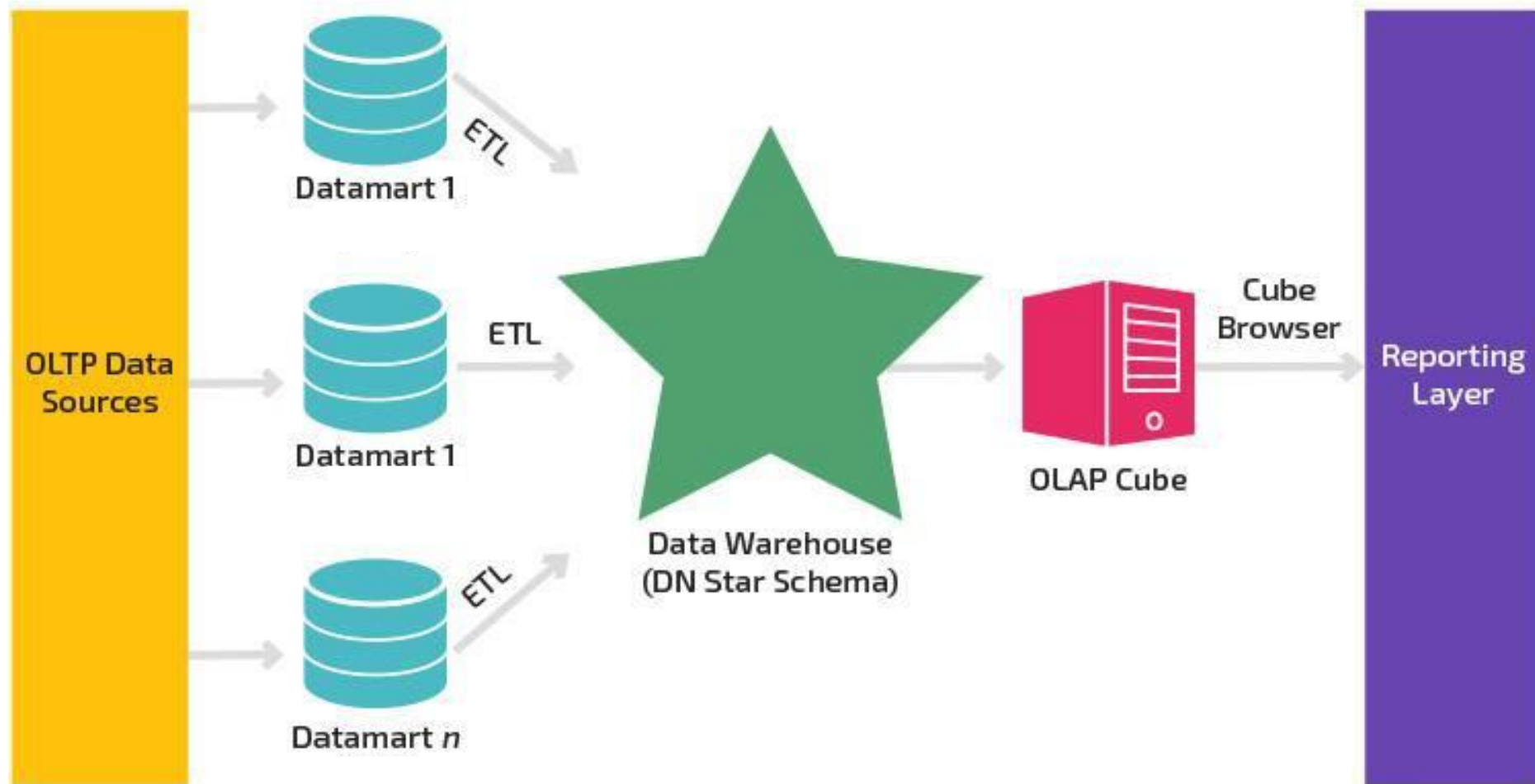
Inmon Model



Ralph Kimball's approach

- **Ralph Kimball's** data warehouse design starts with the most important business processes. In this approach, an organization creates data marts that aggregate relevant data around subject-specific areas.
- The data warehouse is the combination of the organization's individual data marts.
- Data warehouse is the conglomerate of a number of data marts. This is in contrast to Inmon's approach, which creates data marts based on information in the warehouse.

Kimball Model



Data Marts – Use Cases

- Marketing analysis and reporting favor a data mart approach because these activities are typically performed in a specialized business unit, and do not require enterprise-wide data.
- A financial analyst can use a finance data mart to carry out financial reporting.



Data Warehouse – Use Cases

- A company considering an expansion needs to incorporate data from a variety of data sources across the organization to come to an informed decision. This requires a data warehouse that aggregates data from sales, marketing, store management, customer loyalty, supply chains, etc.
- Many factors drive profitability at an insurance company. An insurance company reporting on its profits needs a centralized data warehouse to combine information from its claims department, sales, customer demographics, investments, and other areas.



Thanks