



DS-2002: Data Systems

Overview of Data Warehouse Systems

Prof. Jon Tupitza



Modern Data Platform: **Solution Scenarios**

Big (Unstructured and/or Poly-Schematic) Data Integration and Advanced Analytics

“We want to integrate all our data into our data warehouse”



Modern Data
Warehousing

“We’re trying to predict which of our customers will churn”



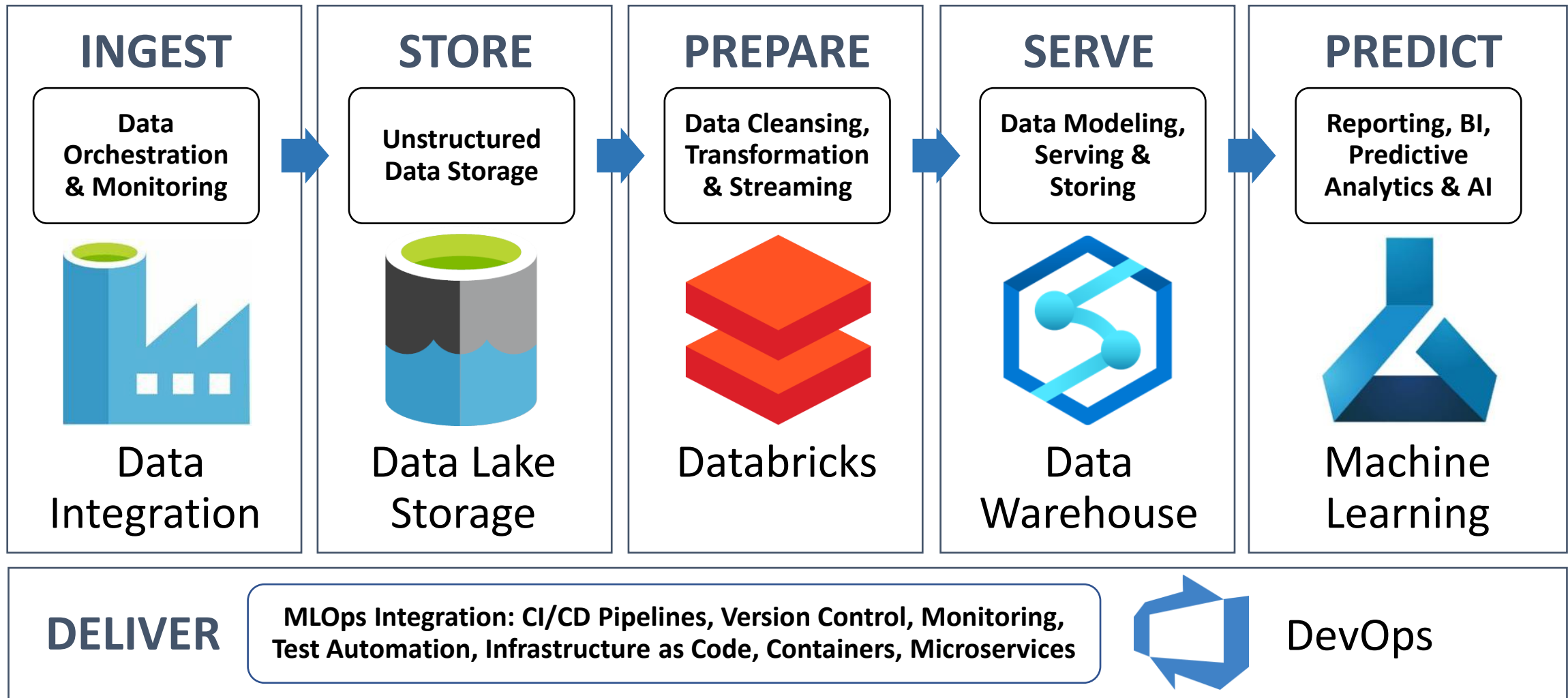
Advanced
Analytics

“We’re trying to get insights from our devices in real-time”



Real-Time
Analytics

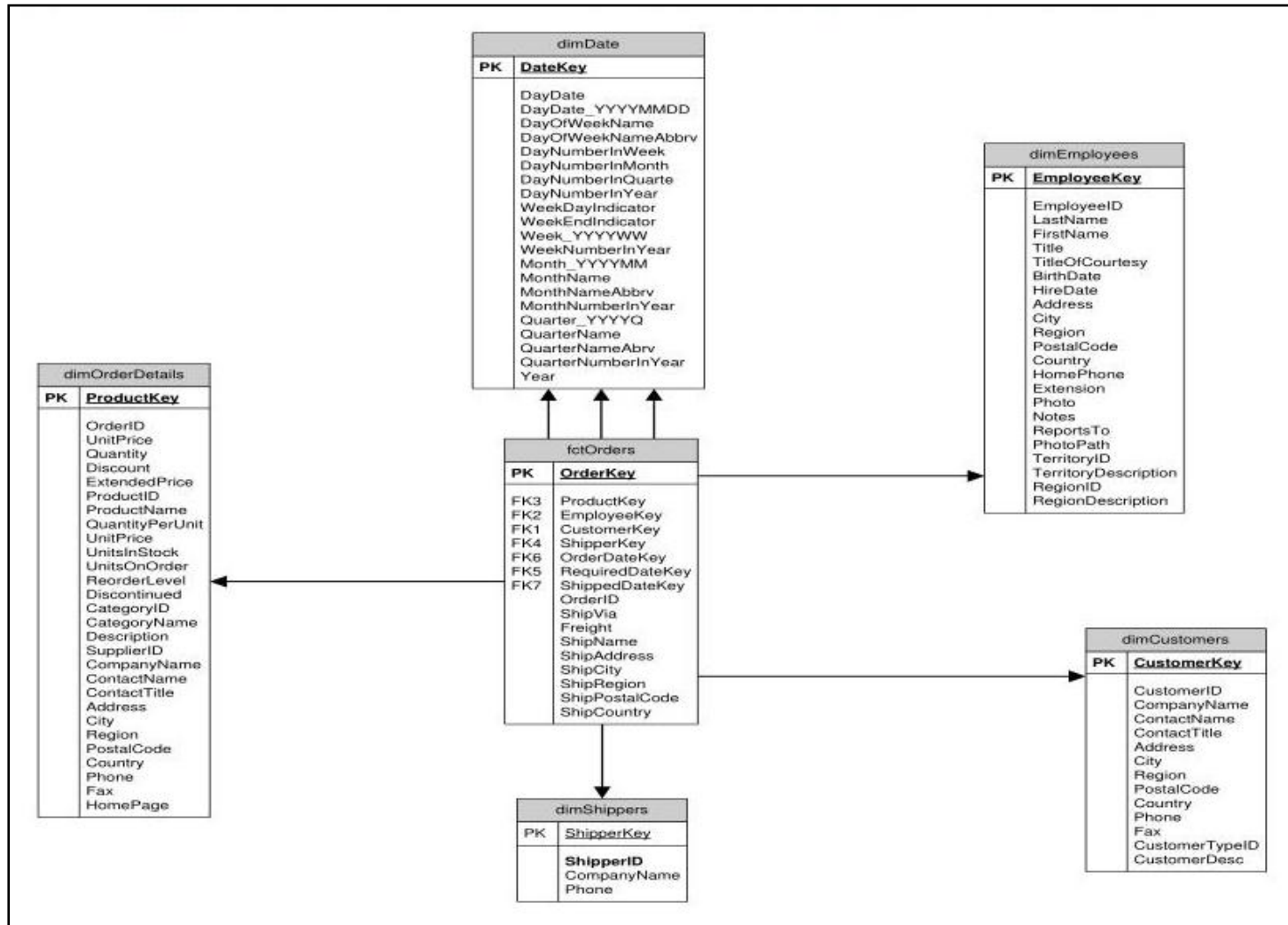
Modern Data Platform: Data Services Pipeline



The Data Warehouse Process

How to Approach Designing and Building a Data Warehouse

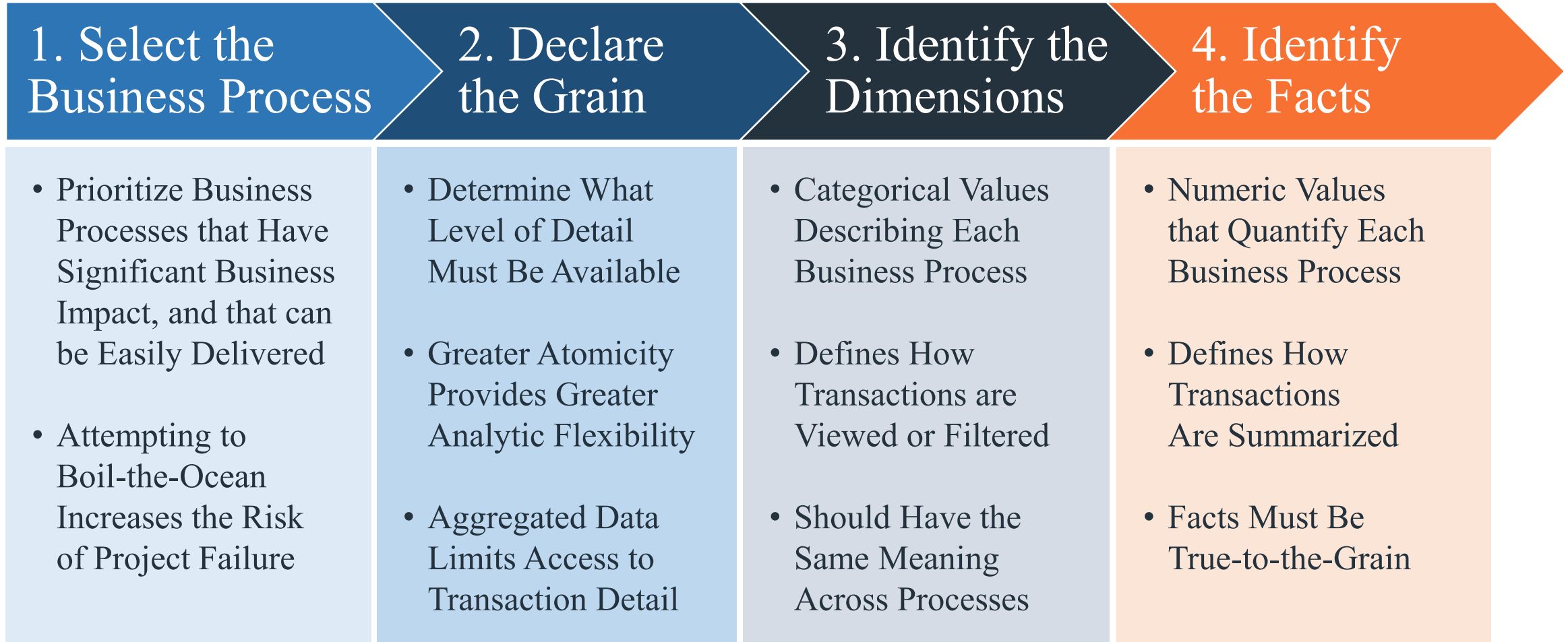
OLAP Database: Multi-Dimensional (Star) Schema





The Four-Step Dimensional Design Process

A Time-Honored and Tested Methodology for Delivering Data Marts & Data Warehouses



The Data Warehouse Toolkit, by Ralph Kimball

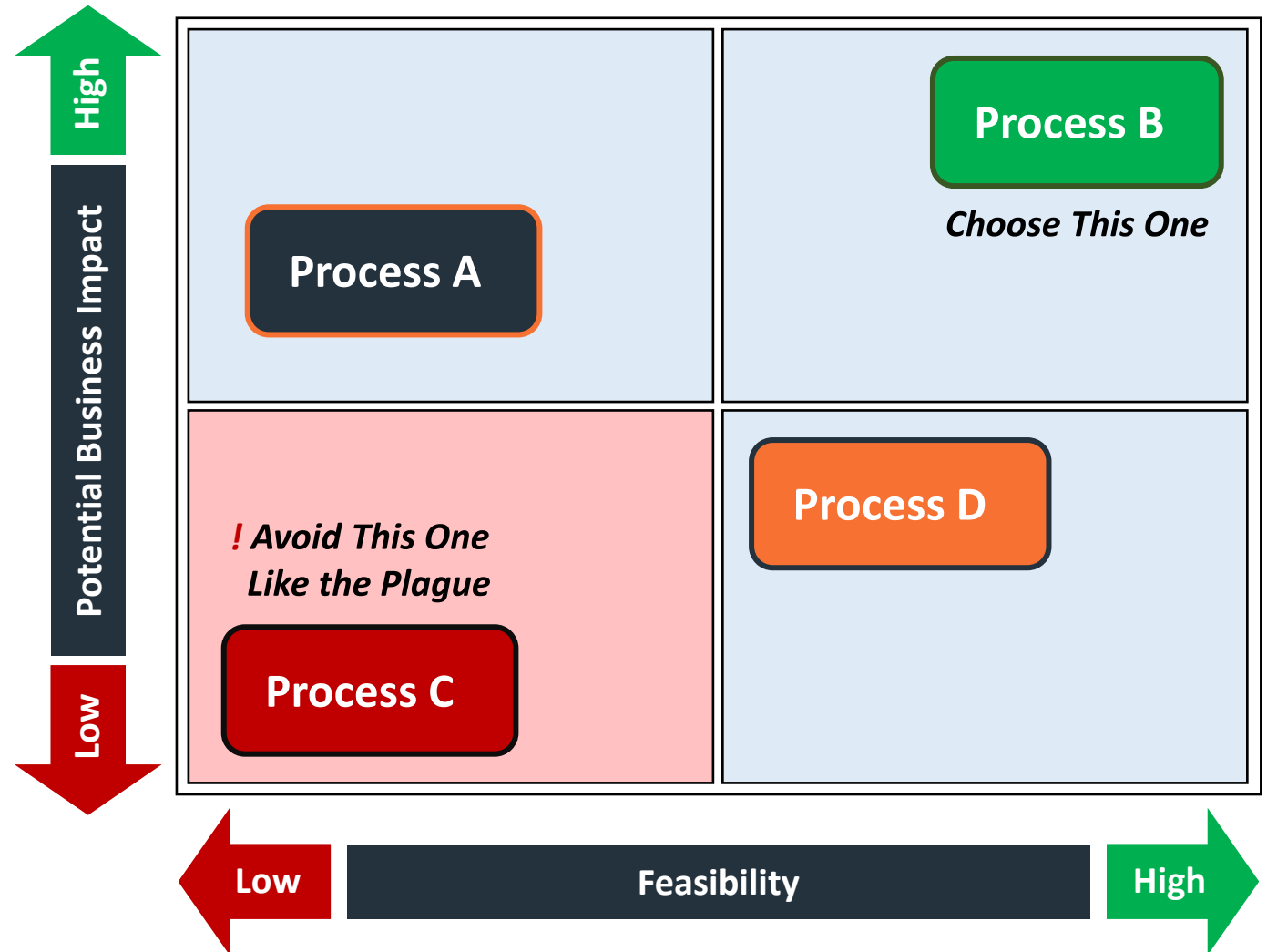
Selecting Business Processes: Prioritizing Requirements



Quadrant Analysis for Prioritizing Requirements:

- **Business Process A:**
 - High Potential Business Impact
 - Extremely Difficult to Implement
- **Business Process B:**
 - High Potential Business Impact
 - Highly Feasible
- **Business Process C:**
 - Very Little Business Impact
 - Extremely Difficult to Implement
- **Business Process D:**
 - Little Business Impact
 - Highly Feasible

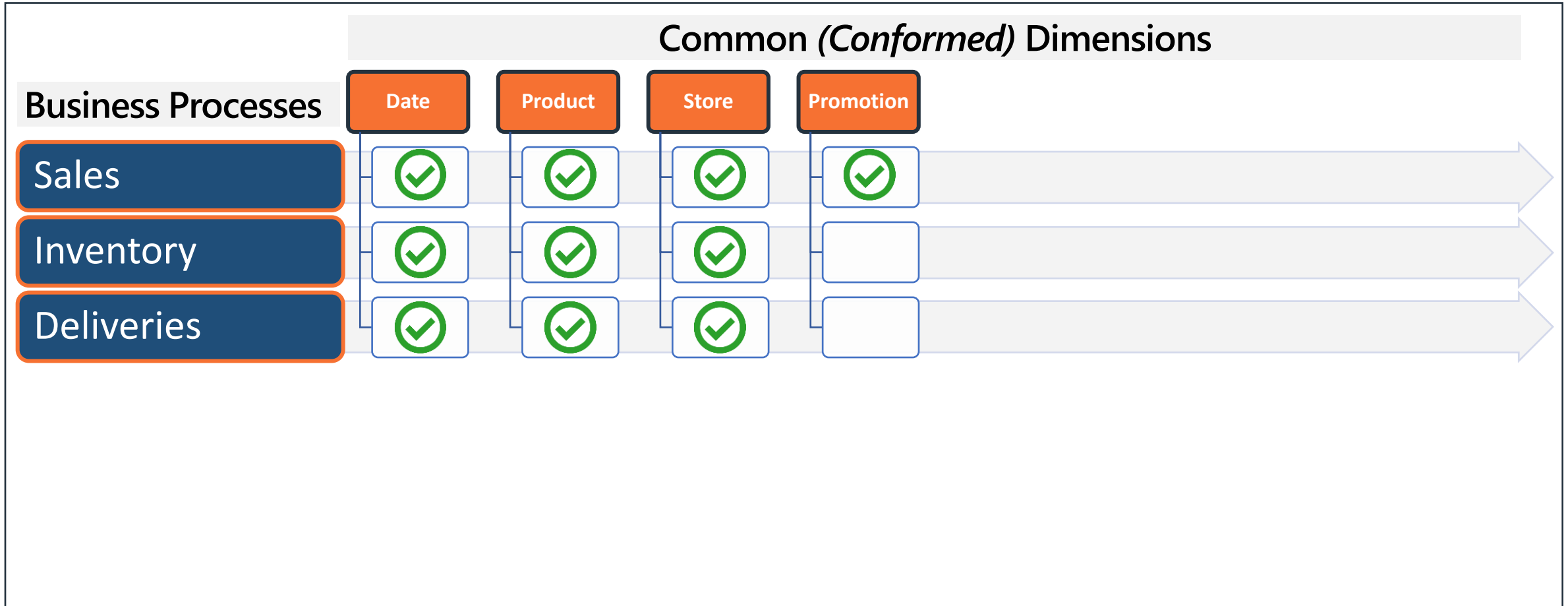
The Data Warehouse Toolkit, by Ralph Kimball



Identifying Dimensions: Data Warehouse Bus Matrix



Using the Same Dimensions Across Multiple Business Processes Enforces a Unified View of the Truth



The Data Warehouse Toolkit, by Ralph Kimball

Identifying Dimensions: Data Warehouse Bus Matrix



Using the Same Dimensions Across Multiple Business Processes Enforces a Unified View of the Truth

Business Processes	Common (<i>Conformed</i>) Dimensions				
	Date	Product	Store	Promotion	
Sales	✓	✓	✓	✓	
Inventory	✓	✓	✓		
Deliveries	✓	✓	✓		
WH Inventory	✓	✓			
WH Deliveries	✓	✓			
Purchase Orders	✓	✓			

The Data Warehouse Toolkit, by Ralph Kimball

Identifying Dimensions: Data Warehouse Bus Matrix



Using the Same Dimensions Across Multiple Business Processes Enforces a Unified View of the Truth

Business Processes	Common (<i>Conformed</i>) Dimensions							
	Date	Product	Store	Promotion	Warehouse	Vendor	Contract	Shipper
Sales	✓	✓	✓	✓				
Inventory	✓	✓	✓					
Deliveries	✓	✓	✓					
WH Inventory	✓	✓			✓	✓		
WH Deliveries	✓	✓			✓	✓		
Purchase Orders	✓	✓			✓	✓	✓	✓

The Data Warehouse Toolkit, by Ralph Kimball

Data Integration

How to Approach Populating a Data Warehouse



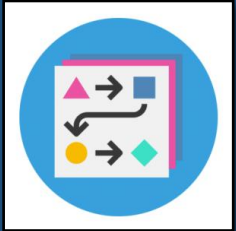
Data Processing: Extract-Transform-Load (ETL)

Frequently, Data Must Be Moved from Sources to a Database and/or Data Lake



Extract

- This is the step where sensors wait for upstream data sources to land. Once available, we transport the data from their source locations to further transformations.



Transform

- The heart of any ETL job: apply business logic, perform actions such as filtering, grouping, and aggregation to translate raw data into analysis-ready datasets.



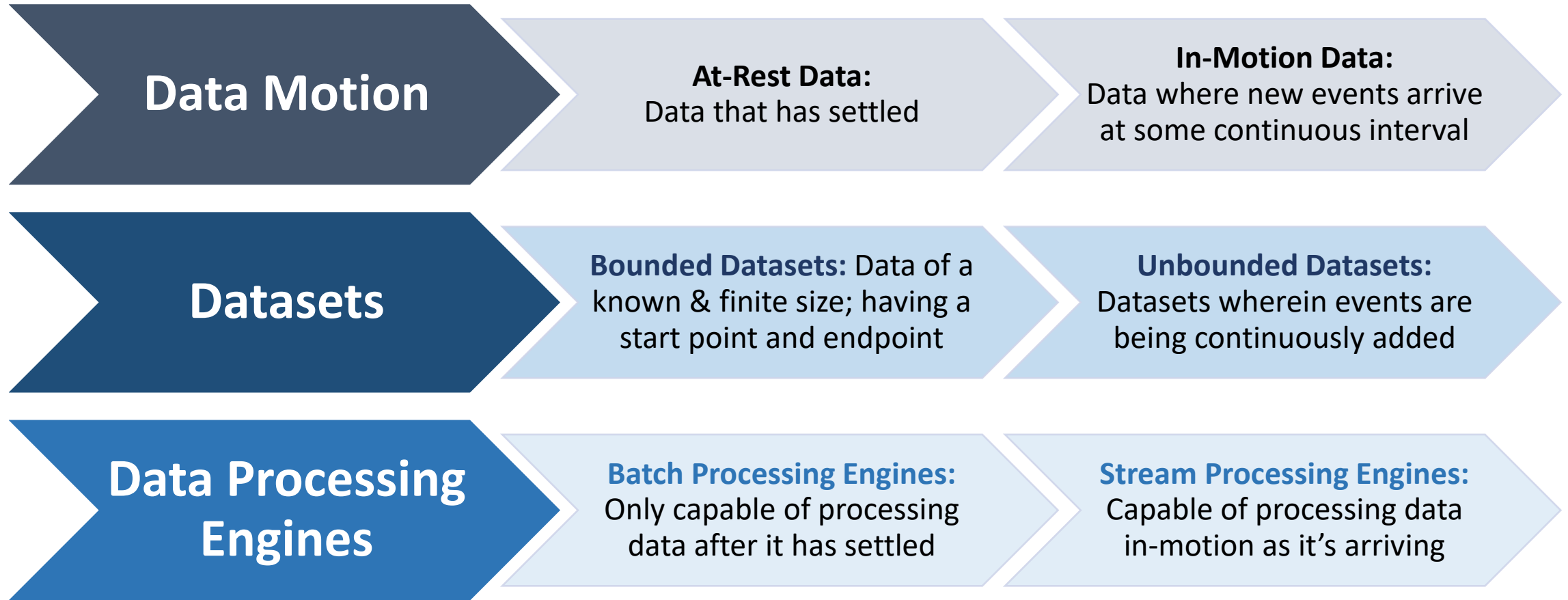
Load

- Load the processed data and transport to a final destination. Can now be consumed directly by end-users or treated as yet another upstream dependency.



Data Processing: **Batching versus Streaming**

The rate at which incoming data must be integrated dictates how it must be processed



Data Processing Paradigms: Latency Requirements



Latency & Response:

The speed at which clients require new insights determines the frequency at which new data must be processed

1. Batch

2. Continuous/Streaming

3. Real-time

10 ms

100 ms

1 sec

1 min

1 hour

1 day

Low-Latency Real-Time

- Spark-less, highly-available prediction server

Real-Time

- Prediction server with Spark

Micro-Batch

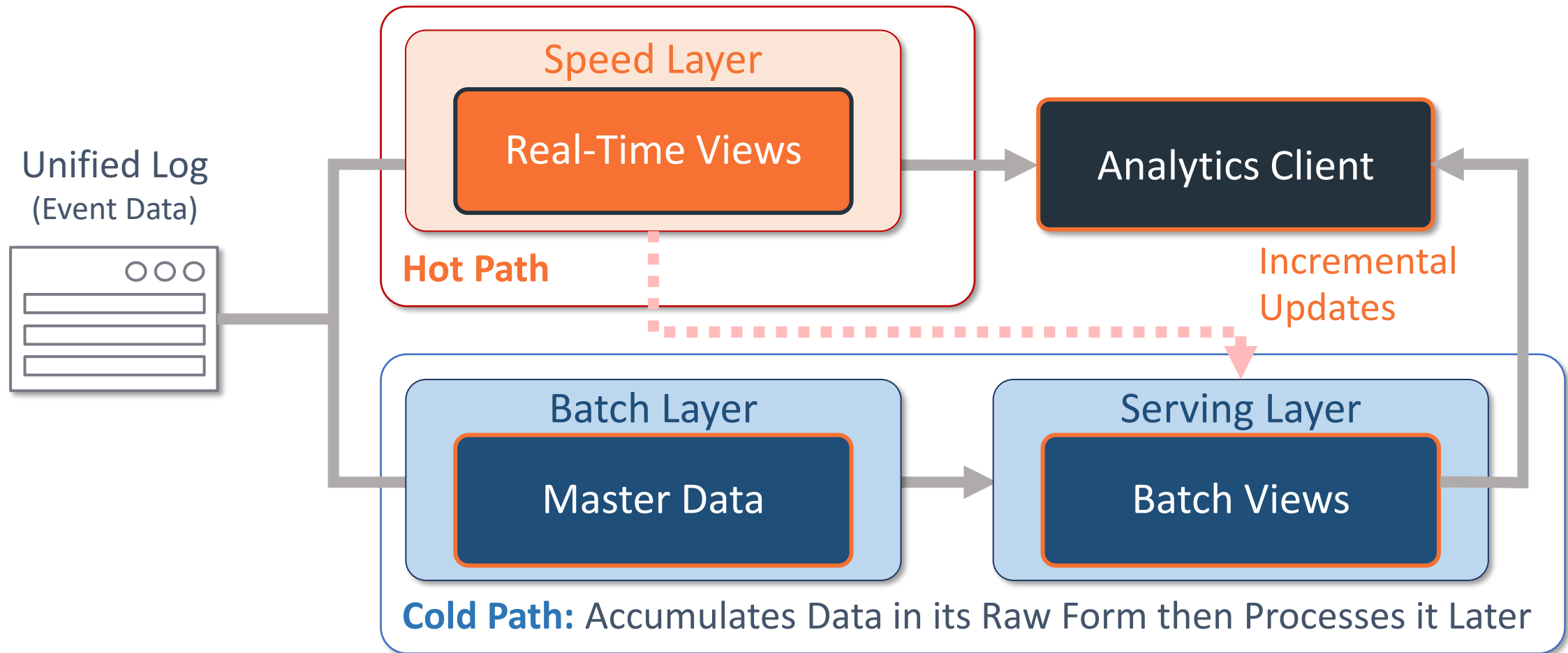
- Structured Streaming

Batch

- Spark batch processing

Data Processing Paradigms: Lambda Architecture

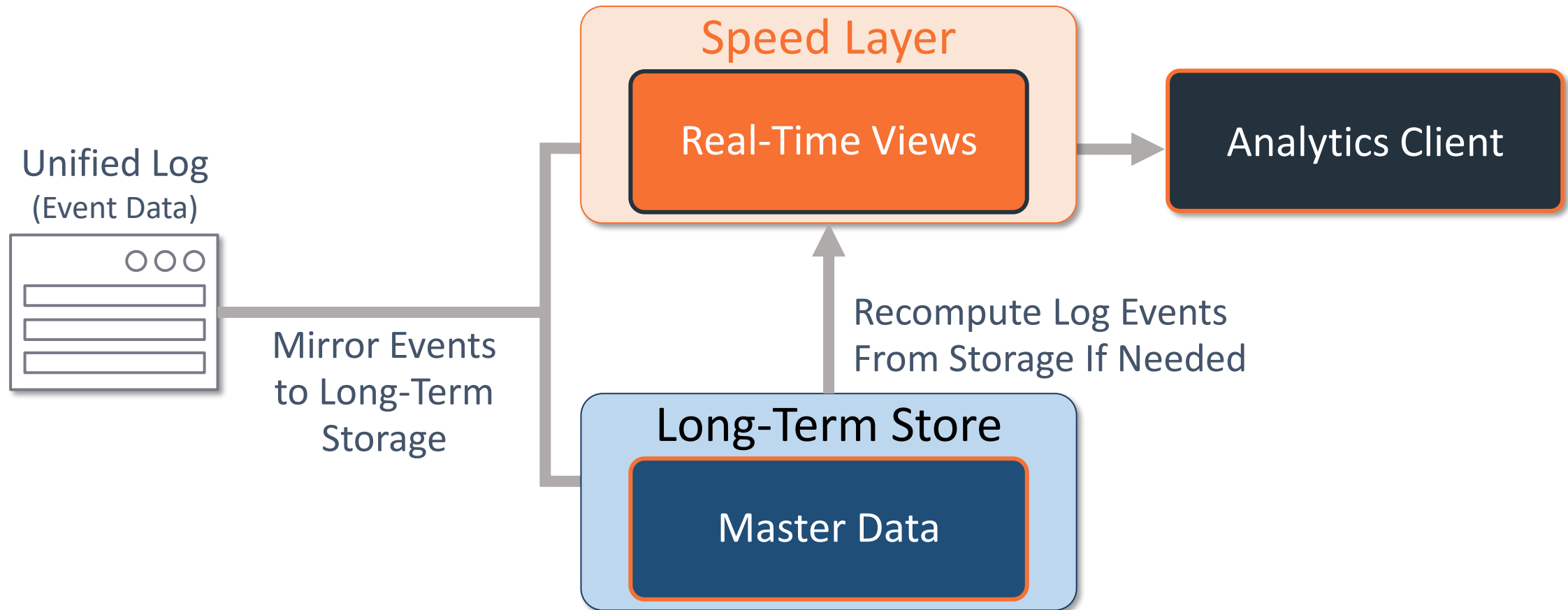
All Data Flows Through One of Two Paths: Hot or Cold





Data Processing Paradigms: Kappa Architecture

All Data Flows Through One of Two Paths: Hot or Cold

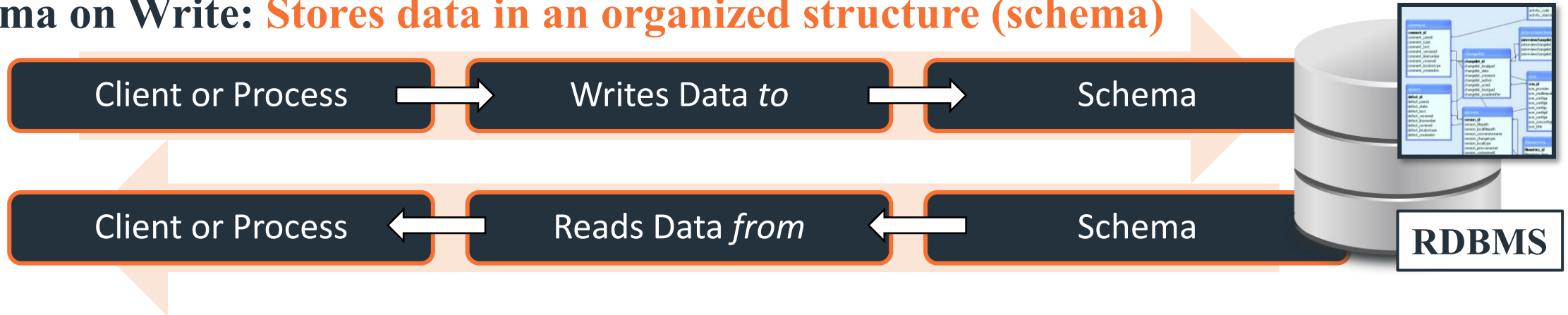




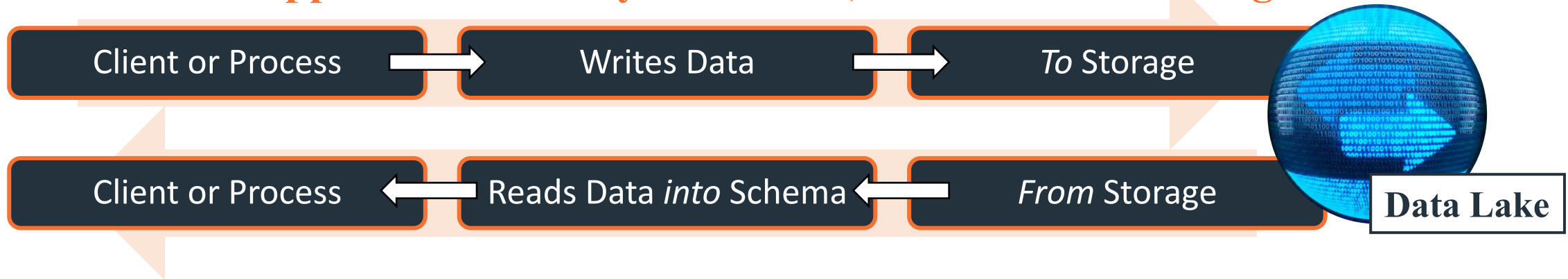
Paradigms: Data Storage and Retrieval

Schema on Write versus Schema on Read

Schema on Write: Stores data in an organized structure (schema)



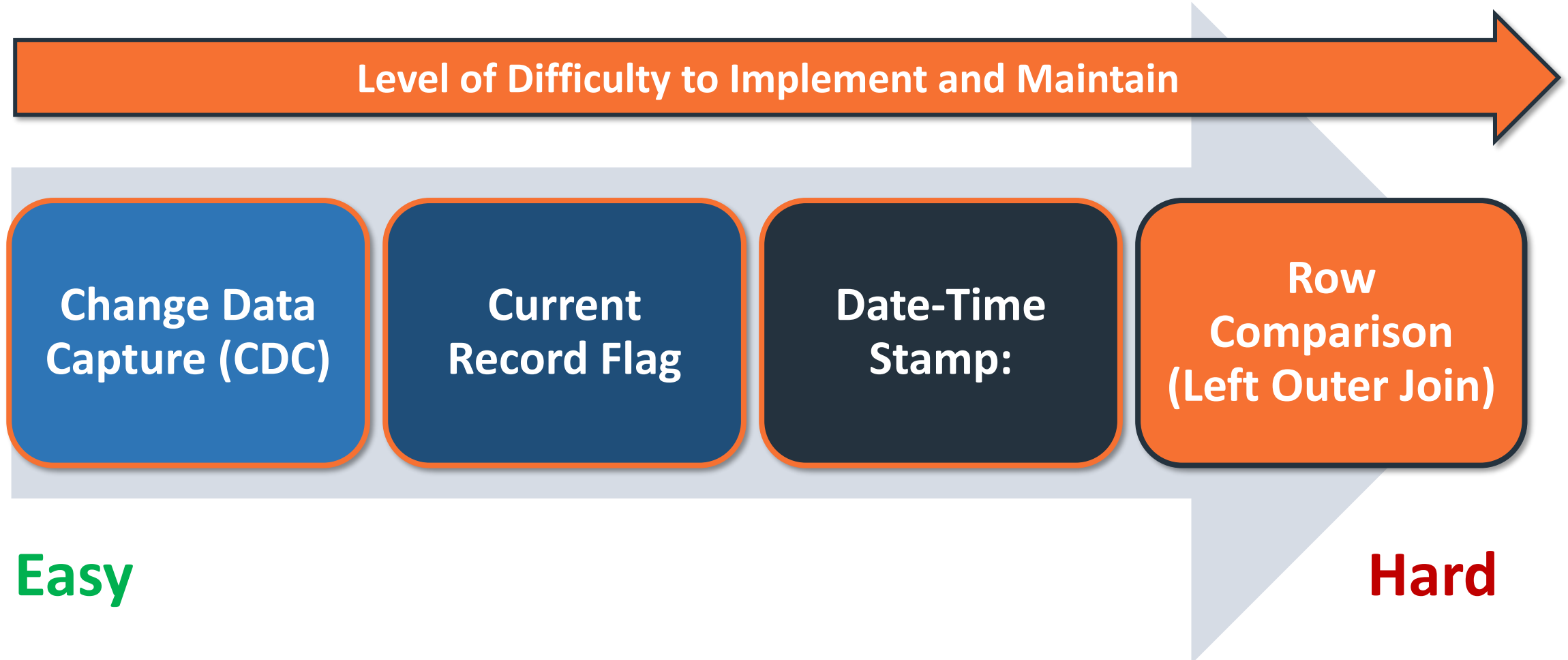
Schema on Read: Applies schema only when read, data stored in its original format





ETL Processing: Incremental Extraction

Techniques for Minimizing Data Movement: Extract Only the Changes





Data Integration Patterns: Dimensional Data

Slowly Changing Dimension Update Strategies: Handling Variable Rates of Change

SCD Type 0

- Data in the Column Never Changes: Ever!
- Only for Static Reference Data

SCD Type 1

- No History is Maintained
- Existing Values are Overwritten by New Values
- **UPDATE**

SCD Type 2

- Historic Values are Maintained
- New Values are Written to a New Row
- *IsCurrent* Flag
- **INSERT**

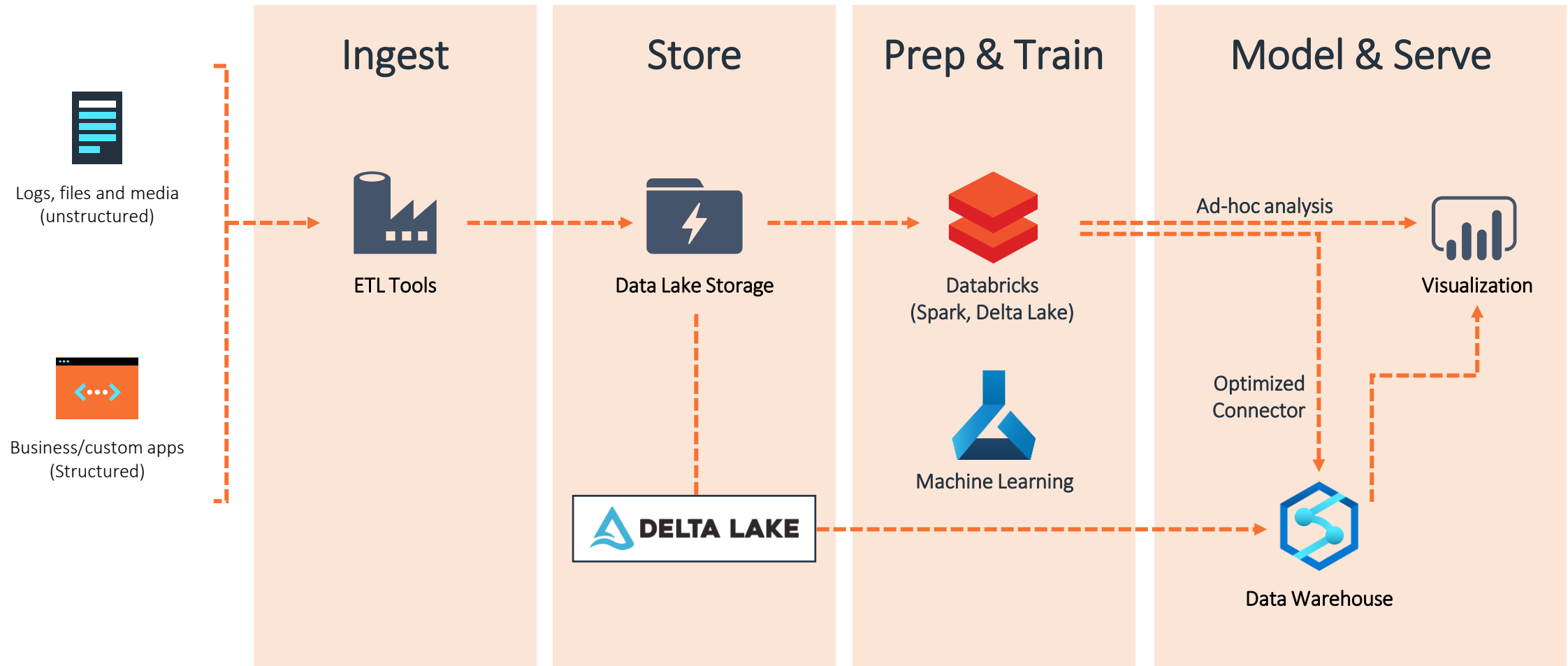
SCD Type 3

- A New Current Value Column is Created in the Existing Record
- Original Column is Also Retained

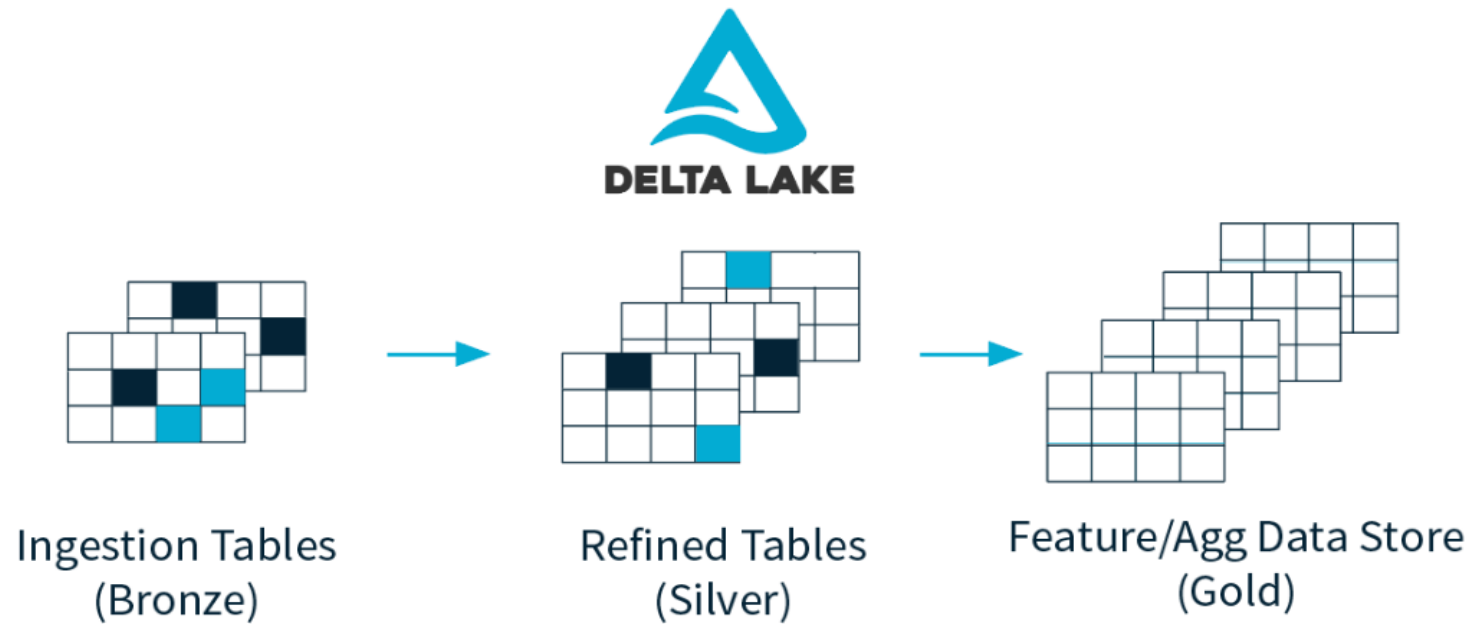
Easier to Implement and Maintain

More Difficult to Implement and Maintain

Data Engineering... for Data Science



Databricks: Delta Lake at Scale



ACID Transaction Guarantees

Atomic, Consistent,
Isolated, Durable

Versioned Parquet Files

Delta transaction log keeps
track of all operations

Efficient Upserts

MERGE, DELETE, UPDATE

Time Travel

Audit history, pipeline
debugging, data
reproducibility

Small file compaction with

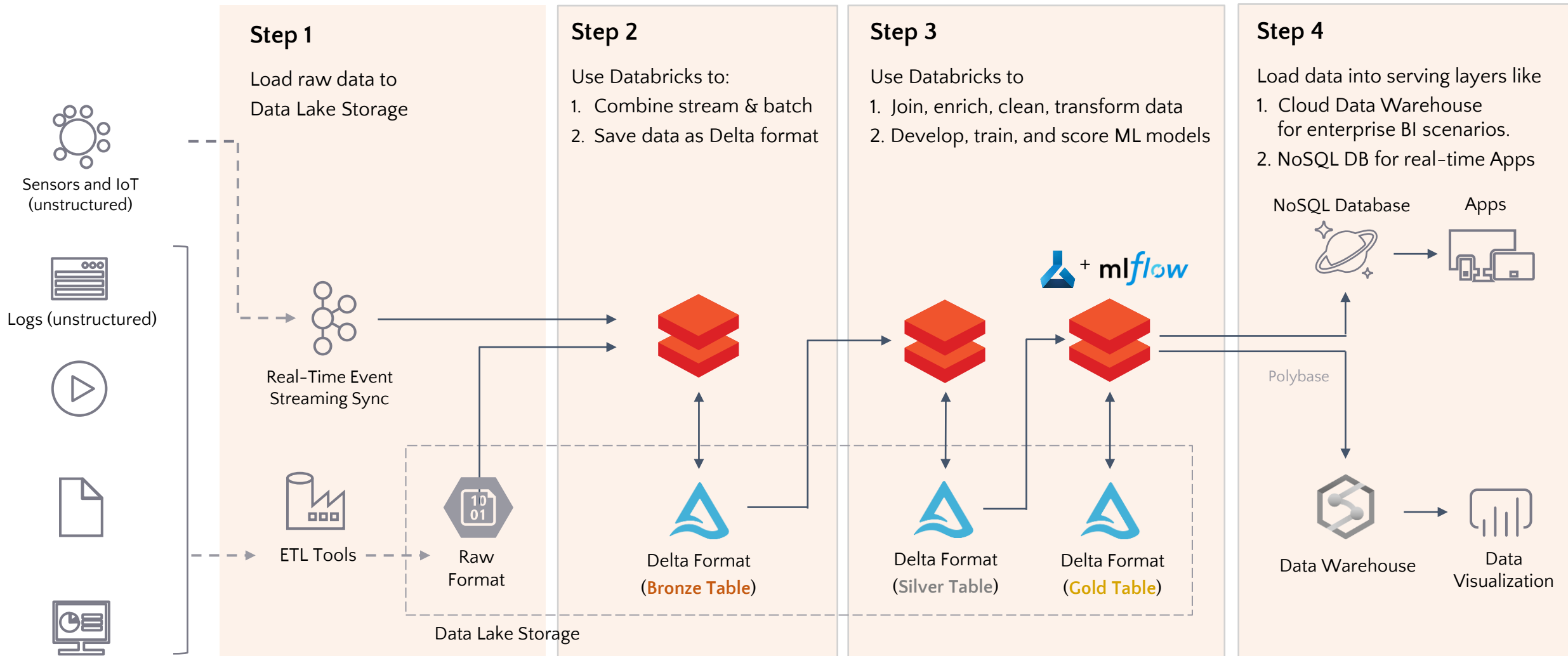
no interrupt to availability

OPTIMIZE and VACUUM

Z-Order partitioning with up to 100x perf

New multidimensional partitioning enables
data skipping

Design Pattern: Modern Data Warehousing



Q & A

An Overview of Data Warehouse Systems

