APAN PS5400: Managing Data

Week 5: Data Warehouses, Data Lakes

Lecturer: François Scharffe

Columbia University
School of Professional Studies

Recap of what we covered last week

- Views
- Functional dependencies
- Normal Forms
- Normalization
- Project 1.2

This week

- Data Warehouses
  - Sections 10.6, 10.7 of *A First Course in Database Systems* (our textbook covers data warehouses)
- Schema-on-write
- Data Lakes
- Schema-on-read
- Data warehouses vs Data Lakes

Transactions Vs. Analysis

- Transactions are jobs that need to be executed
  - Example: Purchasing a product online
- Analysis is to gain insights about patterns in some data
  - Example: To find out which products sell best in which months in which area
  - Actions may be undertaken based on these insights
  - Insights may be needed to decide between alternate actions
  - But the primary intent of analysis is not to execute an action but to gain insight
- Different data infrastructure is needed for analysis vs. transactions.

OLTP vs. OLAP

On-line transaction processing (OLTP)

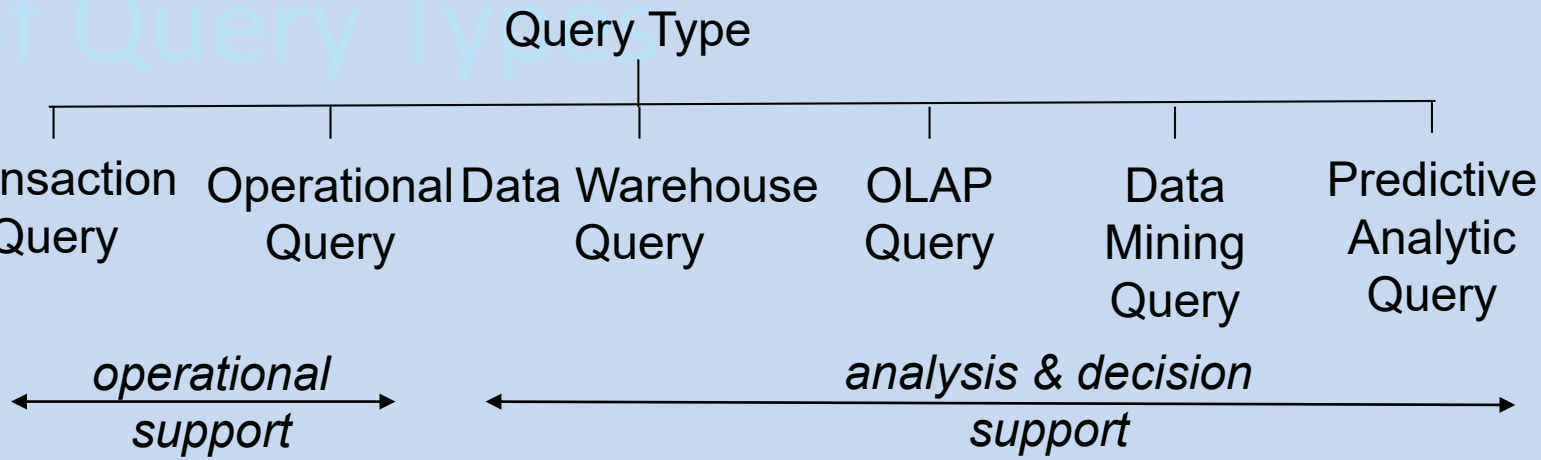
- Each transaction execution involves a small number of data elements (logically single record)
- Small number of transactions types
- Large number of transaction executions
- Comparatively static requirements and data schema
- Primitive CRUD (create, read, update, delete) operations
- Performance and throughput comparatively important

On-line analytical processing (OLAP)

- Each transaction execution may involve a very large number of data elements
- Large number of different types of queries, including ad/hoc
- Small number of long running transactions
- Comparatively dynamic (analytical requirements and data)
- Generally read-only, but increasingly dynamic calculations on the fly (e.g., supporting what-if scenarios)
- Performance and throughput relatively less important

Courtesy of Wayne Huang

Variety of Query Types



- Transaction: Locate the address of a client
- Operational: Uncover the audit trail of a lost item (trade, merchandise)

} Oper..

- DW: Find the trend in sales of a particular product in a specific region
- OLAP: Find the sales of all products by month and region

} OLAP

- DM: What products have similar sales patterns
- PA: Find the rules that predict the sale of a certain product to a certain customer segment (Predictive Analytics)

} Data Mining

Courtesy of Wayne Huang

Data infrastructure for analysis

- Data warehouses and data lakes both provide infrastructure for analysis as opposed to transactions
- Databases provide infrastructure for transactions
- We need to understand for which type of analysis of which type of data for what purpose to decide whether to use a data warehouse or a data lake.

Data warehouses

- Different parts of an enterprise have data in different databases
  - Human resources, marketing dept, sales, public relations, customer service, etc.
  - Data from different branches of the enterprise
- As new data accumulates older data may be archived
- Relevant external data may also be useful
- Bringing all this data in one common repository under one schema can yield new insights
- A data warehouse is where this data is brought together under one common schema

Data Warehousing Terms

Enterprise Data Warehouse:

- Repository of (much of) the firm's data
- Historic dimension: integrated from many Transaction Processing System (TPS) and external sources
- Subject-oriented (targets one or more several subjects of analysis), integrated (contents result from integration of data from several sources), consistent, non-volatile (accumulates data over a long period of time), time-oriented (keeps track of how data has evolved over a long period of time).
- Structured for ease of retrieval
- May take up to 3 years to build, costs $3 to $5 million

Data Mart:

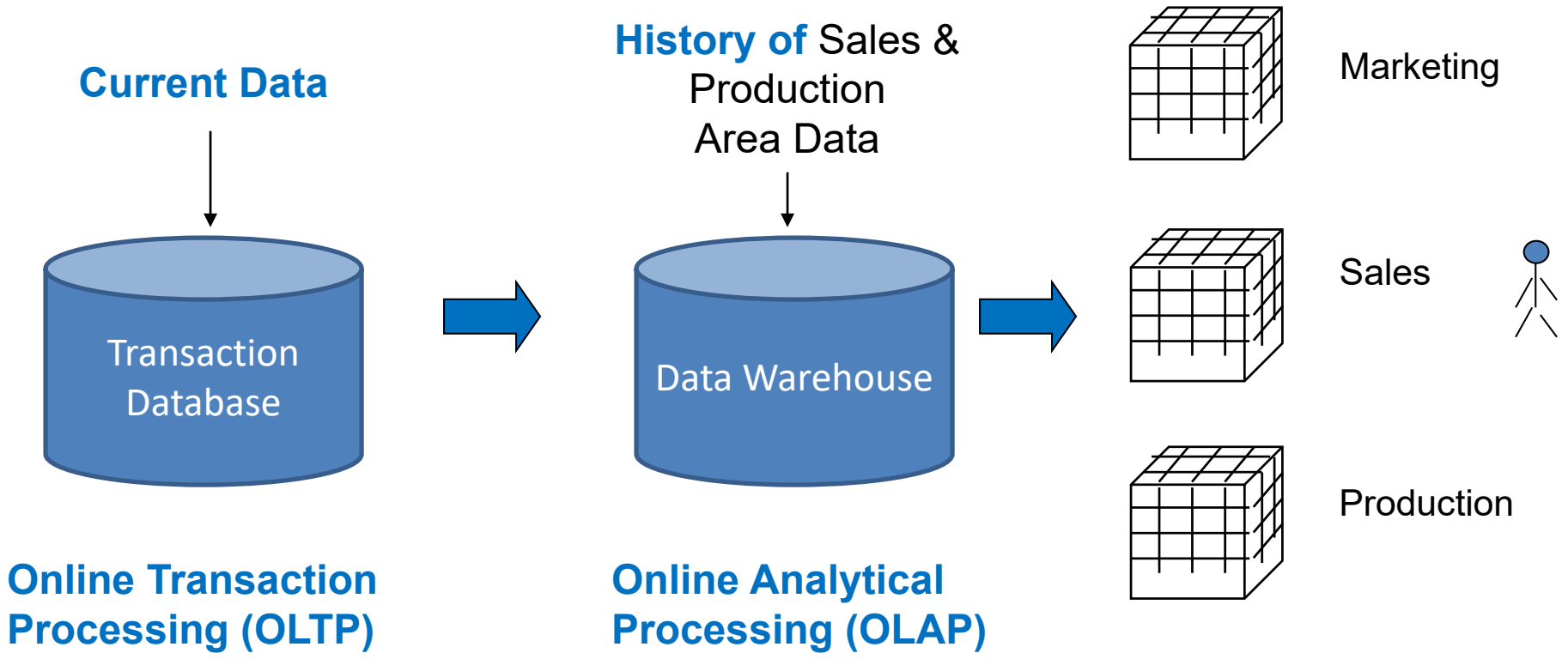
- A repository of all the data for a specific application area.
- Less than 100 GB of data. 10-20 users.

Incremental Development of an Enterprise Data Warehouse:

- Connect separate data marts (or warehouses) together through **conformed dimensions** - i.e., dimensions that are shared across several DWs

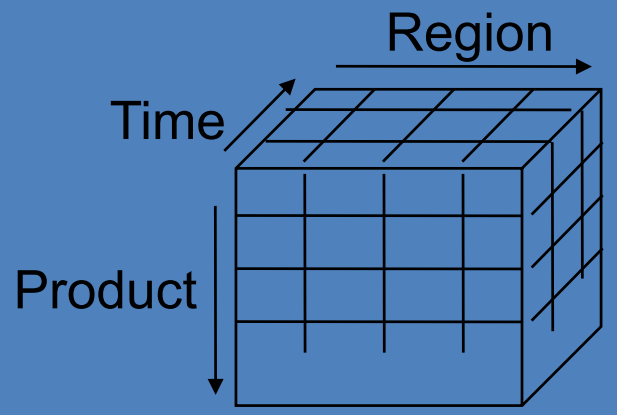
Courtesy of Wayne Huang

Data Warehousing: Process



Data Warehousing/OLAP

Data Warehouse: Large database containing terabytes of information in a form that is suitable for management analysis.



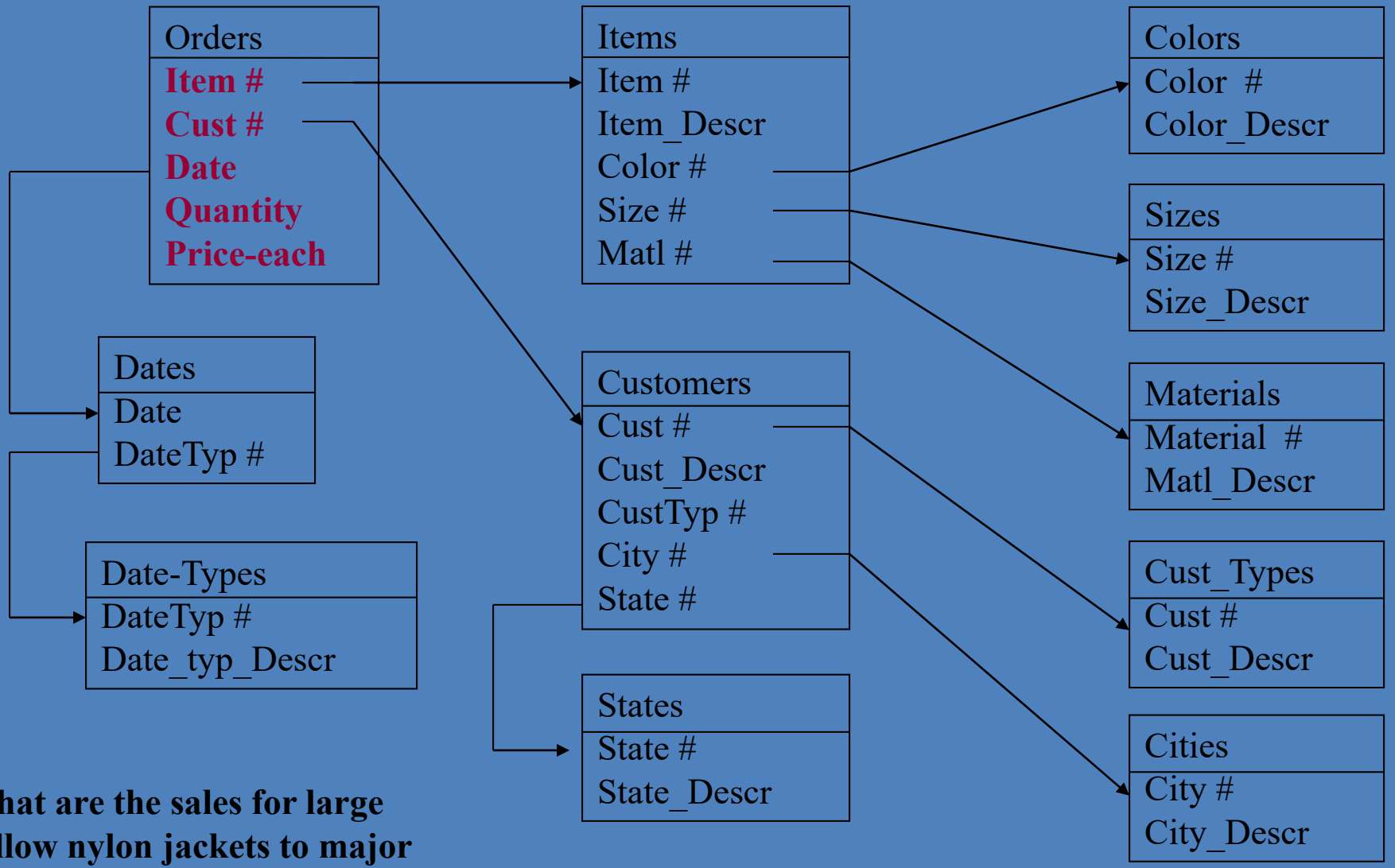
Typical Query: "I want to see (costs) by time (e.g. month), by product (e.g. cola), by channel (e.g. fountain), by sales district."

Data Cube: The user can generate many reports using the "dimensions" (Region, Time, Product) of cube. The "facts" are the information stored in the cells of the cube.

Example: IBM COGNOS (OLAP Tool)

Courtesy of Wayne Huang

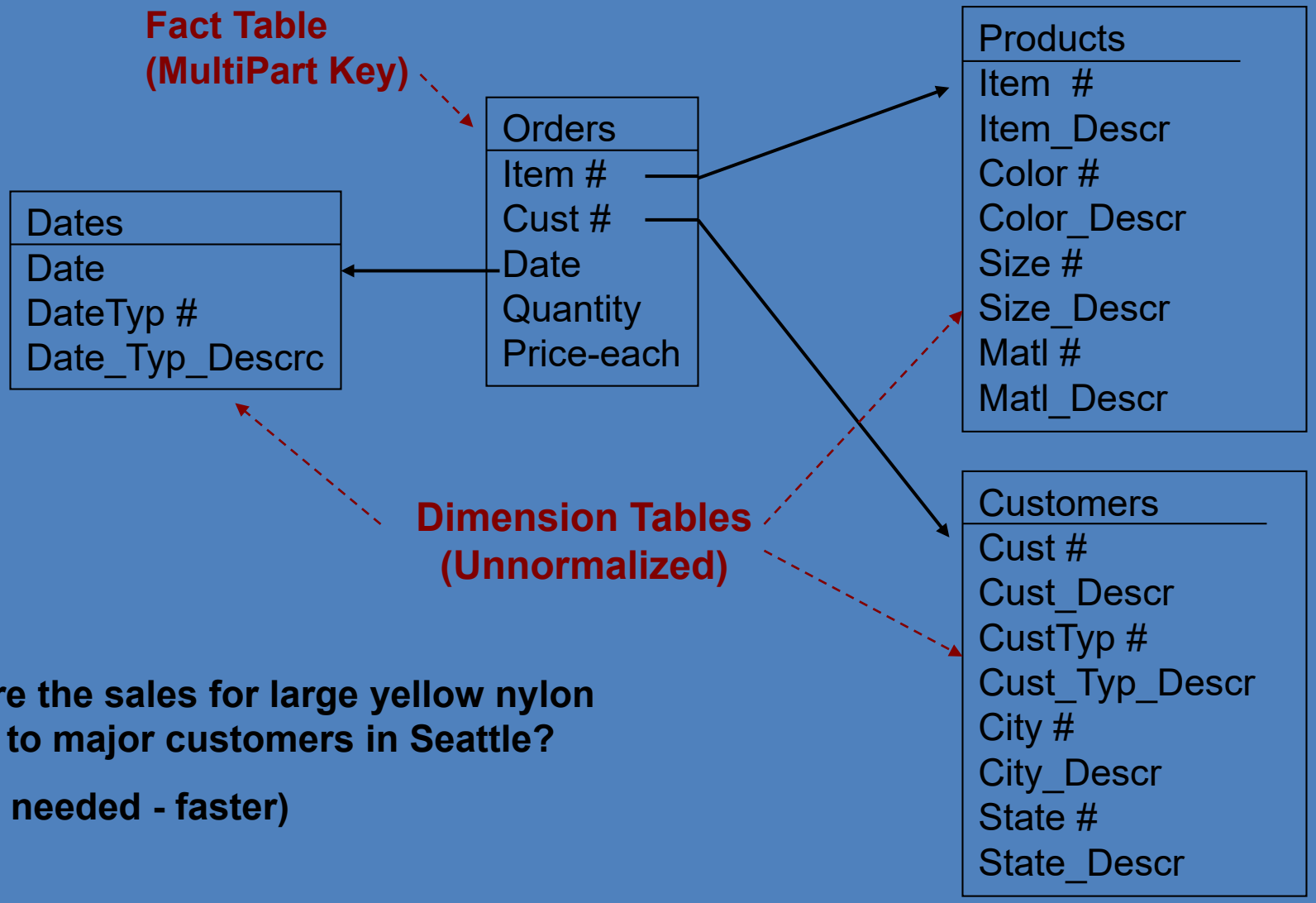
Normalized Design for TP



What are the sales for large yellow nylon jackets to major customers in Seattle? (9 joins - slow!)

Courtesy of Wayne Huang

Star Schema: Un-normalized Design



What are the sales for large yellow nylon jackets to major customers in Seattle?

(4 joins needed - faster)

Courtesy of Wayne Huang

Steps in Star Schema Design Process

1. Choose a *business process* and its *data*
   - E.g., orders, sales, invoices, shipments, etc.
2. Choose the *grain* of the business process
   - Fundamental *atomic level of data* to be represented in the fact table for this process. Important: This is required before step #3
   - Caution: choose the lowest possible level - we can always aggregate to a higher level but can never go lower than what is initially chosen and stored
   - E.g., individual transactions, individual daily snapshots, monthly snapshots, etc
3. Choose the *dimensions* of the business process
   - Dimensions qualify the fact table and represent query **entry points** into the star schema
   - E.g., time, product, customer, promotion, warehouse, transaction-type, status, etc.
   - For each dimension selected, describe all discrete text-like dimensional attributes (fields)
4. Choose the *measured facts* (numerical fields) that will populate each fact table record
   - E.g., quantity sold, dollars sold, etc.

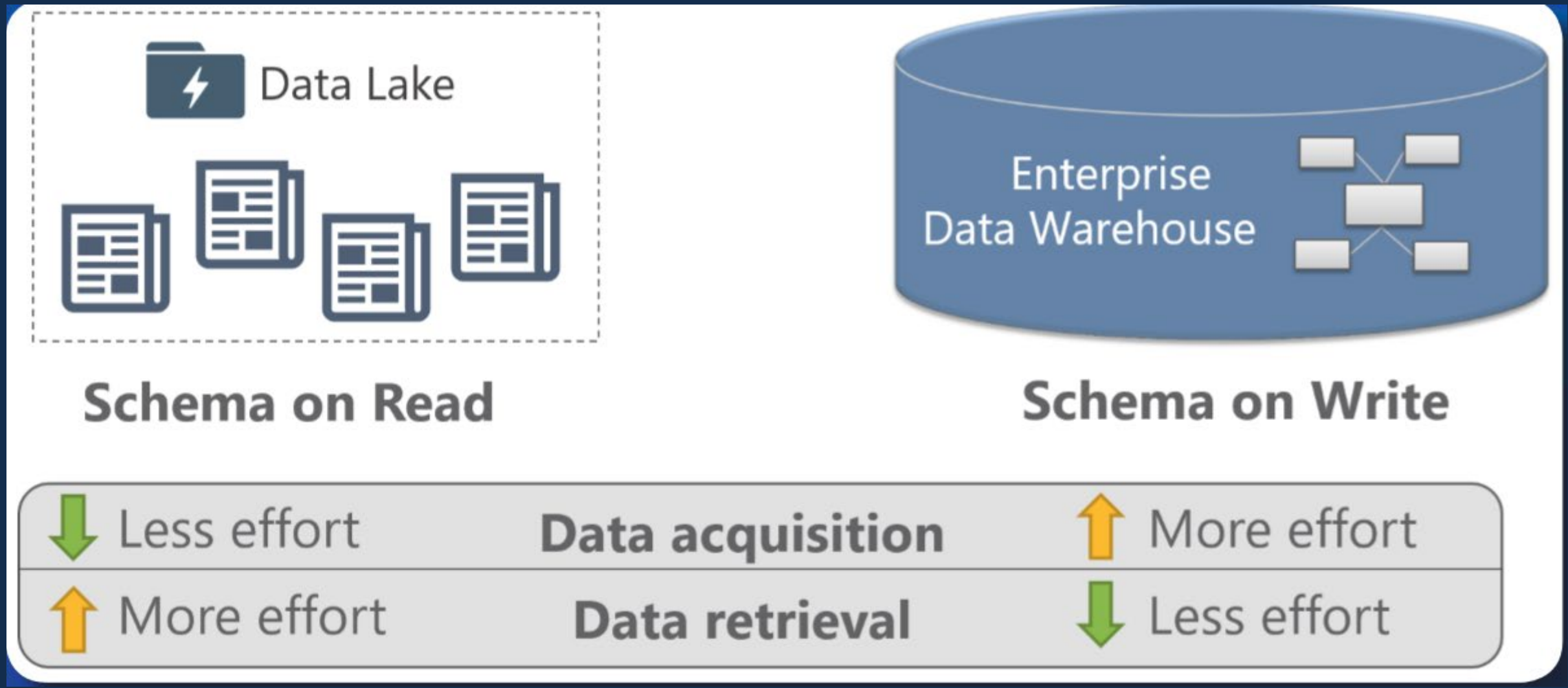
Courtesy of Wayne Huang

Shortcomings of Data Warehouses

- Data ingestion is too slow
  - IoT, social media, sensors generate vast quantities of data quickly
  - Many varieties of data
  - ETL procedures and schema-on-write will not work in these contexts
- Making data available for analysis too slow
  - Fitting rapidly accumulating data into a schema for analysis not a good idea.
  - Takes too long
  - Schema may need to be modified frequently as more types of data are ingested
  - Many applications need to use data soon after ingestion

Motivation for Data Lakes

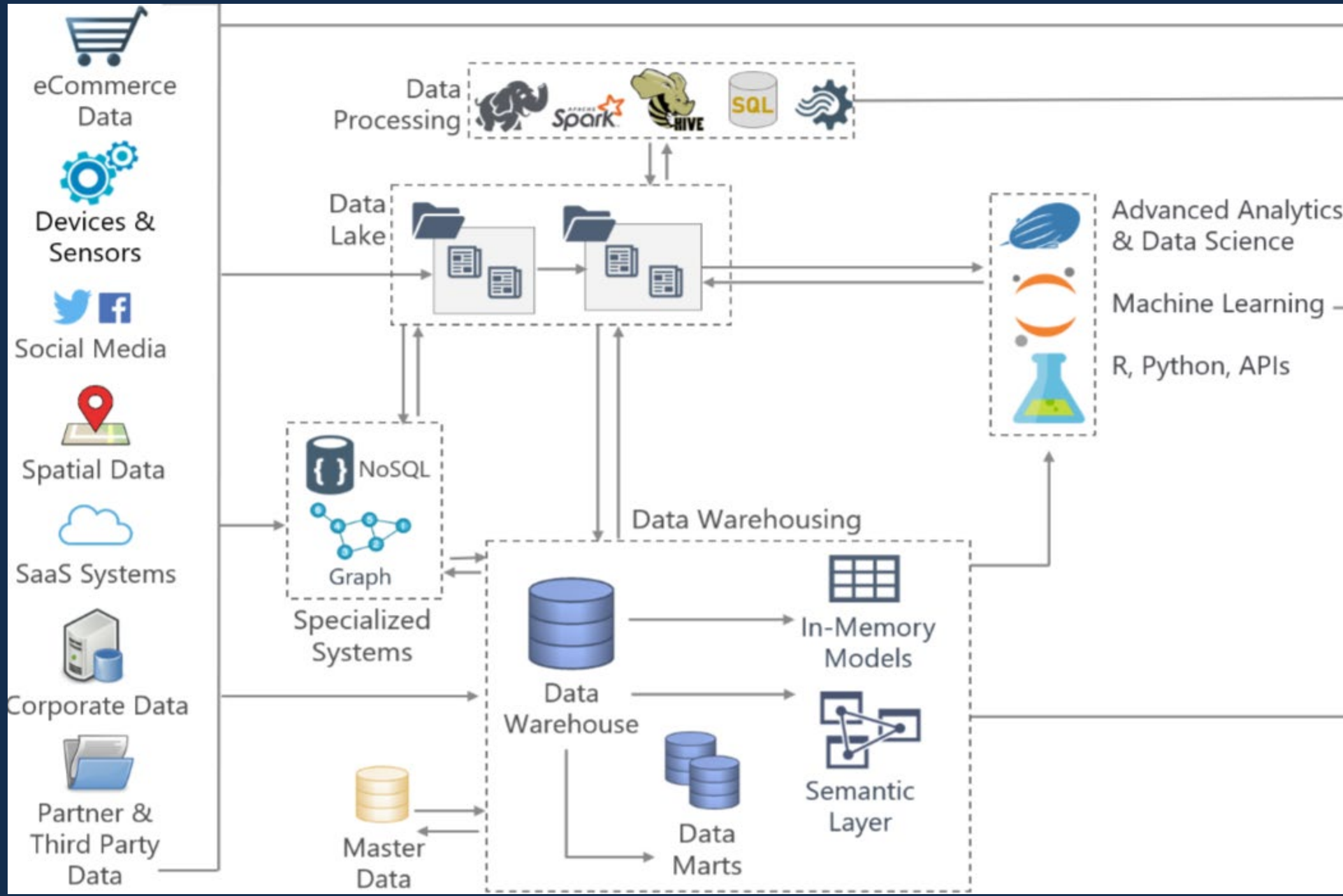
- Need a data repository for analysis which
  - Supports quick data ingestion
    - Raw data, slightly cleaned data
  - Thus, no schema
    - Schema-on-read instead of schema-on-write
    - Different types of data will have to be stored in the repository separately (different zones)
  - If no schema, then applications needing the data will have to create their own interfaces for accessing the data, structuring the data



Adapted from Data Lakes in a Modern Data Architecture by Blue Granite

Modern Data Architecture

Note that it contains both data warehouses and data lakes.



Adapted from Data Lakes in a Modern Data Architecture by Blue Granite

TIPS FOR DESIGNING A DATA LAKE

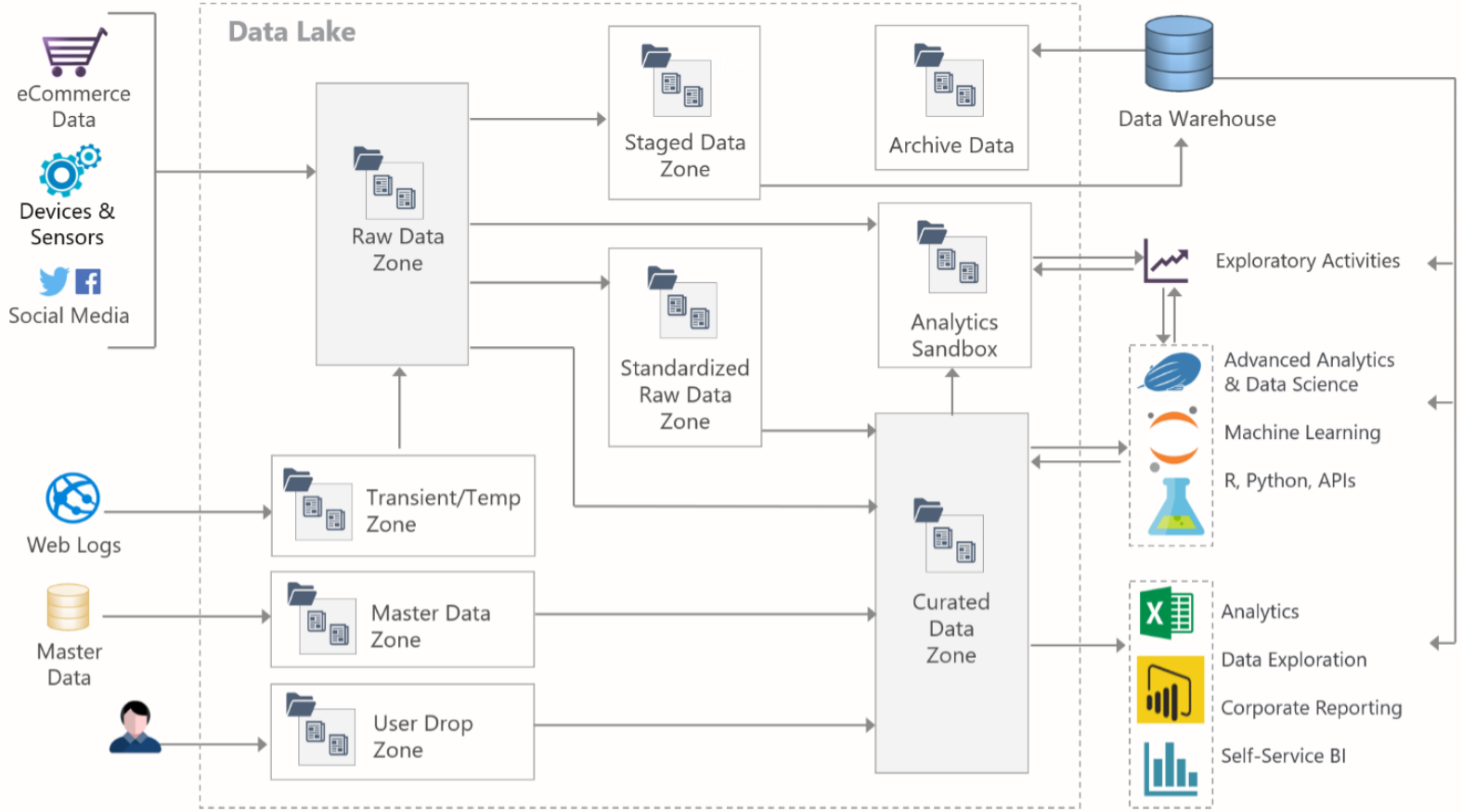
The importance of purposefully organizing the data lake cannot be overstated. The concept of zones in a data lake is important. Zones can be physical, or can just be conceptual. The following high-level zones are commonly used:

The raw data zone is storage for new data acquired, in its native format. This is an exact copy from the source, often in a normalized format, and is immutable to change.

The transient zone is selectively utilized when data quality validations are required before the data may land in the raw data zone. It is also helpful when you temporarily need to segregate a “new data zone” separate from the “raw data zone.”

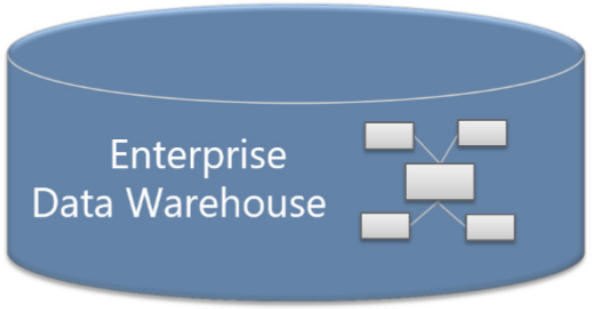
The master data zone contains reference data which augments and aids analytical activities.

The standardized raw data zone contains data which has been standardized into a structure (aka semantic normalization), but otherwise is still in raw form.



The curated data zone is the serving layer location for data which has been cleansed, transformed as necessary, and structured for optimal delivery

Adapted from Data Lakes in a Modern Data Architecture by Blue Granite



	Data Lake	Enterprise Data Warehouse
Type of data stored:	Any type of data structure, any format	Structured data (most often in columns & rows in a relational database)
Best way to ingest data:	Streaming, micro-batch, or batch processes	Batch processes
Typical load pattern:	ELT (Extract, Load, and Transform at the time the data is loaded)	ETL (Extract, Transform, then Load)
Analysis pattern:	Acquire data, analyze it, then iterate to determine its final structured form	Determine structure, acquire data, then analyze it; iterate back to change structure as needed
When structure is applied:	At query time: Schema-on-read or when "curated" data is generated	At data load time: Schema-on-write

Adapted from Data Lakes in a Modern Data Architecture by Blue Granite

When to use which

Data Warehouse

- All the data is more or less of the same type
- Can be easily fitted into a common schema
- Does not change rapidly
- Analytical needs are predictable
- Deployment is not urgent

Data Lake

- Many types of data
- Rapid accumulation
- Analytical needs can alter quickly, unpredictable
- Data/analysis needs to be served very soon after ingestion

What we covered this week

- Data warehouses
- Data lakes
- Comparison of the two
- Modern data architecture

Next week

- Data quality
- The need for data governance
- Data governance policies