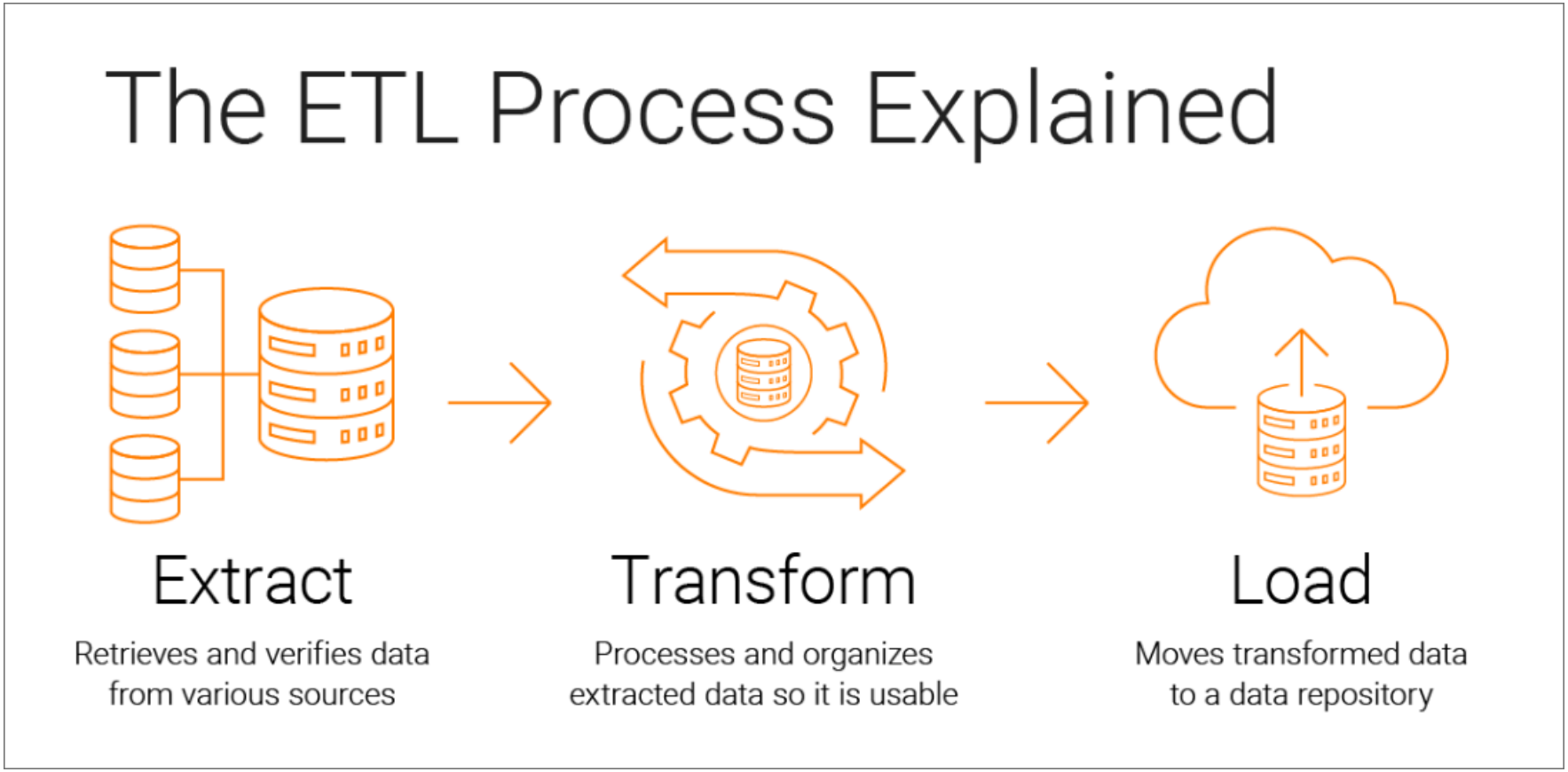
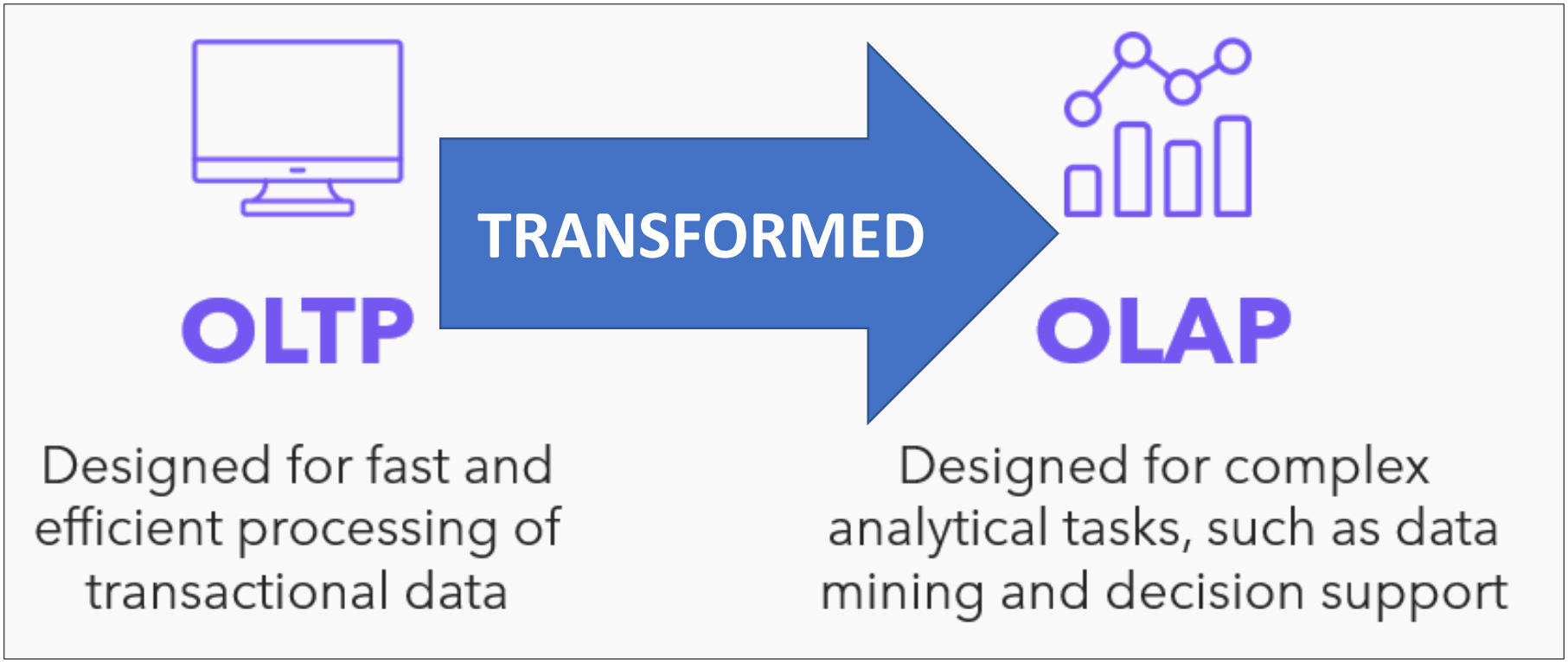
Data Warehouse Final Review

ETL: Extract – Transform - Load



OLTP – Online Transaction Processing; extract from
OLAP – Online Analytical Processing; load into



Extract Transform Load [ETL] Pipeline

Sources of Data

- OLTP databases
- MySQL databases
- NoSQL databases
- Spreadsheets
- CSV files
- Web services/APIs

Extract → Staging database → Transform & Load → Data Warehouse → Move data → Business Intelligence Applications

Flow Of Data

Extract Load Transform [ELT] Pipeline

Sources of Data

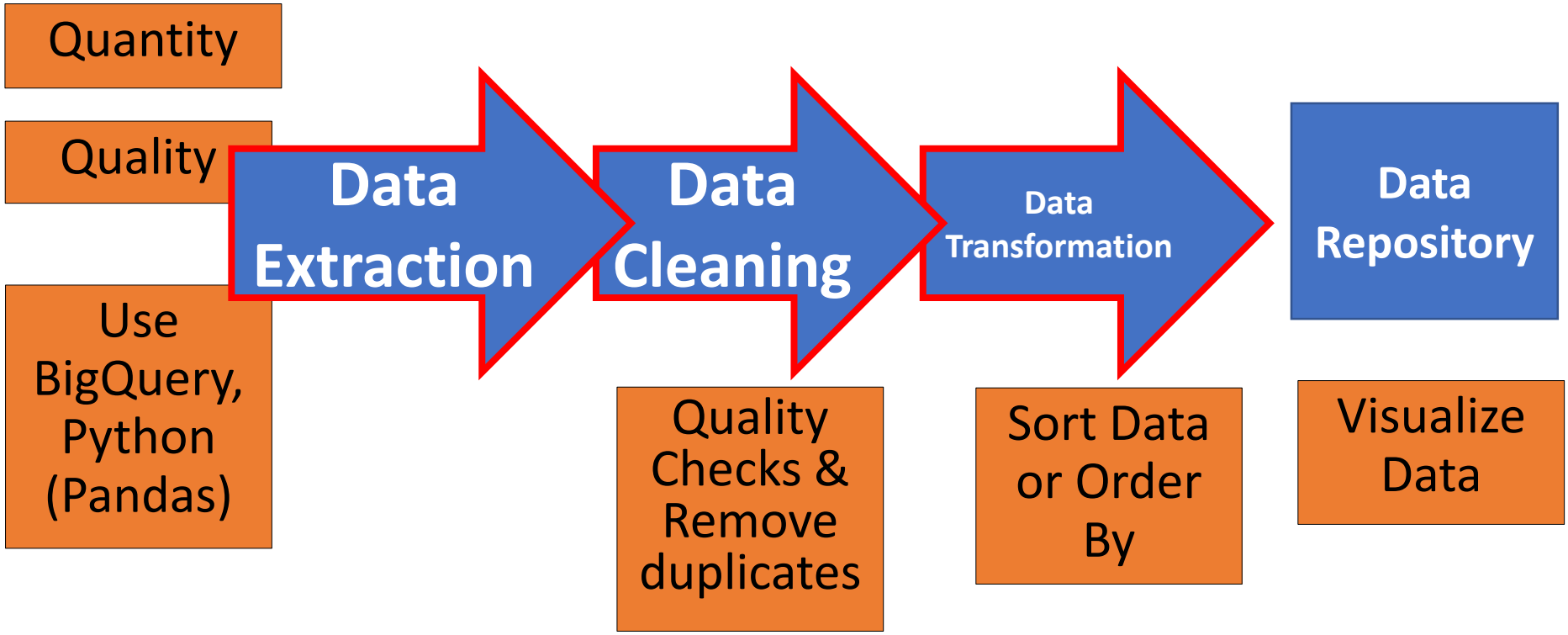
- OLTP databases
- MySQL databases
- NoSQL databases
- Spreadsheets
- CSV files
- Web services/APIs

Extract → Load → Data Warehouse → Transform → Move data → Business Intelligence Applications

Flow Of Data

Data Profiling

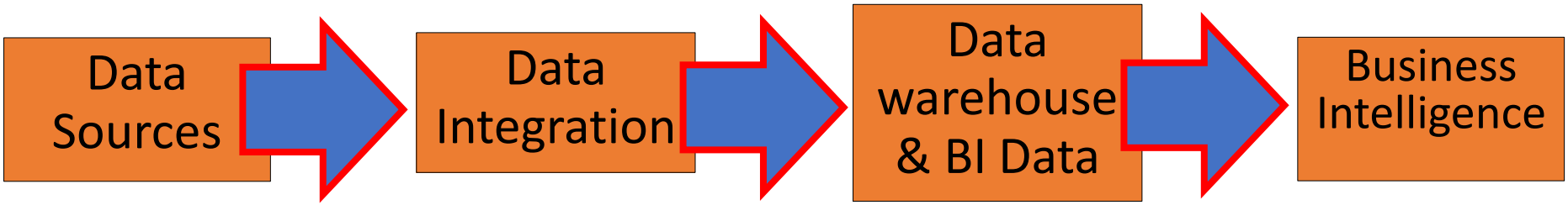
Process of analyzing and creating useful summaries of data



Kimball has 34 steps in this process

Data Warehouse Architecture

Technical Architecture defines the technologies used to implement and support a BI solution



Technology Platforms

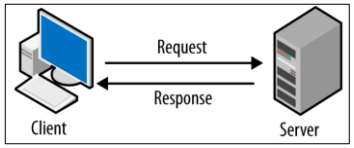
Evolution not Revolution

Mainframe computers

Minicomputers

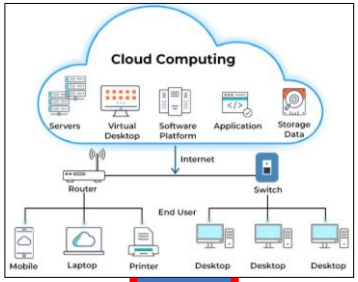
PCs

Client/Server systems



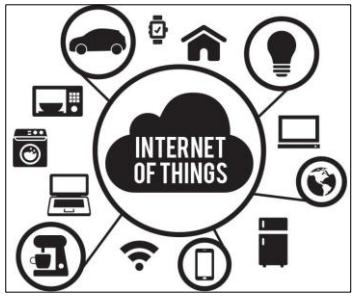
Web-based service oriented architecture

Cloud computing



Big Data

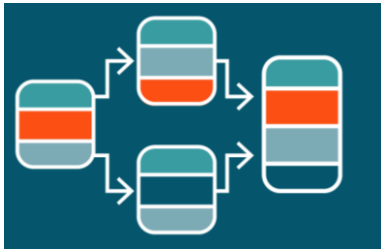
Internet of Things (IoT)



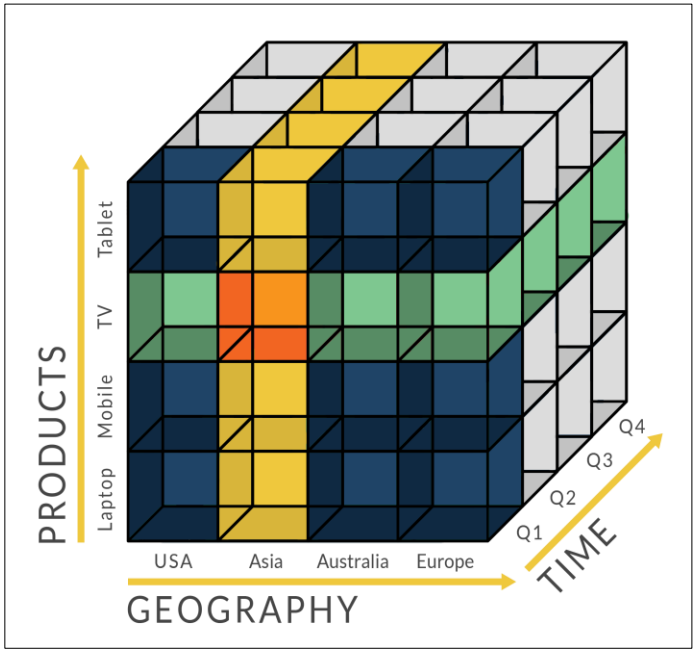
Older to Newer

Architecture of Databases

Relational Databases 1970s



OLAP: multidimensional databases 1990s



Column-oriented databases 2000s

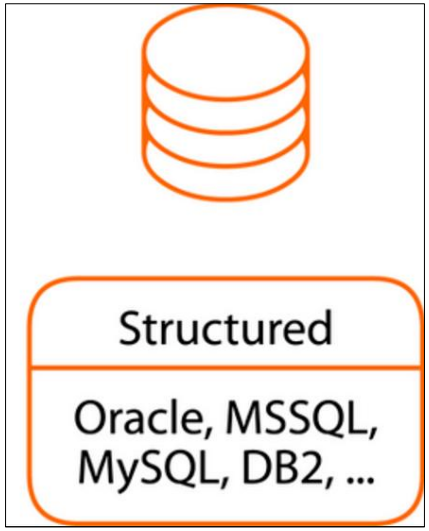
Column oriented

Students		
ID	**First name**	**Last name**
1	Luna	Lovegood
2	Hermione	Granger
3	Ron	Weasley

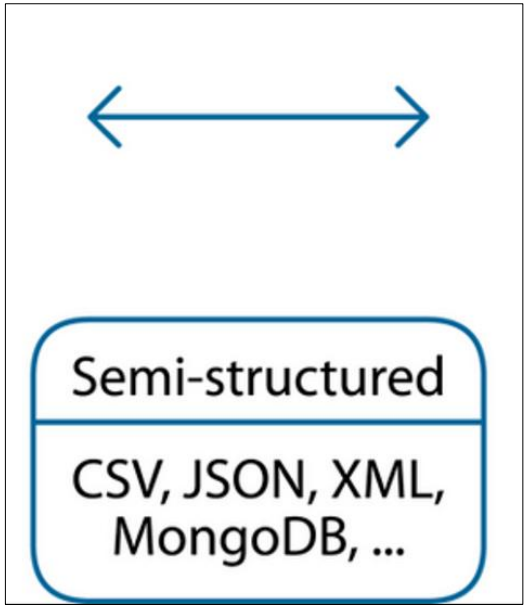
Older to Newer

Big Data Database Types

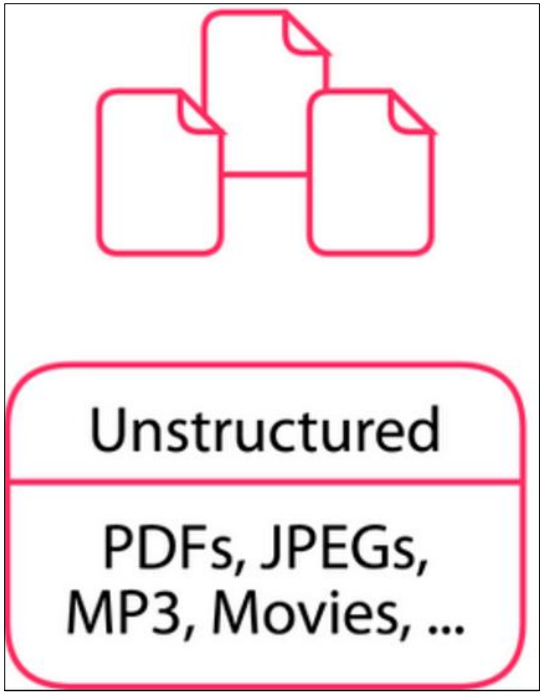
Structured:
data from enterprise applications



Semi-Structured:
Machine data from Internet of Things (IoT)

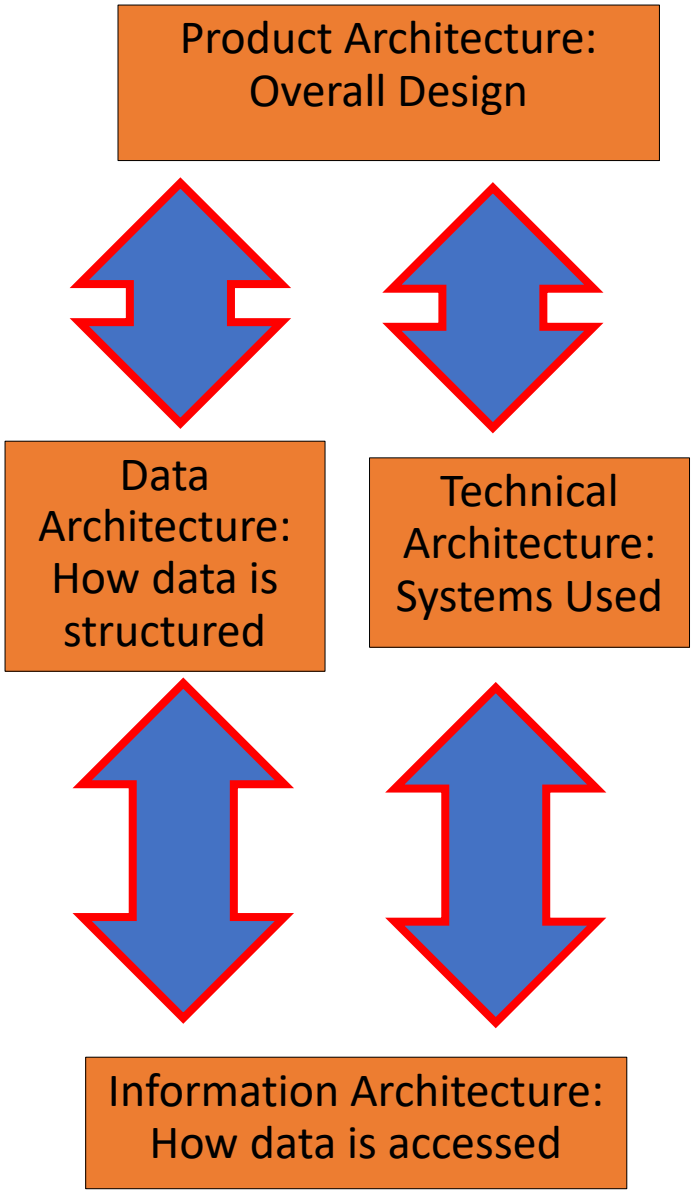


Unstructured:
Text, audio, video from Web



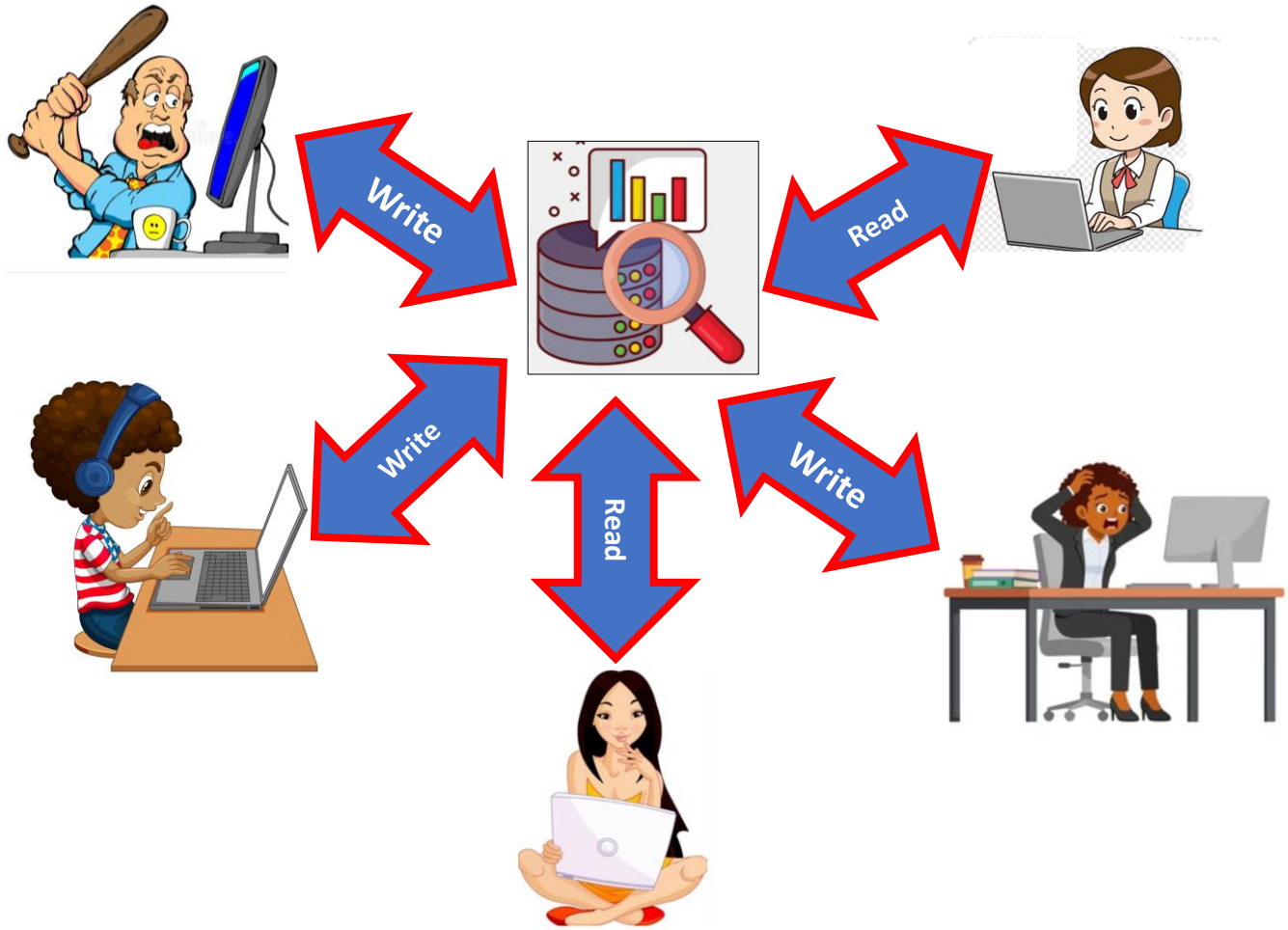
More to less structure

Product Architecture



CAP Theorem

Multi-User Databases
More than 1 person accessing and modifying data at the same time



Transactions: read & write operations that either commit or abort
Transactions need to be controlled: concurrency

ACID Properties

Focus on consistency

Atomic:
Commits
or
Aborts

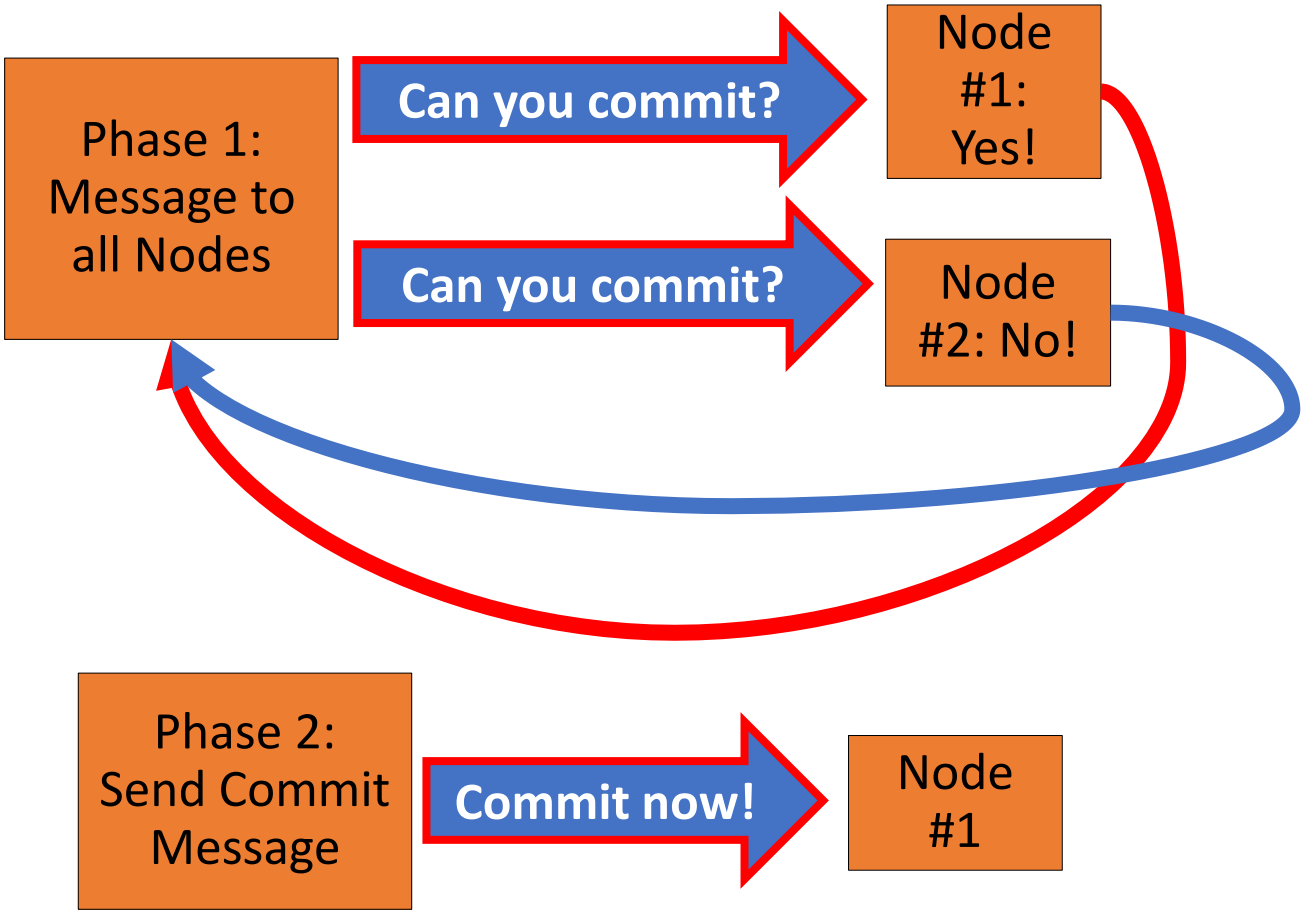
Consistency:
1 consistent
state to another
consistent state

Isolation:
Remains
unknown until
commit, ie:
locked

Durable:
Work not lost
even if future
failure

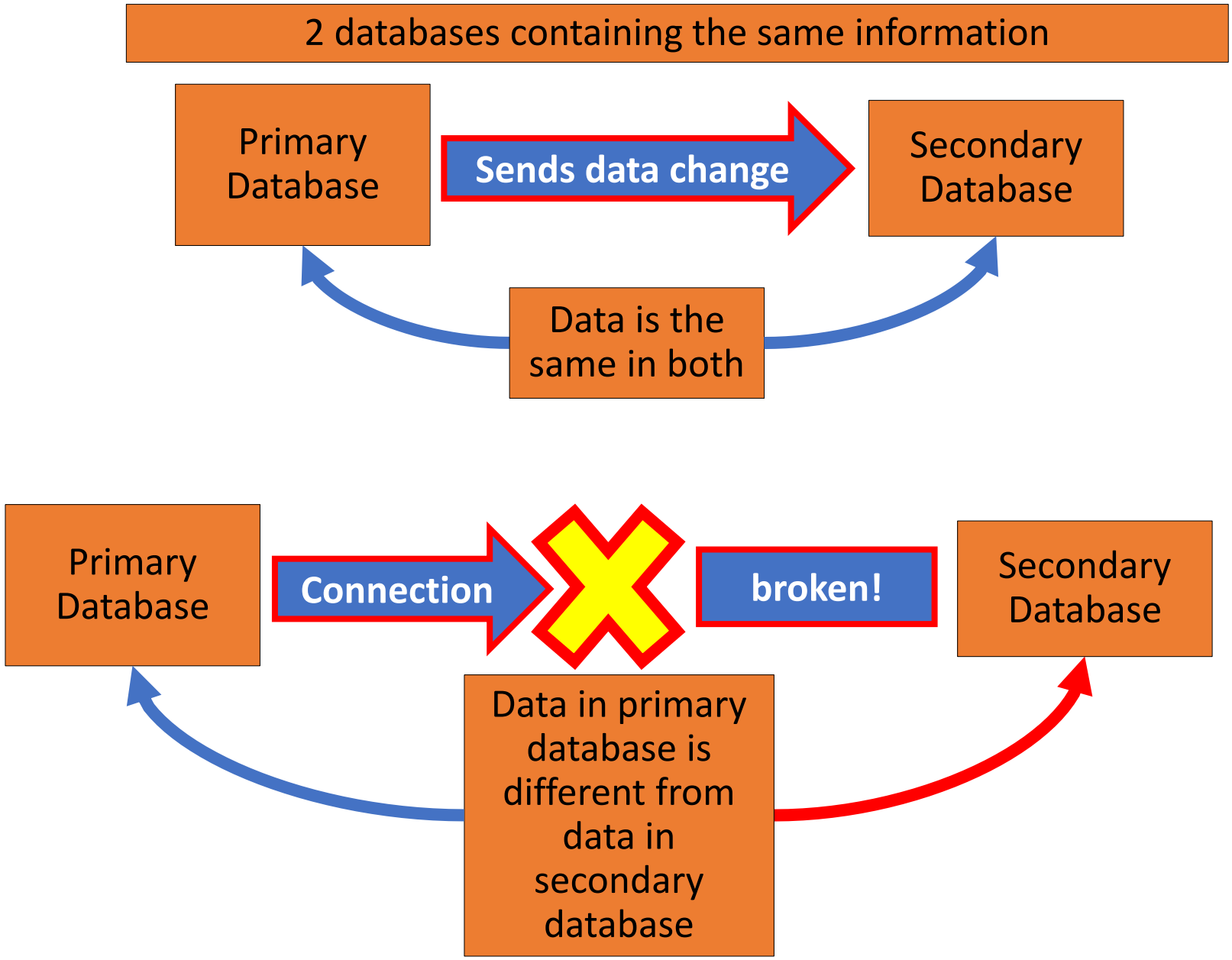
Distributed Commit Protocol

2 Phase Commit aka Synchronous Replication Protocol



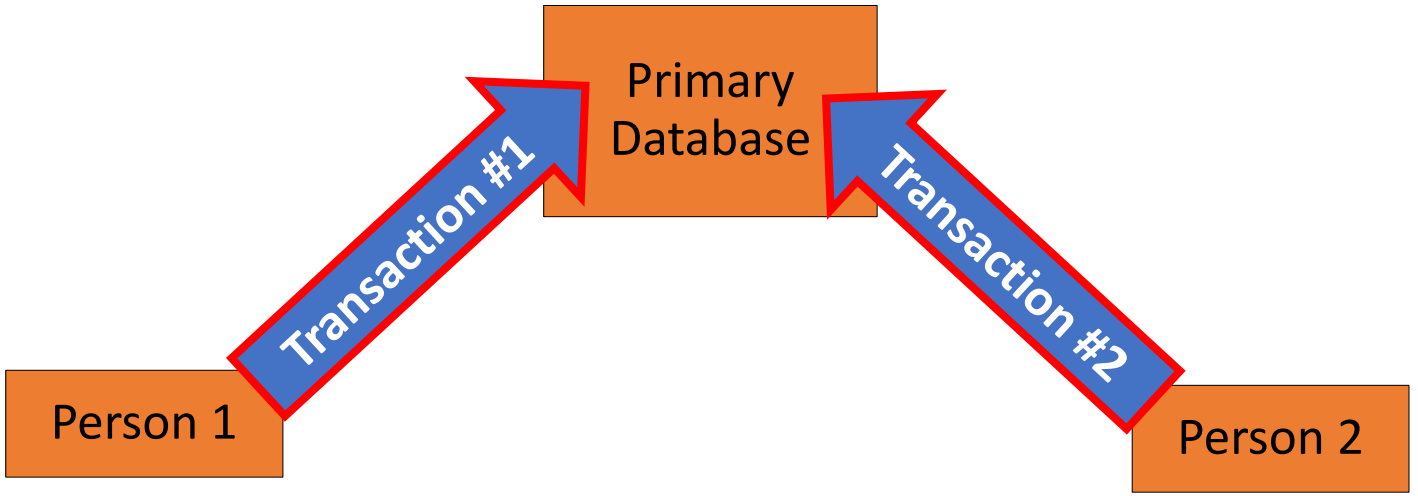
Guarantees all replicas are consistent

Distributed Database

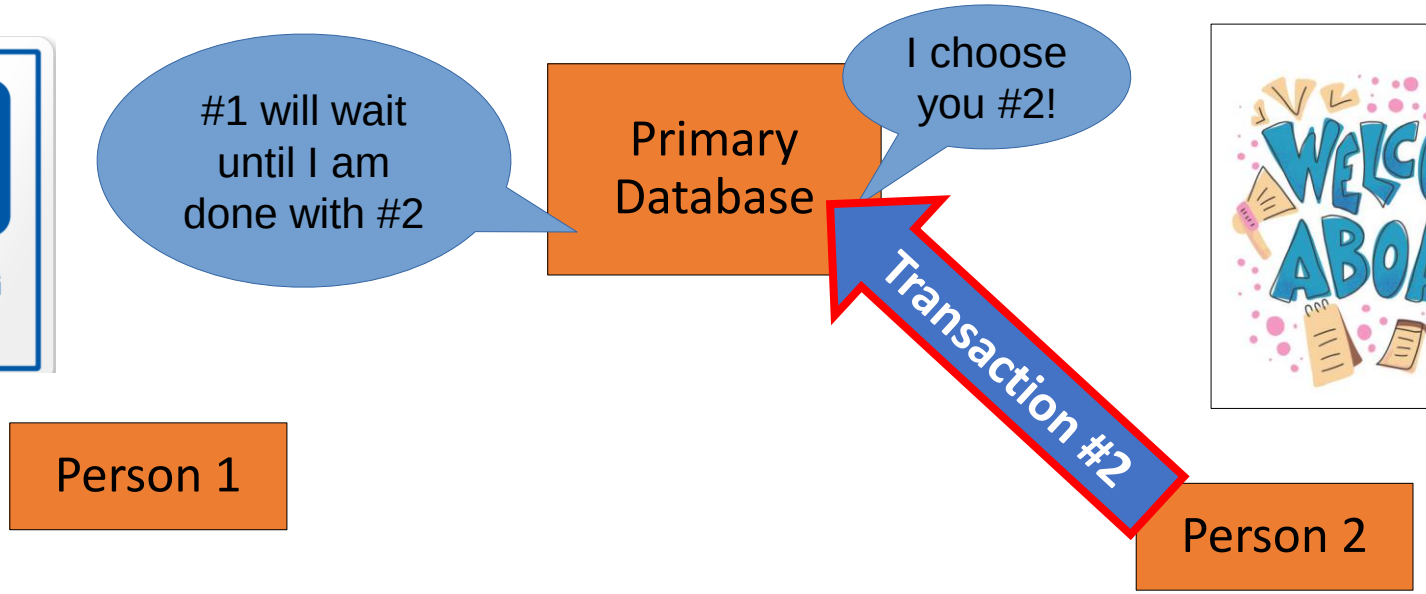


Concurrent Transactions [Isolation]

2 transactions occur at the same time; what to do?



Database will choose ONE of the transactions



CAP Theorem aka Brewer's Theorem

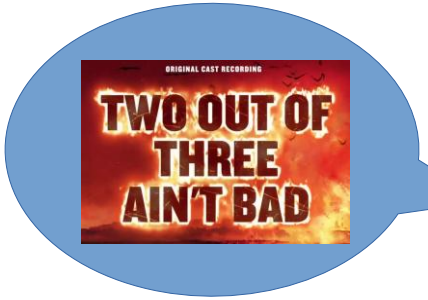
C A P

Consistency

Availability

Partition Tolerance

Distributed database can only have 2 out of 3 of the CAP



Primary Database

Song:
"Two Out Of Three Ain't Bad" by Meatloaf

Consistency: All nodes have same data @ same time

Availability: All requests responded to. But NO guarantee of returning most recent write

Partition Tolerance: System stays up in spite of netowrk failures

Networks FAIL!

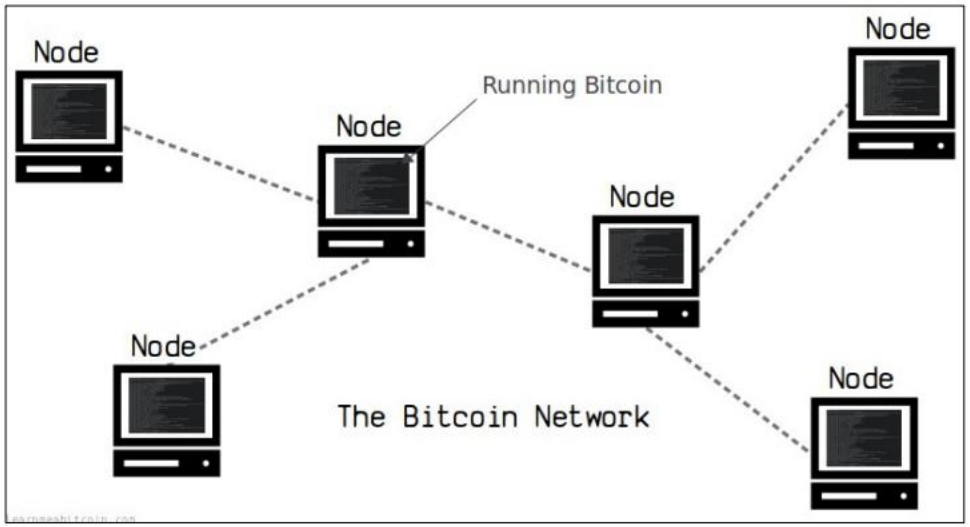
Slow or unavailable connections

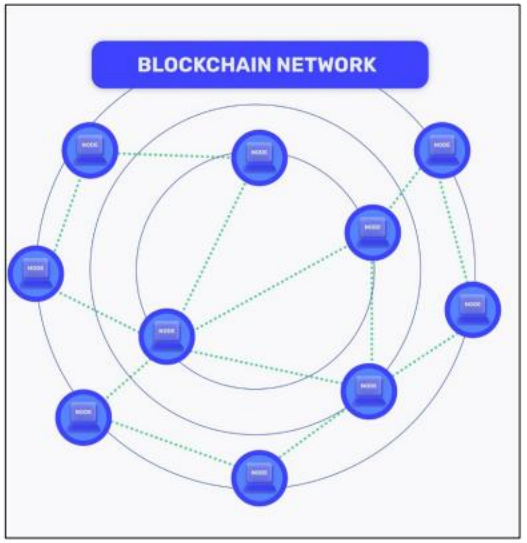
In distributed systems, **P**artition Tolerance is a MUST!
Must choose between **C**onsistency and **A**vailability

Remember, 2 out 3 ain't bad...

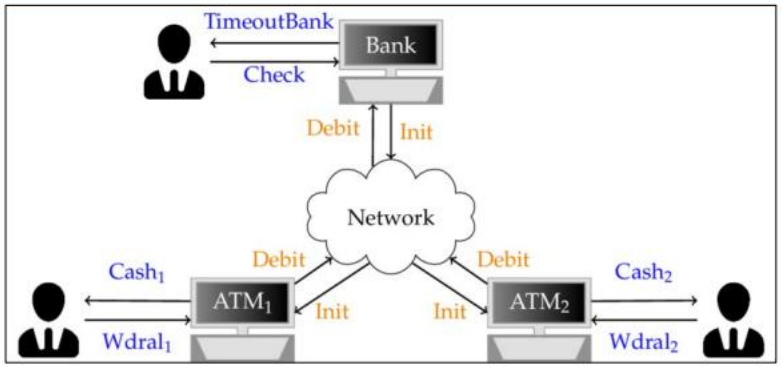
Availability vs Partition Tolerance

Partition Tolerance: Network with broken connection but nodes still operate, eg: Bitcoin & blockchain network





Availability: All requests getting responses within acceptable time eg: Bank ATM networks



Databases

Consistency and Patition Tolerance

Consistency and Availability

Availability and Partition Tolerance

Consistency in CAP is different than Consistency in ACID

If distributed database guarantees ACID →
Must choose Consistency over Availability (CP)

If a distributed database chooses Availability over Consistency (AP)
→ cannot provide ACID

Trade Offs in Distributed Systems is Real
No Right Answers!

Depends Upon Situation

Sometimes availability is more important
(e.g., financial transactions and compensation fees)
→ will get eventual consistency

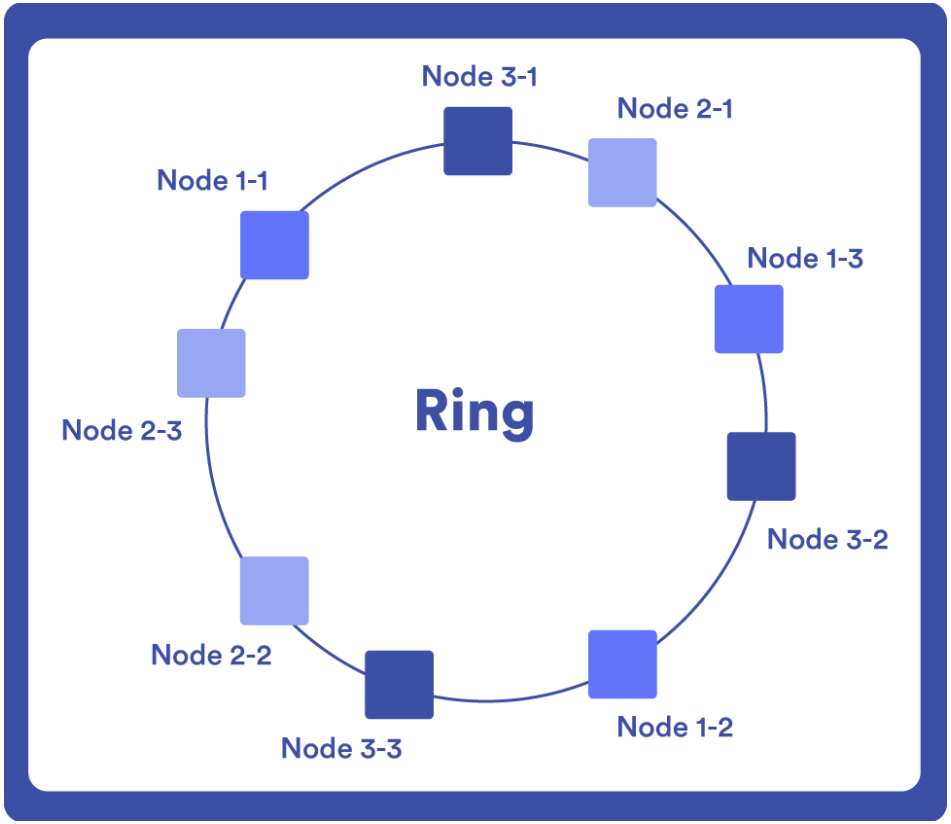
Other times consistency is more important
(e.g., multiples users with the same view, medical records)
→ cost of inconsistency is higher than unavailablity

NoSQL Databases

Non-relational, dynamic schema

MongoDB.

Scale out – add nodes



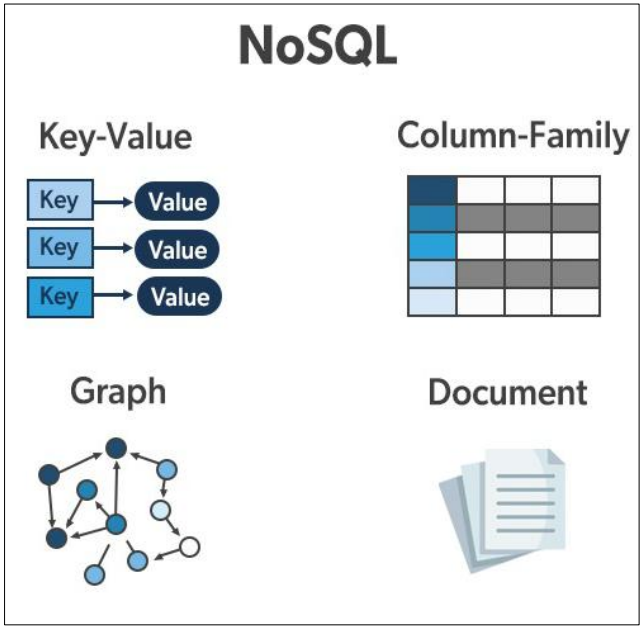
NoSQL Databases – 4 Core Types

Key-Value pair, eg: Riak

Document Store, eg: MongoDB

Column-Store, eg: Cassandra

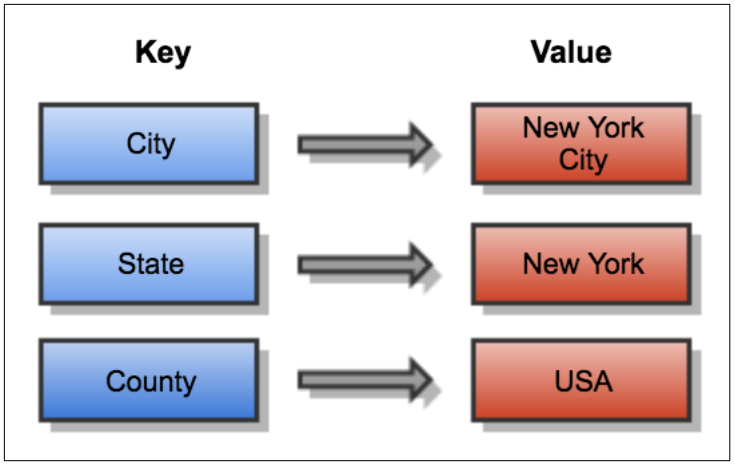
Graph, eg: Neo4J



Key-Value pair, eg: Riak

Simplest type of NoSQL

Each item only has two fields: unique key and value



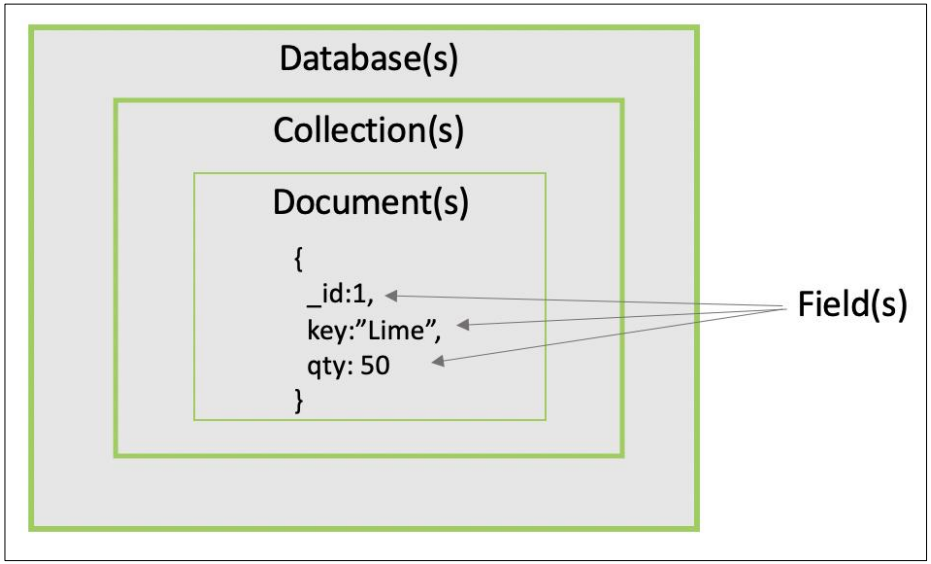
Key can be simple sequential number

Due to simplicity, key-value has excellent performance

Document: set of ordered key-value pairs

Collection: group of documents

Collection contains related documents, ie: inventory



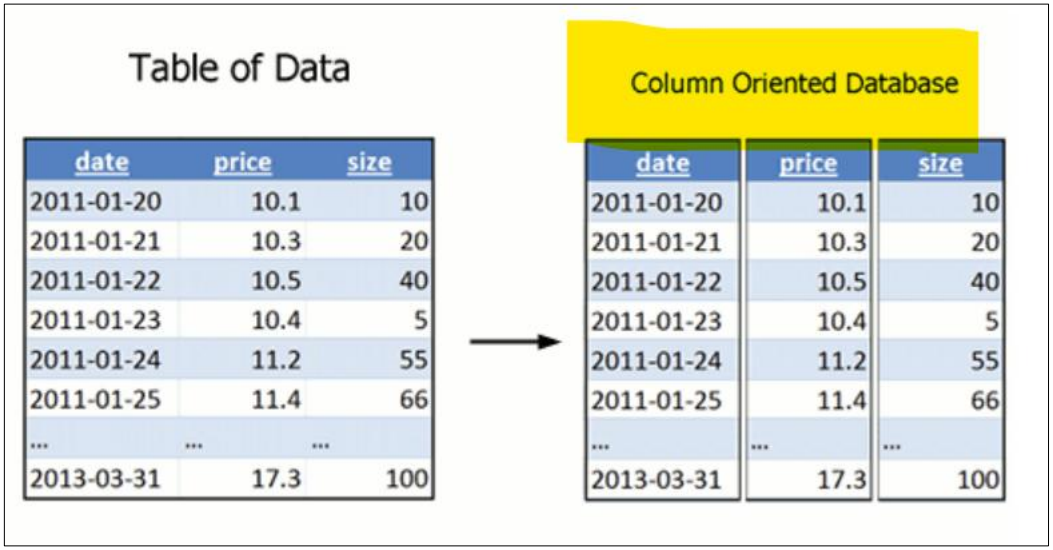
Schemaless – Provide flexibility

MongoDB is an example

Column store database

Column operations are faster

Column: data structure for storing single value



date	price	size
2011-01-20	10.1	10
2011-01-21	10.3	20
2011-01-22	10.5	40
2011-01-23	10.4	5
2011-01-24	11.2	55
2011-01-25	11.4	66
...	...	...
2013-03-31	17.3	100

date	price	size
2011-01-20	10.1	10
2011-01-21	10.3	20
2011-01-22	10.5	40
2011-01-23	10.4	5
2011-01-24	11.2	55
2011-01-25	11.4	66
...	...	...
2013-03-31	17.3	100

Set of columns make up a row

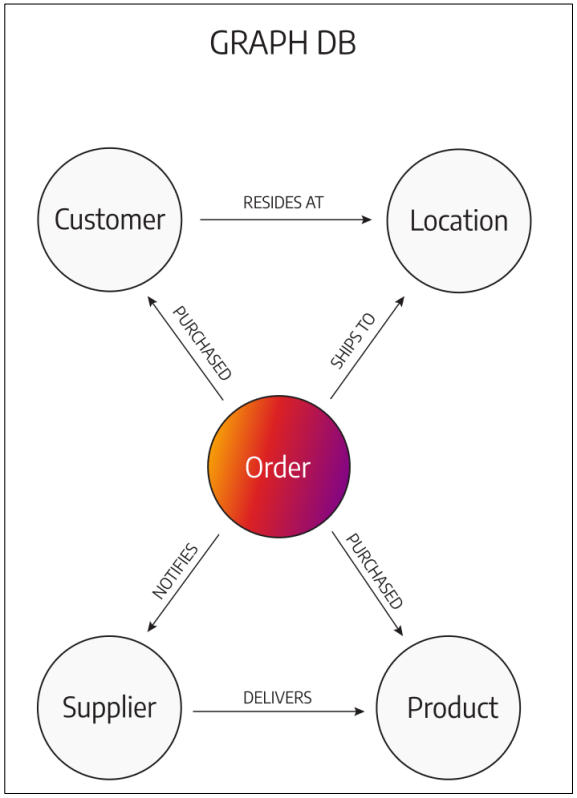
Each row does NOT require a single value/column

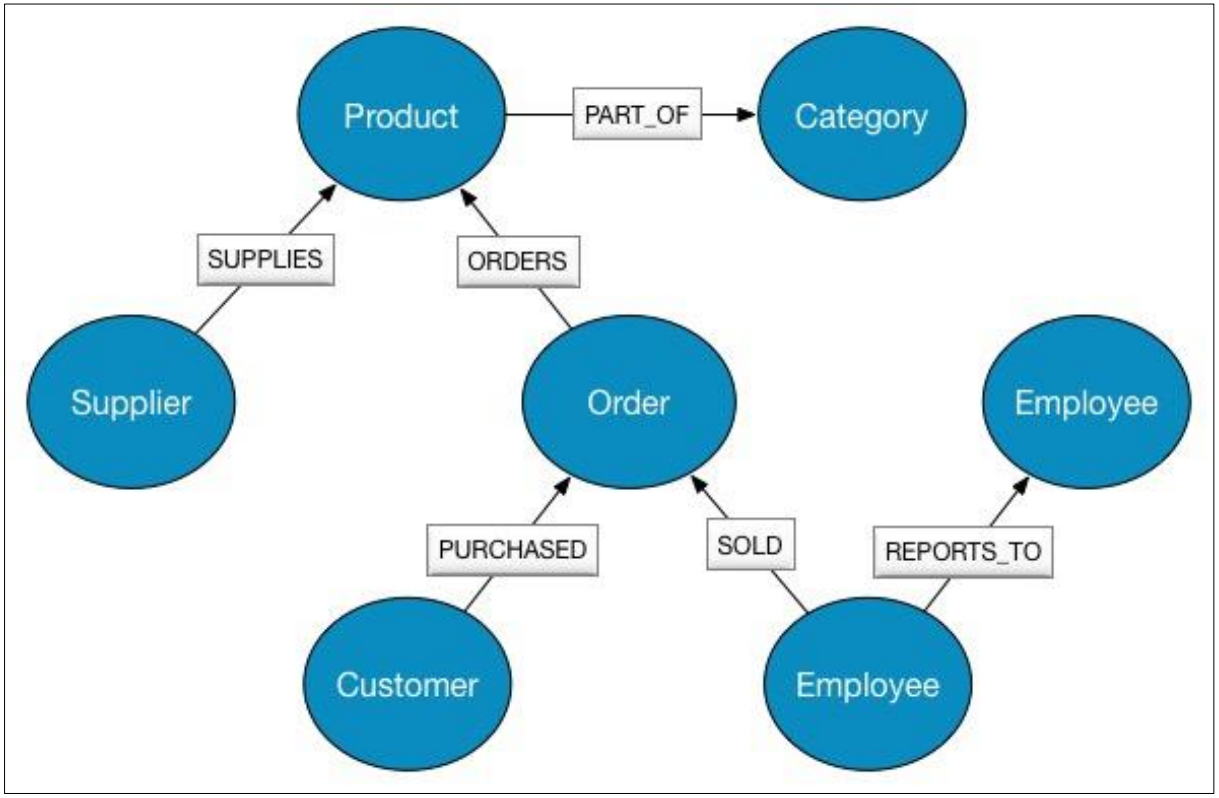
Suited for Frequent Reads

Graph database

Relationships representated as graphs with nodes and connections

Node is given ID and set of attributes



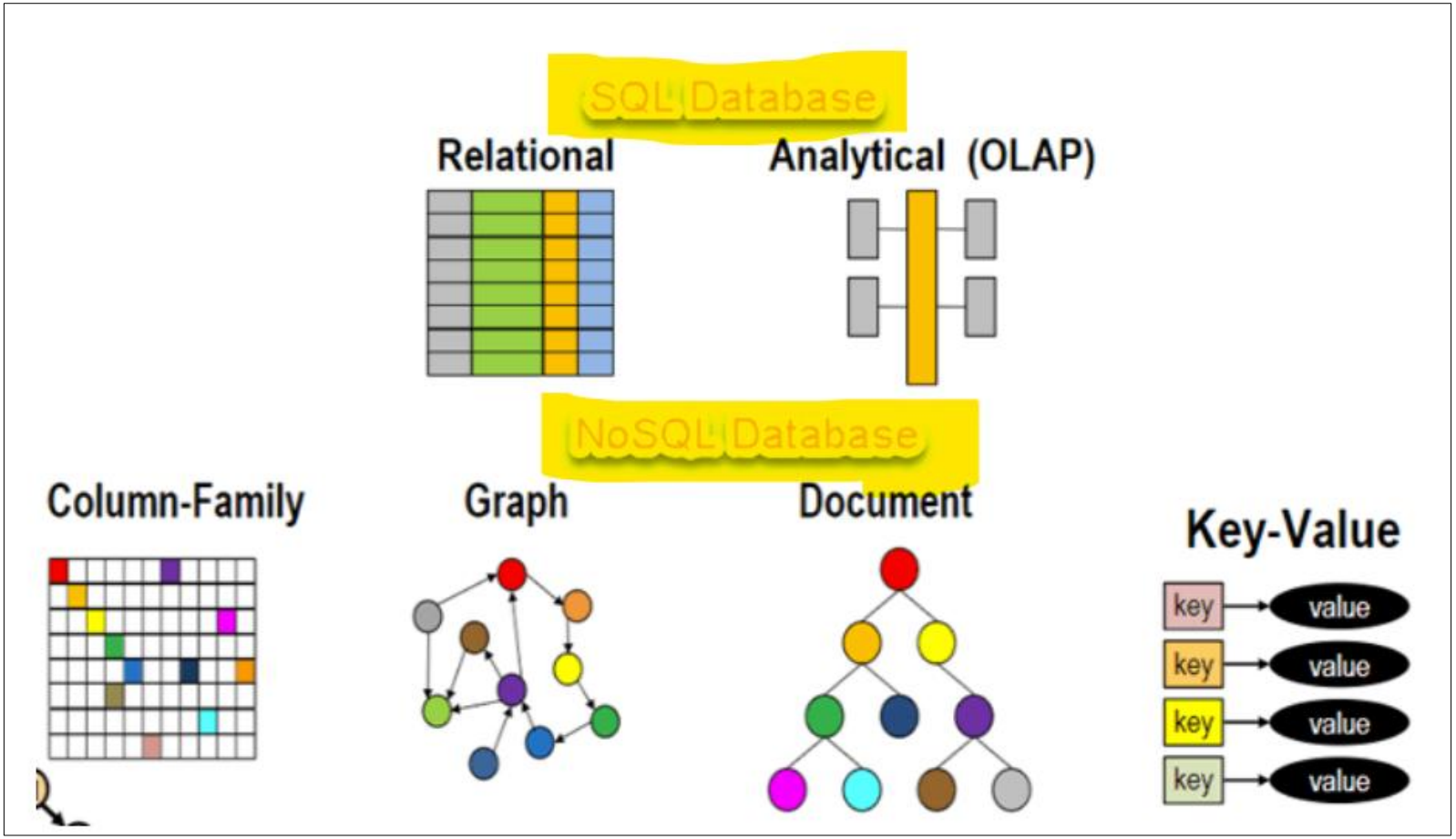


Properties for nodes/relationships are key:value pairs

NoSQL and Relational DBs are **complementary**

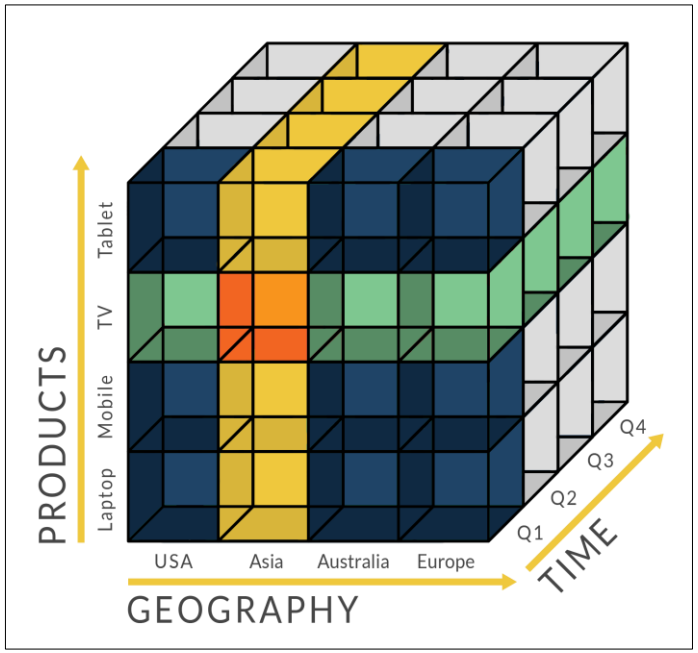
Relational databases provide data integrity

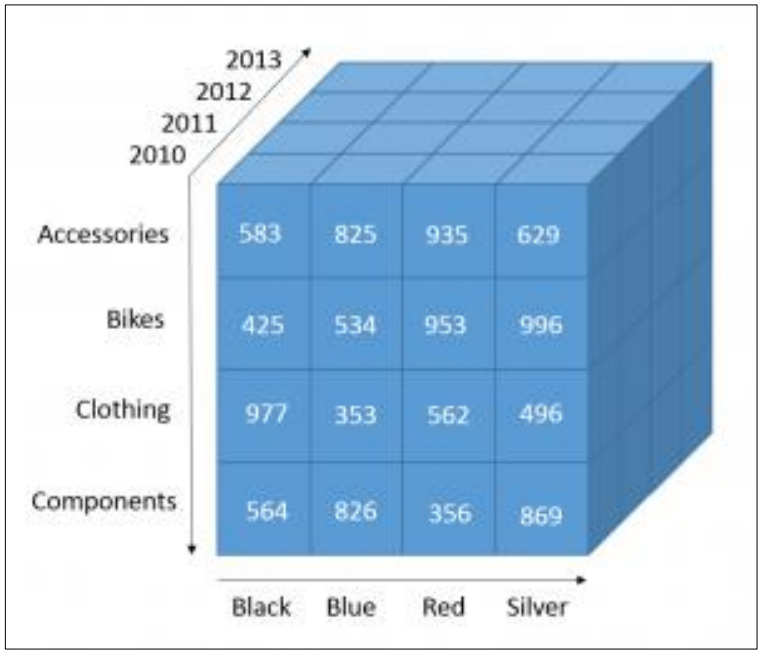
NoSQL provide high performance

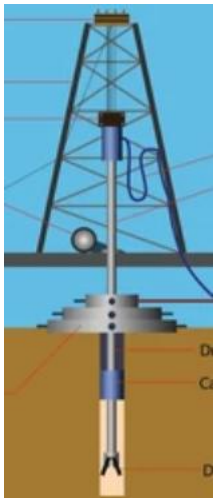


Multi-Dimensional Cubes and OLAP

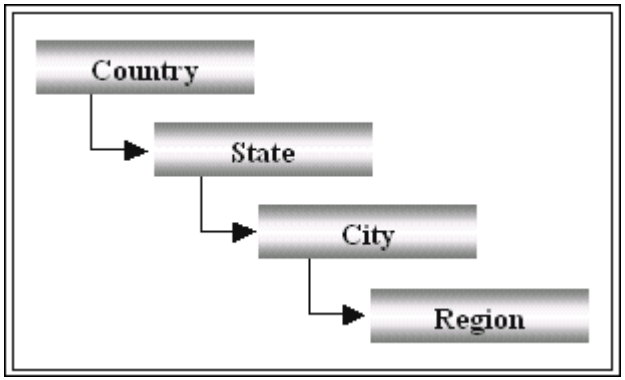
Multidimensional view of data is the foundation of OLAP





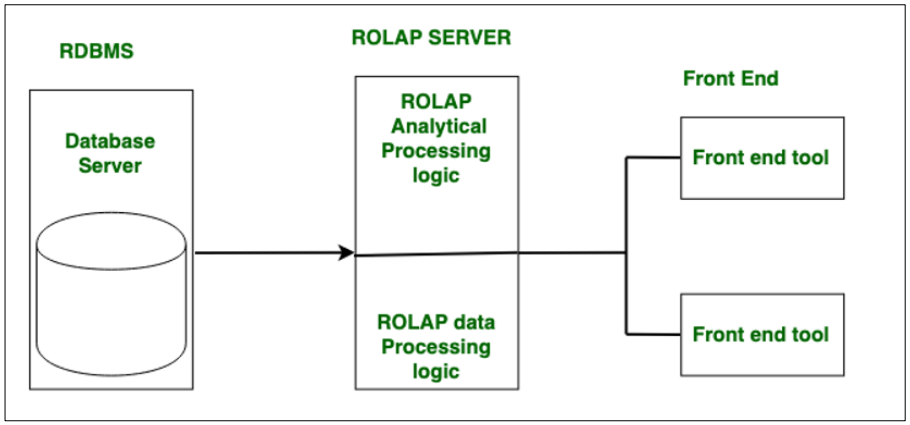


Can drill down for more detail



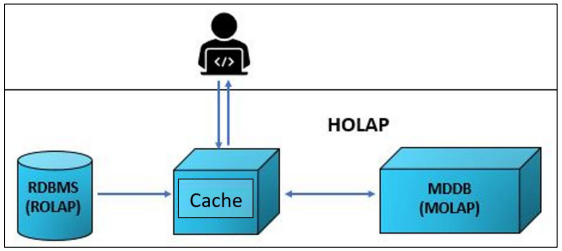
3 Types of OLAP

Relational OLAP [ROLAP]: done on relational DBMS



Multidimensional OLAP [MOLAP]: physical cubes

Hybrid OLAP [HOLAP]:
ROLAP for detail data
MOLAP for aggregated data



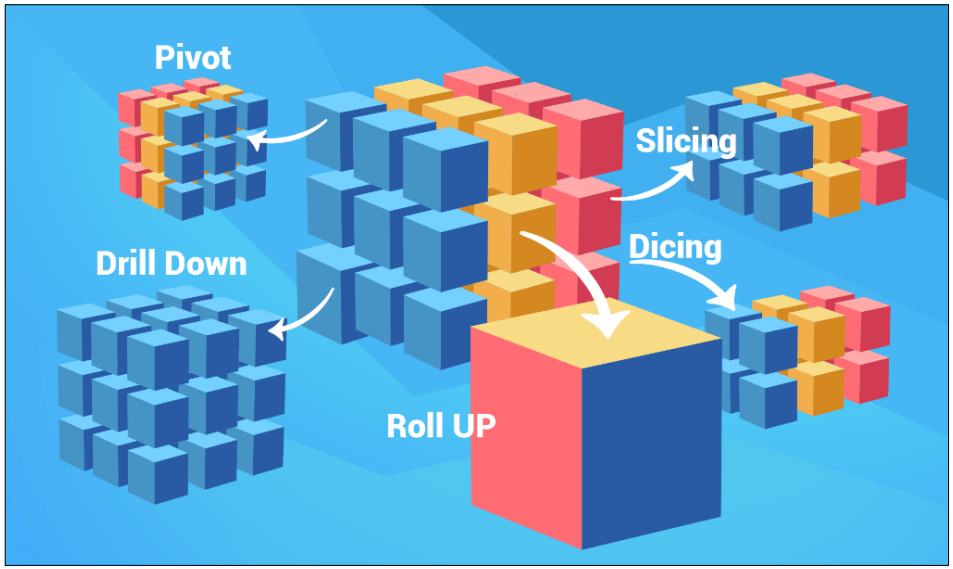
Relational OLAP [ROLAP]

- Familiar relational DBMS
- SQL
- Existing tools

However: inefficient & data volumes limited

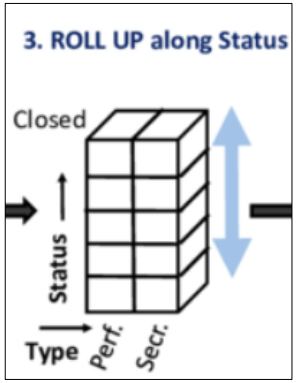
Multidimensional OLAP [MOLAP]:

- Uses Multidimensional DB Mngmnt Systems
- Data pre-computed & pre-summarized
- Data cubes have dimensions
- Uses Multidimensional Expressions [MDX] queries

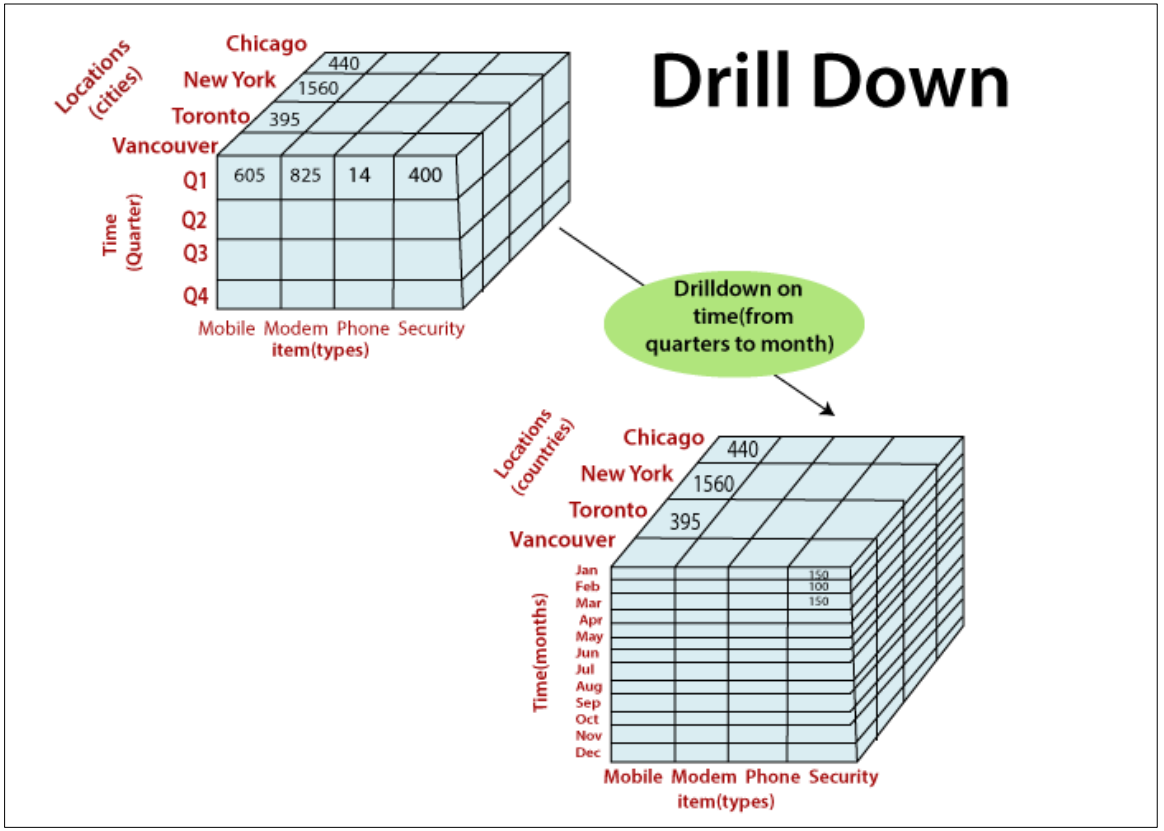


Multidimensional OLAP Operations

Roll-up: Aggregation, dimension reduction



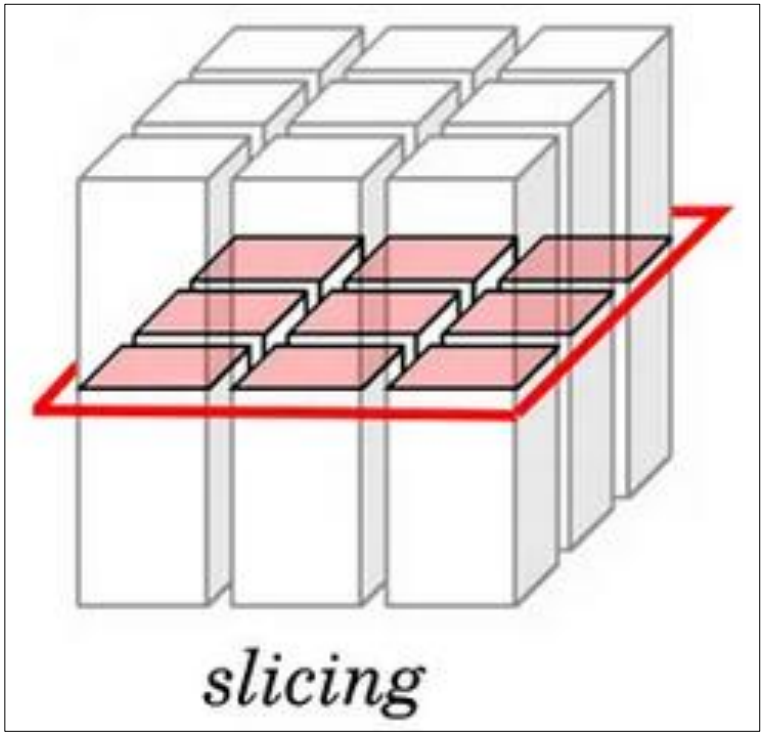
Drill-down: detailed data

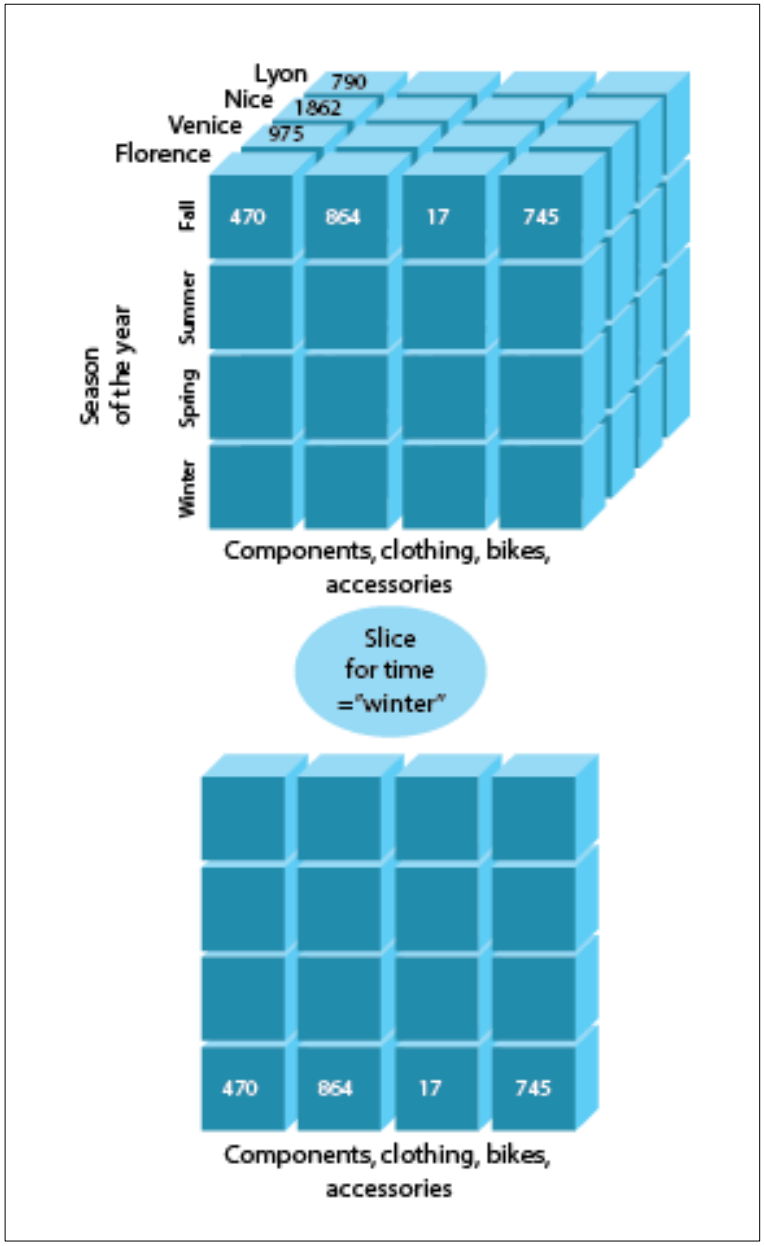


Multidimensional OLAP Operations

Slice: defines subcube;

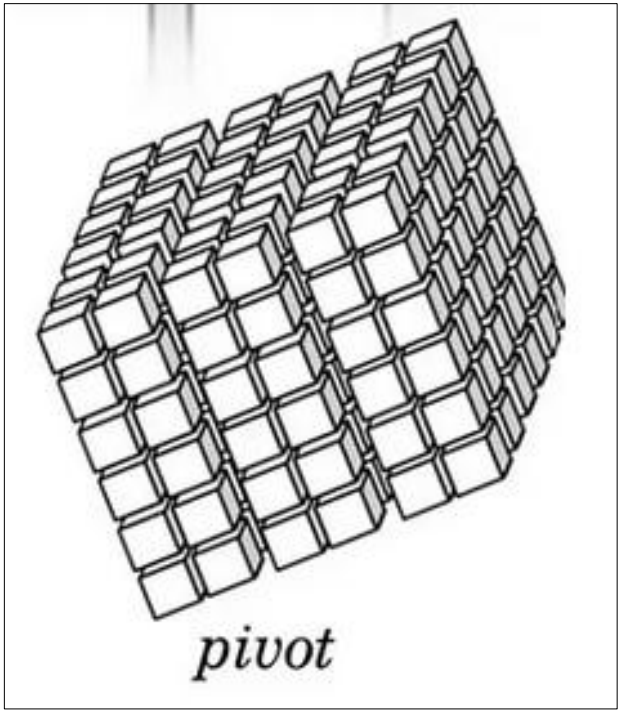
Fix one value of a dimension, eg: winter





Multidimensional OLAP Operations

Pivot: rotate cube around axis



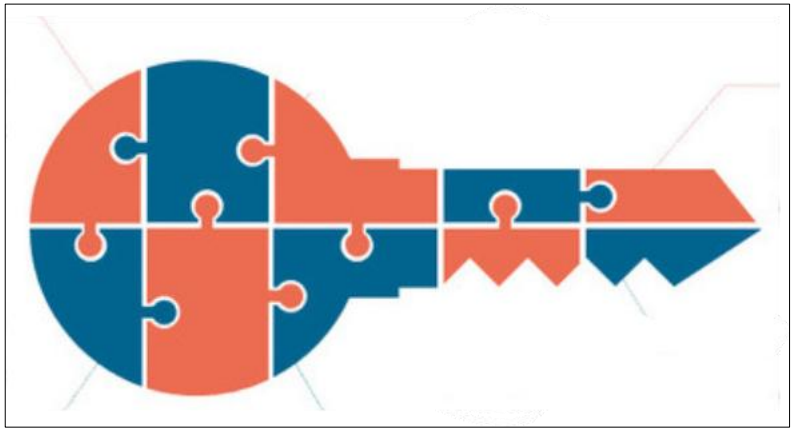
Multidimensional OLAP Operations

Advantages:

- Powerful, efficient engines
- Complex Calculations; slice and dice

Disadvantages:

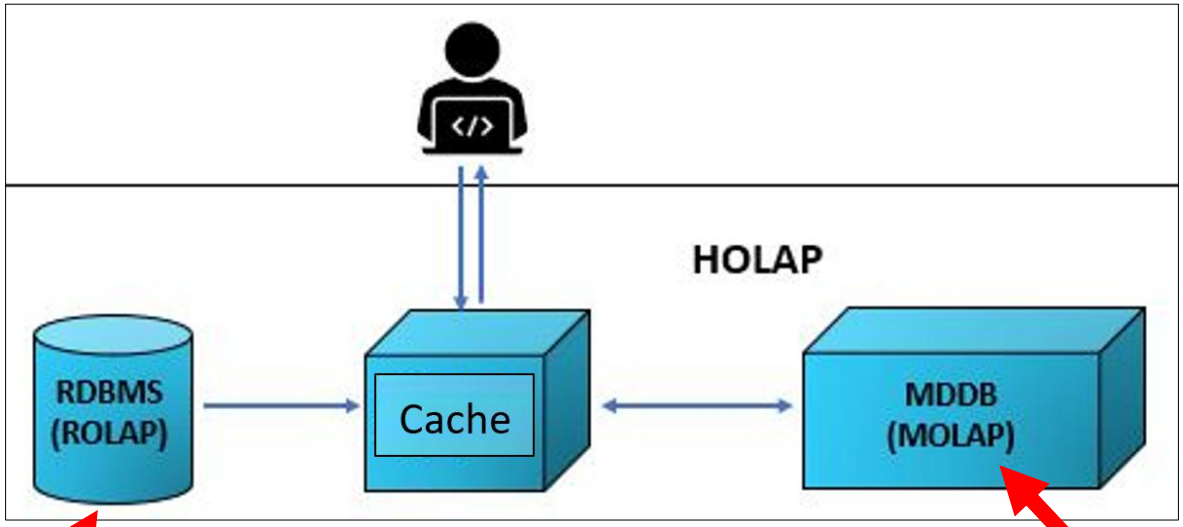
- Proprietary Structure
- Not for transaction processing



Hybrid OLAP [HOLAP]:

Combine advantages of ROLAP and MOLAP

Allows for more flexibility



Relational:
Holds larger quantities of detailed data

Specialized storage of less detailed data

BI Design and Development

End users interact with BI applications to analyze data

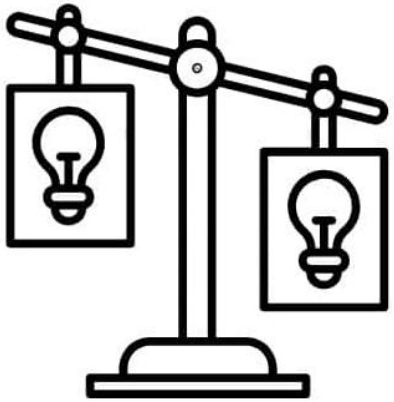
- Casual consumers
- Power users
- Data analysts
- Data Scientists

Types Of Analysis

- Comparative analysis
- Time-series or trending analysis
- Contribution analysis
- Correlation analysis
- Geographic data
- Distribution analysis

Types of Analysis

Comparative



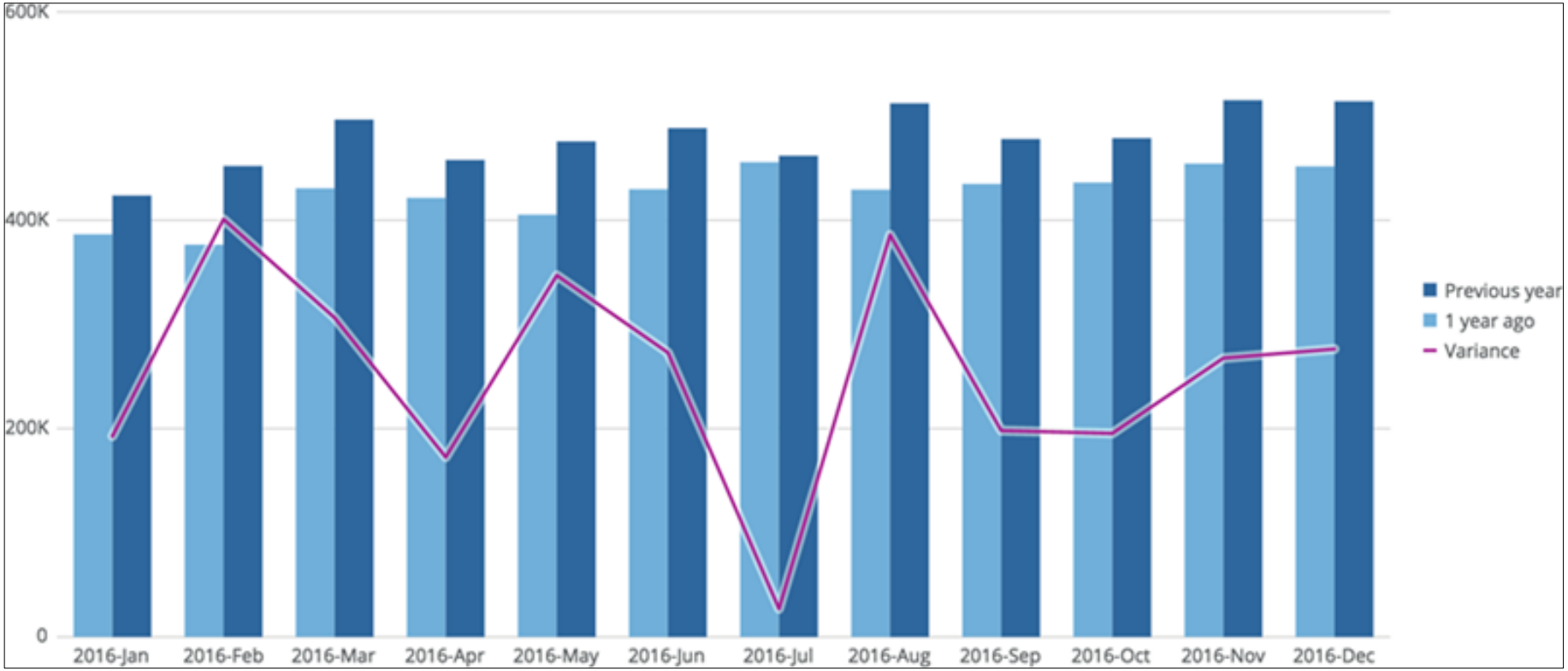
Comparative Analysis

The process of comparing and contrasting entities, variables, or phenomena to uncover insights.

Comparative analysis is a research methodology that involves comparing two or more data sets to draw meaningful conclusions.

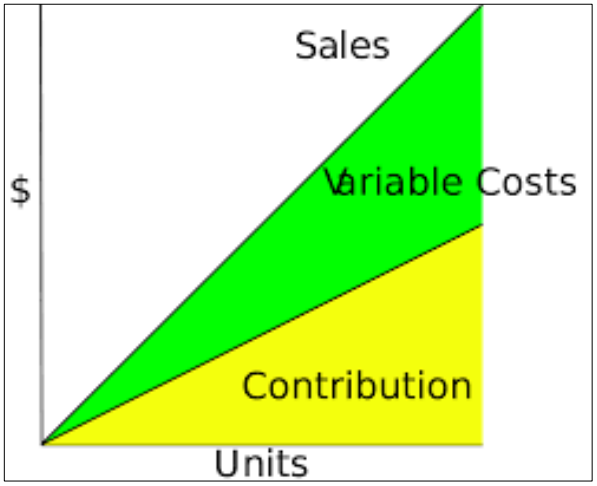
Time Series/Trending

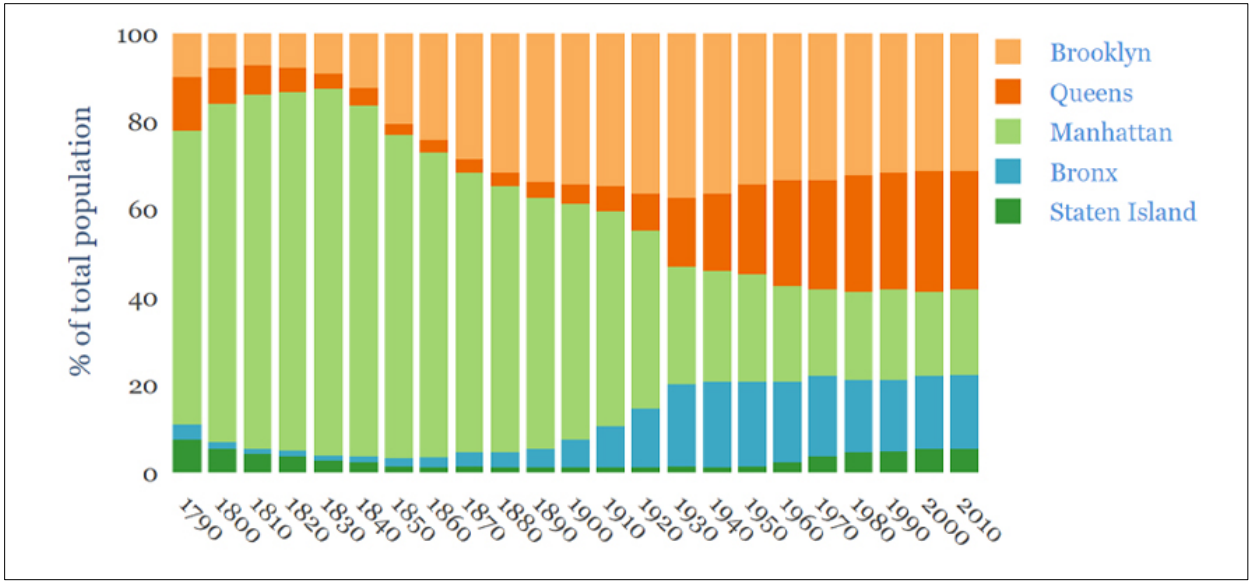
Compare data over time

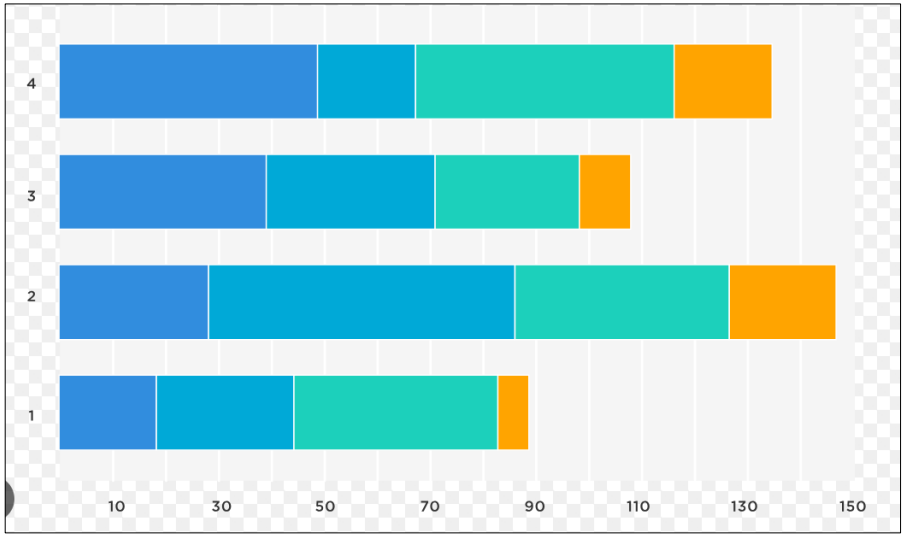


Contribution

Indicate amount added to the whole

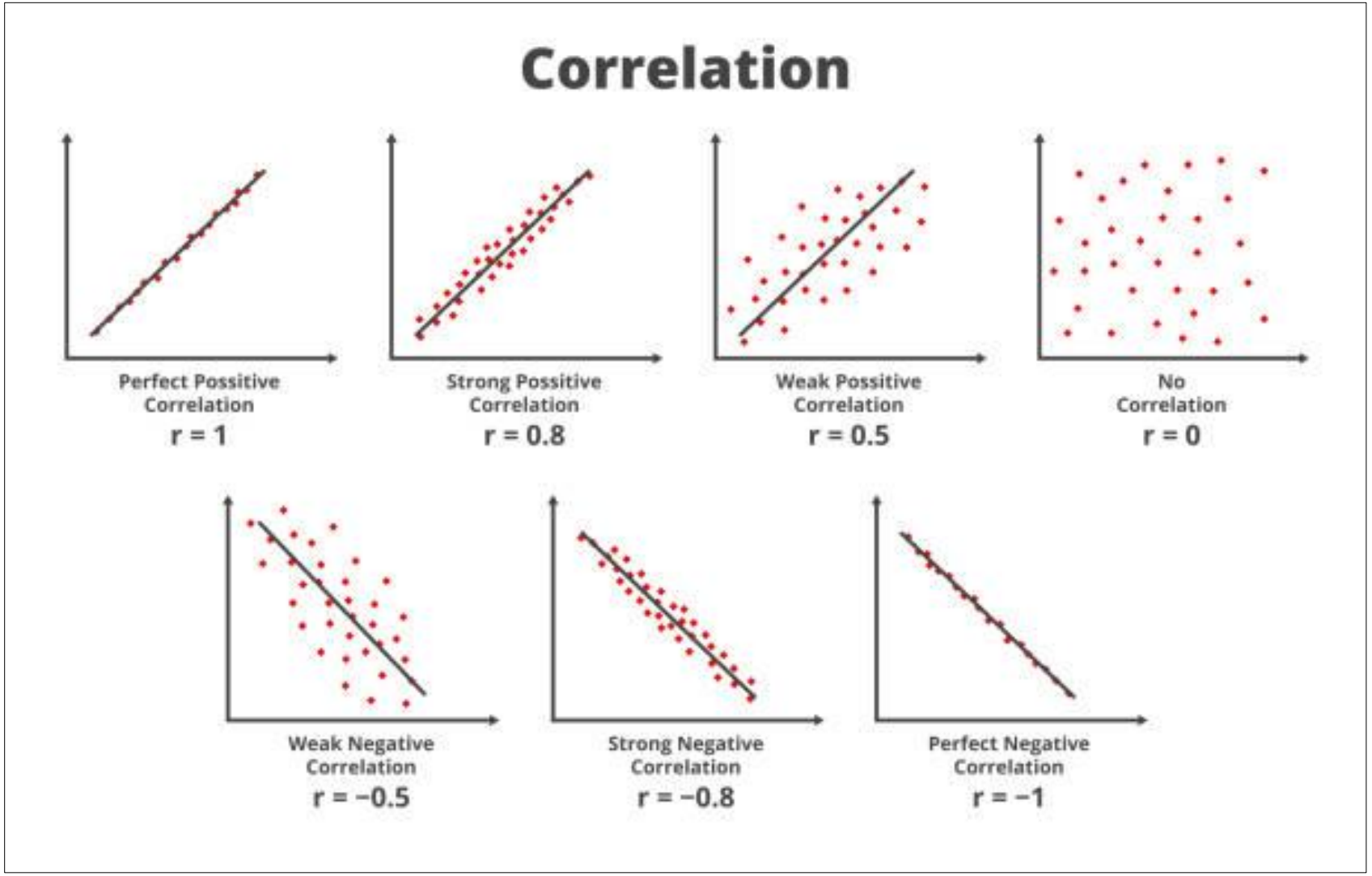






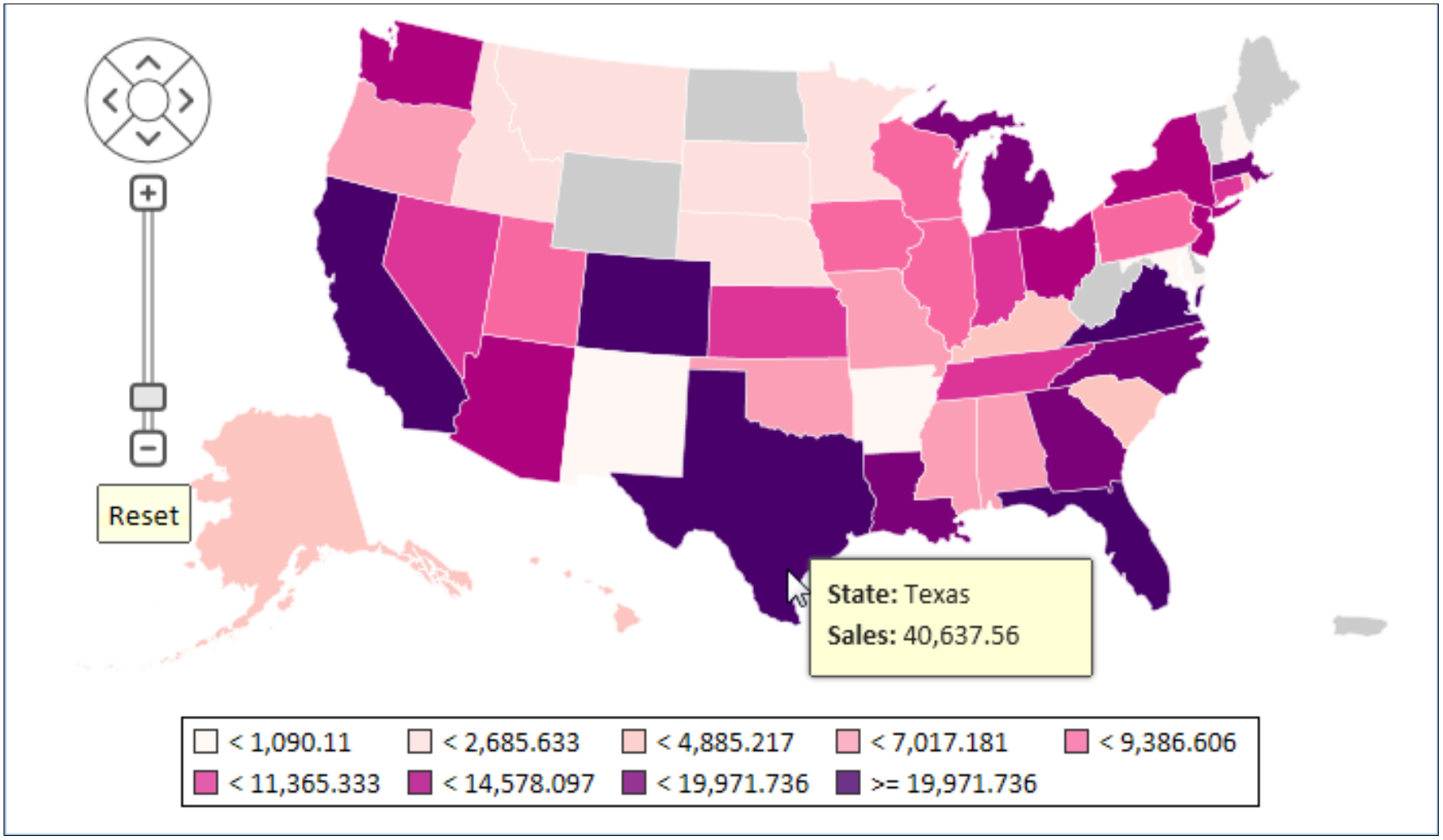
Correlation

Relationship between 2 sets of data



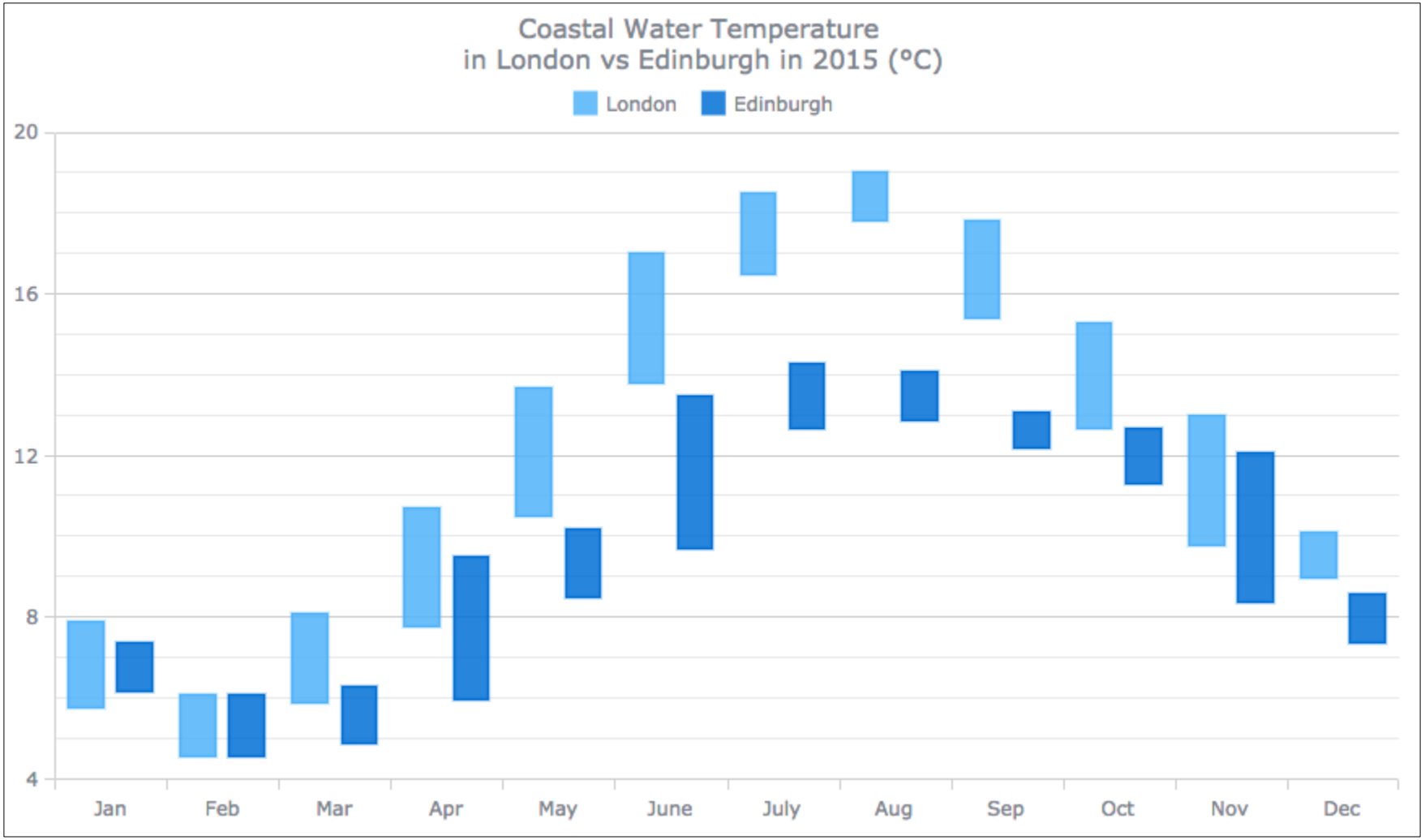
Geographic

Visualize by location



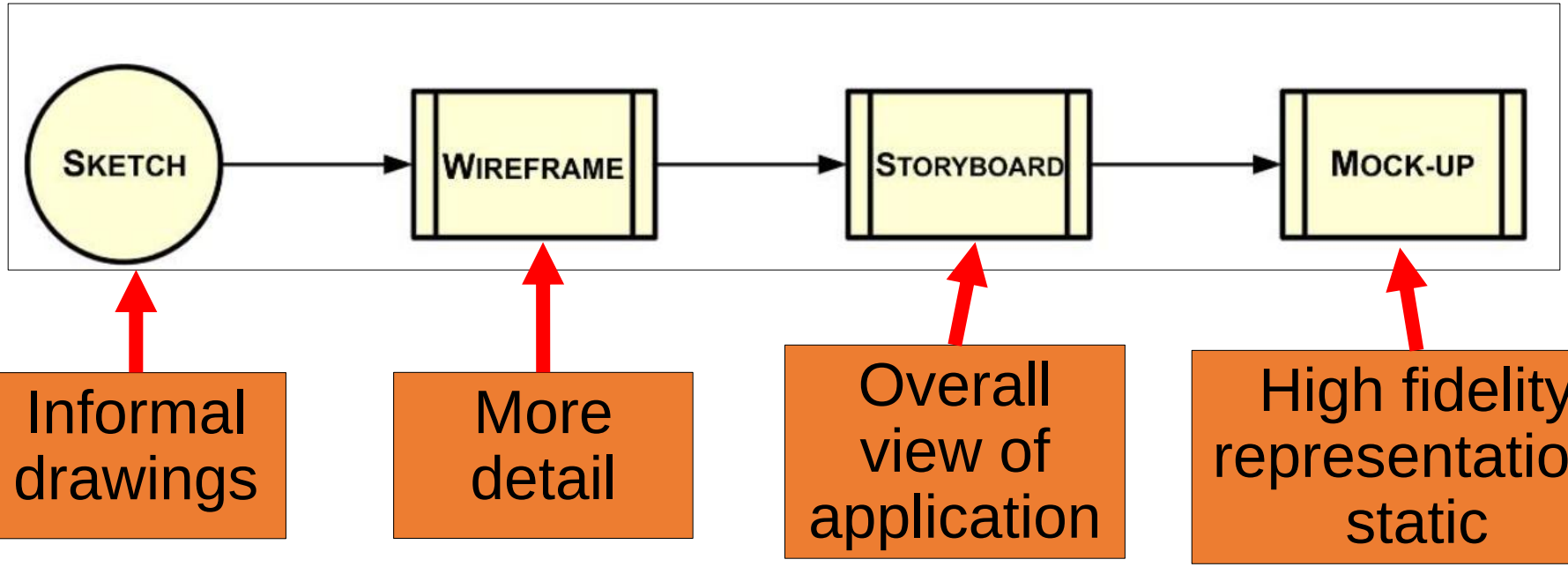
Distribution

How data falls around an average



BI Visual Design Methods

Encourages feedback

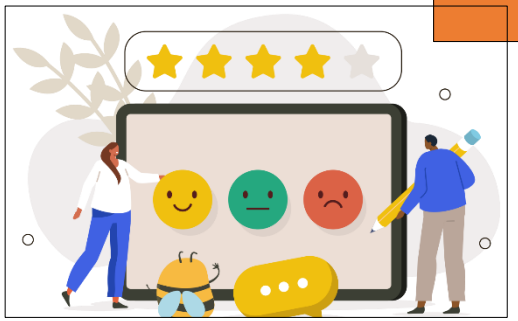


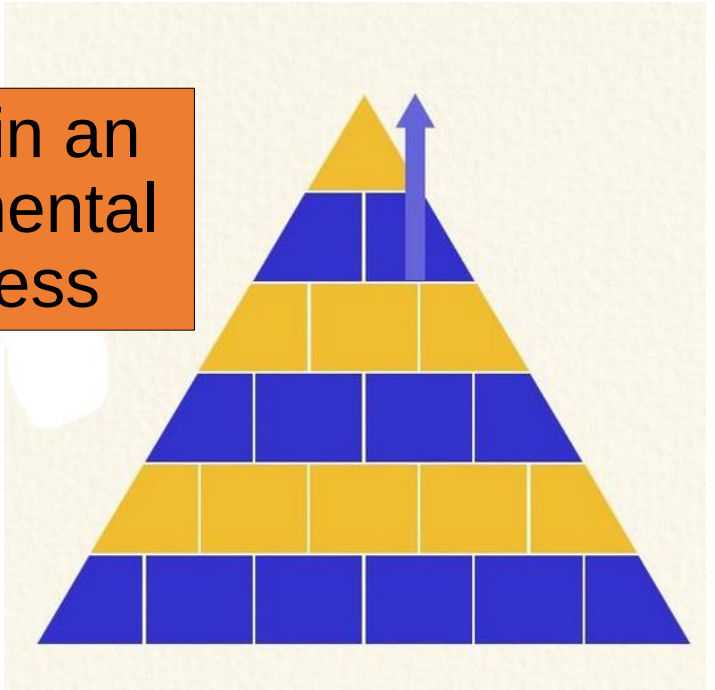
BI Prototyping

Build specific portions of the application

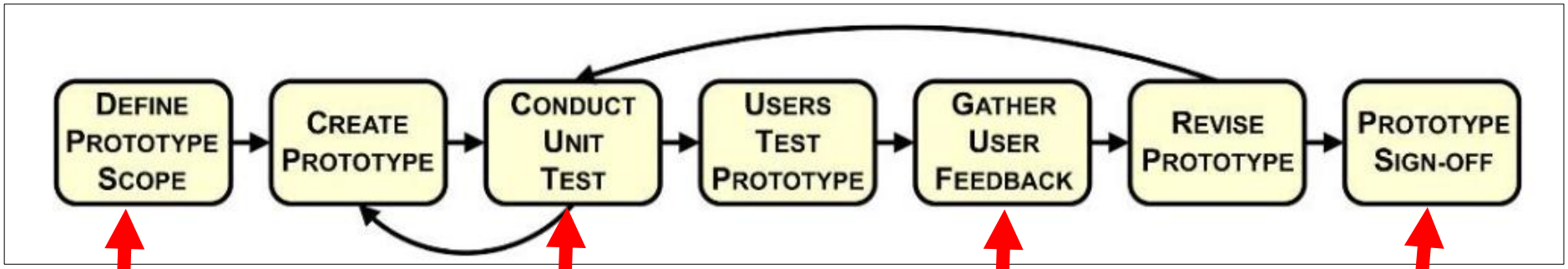
Two objectives

- Obtain feedback from users
- Build in an incremental process





Prototype Lifecycle

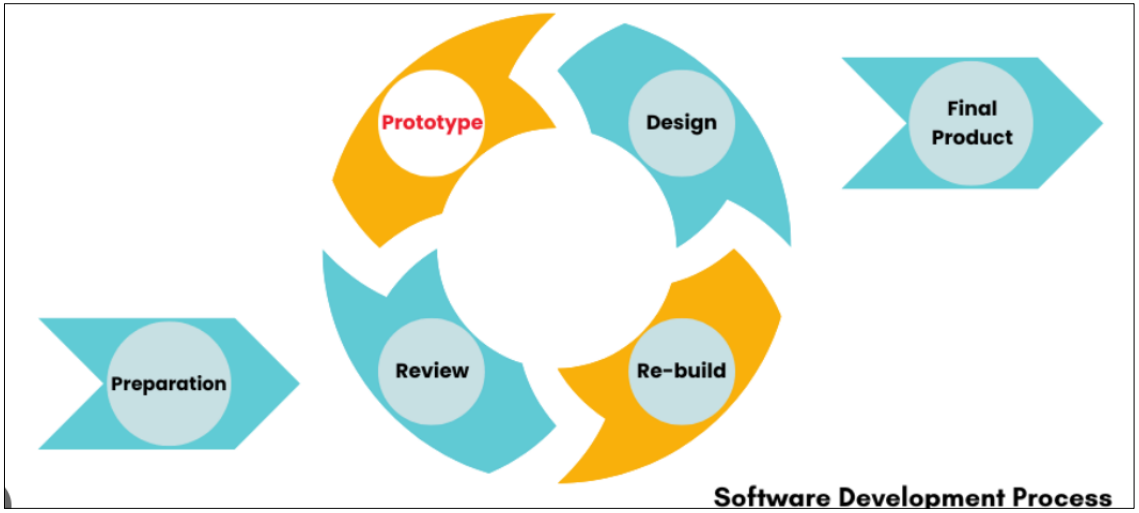


Determine objectives

Test prototype

Involve users in process

Application validated



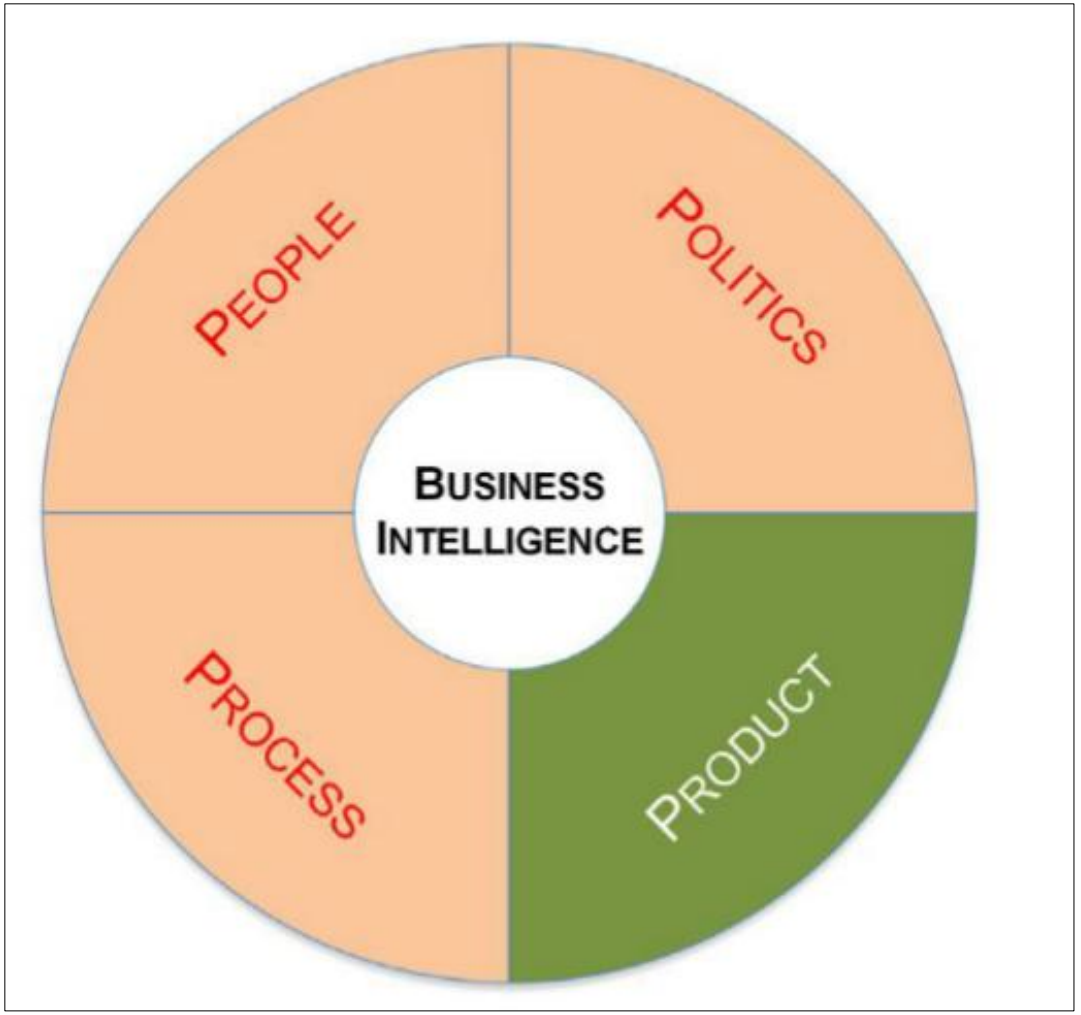
Application Testing Phase

Developer unit tests → End user tests → Developer integration tests [use “real” data] → User acceptance testing [also use “real” data]

REALDATA

DW Administration

People, Process and Politics



Meeting Expectations

Why do they fail?
Failure to meet expectations

- Unpopular solution
- Difficult analytics
- Information shortfall

"They're shutting down our Hopes and Dreams Division. It failed to meet expectations."

Communication is Key

Communicate with all users

Simple feedback loop

Use clear language

Departmental Roles

Business [Front Office]
- Business analysis
- Defines solution requirements

IT [Back Office]
- Create infrastructure
- Integration of data

BI Team

Sponsorship
- Commit business resources
- Financial support

Project Management
- Day-to-day tasks
- Report status

Development Team
- Power Users
- Integration/ETL

Extended Team
- QA
- Operations

BI Training
Necessary to fulfill its potential

- Foundational
- Tool-specific resources
- Instruct with use cases

IT Group
- ETL
- Database
- SQL

Business Group
- Analytics
- Functions
- Use cases

Data Governance

- Process of managing the availability and security
- Control usage
- Enforces definitions and rules

Project Management

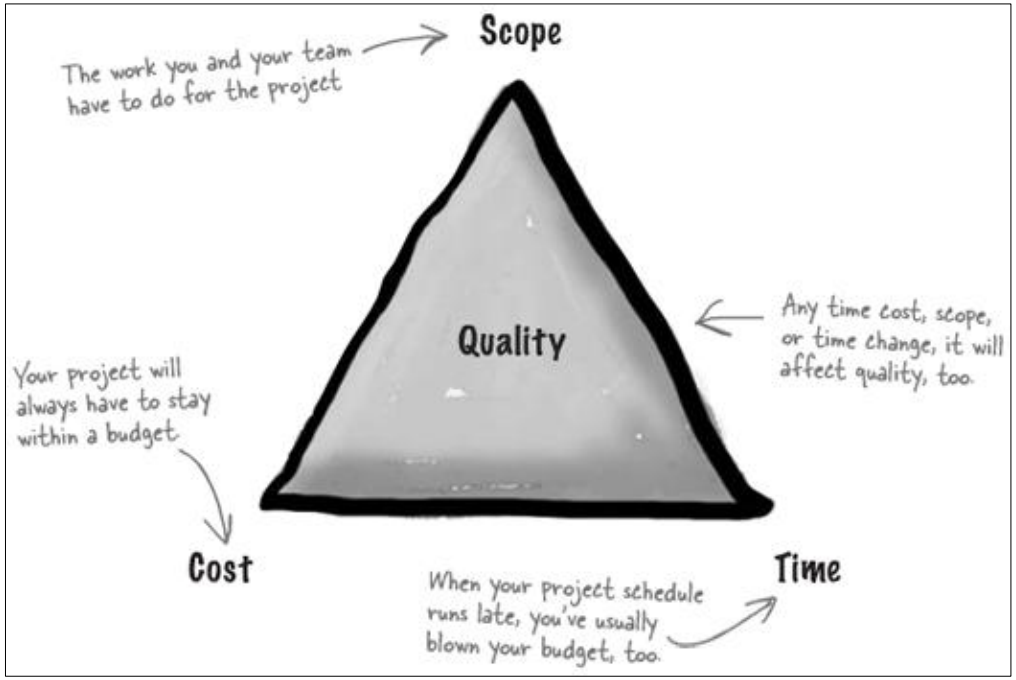
Necessary to prevent:
- Lateness
- Budget overruns
- Low quality
- Failure to meet expectations

Business Strategy

Strategy drives BI Program:
- Sponsorship [CFO]
- Governance
- Participation

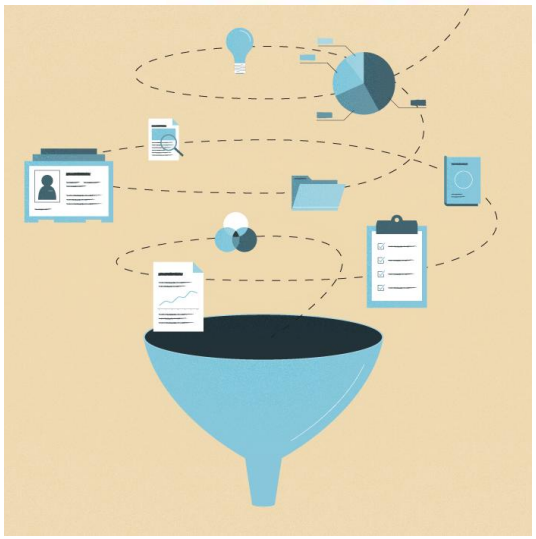
PM Is a Balancing Act

- Results
- Money
- Time



Other Factors Affecting Scope

- Analytic complexity
- Amount of data
- Age of data
- Integration complexity
- Number sources



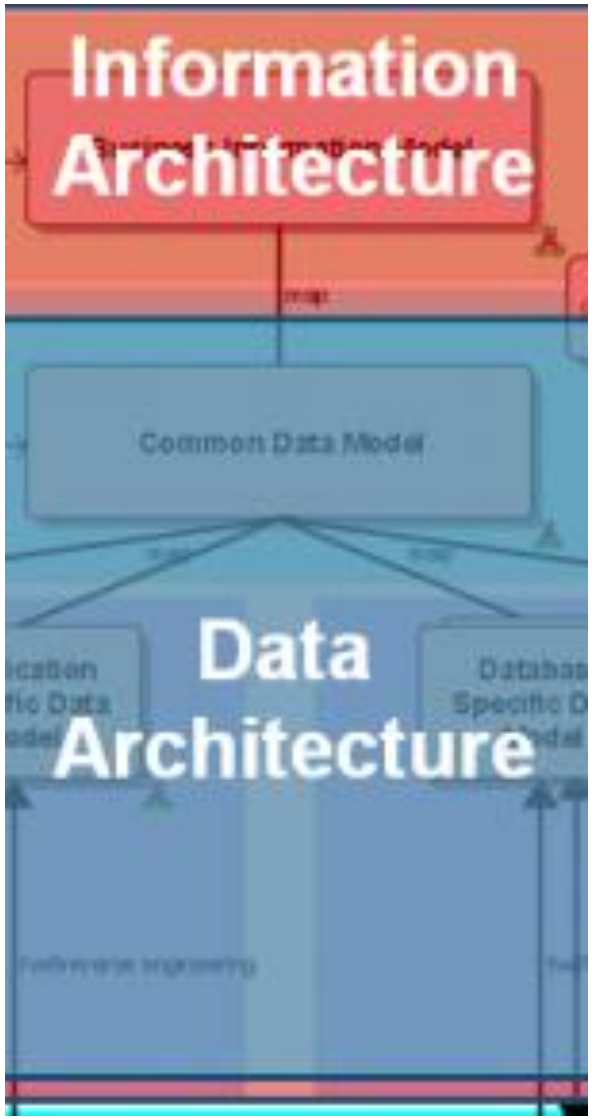
3 Phases BI Assessment

- Discovery [current]
- Analysis [ID gaps]
- Recommendations [Priorities]

Discovery

Analysis

BI Project Phases

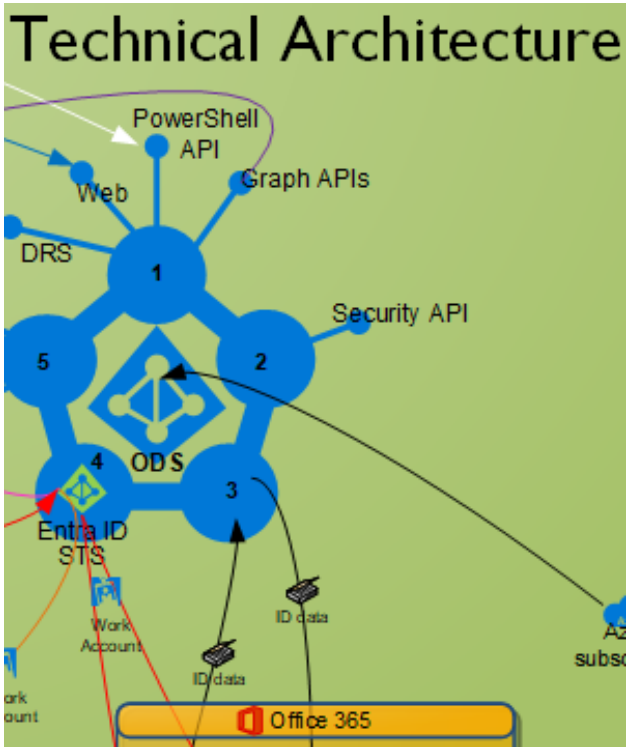


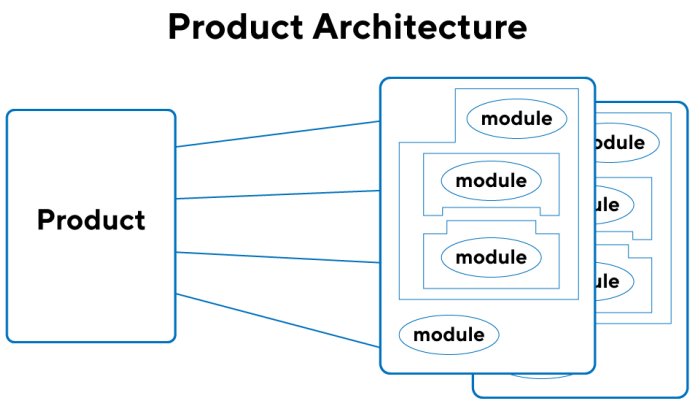
Information Architecture

Data Architecture

Technical Architecture

Product Architecture

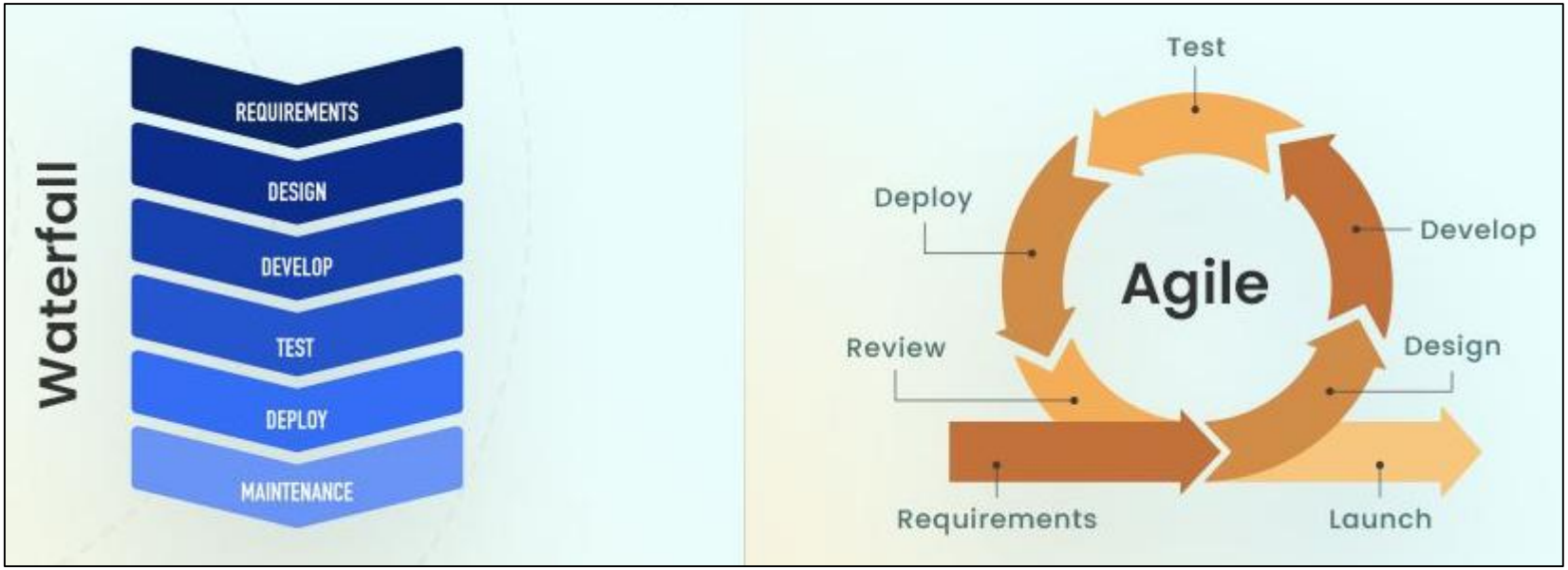


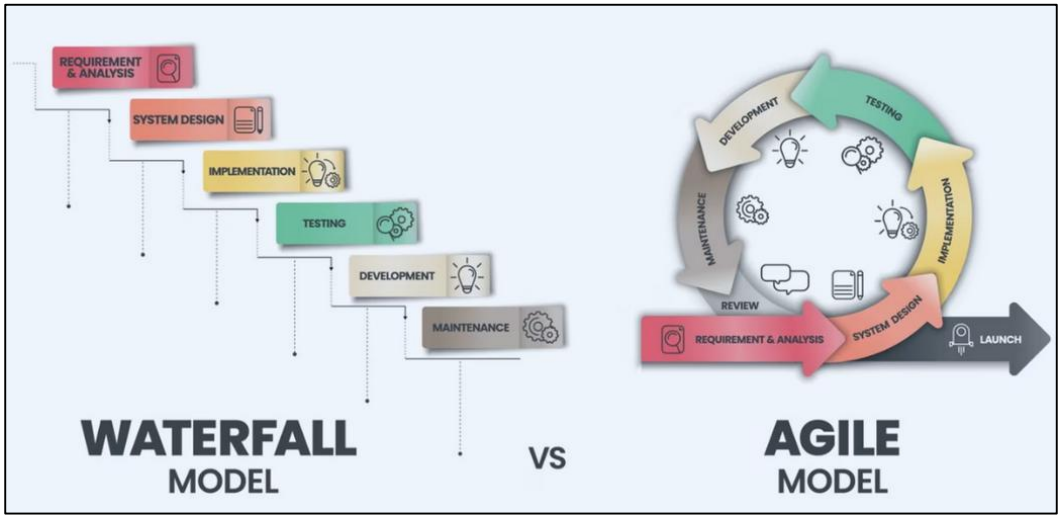


Project Methodologies

Waterfall

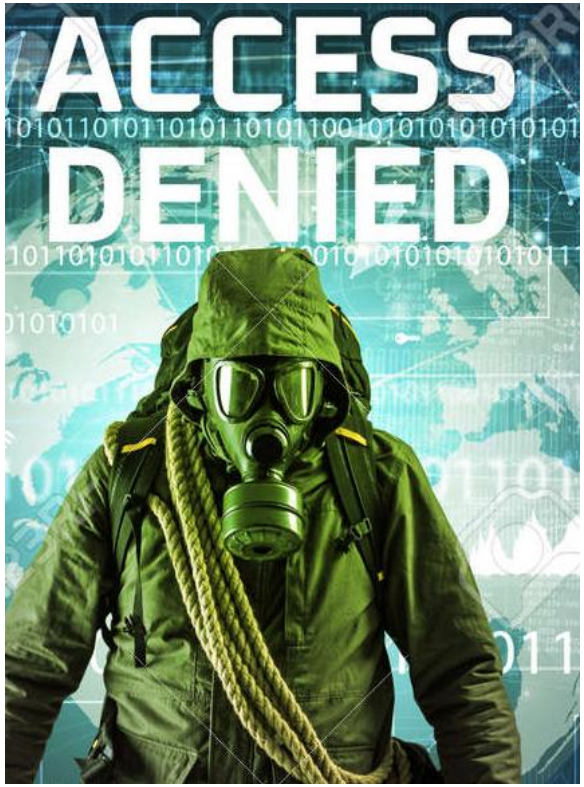
Agile





DW Security

DWs are lucrative targets for malicious actors



Stop access to unauthorized users

Available to right users at the right time

Keep record of activities [log]

```
2015-10-17 15:45:11,258 INFO [main] org.apache.hadoop.metrics2.impl.MetricsConfig: loaded properties from hadoop-metrics2.properties
2015-10-17 15:45:11,399 INFO [main] org.apache.hadoop.metrics2.impl.MetricsSystemImpl: Scheduled snapshot period at 10 second(s).
2015-10-17 15:45:11,399 INFO [main] org.apache.hadoop.metrics2.impl.MetricsSystemImpl: MapTask metrics system started
2015-10-17 15:45:11,430 INFO [main] org.apache.hadoop.mapred.YarnChild: Executing with tokens:
2015-10-17 15:45:11,430 INFO [main] org.apache.hadoop.mapred.YarnChild: Kind: mapreduce.job, Service: job_1445062781478_0015, Ident: (org.apache$
2015-10-17 15:45:11,602 INFO [main] org.apache.hadoop.mapred.YarnChild: Sleeping for 0ms before retrying again. Got null now.
2015-10-17 15:45:12,196 INFO [main] org.apache.hadoop.mapred.YarnChild: mapreduce.cluster.local.dir for child: /tmp/hadoop-msrabi/nm-local-dir/u$
2015-10-17 15:45:12,711 INFO [main] org.apache.hadoop.conf.Configuration.deprecation: session.id is deprecated. Instead, use dfs.metrics.session$
2015-10-17 15:45:13,602 INFO [main] org.apache.hadoop.yarn.util.ProcfsBasedProcessTree: ProcfsBasedProcessTree currently is supported only on Li$
2015-10-17 15:45:13,618 INFO [main] org.apache.hadoop.mapred.Task:  Using ResourceCalculatorProcessTree : org.apache.hadoop.yarn.util.WindowsBas$
2015-10-17 15:45:14,008 INFO [main] org.apache.hadoop.mapred.MapTask: Processing split: hdfs://msra-sa-41:9000/pageinput2.txt:402653184+134217728
2015-10-17 15:45:14,102 INFO [main] org.apache.hadoop.mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)
2015-10-17 15:45:14,102 INFO [main] org.apache.hadoop.mapred.MapTask: mapreduce.task.io.sort.mb: 100
2015-10-17 15:45:14,102 INFO [main] org.apache.hadoop.mapred.MapTask: soft limit at 83886080
2015-10-17 15:45:14,102 INFO [main] org.apache.hadoop.mapred.MapTask: bufstart = 0; bufvoid = 104857600
2015-10-17 15:45:14,102 INFO [main] org.apache.hadoop.mapred.MapTask: kvstart = 26214396; length = 6553600
2015-10-17 15:45:14,118 INFO [main] org.apache.hadoop.mapred.MapTask: Map output collector class = org.apache.hadoop.mapred.MapTask$MapOutputBuf$
2015-10-17 15:45:17,305 INFO [main] org.apache.hadoop.mapred.MapTask: Spilling map output
2015-10-17 15:45:17,305 INFO [main] org.apache.hadoop.mapred.MapTask: bufstart = 0; bufend = 48271024; bufvoid = 104857600
2015-10-17 15:45:17,305 INFO [main] org.apache.hadoop.mapred.MapTask: kvstart = 26214396(104857584); kvend = 17310640(69242560); length = 8903755
2015-10-17 15:45:17,305 INFO [main] org.apache.hadoop.mapred.MapTask: (EQUATOR) 57339776 kvi 14334940(57339760)
2015-10-17 15:45:26,696 INFO [SpillThread] org.apache.hadoop.mapred.MapTask: Finished spill 0
2015-10-17 15:45:26,696 INFO [main] org.apache.hadoop.mapred.MapTask: (RESET) equator 57339776 kv 14334940(57339760) kvi 12140764(48563056)
2015-10-17 15:45:30,603 INFO [main] org.apache.hadoop.mapred.MapTask: Spilling map output
2015-10-17 15:45:30,603 INFO [main] org.apache.hadoop.mapred.MapTask: bufstart = 57339776; bufend = 743078; bufvoid = 104857600
2015-10-17 15:45:30,603 INFO [main] org.apache.hadoop.mapred.MapTask: kvstart = 14334940(57339760); kvend = 5428644(21714576); length = 8906297/$
2015-10-17 15:45:30,603 INFO [main] org.apache.hadoop.mapred.MapTask: (EQUATOR) 9811814 kvi 2452948(9811792)
2015-10-17 15:45:39,525 INFO [SpillThread] org.apache.hadoop.mapred.MapTask: Finished spill 1
2015-10-17 15:45:39,525 INFO [main] org.apache.hadoop.mapred.MapTask: (RESET) equator 9811814 kv 2452948(9811792) kvi 244148(976592)
2015-10-17 15:45:43,307 INFO [main] org.apache.hadoop.mapred.MapTask: Spilling map output
2015-10-17 15:45:43,307 INFO [main] org.apache.hadoop.mapred.MapTask: bufstart = 9811814; bufend = 58036090; bufvoid = 104857600
2015-10-17 15:45:43,307 INFO [main] org.apache.hadoop.mapred.MapTask: kvstart = 2452948(9811792); kvend = 19751984(79007616); length = 8915445/6$
2015-10-17 15:45:43,307 INFO [main] org.apache.hadoop.mapred.MapTask: (EQUATOR) 67104842 kvi 16776204(67104816)
```

Consolidated DW

Consistent Security

Security must be centralized

Fewer points of attack