Data Warehousing

Lecture 6 Metadata and Data Marts

CITS3401
CITS5504

Sirui Li

Computer Science and Software Engineering

School of Physics Mathematics and Computing

Acknowledgement: Some lecture slides are based on online sources.

Outline

- **Metadata**
  - ➢ Introduction to Metadata and Metadata Examples
  - ➢ Data Warehouse and Metadata
  - ➢ Metadata Repository and Category
  - ➢ Challenges of Metadata Management
- **Data Mart**
  - ➢ Introduction to Data Mart
  - ➢ Types of Data Mart
  - ➢ Inmon and Kimball’s Data Warehouses
  - ➢ Implementing Data Mart
  - ➢ Advantage/ Disadvantage of Data Mart
  - ➢ Use Cases Example

Metadata

What is metadata?

Metadata is simply **data about data**.

- leads us to detail and represent data
- helps organise, find and understand data.
- index of a book serves as a ‘metadata’ for the contents in the book

Relation between Metadata and Data warehouse

- Metadata is the *road-map* to a data warehouse
- Metadata in a data warehouse defines the warehouse *objects*.
- Metadata acts as a *directory* to locate the *contents* of a data warehouse

Typical metadata elements

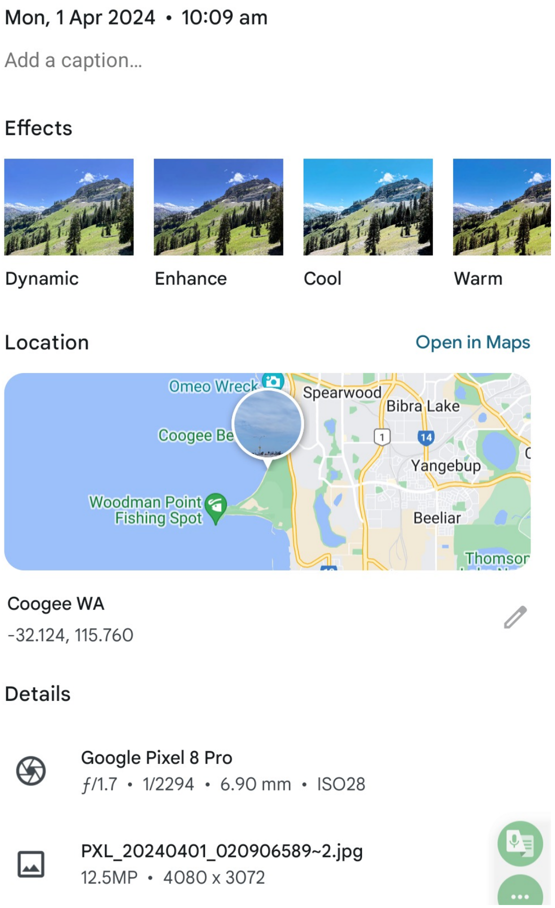
- Title and description
- Tags and categories
  - tags of a picture, a blog
- Who created and when
  - Username and Posting time of a tweet
- Who last modified and when
  - File system
  - Collaborative editing
- Who can access or update
  - Database Management (Roles)

Metadata Examples: Photo

Every time you take a photo with today's cameras a bunch of metadata is gathered and saved

- date and time
- filename
- camera settings
- geolocation

A photo



Mon, 1 Apr 2024 • 10:09 am

Add a caption...

Effects

Dynamic Enhance Cool Warm

Location Open in Maps

Coogee WA

-32.124, 115.760

Details

Google Pixel 8 Pro

ƒ/1.7 • 1/2294 • 6.90 mm • ISO28

PXL_20240401_020906589~2.jpg

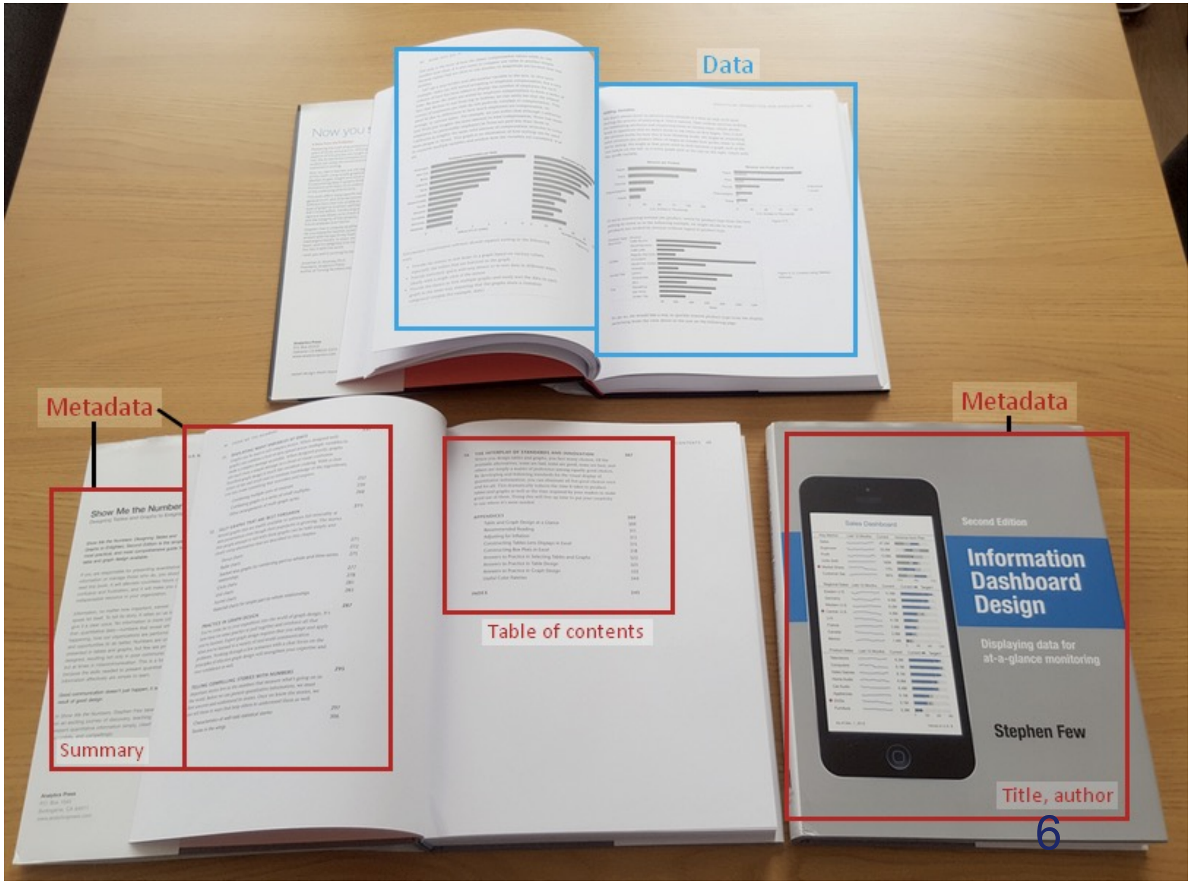
12.5MP • 4080 x 3072

Metadata Examples: Book

Each book has a number of standard metadata on the covers and inside. This includes:

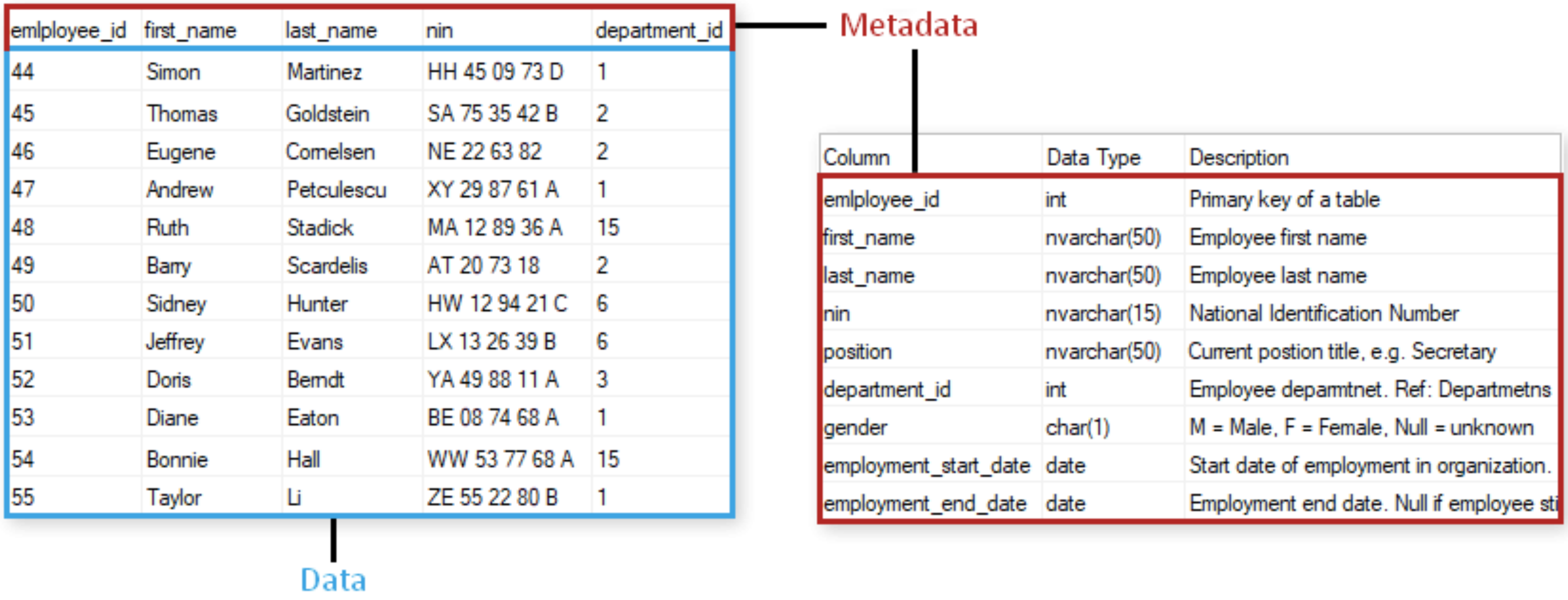
- A title
- author name
- publisher and copyright details
- description on a back
- table of contents
- index
- page numbers

A book



Metadata Examples: Database Table

Relational database



emlployee_id	first_name	last_name	nin	department_id
44	Simon	Martinez	HH 45 09 73 D	1
45	Thomas	Goldstein	SA 75 35 42 B	2
46	Eugene	Cornelsen	NE 22 63 82	2
47	Andrew	Petculescu	XY 29 87 61 A	1
48	Ruth	Stadick	MA 12 89 36 A	15
49	Barry	Scardelis	AT 20 73 18	2
50	Sidney	Hunter	HW 12 94 21 C	6
51	Jeffrey	Evans	LX 13 26 39 B	6
52	Doris	Berndt	YA 49 88 11 A	3
53	Diane	Eaton	BE 08 74 68 A	1
54	Bonnie	Hall	WW 53 77 68 A	15
55	Taylor	Li	ZE 55 22 80 B	1

Column	Data Type	Description
emlployee_id	int	Primary key of a table
first_name	nvarchar(50)	Employee first name
last_name	nvarchar(50)	Employee last name
nin	nvarchar(15)	National Identification Number
position	nvarchar(50)	Current postion title, e.g. Secretary
department_id	int	Employee deparmtnet. Ref: Departmetns
gender	char(1)	M = Male, F = Female, Null = unknown
employment_start_date	date	Start date of employment in organization.
employment_end_date	date	Employment end date. Null if employee sti

Metadata Examples: Database Table

Relational databases (most common type of database) store and provide access not only data but also metadata in a structure called **data dictionary or system catalog**. It holds information about:

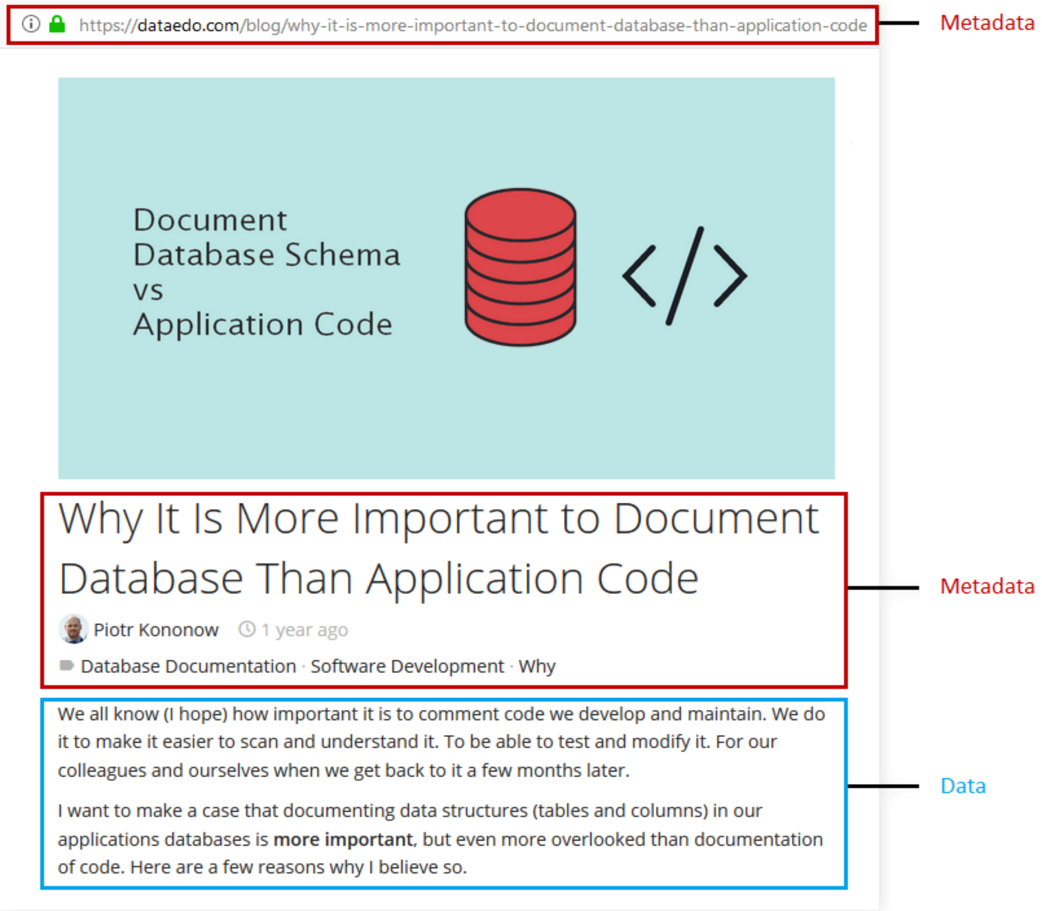
- tables
- columns,
- data types,
- constraints
- table relationships,
- and many more

Metadata Examples: Webpage

Every blog post has standard metadata fields that are usually at before first paragraph. This includes:

- A title
- Author id/ name
- published time
- Category
- Tags

A blog post



https://dataedo.com/blog/why-it-is-more-important-to-document-database-than-application-code

Document
Database Schema
vs
Application Code

Why It Is More Important to Document Database Than Application Code

Piotr Kononow · 1 year ago

Database Documentation · Software Development · Why

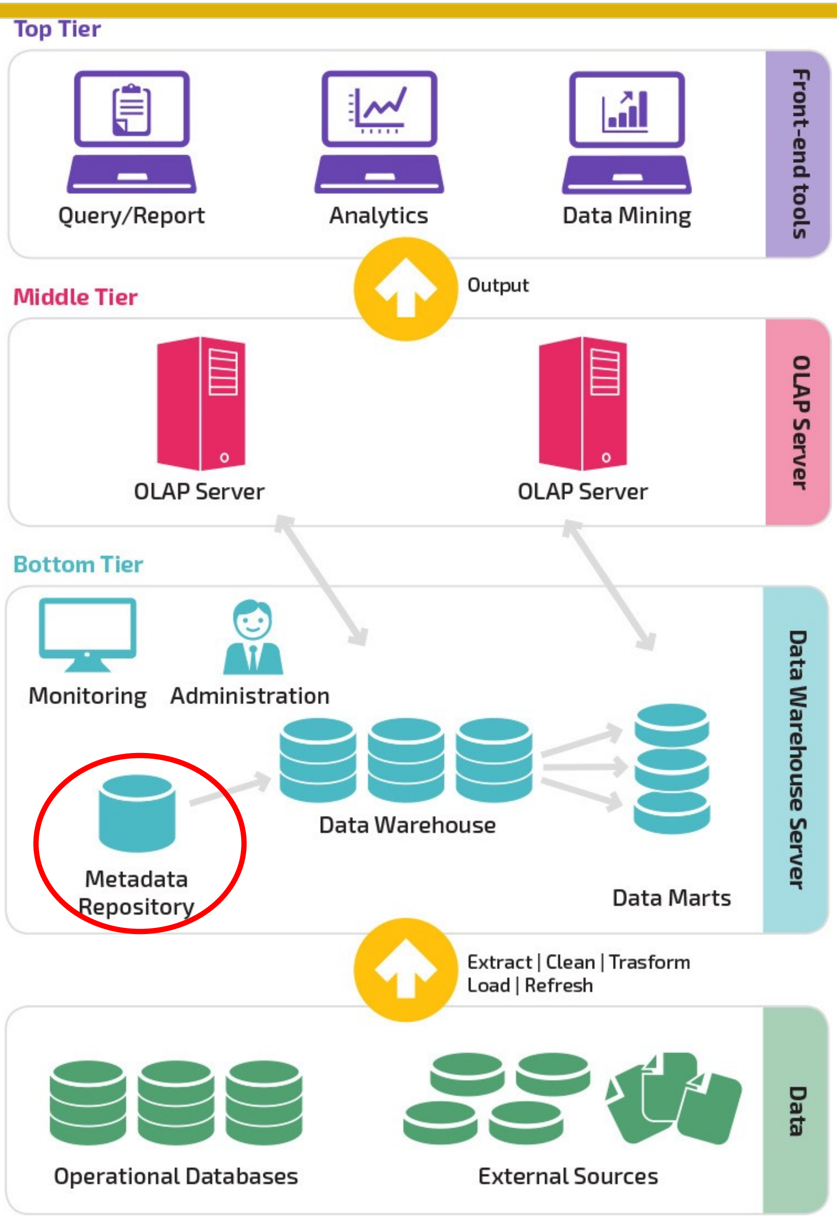
We all know (I hope) how important it is to comment code we develop and maintain. We do it to make it easier to scan and understand it. To be able to test and modify it. For our colleagues and ourselves when we get back to it a few months later.

I want to make a case that documenting data structures (tables and columns) in our applications databases is **more important**, but even more overlooked than documentation of code. Here are a few reasons why I believe so.

Outline

- **Metadata**
  - Introduction to Metadata and Metadata Examples
  - **Data Warehouse and Metadata**
  - Metadata Repository and Category
  - Challenges of Metadata Management
- **Data Mart**
  - Introduction to Data Mart
  - Types of Data Mart
  - Inmon and Kimball’s Data Warehouses
  - Implementing Data Mart
  - Advantage/ Disadvantage of Data Mart
  - Use Cases Example

DW Architecture and Metadata



Data Warehouse Metadata

A Data Warehouse has specific metadata requirements.

Metadata that describes tables typically includes:

- Physical and Logical Name
- Type: Fact, Dimension
- Role: Legacy, OLTP, Stage
- DBMS: DB2, Informix, MS SQL Server, Oracle,
- Definition/Description
- Location

Metadata describes columns within tables:

- Physical and Logical Name
- Order in Table
- Datatype
- Length
- Default Value
- Nullable/Required
- Edit Rules

Metadata in Data Warehousing

In general, we have seven kinds of metadata:

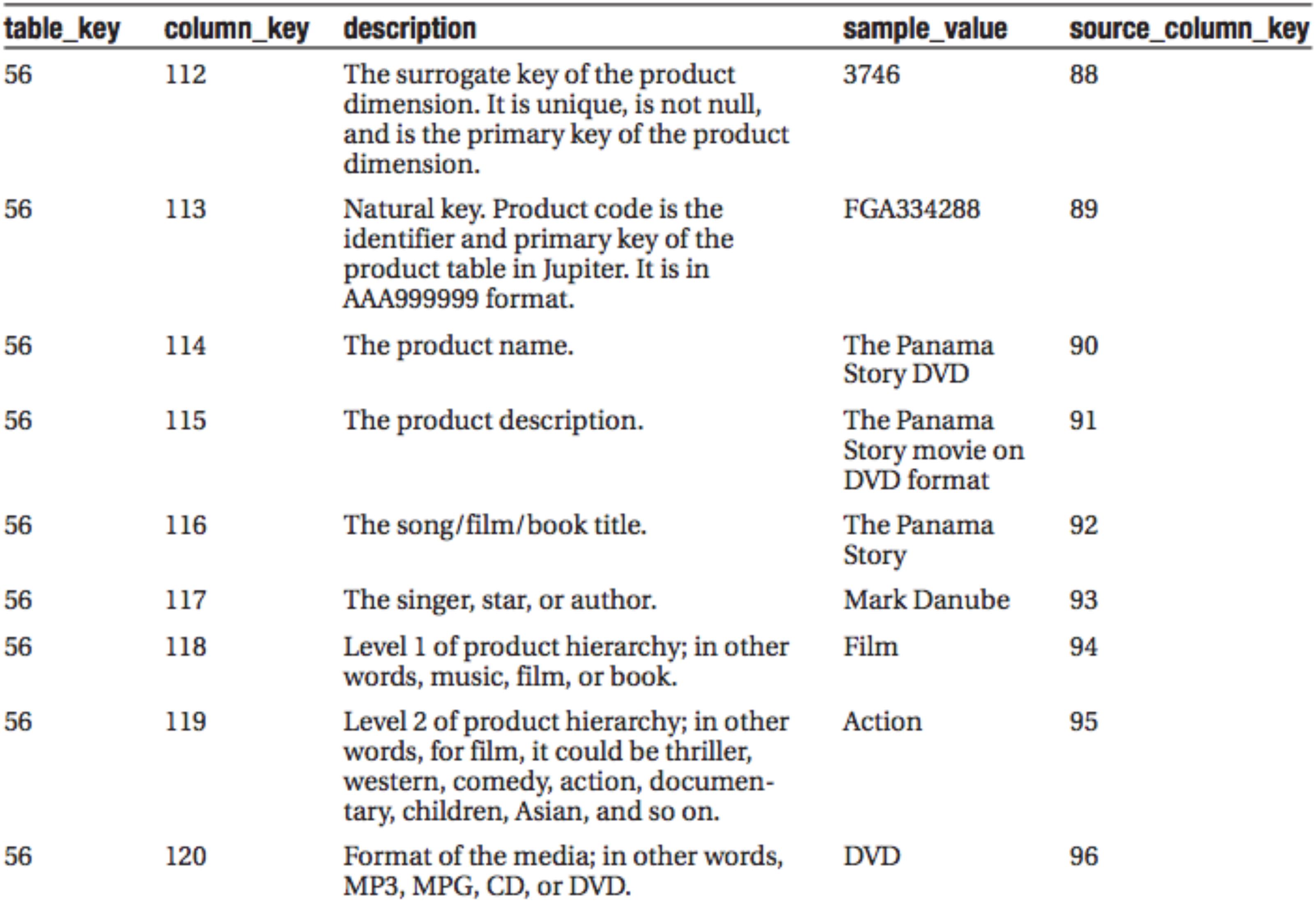
- **Data definition and mapping metadata** contains the meaning of each fact and dimension column and where the data is coming from.
- **Data structure metadata** describes the structure of the tables in each data store.
- **Source system metadata** describes the data structure of source system databases.
- **ETL process metadata** describes each data flow in the ETL processes.
- **Data quality metadata** describes data quality rules, their risk levels, and their actions.
- **Audit metadata** contains a record of processes and activities in the data warehouse.
- **Usage metadata** contains an event log of application usage**.**

Data Definition and Mapping Metadata

- **Data definition metadata** is a list of all columns from every table in the DDS, ODS, and NDS along with their meanings and sample values. Instead of mentioning the data store names, table names, and column names, data definition metadata uses the **table key** and **column key** defined in the *data structure metadata*.

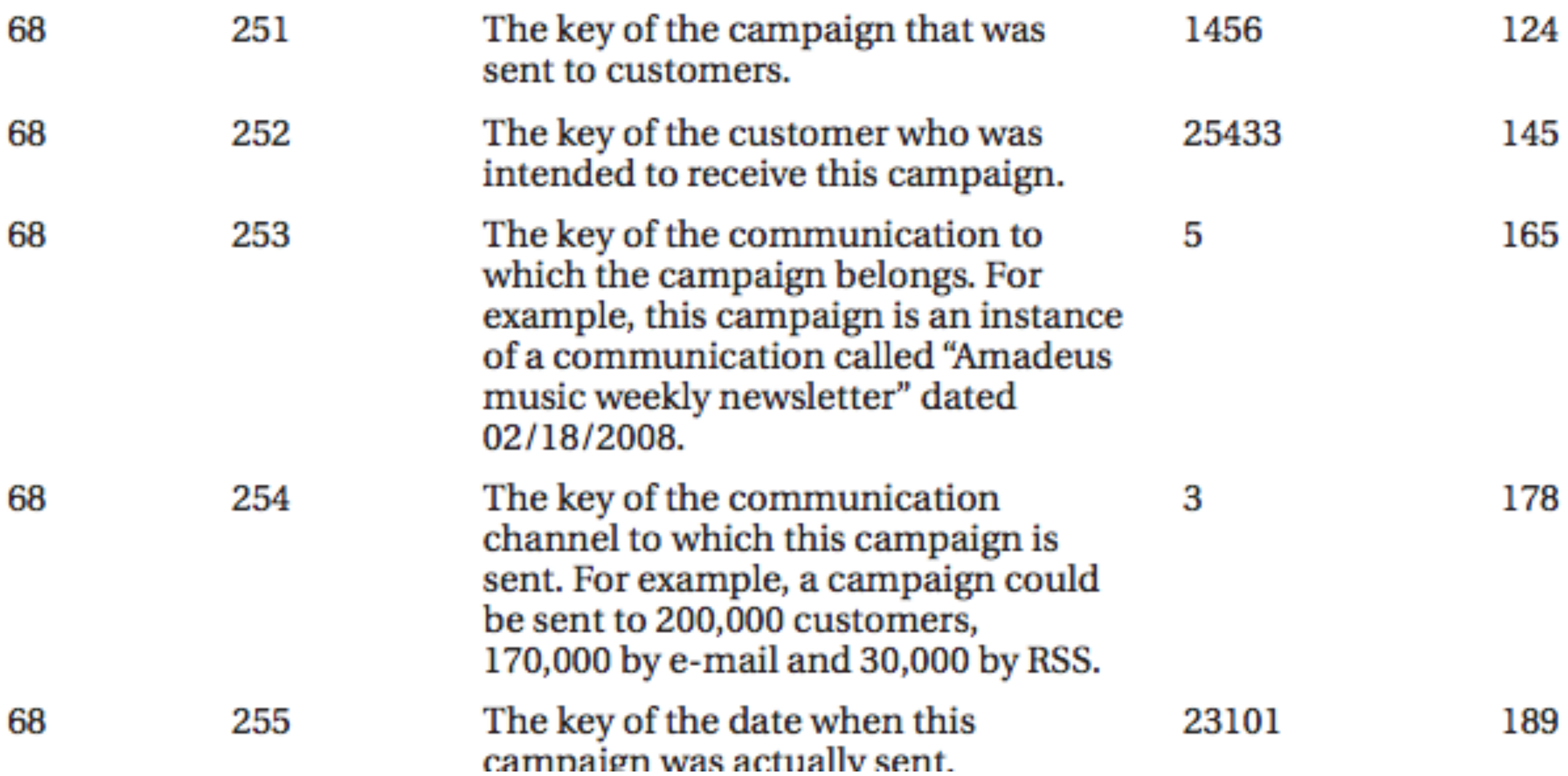
- **Mapping metadata** describes where each piece of data comes from in the source system. Mapping metadata is also known as **data lineage metadata**.

Data Definition Metadata Examples



table_key	column_key	description	sample_value	source_column_key
56	112	The surrogate key of the product dimension. It is unique, is not null, and is the primary key of the product dimension.	3746	88
56	113	Natural key. Product code is the identifier and primary key of the product table in Jupiter. It is in AAA999999 format.	FGA334288	89
56	114	The product name.	The Panama Story DVD	90
56	115	The product description.	The Panama Story movie on DVD format	91
56	116	The song/film/book title.	The Panama Story	92
56	117	The singer, star, or author.	Mark Danube	93
56	118	Level 1 of product hierarchy; in other words, music, film, or book.	Film	94
56	119	Level 2 of product hierarchy; in other words, for film, it could be thriller, western, comedy, action, documentary, children, Asian, and so on.	Action	95
56	120	Format of the media; in other words, MP3, MPG, CD, or DVD.	DVD	96

Data Definition Metadata Examples

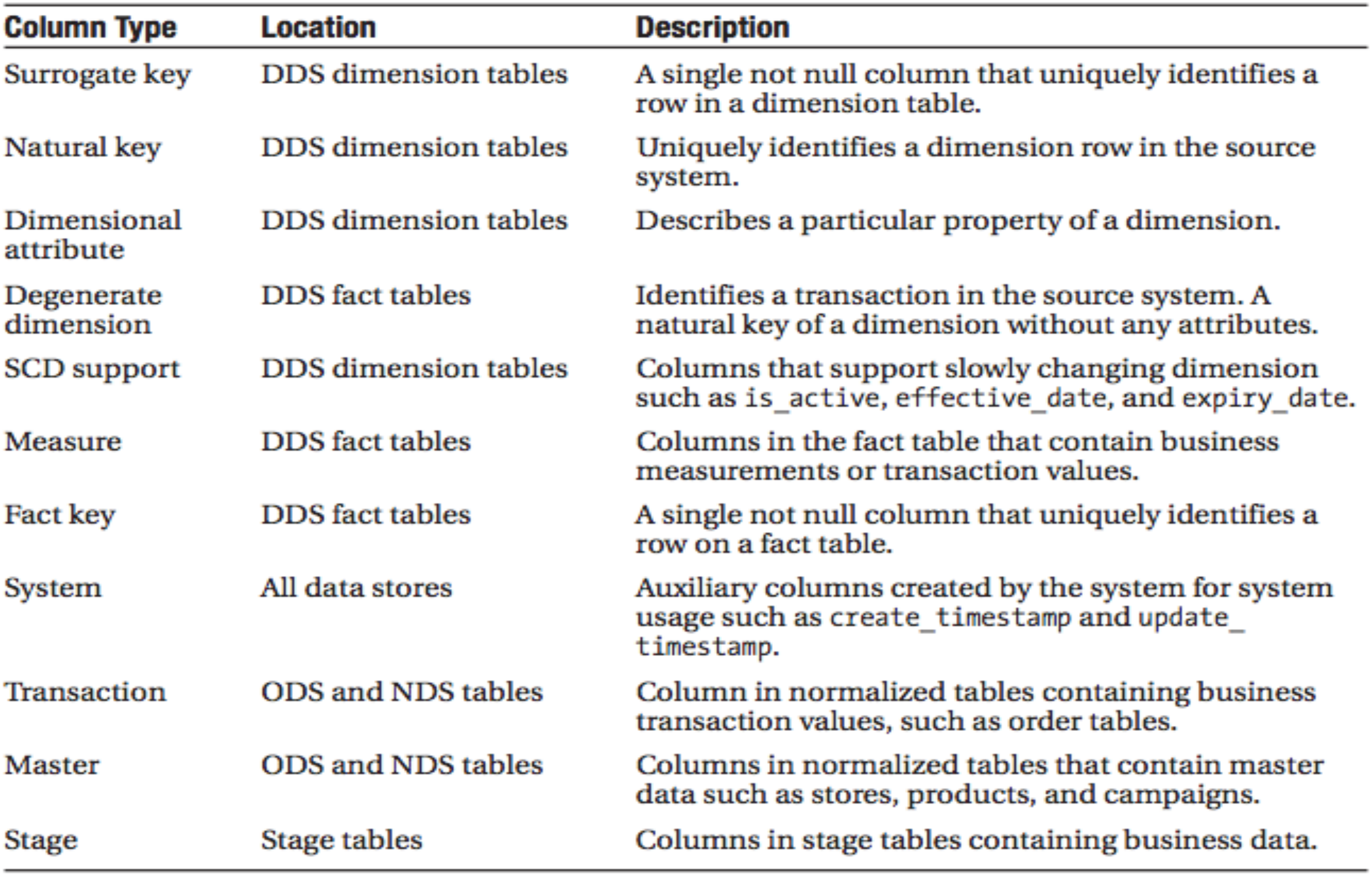


68	251	The key of the campaign that was sent to customers.	1456	124
68	252	The key of the customer who was intended to receive this campaign.	25433	145
68	253	The key of the communication to which the campaign belongs. For example, this campaign is an instance of a communication called "Amadeus music weekly newsletter" dated 02/18/2008.	5	165
68	254	The key of the communication channel to which this campaign is sent. For example, a campaign could be sent to 200,000 customers, 170,000 by e-mail and 30,000 by RSS.	3	178
68	255	The key of the date when this campaign was actually sent.	23101	189

This and the previous slide contain an example of **data definition and mapping metadata**. Table Key **56** refers to the **product table**, and Table Key **68** refers to **the campaign result fact table**. The **source column** is located on the database one step earlier in the process

Data Definition Metadata Examples

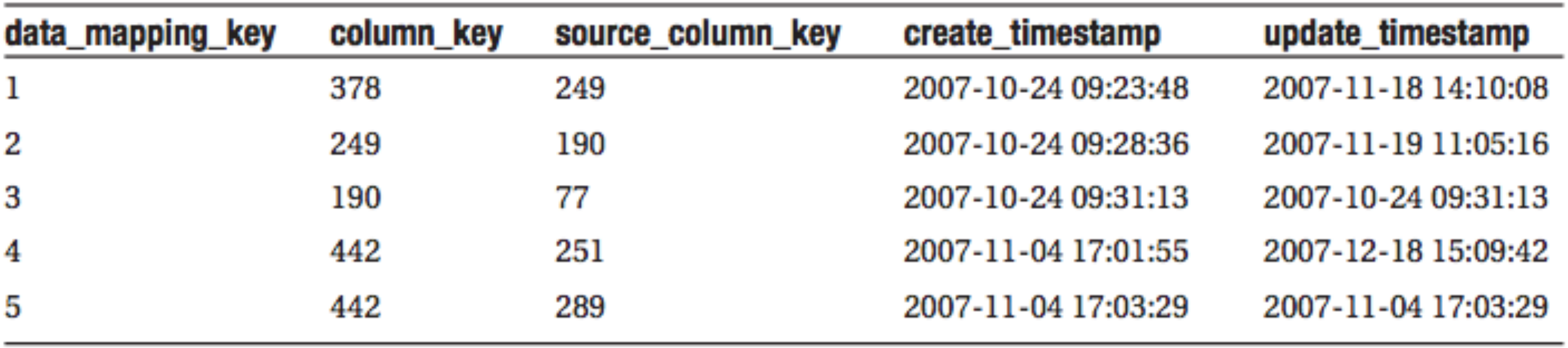
Column Type Metadata



Column Type	Location	Description
Surrogate key	DDS dimension tables	A single not null column that uniquely identifies a row in a dimension table.
Natural key	DDS dimension tables	Uniquely identifies a dimension row in the source system.
Dimensional attribute	DDS dimension tables	Describes a particular property of a dimension.
Degenerate dimension	DDS fact tables	Identifies a transaction in the source system. A natural key of a dimension without any attributes.
SCD support	DDS dimension tables	Columns that support slowly changing dimension such as `is_active`, `effective_date`, and `expiry_date`.
Measure	DDS fact tables	Columns in the fact table that contain business measurements or transaction values.
Fact key	DDS fact tables	A single not null column that uniquely identifies a row on a fact table.
System	All data stores	Auxiliary columns created by the system for system usage such as `create_timestamp` and `update_timestamp`.
Transaction	ODS and NDS tables	Column in normalized tables containing business transaction values, such as order tables.
Master	ODS and NDS tables	Columns in normalized tables that contain master data such as stores, products, and campaigns.
Stage	Stage tables	Columns in stage tables containing business data.

Data Definition Metadata Examples

Data Mapping Table for the Data Flow



data_mapping_key	column_key	source_column_key	create_timestamp	update_timestamp
1	378	249	2007-10-24 09:23:48	2007-11-18 14:10:08
2	249	190	2007-10-24 09:28:36	2007-11-19 11:05:16
3	190	77	2007-10-24 09:31:13	2007-10-24 09:31:13
4	442	251	2007-11-04 17:01:55	2007-12-18 15:09:42
5	442	289	2007-11-04 17:03:29	2007-11-04 17:03:29

Outline

- **Metadata**
  - ✓ Introduction to Metadata and Metadata Examples
  - ✓ Data Warehouse and Metadata
  - ✓ **Metadata Repository and Category**
  - ✓ Challenges of Metadata Management
- **Data Mart**
  - ✓ Introduction to Data Mart
  - ✓ Types of Data Mart
  - ✓ Inmon and Kimball’s Data Warehouses
  - ✓ Implementing Data Mart
  - ✓ Advantage/ Disadvantage of Data Mart
  - ✓ Use Cases Example

Metadata Repository

- Metadata is collectively organised in a catalogue called metadata repository.
- It stores **descriptive information** about data model used to store and share.

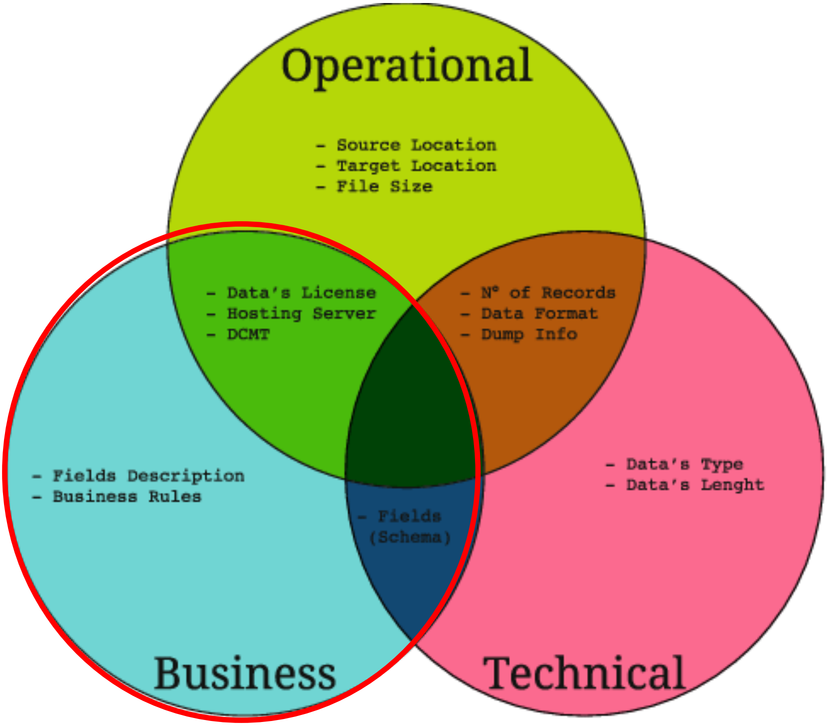
Each Data Warehouse has one or multiple repositories that hold the following metadata in them:

- **Definition** of the structure of the data warehouse
- **Structure** of the **tables** involved in the warehouse
- **Description** of the dataflow through the warehouse
- **Outcomes** of all querying process e.g., indexing, ETL, etc.
- Information about **who** accesses objects and when
- includes **aggregation**, **summarising**, etc.

Metadata Categories

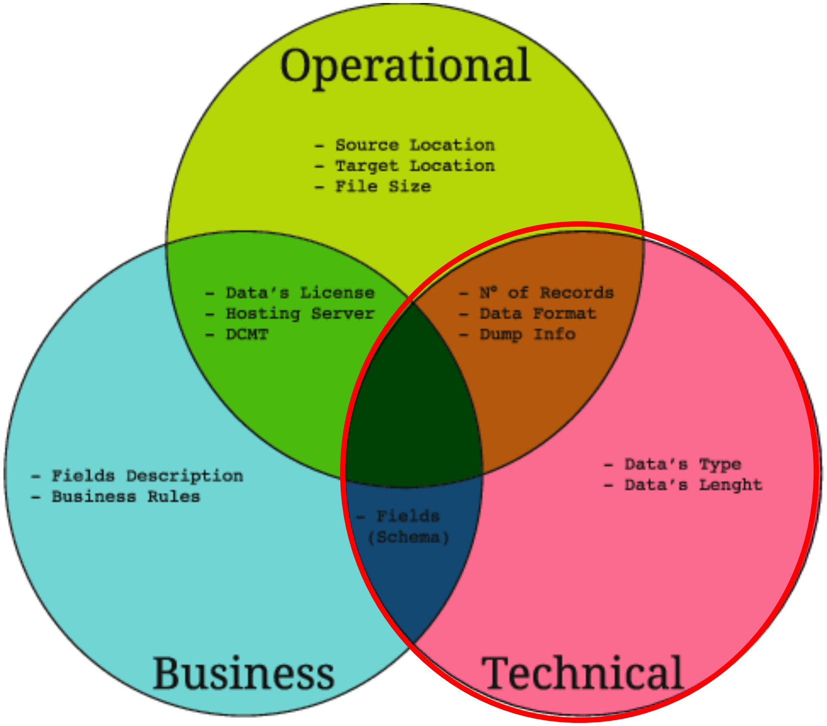
Broadly categorised into three categories: Business metadata, Operational Metadata, and Technical Metadata

- **1) Business Metadata** includes data ownership information, business definition, and changing policies.
  - ❑ Description of information from the business perspective (e.g., weekly sales, or budget variance reports)
  - ❑ Contains high-level definitions of all fields present in the data warehouse, information about cubes, aggregates, Data Marts.
  - ❑ addressed to and used by the data warehouse users
  - ❑ report querying authors, cubes creators, data managers, testers, analysts



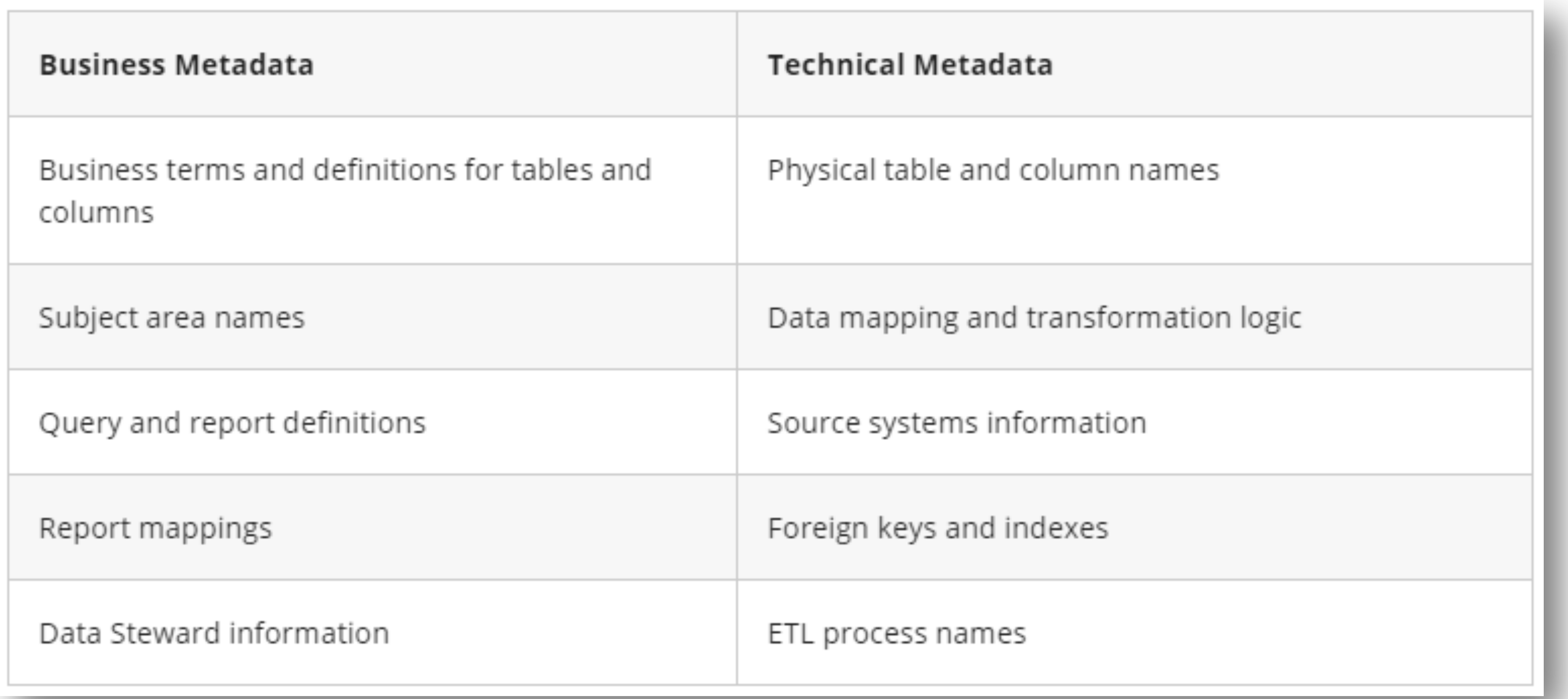
Metadata Categories

- **2) Technical Metadata** includes structural information such as primary and foreign key attributes and indices, data types and values.
  - ❑ gives information about the **structure** of data, where it **resides**, and other **technical** details related to finding data in its native database.
  - ❑ mainly used by **software tools** to understand and process data
  - ❑ stores data **mapping** and **transformations** from source systems to the data warehouse
  - ❑ illustrates information related to **system functions** and **metadata**



Metadata Categories

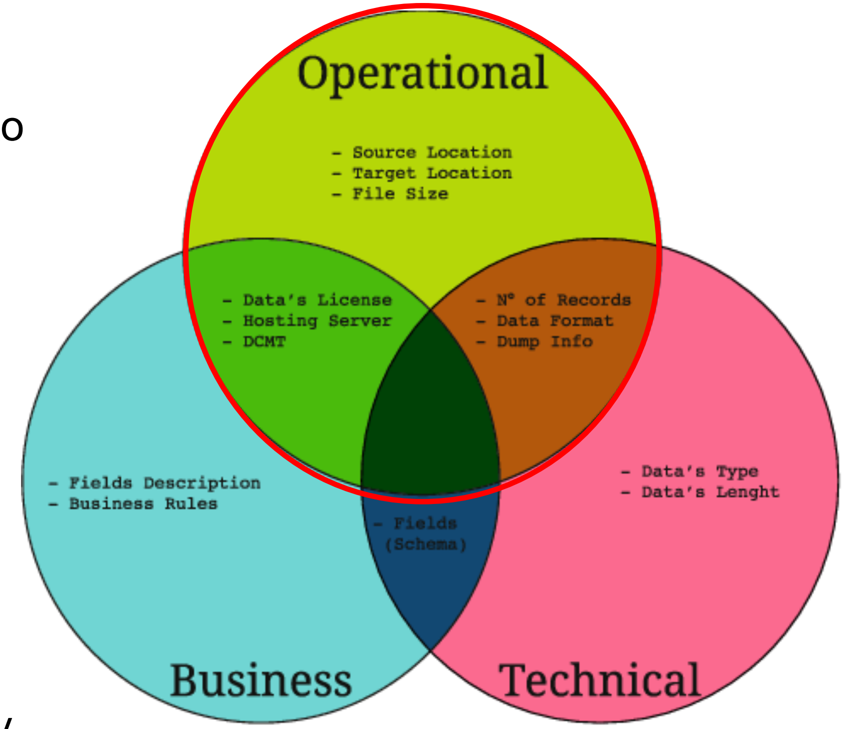
Difference between Business and Technical Metadata



Business Metadata	Technical Metadata
Business terms and definitions for tables and columns	Physical table and column names
Subject area names	Data mapping and transformation logic
Query and report definitions	Source systems information
Report mappings	Foreign keys and indexes
Data Steward information	ETL process names

Metadata Categories

- **3) Operational Metadata** offers a connection between the metadata repository and the data warehouse
  - ❑ Allows adding physical database columns to the data warehouse tables
  - ❑ includes currency of data and data lineage and enables easier use for **business** and **technical** consumers.
    - ➢ Currency of data means whether the data is active, archived, or purged
    - ➢ Lineage of data means the history of data migrated and transformation applied on it.
  - ❑ **Benefits** of operational metadata
    - ❑ referenced at a row level of granularity in DW (unlike in meta data repository)
    - ❑ provides a detailed row level explanation of actual information content



Application of Metadata in DW

Data Warehouse users can **use metadata** in a variety of situations to build, maintain and manage the system

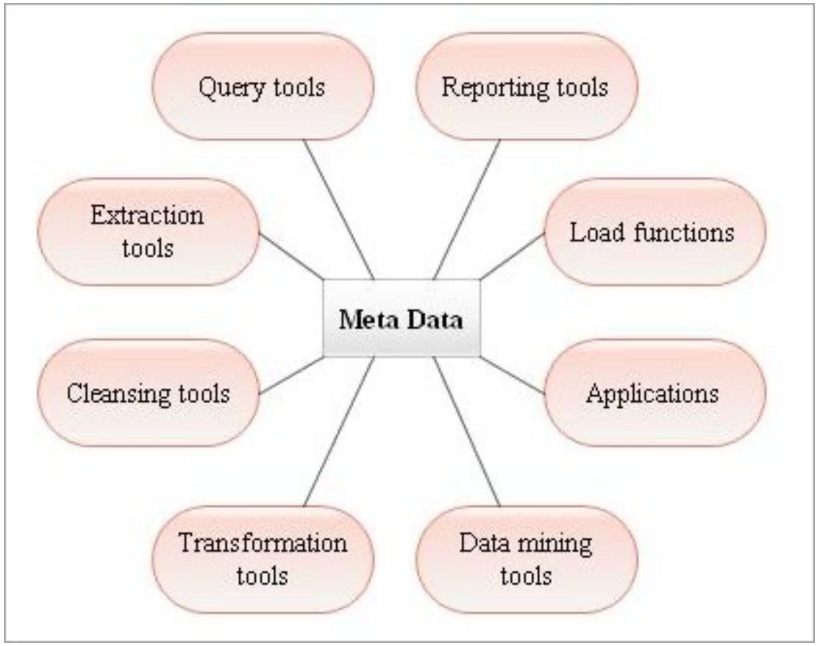
- **Finding Data**
  - ❑ Data in warehouses may reach terabyte levels
  - ❑ Metadata serves as a roadmap during the development of a DW, making the process of finding a relevant object considerably easier
  - ❑ Search data is faster due to the small size of metadata

- **Using Data**
  - ❑ helps utilise large sums of data present in the data warehouse without using the actual dataset.
  - ❑ gives information about the retrieval, structure, terminology, and regulations governing the data warehouse

Roles of Metadata

Metadata has a very important **role** in a **data warehouse:**

- helps the decision support system to **locate** the **contents** of the data warehouse
- helps in **decision support system** for mapping of data when data is **transformed** from **operational** environment to **data warehouse** environment
- used for **query tools**
- used in **extraction**, **cleansing, reporting**, and **transformation** tools.
- plays an important role in **loading functions**



DVC by iterative.ai

Outline

- **Metadata**
  - ✓ Introduction to Metadata and Metadata Examples
  - ✓ Data Warehouse and Metadata
  - ✓ Metadata Repository and Category
  - ✓ **Challenges of Metadata Management**
- **Data Mart**
  - ✓ Introduction to Data Mart
  - ✓ Types of Data Mart
  - ✓ Inmon and Kimball’s Data Warehouses
  - ✓ Implementing Data Mart
  - ✓ Advantage/ Disadvantage of Data Mart
  - ✓ Use Cases Example

How can DW Metadata be managed?

Data warehousing metadata is best managed through a combination of concerned *individual*, *process* and *tools*:

- The **individual** side requires that people be trained in the importance and use of metadata.
- The **process** side incorporates metadata management into the data warehousing and business intelligence life cycle.
- Metadata can be managed through individual **tools**:
  - Metadata manager/repository
  - Metadata extract tools
  - Data modelling and ETL tools
  - BI Reporting tools

Challenges for Metadata Management

Consider the following questions:

- **How do you ensure** that you are exploiting the metadata you are collecting to the **fullest**, **possible extent**?

- **How do you make sure** that your metadata is **easily accessible** and **effectively used** across your organisation?

- **How do you ensure** that it is kept **up-to-date** so that new metadata about new data is incorporated?

Challenges for Metadata Management

- **Disparate information sources**
  - Wide **variety** of sources, in a big organisation, metadata is **scattered** across the organisation
  - Significant portion of every organisation's vital data resides **outside** of its databases
  - Hard to maintain **consistent** and **easy-to-understand** format
- **Enforcing business rules for metadata**
  - Creating a context of enforceable **business rules** around the metadata is an important aspect of maintaining data integrity and usability.
  - Data repositories do not help you understand the **relationships** around the data
  - Clarifying the **dependencies** associated with data is challenging
- **Effective communication**
  - Information about **how to use data** is hard to find or hard to use
  - There are no industry-wide accepted **standards**
  - Clear **communication** is vital to leveraging metadata

Outline

- **Metadata**
  - ✓ Introduction to Metadata and Metadata Examples
  - ✓ Data Warehouse and Metadata
  - ✓ Metadata Repository and Category
  - ✓ Challenges of Metadata Management
- **Data Mart**
  - ✓ **Introduction to Data Mart**
  - ✓ Types of Data Mart
  - ✓ Inmon and Kimball’s Data Warehouses
  - ✓ Implementing Data Mart
  - ✓ Advantage/ Disadvantage of Data Mart
  - ✓ Use Cases Example

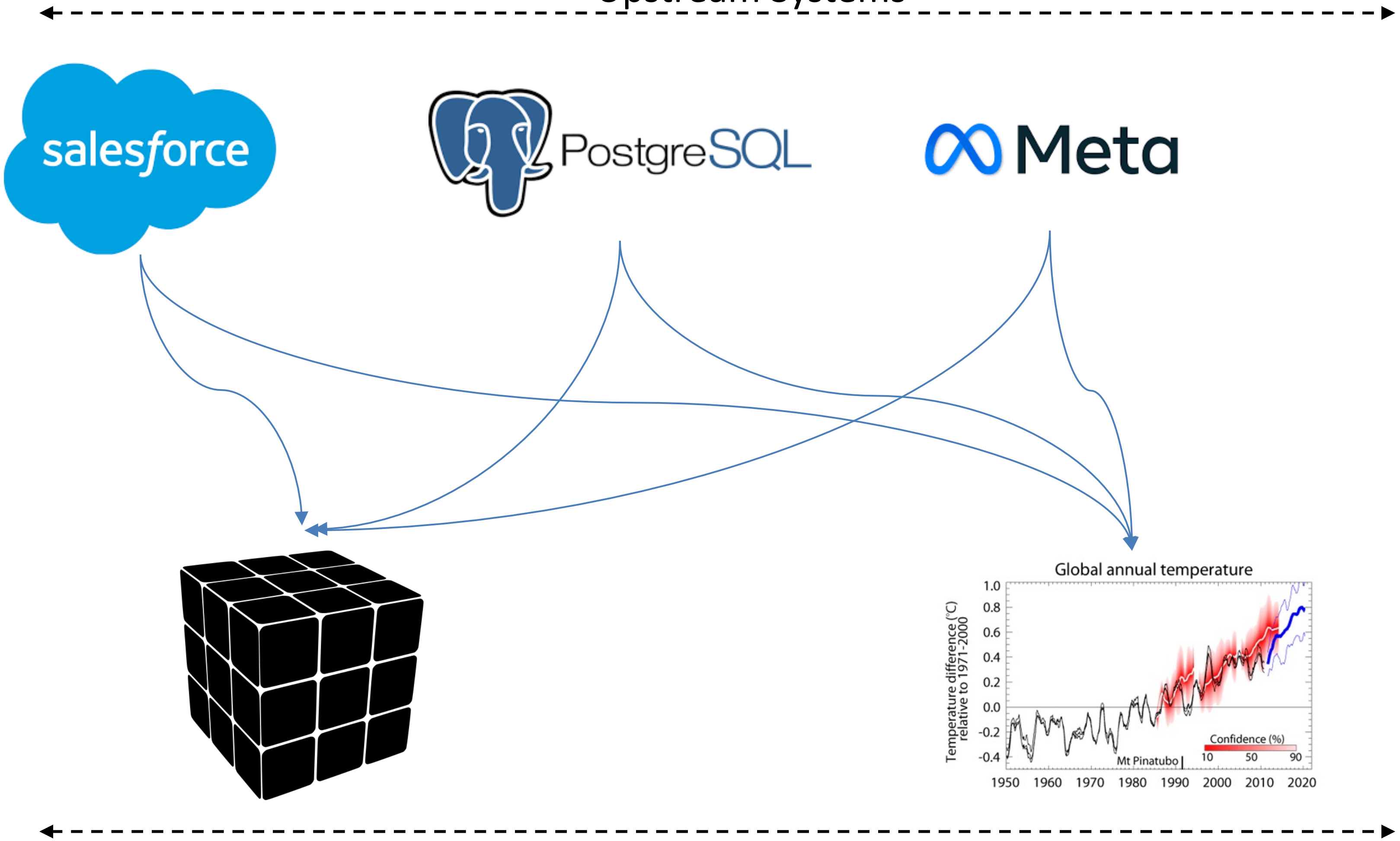
What is a Data Lake?

- A **data lake** is a storage platform for
  - for semi-structured, structured, unstructured, and binary data, at any scale, and
  - has the specific purpose of supporting the execution of analytics workloads.
- Data is loaded and stored in “raw” format in a data lake, with no indexing or prepping required.
- This allows the flexibility to perform many types of analytics
  - exploratory data science, big data processing, machine learning, and real-time analytics
  - from the most comprehensive dataset, in one central repository.

StreamSets
A SOFTWARE AG COMPANY

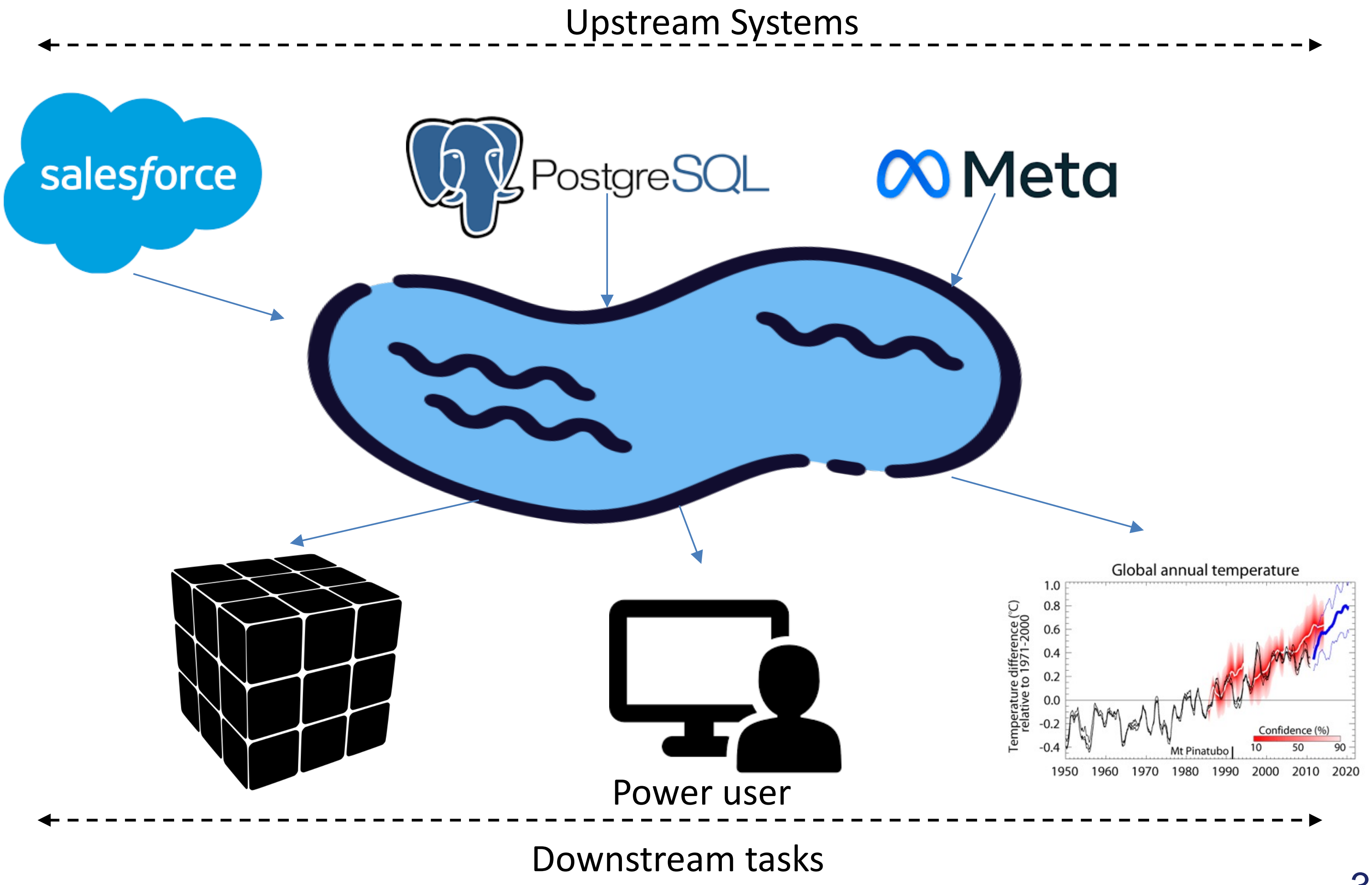
Without Data Lake

Upstream Systems



Downstream tasks

With Data Lake



Upstream Systems

Power user

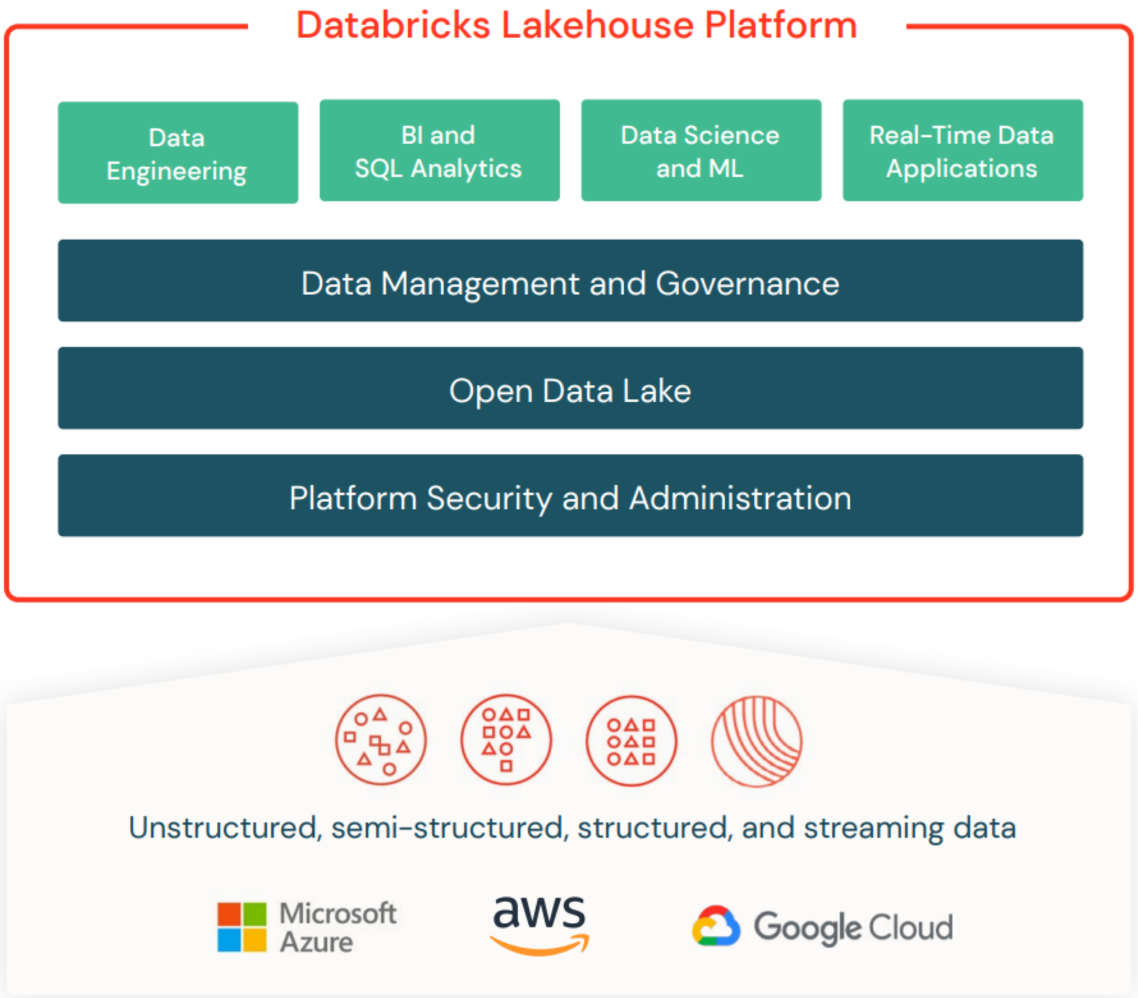
Downstream tasks

Data Lake or (?) Data Warehouse

- Data Lake IS NOT equal to Data Warehouse
- Data Lake is a Complement to Data Warehouse
  - Data lakes enhanced the utility of data warehouses.
- Data lakes allow organizations to stage swathes of
  - unstructured, semi-structured and structured data from multiple sources that they can then route to multiple purpose-built data warehouses.
- Data lakes, together with Data Warehouse facilitates seamless *data staging and storage* between sources and destinations.
  - Modern data infrastructure responds and grows with evolving use cases, business priorities, and technologies.

StreamSets
A SOFTWARE AG COMPANY

Data Lakehouse



Databricks Lakehouse Platform

Data Engineering

BI and SQL Analytics

Data Science and ML

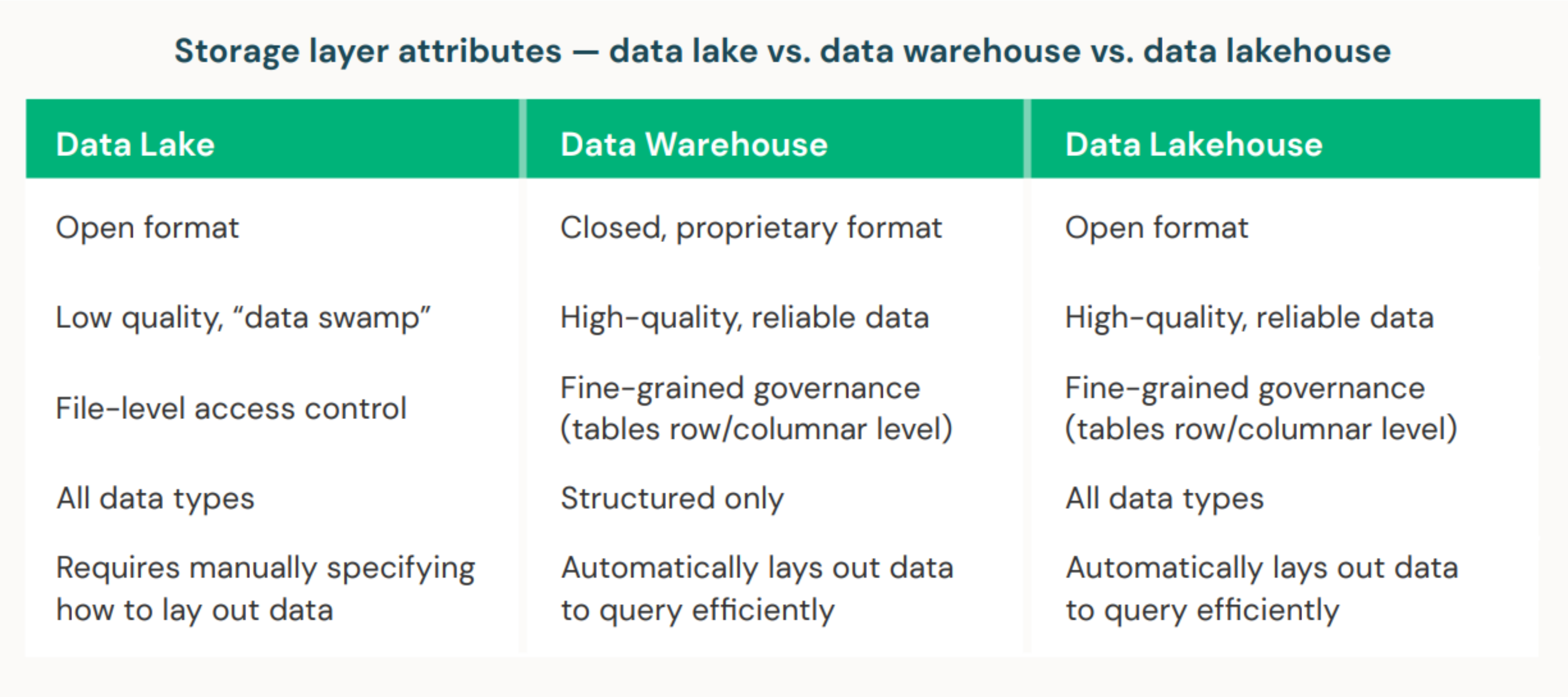
Real-Time Data Applications

Data Management and Governance

Open Data Lake

Platform Security and Administration

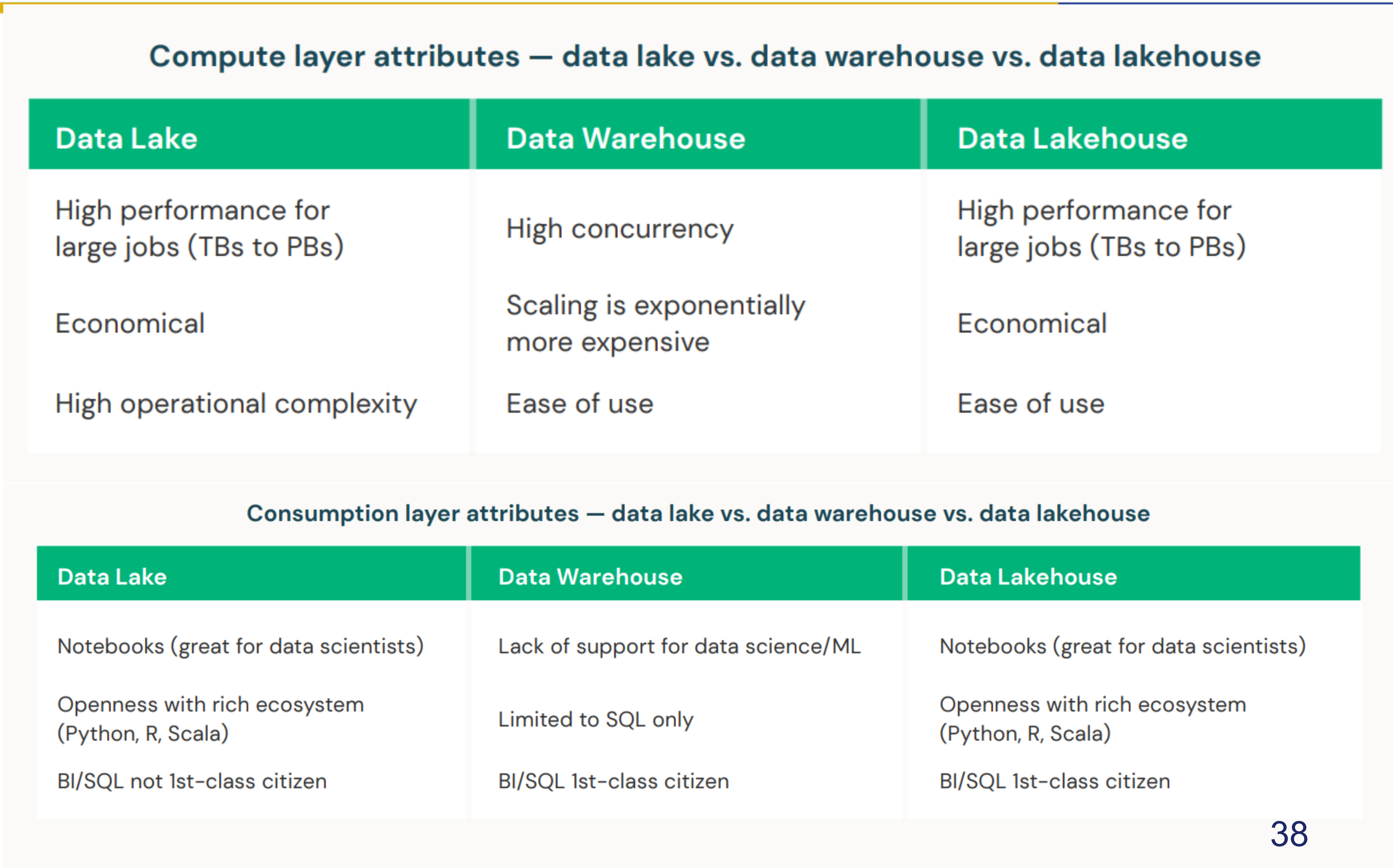
Unstructured, semi-structured, structured, and streaming data



Storage layer attributes — data lake vs. data warehouse vs. data lakehouse

Data Lake	Data Warehouse	Data Lakehouse
Open format	Closed, proprietary format	Open format
Low quality, “data swamp”	High-quality, reliable data	High-quality, reliable data
File-level access control	Fine-grained governance (tables row/columnar level)	Fine-grained governance (tables row/columnar level)
All data types	Structured only	All data types
Requires manually specifying how to lay out data	Automatically lays out data to query efficiently	Automatically lays out data to query efficiently

databricks



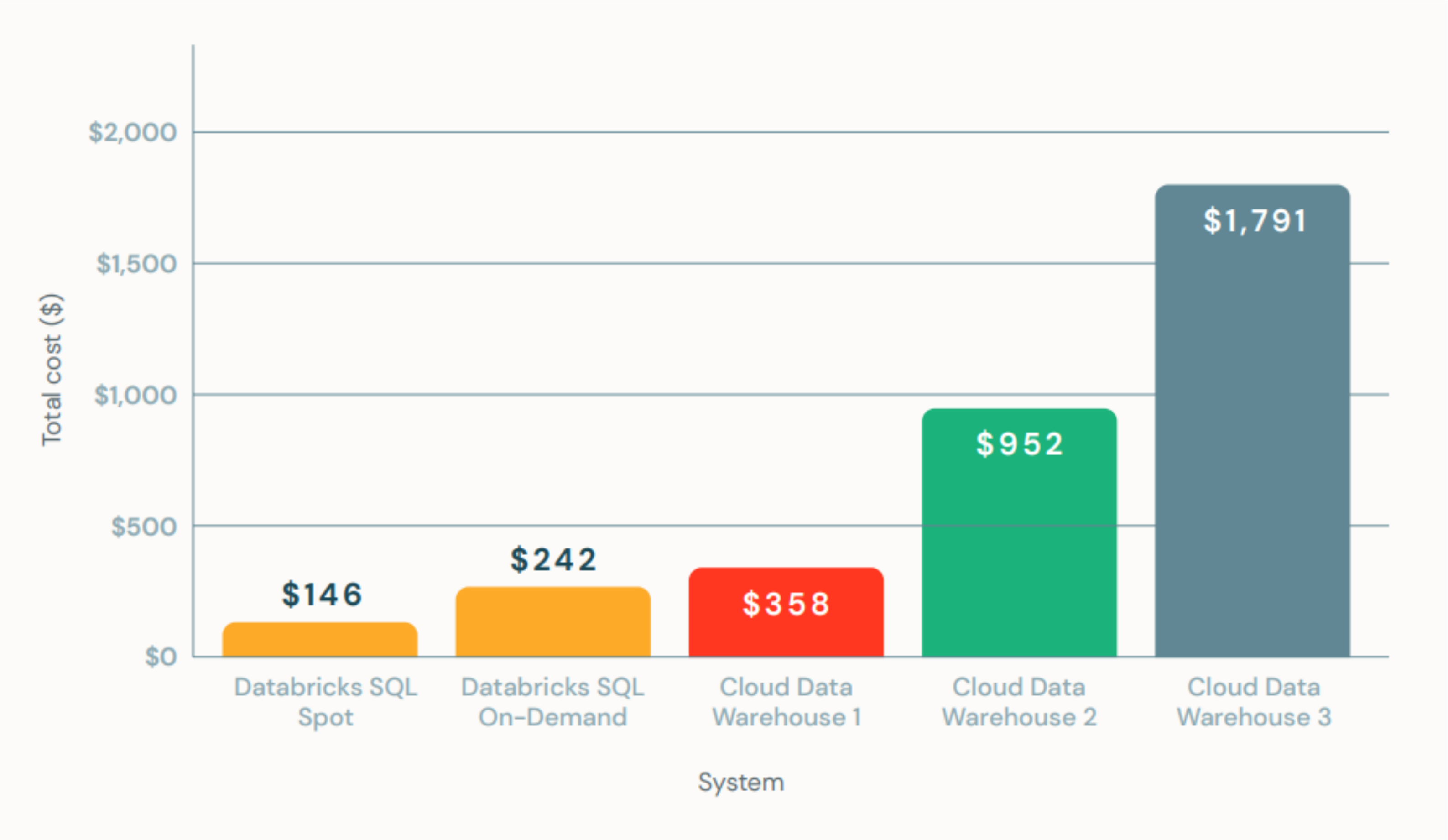
Compute layer attributes — data lake vs. data warehouse vs. data lakehouse

Data Lake	Data Warehouse	Data Lakehouse
High performance for large jobs (TBs to PBs)	High concurrency	High performance for large jobs (TBs to PBs)
Economical	Scaling is exponentially more expensive	Economical
High operational complexity	Ease of use	Ease of use

Consumption layer attributes — data lake vs. data warehouse vs. data lakehouse

Data Lake	Data Warehouse	Data Lakehouse
Notebooks (great for data scientists)	Lack of support for data science/ML	Notebooks (great for data scientists)
Openness with rich ecosystem (Python, R, Scala)	Limited to SQL only	Openness with rich ecosystem (Python, R, Scala)
BI/SQL not 1st-class citizen	BI/SQL 1st-class citizen	BI/SQL 1st-class citizen

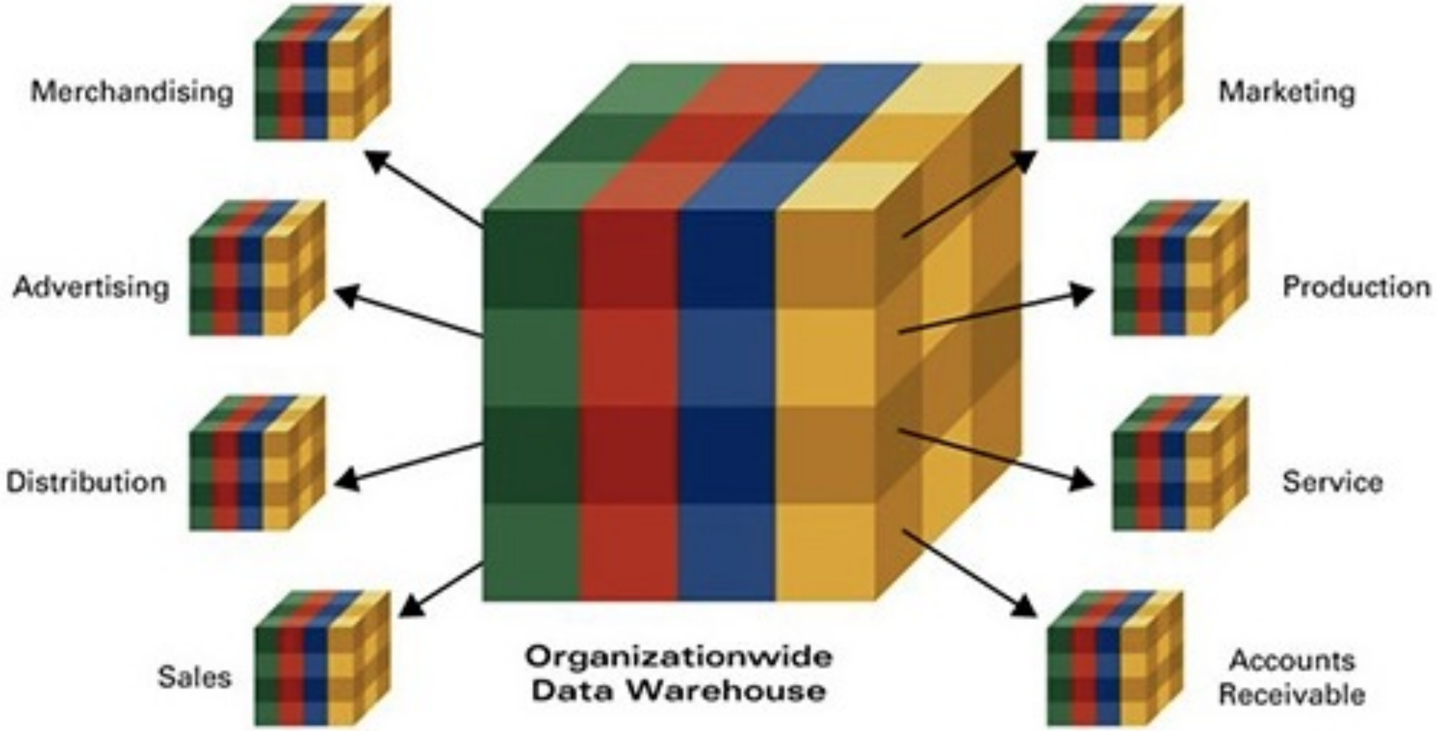
Performance comparison



100TB TPC-DS price/performance benchmark (lower is better)

databricks

Data Mart Overview

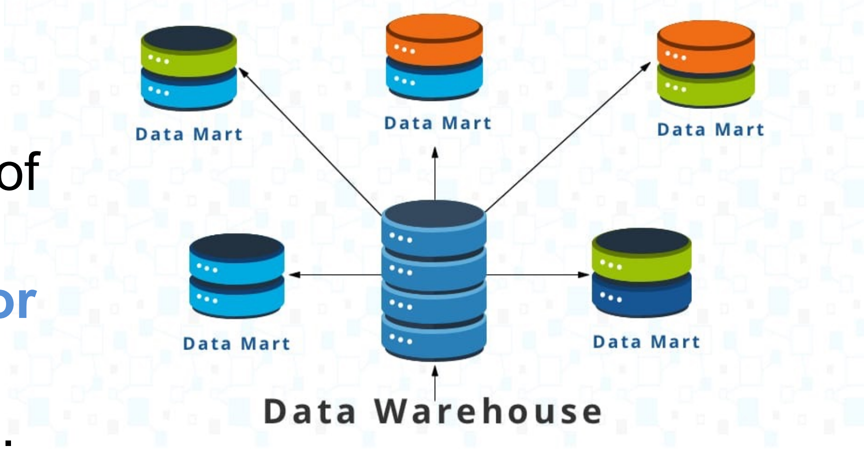


What is Data Mart

- A data mart is a **subset** of a data warehouse oriented to a specific business line.
  - Focuses on particular business domain as marketing or sales.

- Data marts contain repositories of summarised data collected for analysis on a **specific section or unit** within an organisation, for example, the **sales department**.

- A data mart can be created from an existing data warehouse - the top-down approach



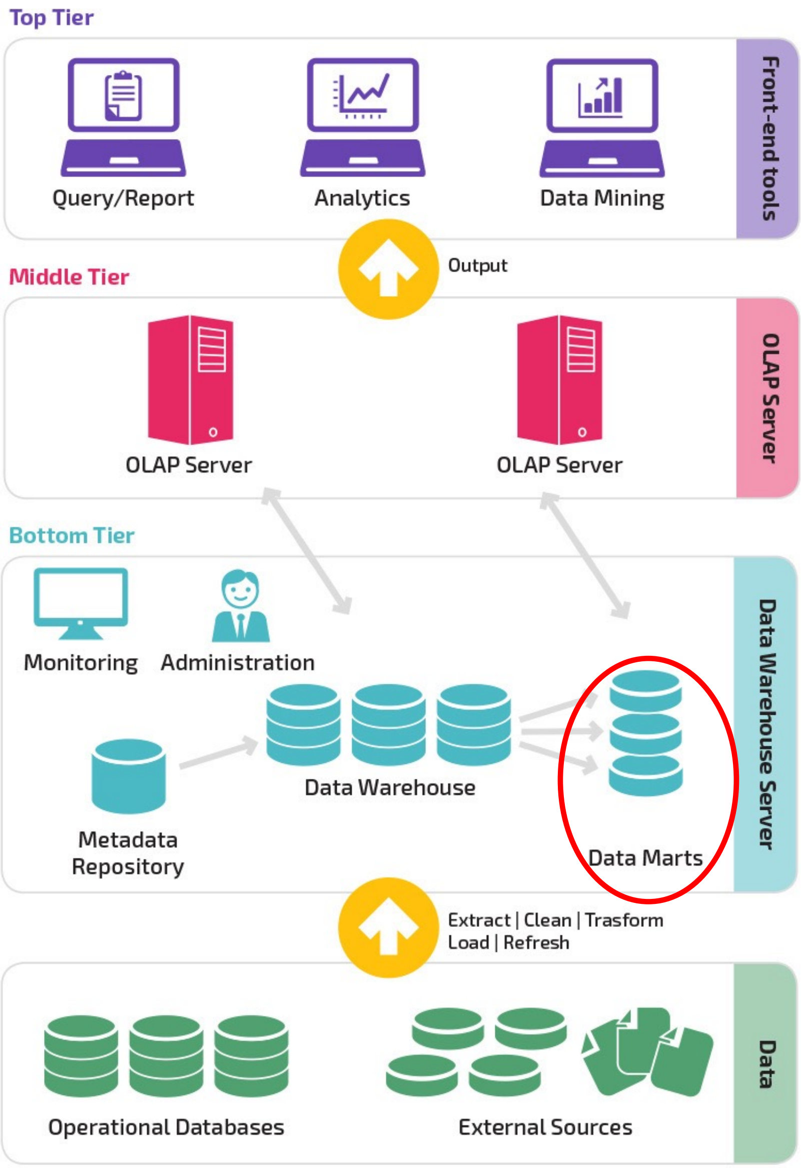
When to consider a Data Mart

- If you want to partition the data with a set of user access control strategy.
- If a particular department wants to see the query results much faster instead of scanning huge Data Warehouse data.
- If a department wants data to be built on other hardware (or) software platforms.
- If a department wants data to be designed in a manner that is suitable for its tools.

Why Data Mart?

- Data Mart focuses only on functioning of **particular** department of an organisation.
- The data mart can **improve** the **response time** of users due to the reduction of data.
- It provides **easy access** to frequently requested data.
- It is **easier** to **implement** when compare to corporate data warehouse. (The cost of implementing Data Mart is certainly lower)
- Compared to Data Warehouse, a data mart is **agile**. In case of change in model, data mart can be built **quicker** due to a smaller size.
- Data is partitioned and allows very **granular** access control privileges.

DW Architecture and DataMart



Characteristics of a Data Mart

- Dedicated **single subject matter**

- Focuses in on the subject matter by **consolidating** and **integrating** information from **various sources**.

- Usually dedicated for a **specific** business function or purpose.

- Built using a dimensional model with **star schema**. This allows data marts to have **multidimensional** analytical capabilities.

Outline

- **Metadata**
  - ✓ Introduction to Metadata and Metadata Examples
  - ✓ Data Warehouse and Metadata
  - ✓ Metadata Repository and Category
  - ✓ Challenges of Metadata Management
- **Data Mart**
  - ✓ Introduction to Data Mart
  - ✓ **Types of Data Mart**
  - ✓ Inmon and Kimball’s Data Warehouses
  - ✓ Implementing Data Mart
  - ✓ Advantage/ Disadvantage of Data Mart
  - ✓ Use Cases Example

Data Mart – Types of Data Mart

There are three main types of data mart:

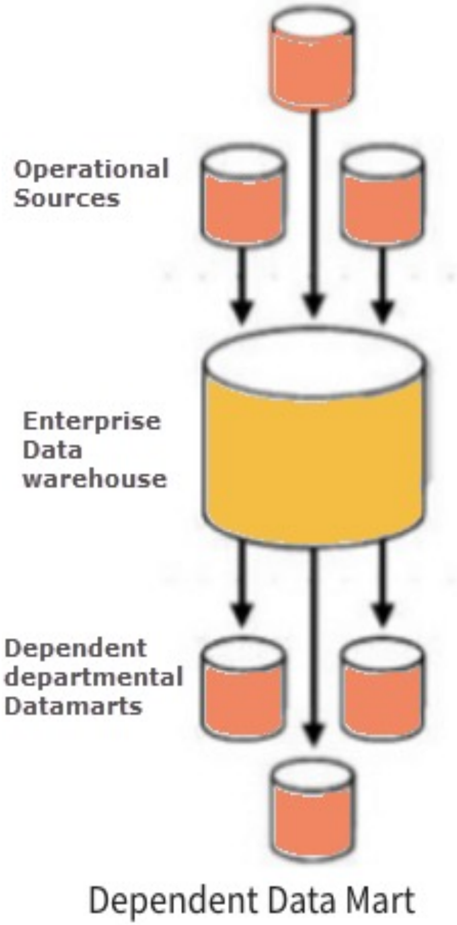
- **Dependent**: Dependent data marts are created by drawing data directly from operational, external or both sources.

- **Independent**: Independent data mart is created without the use of a central data warehouse.

- **Hybrid**: This type of data marts can take data from data warehouses or operational systems.

Dependent Data Mart

- A dependent data mart allows sourcing organisation's data from a single Data Warehouse.
  - It offers the benefit of **centralisation**.
  - If you need to develop one or more physical data marts, then you need to configure them as dependent data marts.
- Dependent Data Mart in data warehouse can be built in **two different ways**:
  - Either where a user can access both the data mart and data warehouse, depending on need;
  - Or where access is limited only to the data mart.

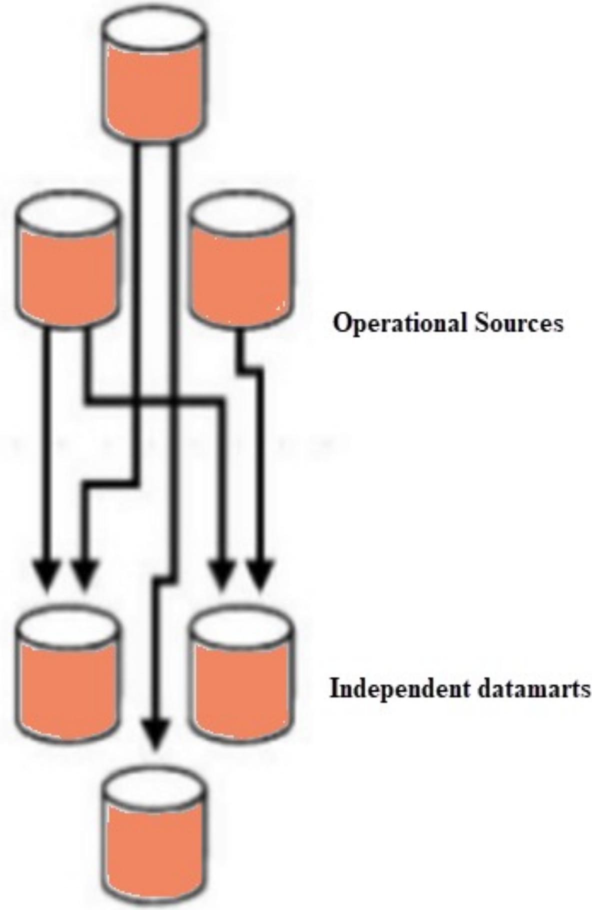


Independent Data Mart

- An independent data mart is created **without** the use of a central Data Warehouse.
  - An ideal option for smaller groups within an organisation.

- An independent data mart has neither a relationship with the enterprise data warehouse nor with any other data mart.
  - Data is input separately, and
  - Analyses are also performed autonomously.

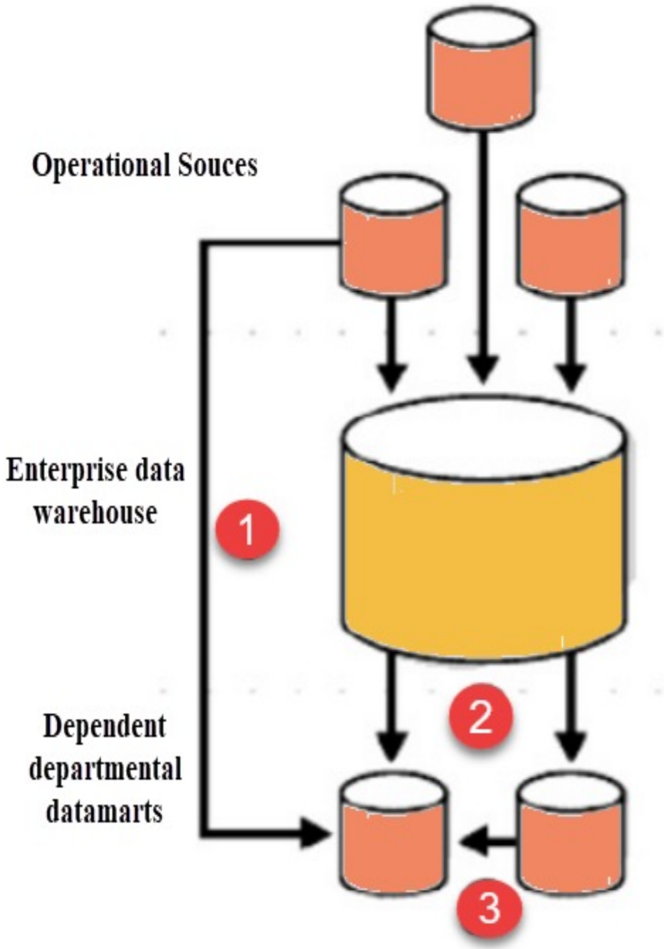
- Implementation of independent data marts is antithetical to the motivation for building a data warehouse.
  - We need a consistent, centralised store of enterprise data which can be analysed by multiple users with different interests who want widely varying information.



Hybrid Data Mart

- A hybrid Data Mart combines input from sources apart from Data Warehouse.
  - This could be helpful when you want ad-hoc integration, like after a new group or product is added to the organisation.

- It is best suited for multiple database environments and fast implementation turnaround for any organisation.
  - Requires least data cleansing effort.
  - Supports large storage structures, and
  - Best suited for flexible for smaller data-centric applications.



Hybrid Data Mart

Dependent vs Independent

Independent Data Mart

Pros:

- Easier and faster to implement
- More flexible

Cons:

- It is easy to form an information island (because it disengages from the data warehouse, when multiple independent data marts grow to a certain scale, enterprises will only add some information islands because there is no centralised data warehouse coordination.)
- Not a true enterprise-wide solution and can become very costly over time as more and more are added.
- They do not provide the historical depth of a true data warehouse.

Dependent vs Independent

Dependent Data Mart

Pros：

- Generally considered a better solution than independent marts. (because dependent marts use the warehouse as their foundation)
- There is no historical limit to the data
- Improve data analysis quality
- Data integrity is ensured.

Cons:

- Take longer and more expensive to implement.

Outline

- **Metadata**
  - ✓ Introduction to Metadata and Metadata Examples
  - ✓ Data Warehouse and Metadata
  - ✓ Metadata Repository and Category
  - ✓ Challenges of Metadata Management
- **Data Mart**
  - ✓ Introduction to Data Mart
  - ✓ Types of Data Mart
  - ✓ **Inmon and Kimball's Data Warehouses**
  - ✓ Implementing Data Mart
  - ✓ Advantage/ Disadvantage of Data Mart
  - ✓ Use Cases Example

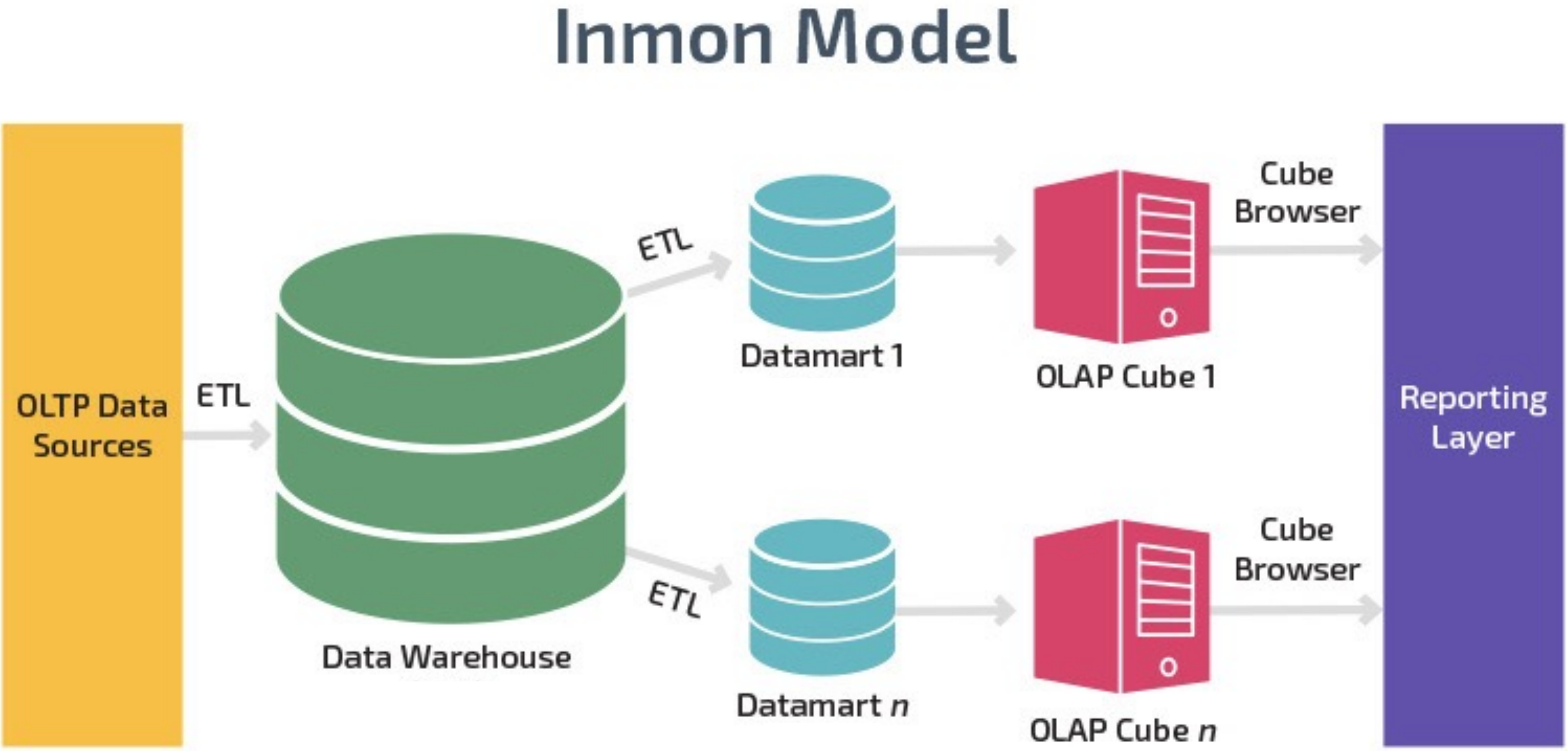
Data Warehouse: Inmon vs. Kimball

Bill Inmon and Ralph Kimball are two outstanding pioneers in the field of data warehouse. They have different views on how data warehouses should be designed from the organisation's perspective.

- **Bill Inmon's** approach favours a **top-down** design in which the data warehouse is the centralised data repository and the most important component of an organisation's data systems.

- The **Inmon** approach first builds the centralised 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 for Data Warehouses

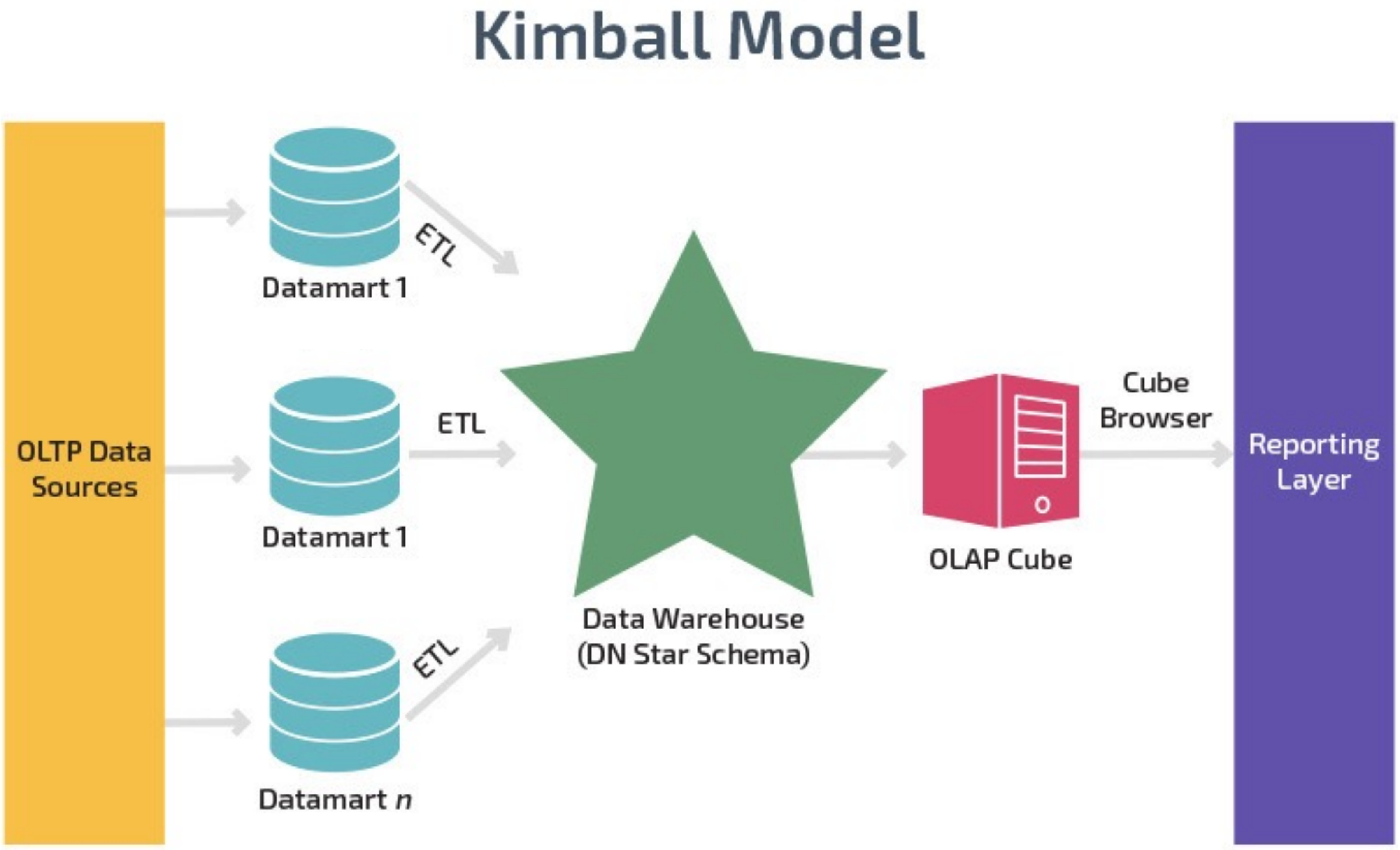


Data Warehouse: Inmon vs. Kimball

- In the Inmon model, data in the data warehouse is integrated, meaning the Data Warehouse is the source of the data that ends up in the different Data Marts. This ensures data integrity and consistency across the organisation.

- **Ralph Kimball's** Data Warehouse design starts with the most important business processes. In this approach, an organisation creates Data Marts that aggregate relevant data around subject-specific areas. The Data Warehouse is the combination of the organisation’s individual data marts.

Kimball’s Model for Data Warehouses



Inmon vs. Kimball

- With the **Kimball** approach, the 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.

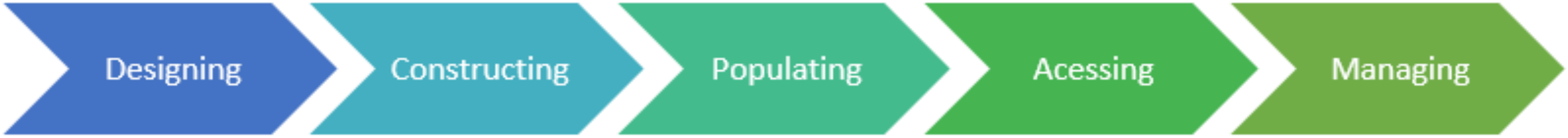
- **Kimball** said “the data warehouse is nothing more than the union of all Data Marts.”

Outline

- **Metadata**
  - ✓ Introduction to Metadata and Metadata Examples
  - ✓ Data Warehouse and Metadata
  - ✓ Metadata Repository and Category
  - ✓ Challenges of Metadata Management
- **Data Mart**
  - ✓ Introduction to Data Mart
  - ✓ Types of Data Mart
  - ✓ Inmon and Kimball’s Data Warehouses
  - ✓ **Implementing Data Mart**
  - ✓ Advantage/Disadvantage of Data Mart
  - ✓ Use Cases Example

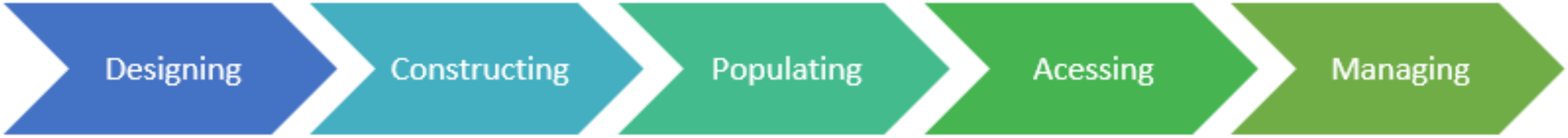
Steps of Implementing a Data Mart

- **Designing**
  - Gathering the business & technical requirements and Identifying data sources.
  - Selecting the appropriate subset of data.
  - Designing the logical and physical structure of the Data Mart.
- **Constructing**
  - Implementing the physical database designed in the earlier phase, e.g., table, indexes, views, etc.
- **Populating**
  - Source data to target data mapping
  - Extraction of source data
  - Cleaning and transformation operations on the data
  - Loading data into the Data Mart
  - Creating and storing metadata



Steps of Implementing a Data Mart

- **Accessing**
  - Querying data, creating reports, charts.
  - Set up and maintain database structures
  - Set up API and interfaces if required.
- **Managing**
  - Ongoing user access management.
  - System optimisations and fine-tuning to achieve the enhanced performance.
  - Adding and managing fresh data into the Data Mart.
  - Planning recovery scenarios and ensure system availability in the case when the system fails.



Advantage of a Data Mart

- **Managing** big data and gaining valuable business insights
- **Efficient access** - A data mart is a time-saving solution for accessing a specific set of data for business intelligence
- **Inexpensive data warehouse alternative** - Data marts can be an inexpensive alternative to developing an enterprise data warehouse, where required data sets are smaller.
- **Improve data warehouse performance** - Significantly can reduce analytics processing
- **Data maintenance** - Different departments can own and control their data.

Disadvantage of a Data Mart

- Many enterprises create too many disparate and **unrelated** Data Marts without much benefit.
  - Creating too many **data marts** become cumbersome sometimes.

- Data Mart cannot provide company-wide data analysis as their **data set is limited**.
  - Data Mart stores the data related only to specific function

- Data Marts are meant for **small business** needs. Increasing the size of data marts will **decrease** its **performance**.

Comparison with Data Warehouse

	Data Mart	Data Warehouse
Focus	A single subject or functional organisation area	Enterprise-wide repository of disparate data sources
Data Sources	Relatively few sources linked to one line of business	Many external and internal sources from different areas of an organisation
Size	Less than 100 GB	100 GB minimum but often in the range of terabytes for large organisations
Normalization	No preference between a normalised and denormalised structure	Modern warehouses are mostly denormalised for quicker data querying and read performance
Decision Types	Tactical decisions pertaining to particular business lines and ways of doing things	Strategic decisions that affect the entire enterprise

Comparison with Data Warehouse

	Data Mart	Data Warehouse
Cost	typically from $10,000 upwards	Varies but often greater than $100,000; for cloud solutions costs can be dramatically lower as organisations pay per use
Setup Time	3-6 months	At least a year for on-premise warehouses; cloud data warehouses are much quicker to set up
Data Held	Typically summarised data	Raw data, metadata, and summary data

Outline

- **Metadata**
  - ✓ Introduction to Metadata and Metadata Examples
  - ✓ Data Warehouse and Metadata
  - ✓ Metadata Repository and Category
  - ✓ Challenges of Metadata Management
- **Data Mart**
  - ✓ Introduction to Data Mart
  - ✓ Types of Data Mart
  - ✓ Inmon and Kimball's Data Warehouses
  - ✓ Implementing Data Mart
  - ✓ Advantage/Disadvantage of Data Mart
  - ✓ **Use Cases Example**

Use Cases of Data Marts

Data Mart

- Marketing analysis and reporting favor a Data Mart approach because these activities are typically performed in **a specialised business unit**, and do not require enterprise-wide data.

- A financial analyst can use a finance Data Mart to carry out **financial reporting**.

Use Cases of Data Warehouses

Data Warehouse

- A company considering an expansion needs to incorporate data from a variety of data sources across the organisation 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 centralised data warehouse to combine information from its claims department, sales, customer demographics, investments, and other areas.

References

References and Readings

- Han et al. Chapter 4.1
- What is Metadata
- Data Mart Tutorial
- Data Mart or Data Warehouse
- Data Lake vs Data Warehouse
- Cloud Data Warehousing for Dummies (Google Snowflake)
- Top Cloud Data Warehouses Comparison by Qlik
- Why the Data Lakehouse is Your Next Data Warehouse (Databricks)

Guest lecture next week

Yuanyi Luo is currently a fourth-year PhD candidate at the Harbin Institute of Technology, specialising in multimodal machine learning and pattern analysis.

With publications in SCI journals and top-ranking conferences, Luo is actively applying his research to advance advertising supervision and healthcare. As a visiting scholar at UWA, he focuses on integrating knowledge graphs with multimodal learning.

Copyright Notice