Collecting Data to Extract Insights

Janani Ravi
CO-FOUNDER, LOONYCORN
www.loonycorn.com

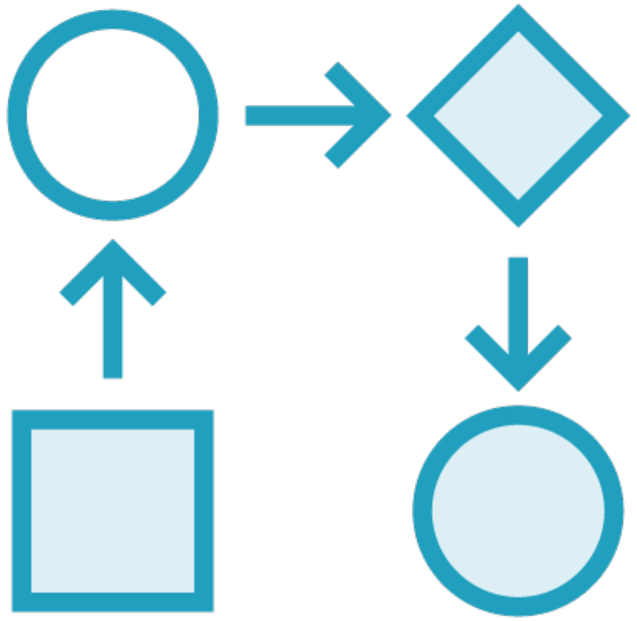
Overview

- Standardization and normalization
- Binning and sampling
- Big data
- Batch vs. streaming data
- Event time and processing time

Two Hats of a Data Professional

Find the Dots

Identify important elements in a dataset



Connect the Dots

Explain those elements via relationships with other elements

Essential Steps in Connecting the Dots

Processing Data for Use in Models

Building and Refining Models

Incorporating Real-world Data into Models

Essential Steps in Connecting the Dots

Processing Data for Use in Models

Building and Refining Models

Incorporating Real-world Data into Models

Not in scope in this course

Essential Steps in Connecting the Dots

Processing Data for Use in Models	**Incorporating Real-world Data into Models**

Essential Steps in Connecting the Dots

Processing Data for Use in Models

Incorporating Real-time Data into Models

Standardizing Data

Standardizing Data

$$\begin{bmatrix} X_{11} & & X_{1k} \\ X_{21} & & X_{2k} \\ X_{31} & \dots & X_{3k} \\ \dots & & \dots \\ X_{n1} & & X_{nk} \end{bmatrix}$$

$\text{avg}(X_1)$ ... $\text{avg}(X_k)$

$\text{stdev}(X_1)$ ... $\text{stdev}(X_k)$

Standardizing Data

$$\begin{bmatrix} \dfrac{x_{11} - avg(X_1)}{stdev(X_1)} & \cdots & \dfrac{x_{1k} - avg(X_k)}{stdev(X_k)} \\ \cdots & & \cdots \\ \dfrac{x_{n1} - avg(X_1)}{stdev(X_1)} & & \dfrac{x_{nk} - avg(X_k)}{stdev(X_k)} \end{bmatrix}$$

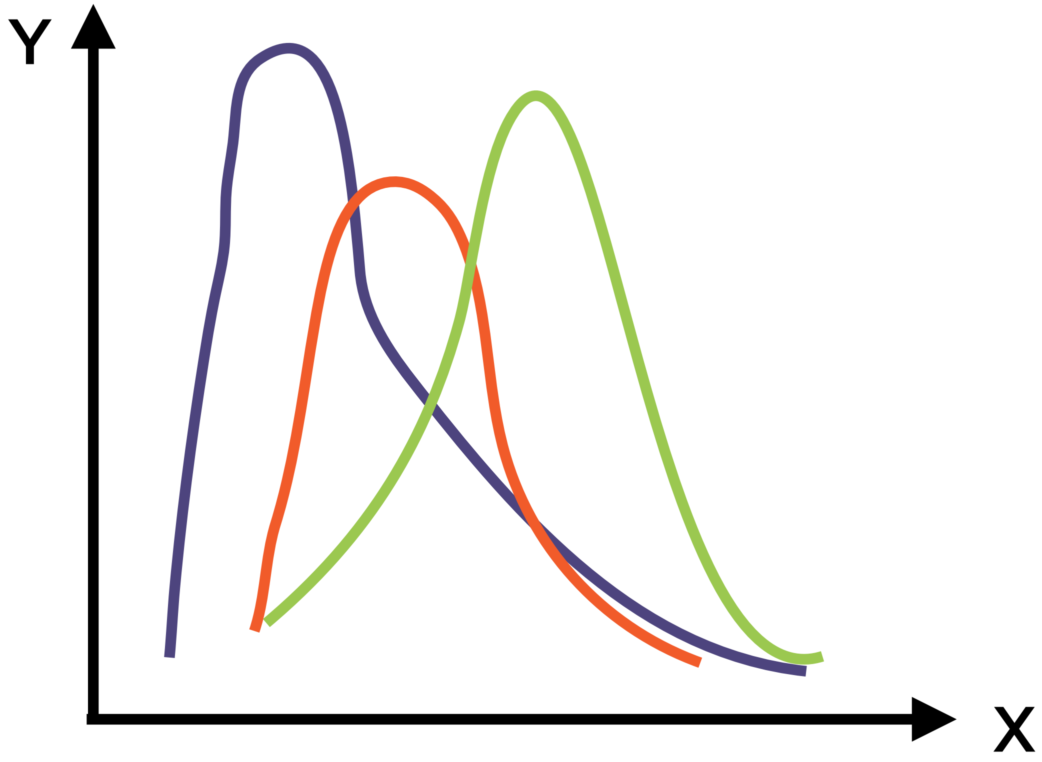
Each column of the standardized data has mean 0 and variance 1

Standardizing Data

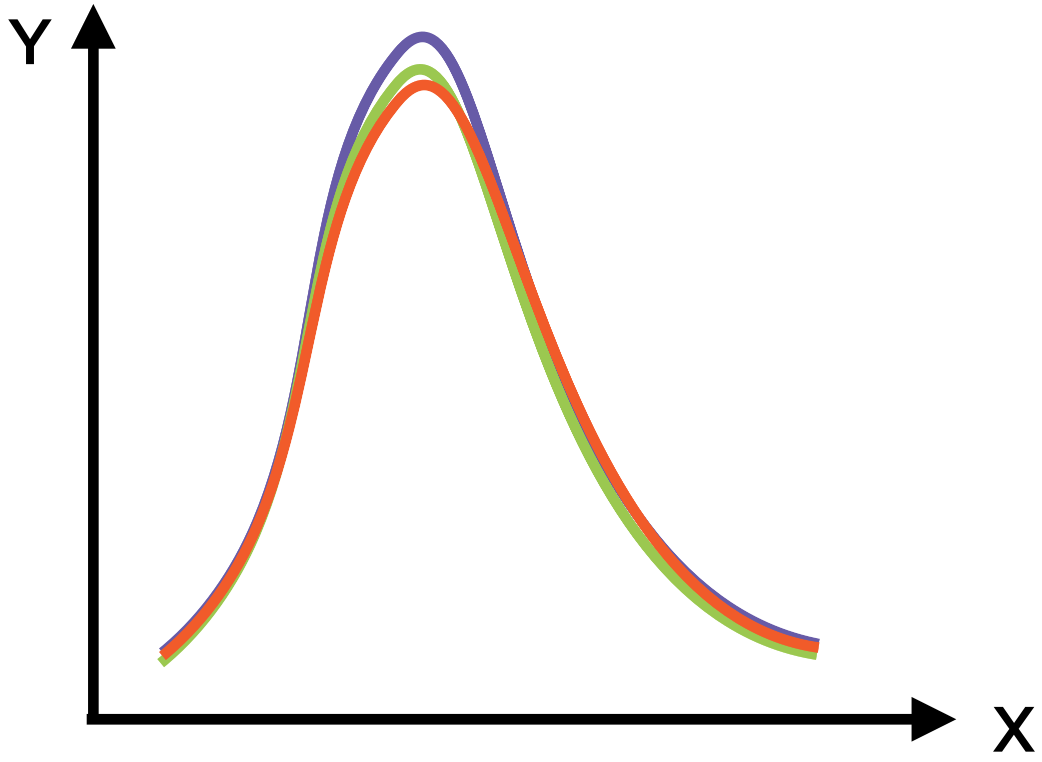
$$z = \frac{x_i - \mathrm{mean}(x)}{\mathrm{stdev}(x)}$$

Standardization operates column-by-column and yields features with zero mean and unit variance

Standardizing Data



Before



After

Standardizing Data

$$z = \frac{x_i - \text{mean}(x)}{\text{stdev}(x)}$$

Mean is a measure of central tendency and standard deviation is a measure of dispersion

Robust Standardization

$$z = \frac{x_i - \text{median}(x)}{\text{Inter-quartile Range}(x)}$$

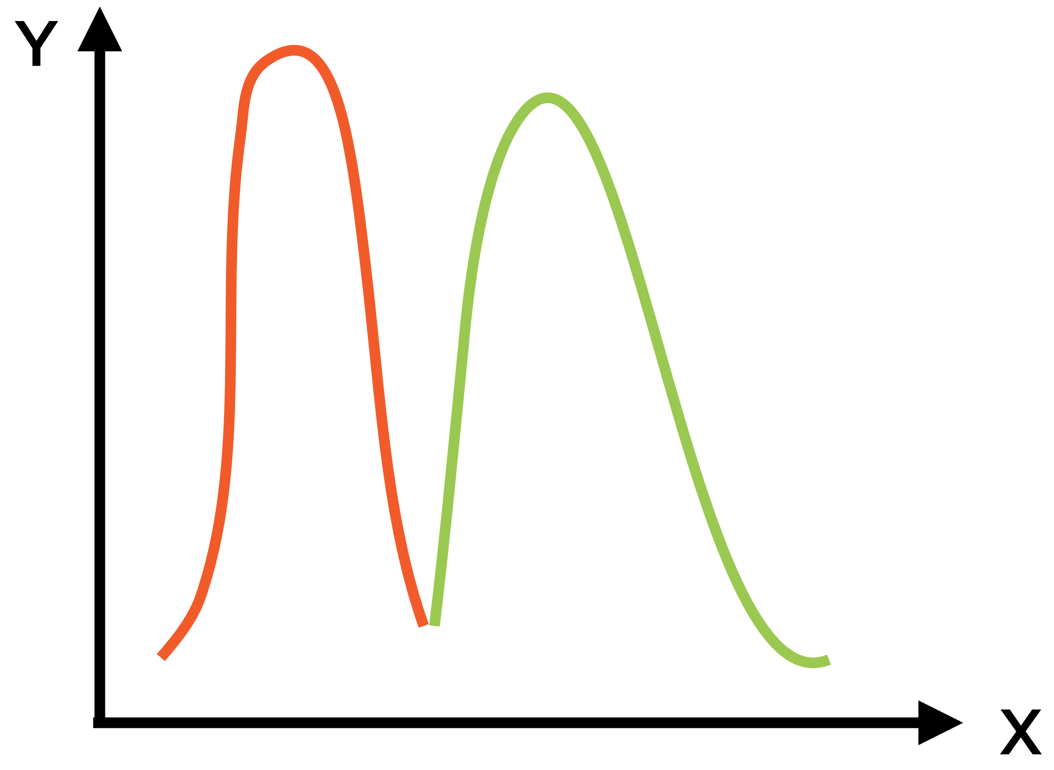
Median is also a measure of central tendency and inter-quartile range is also measure of dispersion

Robust Standardization

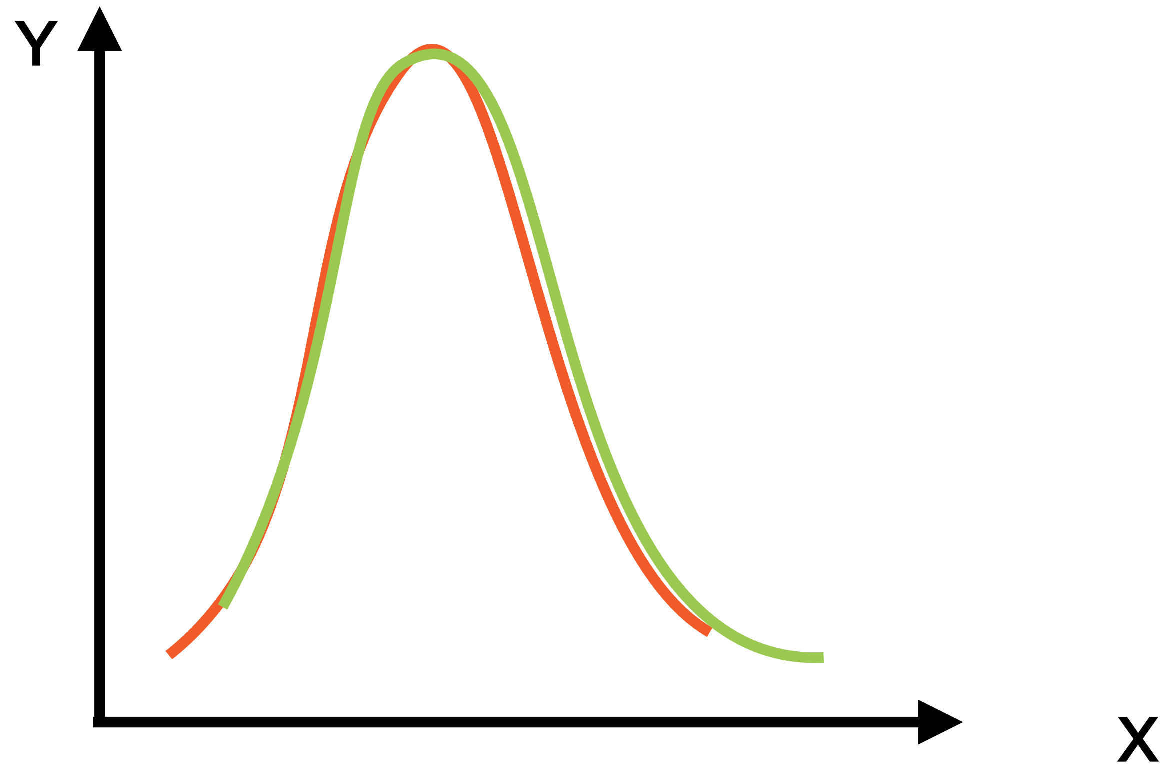
$$z = \frac{x_i - \text{median}(x)}{\text{Inter-quartile Range}(x)}$$

Output does not change much due to outliers

Robust Standardization



Before



After

Normalizing Data

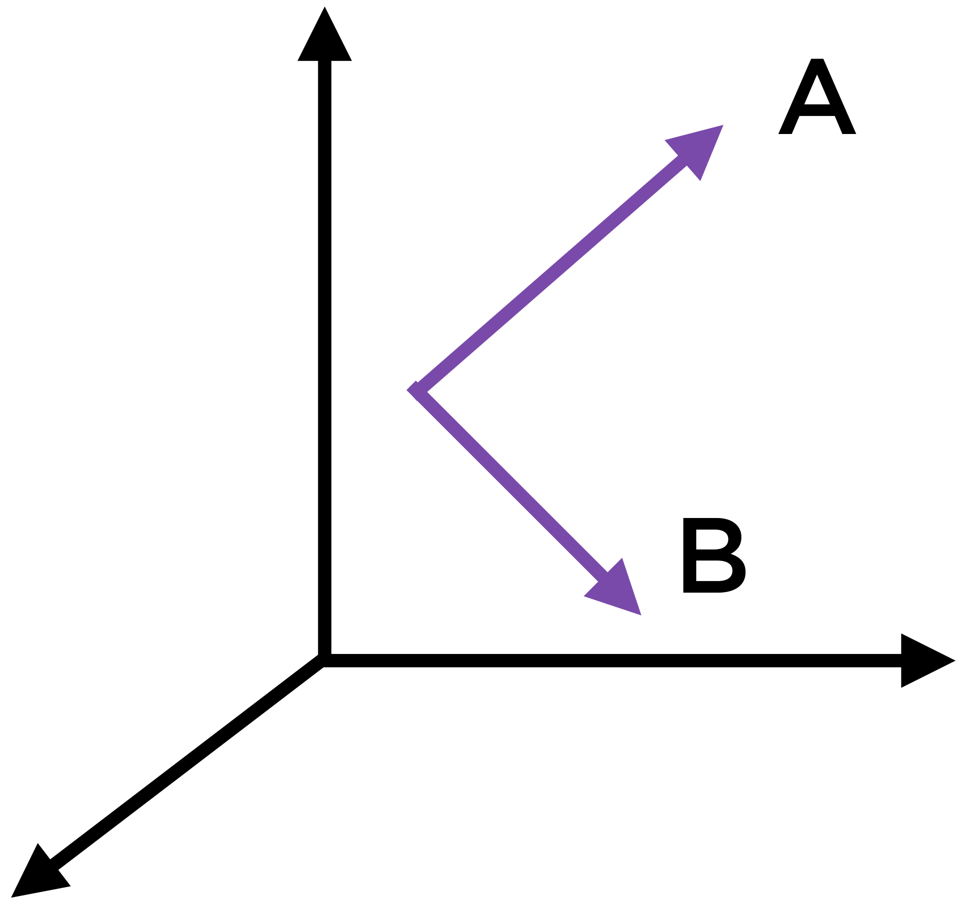
Normalization

Process of scaling input vectors individually to unit norm (unit magnitude), often in order to simplify cosine similarity calculations.

Cosine Similarity

Cosine similarity is a measure of similarity between two non-zero vectors, widely used in ML algorithms - especially in document modeling applications.

Orthogonal Vectors



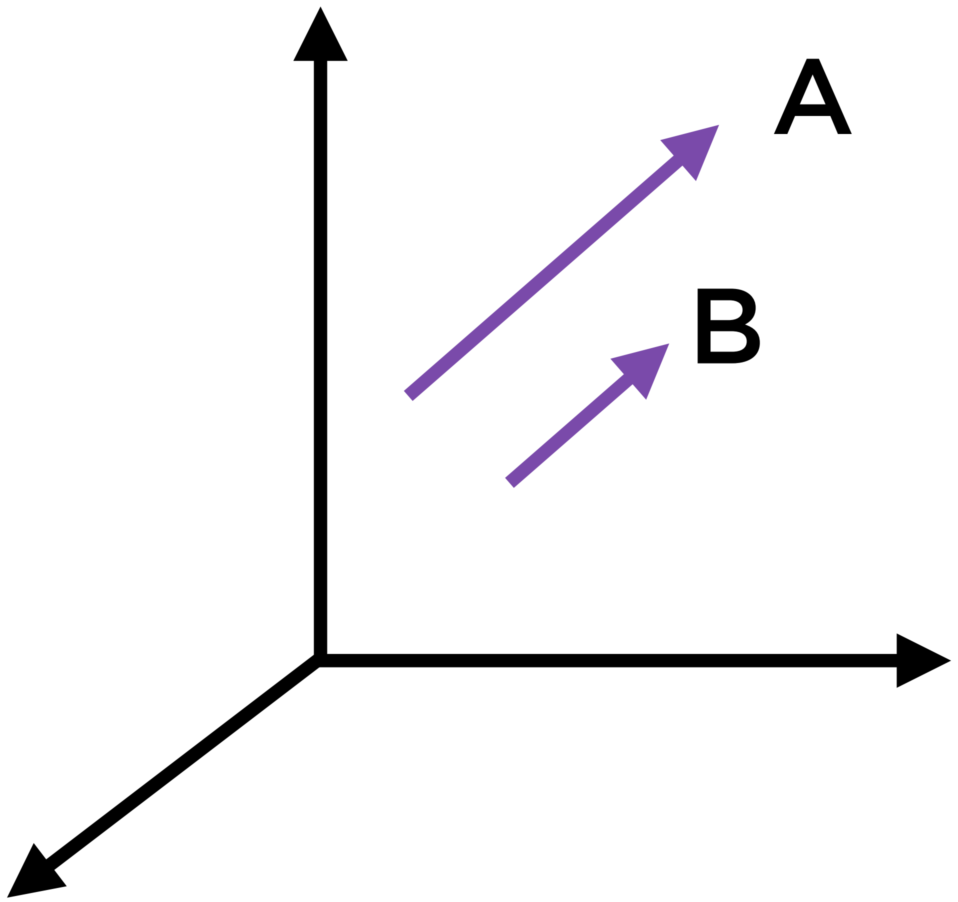
Vectors A and B are at 90 degrees

Orthogonal vectors represent uncorrelated data

A and B are unrelated, independent

Cosine of 90 degrees = 0

Aligned Vectors



Vectors A and B are parallel

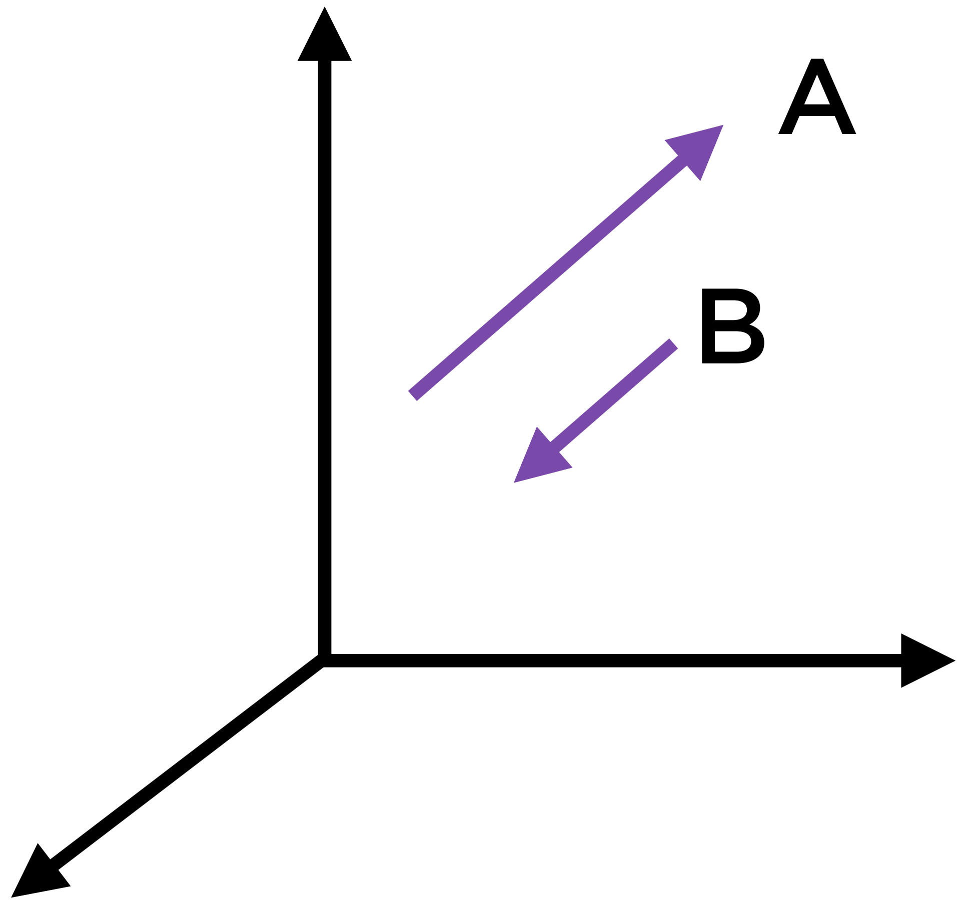
Angle between them is 0 degrees

Perfectly aligned

Correlation of 1 (highest possible)

Cosine of 0 degrees = 1

Opposite Vectors



Vectors A and B point in opposite directions

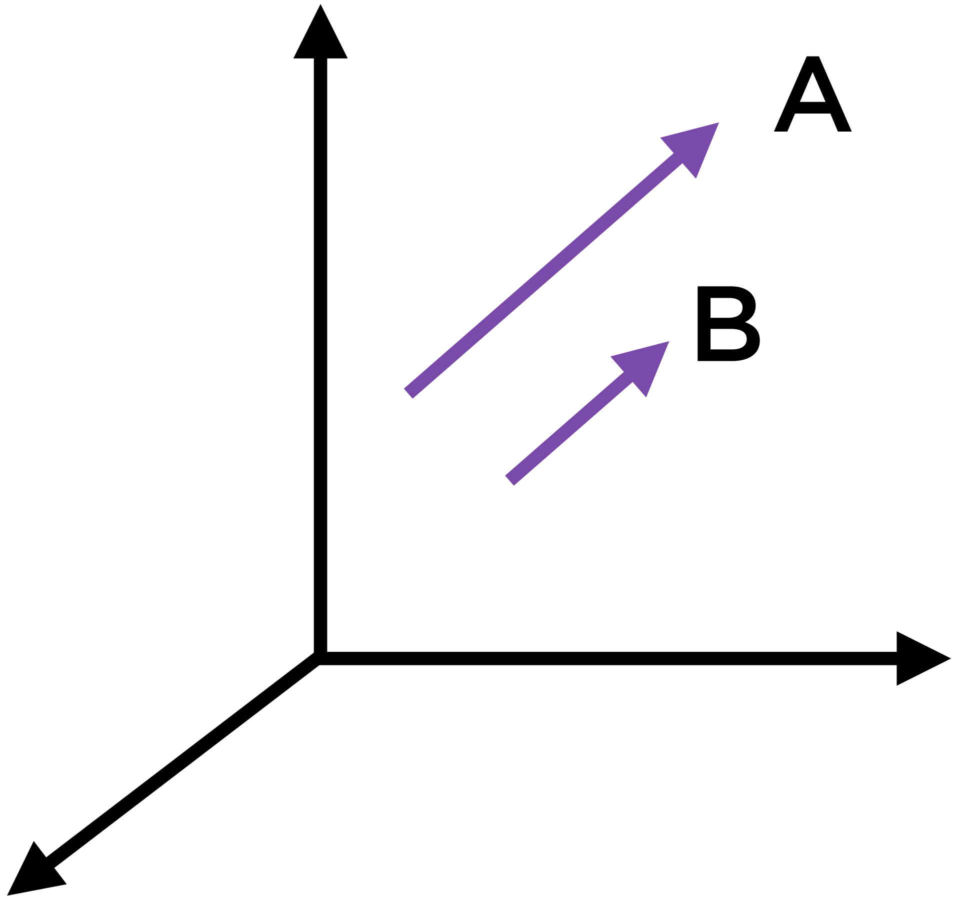
Angle between them is 180 degrees

Perfectly opposed

Correlation of -1 (lowest possible)

Cosine of 180 degrees = -1

Cosine Similarity



Quick and intuitive way to express alignment between two vectors

Each vector represents a single point

In three dimensions, a point is represented as

$$(x_i , y_i , z_i)$$

Cosine Similarity

$$\cos(\theta) = \frac{A \, . \, B}{||A|| \; ||B||}$$

$$||A||^2 = x_A{}^2 + y_A{}^2 + z_A{}^2$$

$$||B||^2 = x_B{}^2 + y_B{}^2 + z_B{}^2$$

$$A.B = x_A x_B + y_A y_B + z_A z_B$$

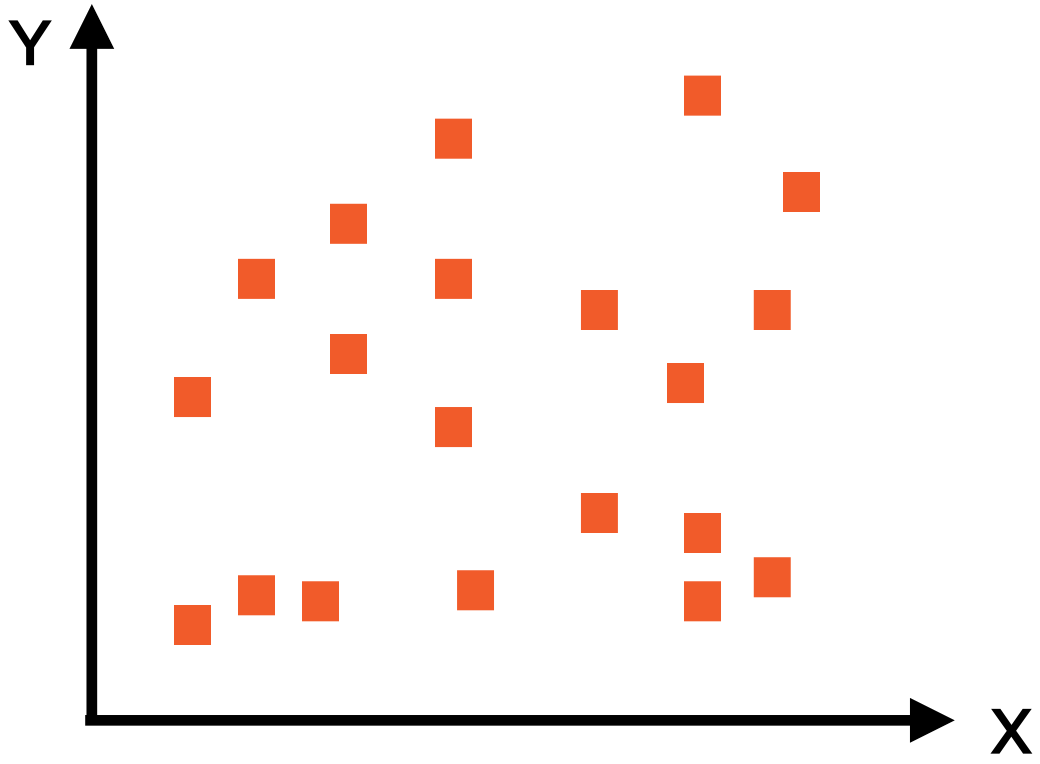
Normalization

Pre-convert A and B to unit norm vectors to simplify calculation

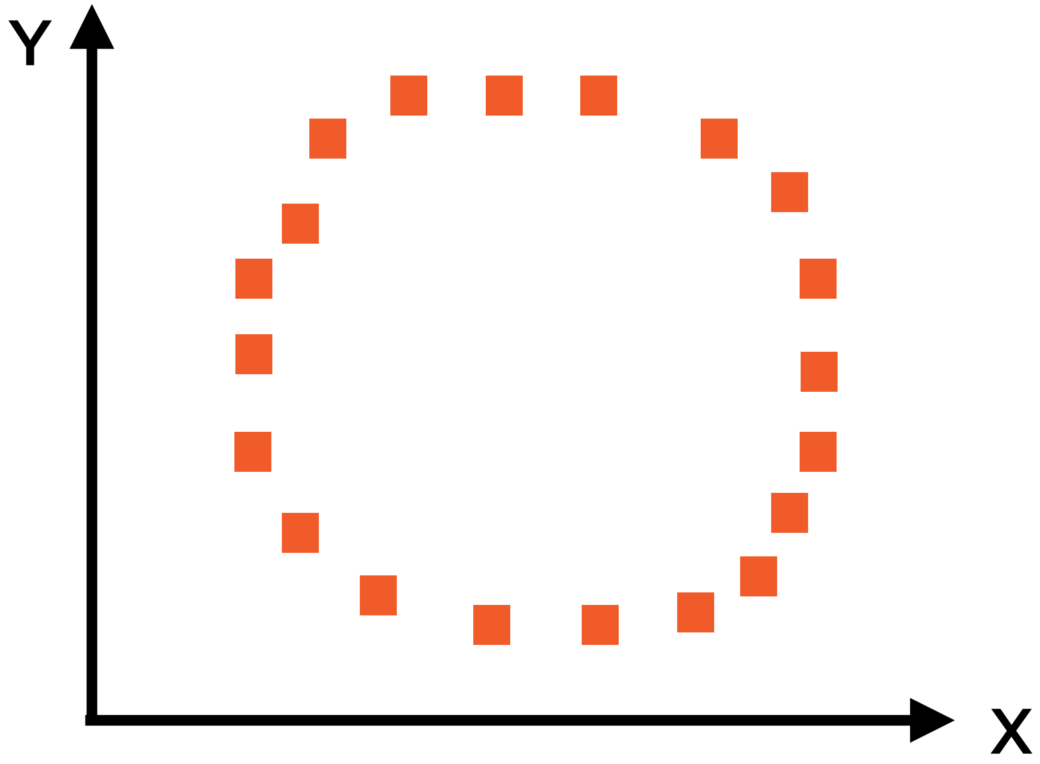
$$a = \frac{A}{||A||} = \frac{(x_A, y_A, z_A)}{sqrt(x_A^2 + y_A^2 + z_A^2)}$$

$$b = \frac{B}{||B||} = \frac{(x_B, y_B, z_B)}{sqrt(x_B^2 + y_B^2 + z_B^2)}$$

Normalization



Before Normalization



After Normalization

Normalizing is a row-wise operation, while scaling is a column-wise operation

Different Norms

L1

Sum of absolute values of components of vector

L2

Traditional definition of vector magnitude

max

Largest absolute value of elements of vector

L1-norm

$$x_{new} = \frac{x}{|x| + |y| + |z|}$$

L2-norm

$$x_{new} = \frac{x}{\mathrm{sqrt}(x^2 + y^2 + z^2)}$$

max norm

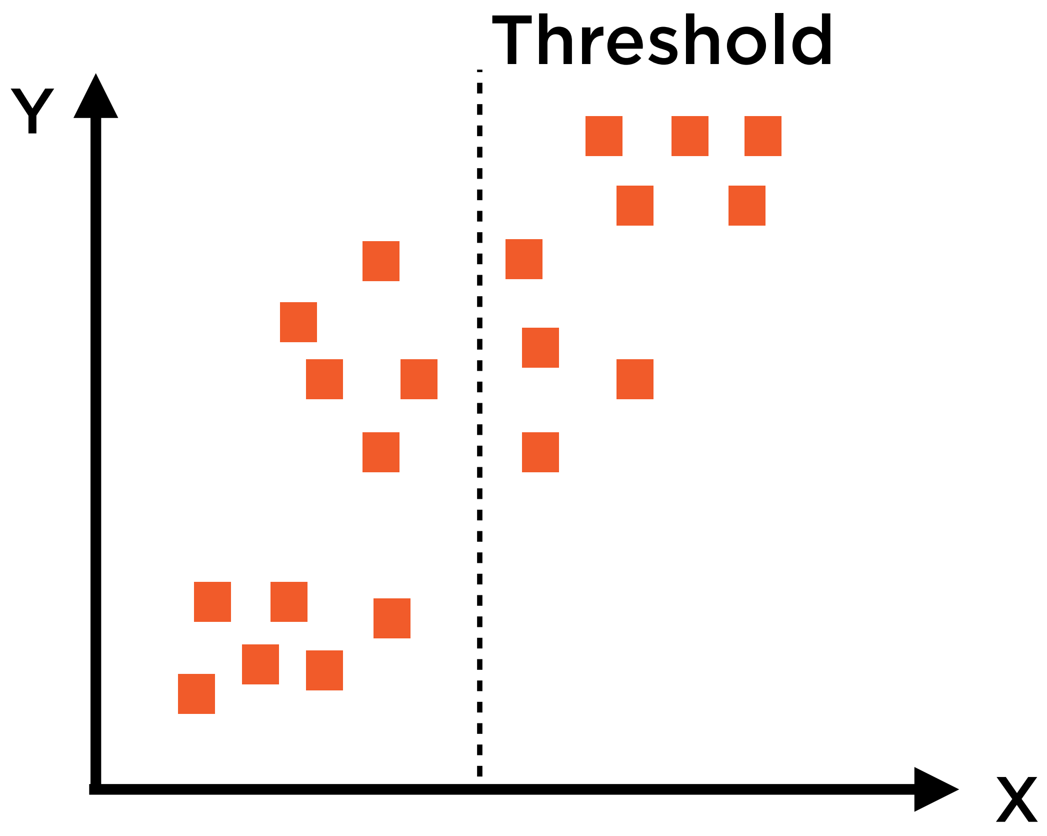
$$x_{new} = \frac{x}{\max(x)}$$

Binning Data

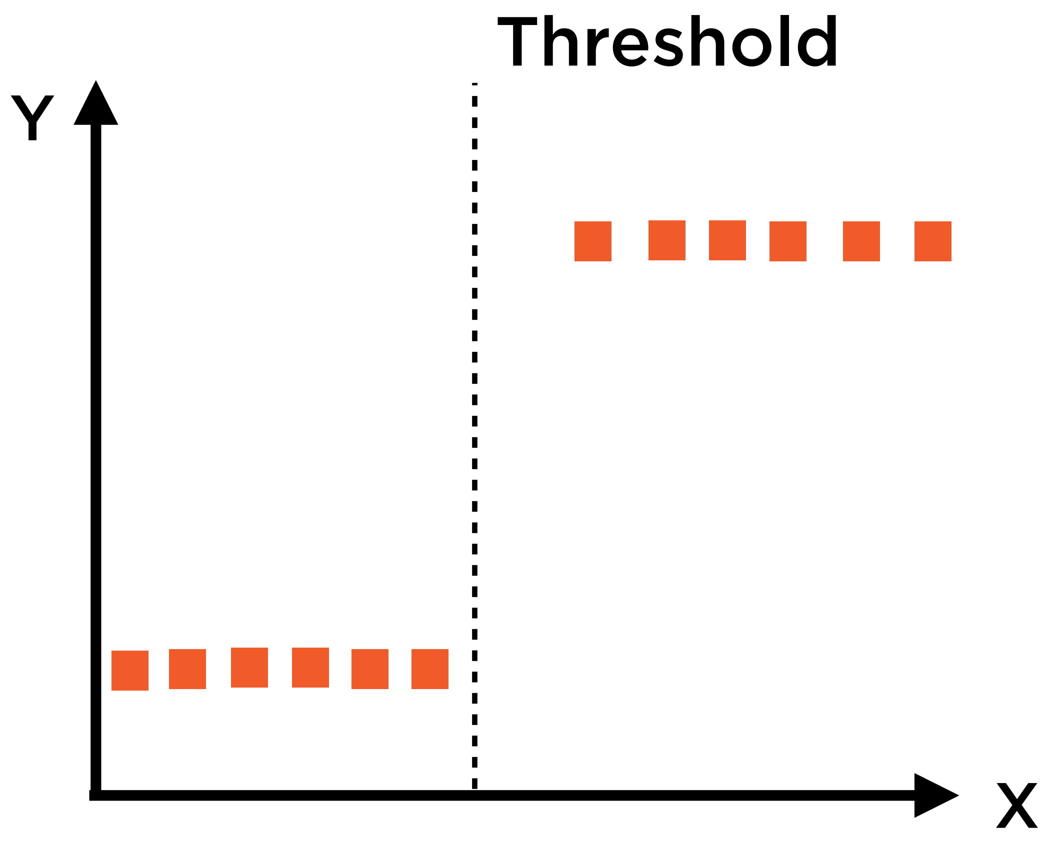
Data Binarization

Converts continuous variable into a binary categorical variable based on a threshold specified by user.

Binarizing Data



Continuous
Input

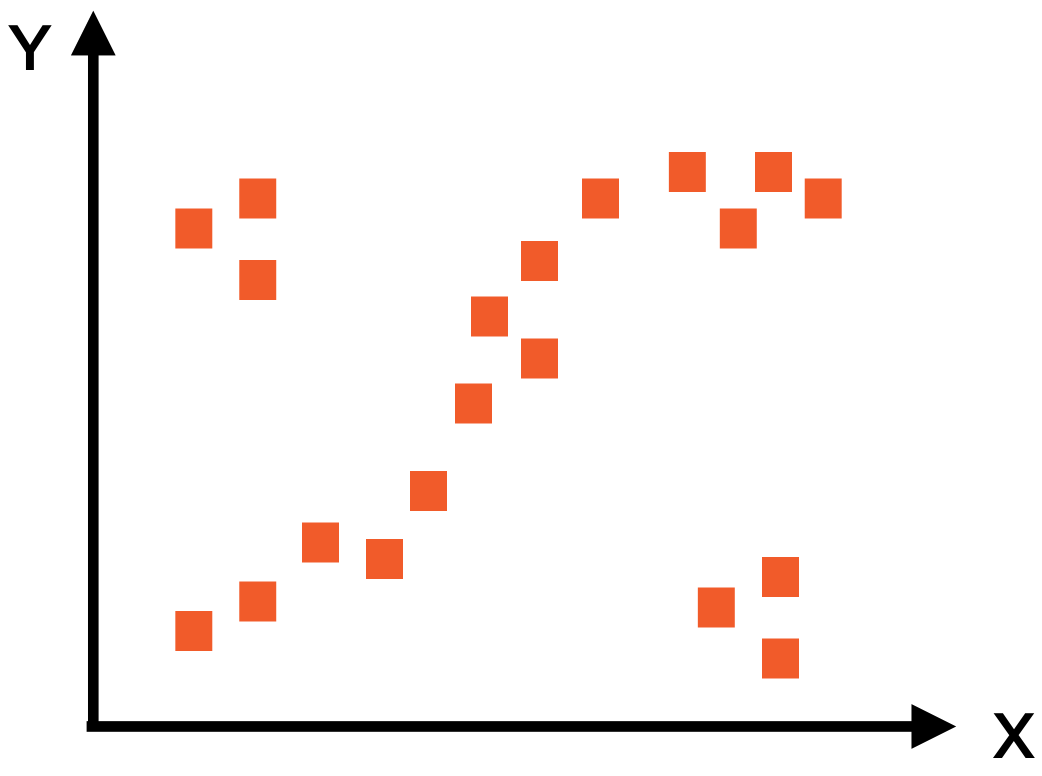


Binary Categorical
Output

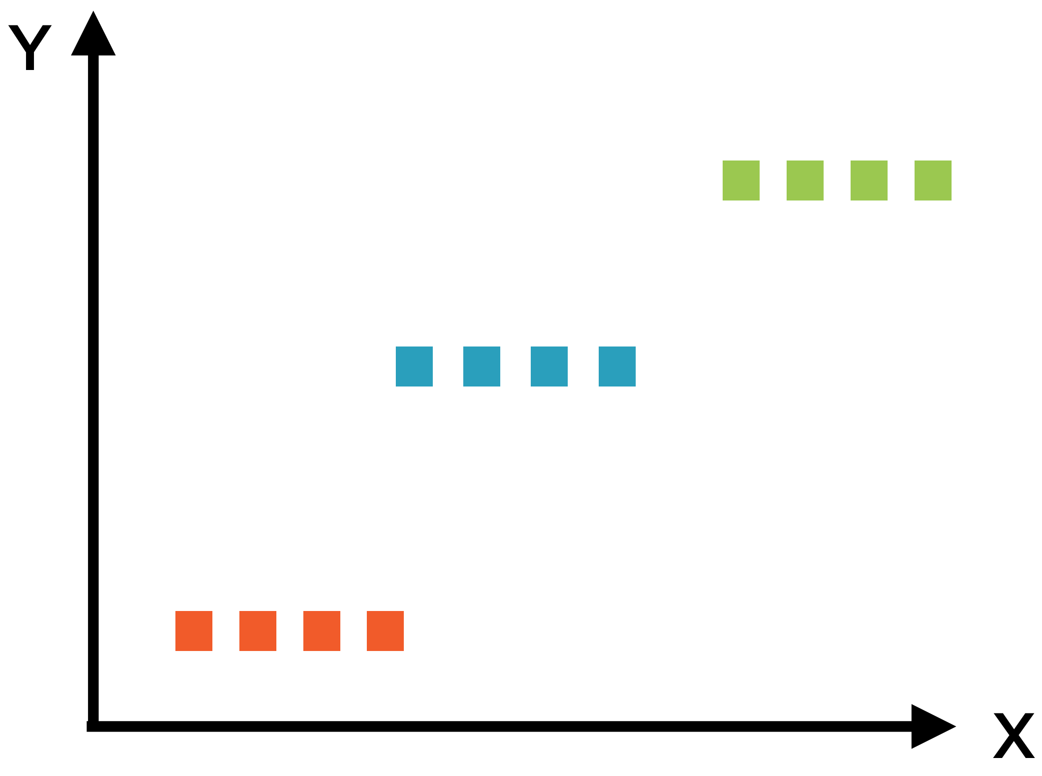
Discretizing Data

Generalizes idea of binarization; converts continuous data into categorical data arranged into a specified number of bins.

Discretizing Data



Before



After (3 Bins)

Binning Strategies

Uniform

Bin widths are constant in each feature

Quantile

All bins in each feature have approximately the same number of samples

K-means

Bins based on the centroids of a K-means clustering procedure

Big Data

Essential Steps in Connecting the Dots

Processing Data for Use in Models

Incorporating Real-world Data into Models

Essential Steps in Connecting the Dots

Processing Data for Use in Models

Incorporating Real-world Data into Models

Transactional and Analytical Processing

Order Management Support

John is responsible for tracking and delivering orders on time

Revenue Analyst

Anna is responsible for tracking and monitoring revenues

Order Management Support

20 deliveries in Kent, WA are delayed

The courier company has had a computer outage

John assigns the orders to another courier company in the region

Revenue Analyst

Her manager wants an update on last month's revenues

Last month was an unusually slow one

Anna pulls up data for the last 5 years to check for seasonal effects

Transactional and Analytical Processing

Transactional Processing

Analytical Processing

Transactional and Analytical Processing

Transactional Processing	Analytical Processing
Ensure correctness of individual entries	Analyzes large batches of data
Access to recent data, from the last few hours or days	Access to older data going back months, or even years
Updates data	Mostly reads data
Fast real-time access	Long running jobs
Usually a single data source	Multiple data sources

Transactional and Analytical Processing

Small Data

Both these objectives could be achieved using the same database system

Small Data

Single machine with backup

Structured, well-defined data

Can access individual records or the entire dataset

No replication, updated data available instantaneously

Different tables store data from different sources

Transactional and Analytical Processing

Big Data

Very hard to meet all requirements with the same database system

Big Data

Data distributed on a cluster with multiple machines

Semi-structured or unstructured data

No random access to data

Data replicated, propagation of updates take time

Different sources may have different unknown formats

Transactional and Analytical Processing

Transactional Processing

Traditional
RDBMS

Analytical Processing

Data Warehouse

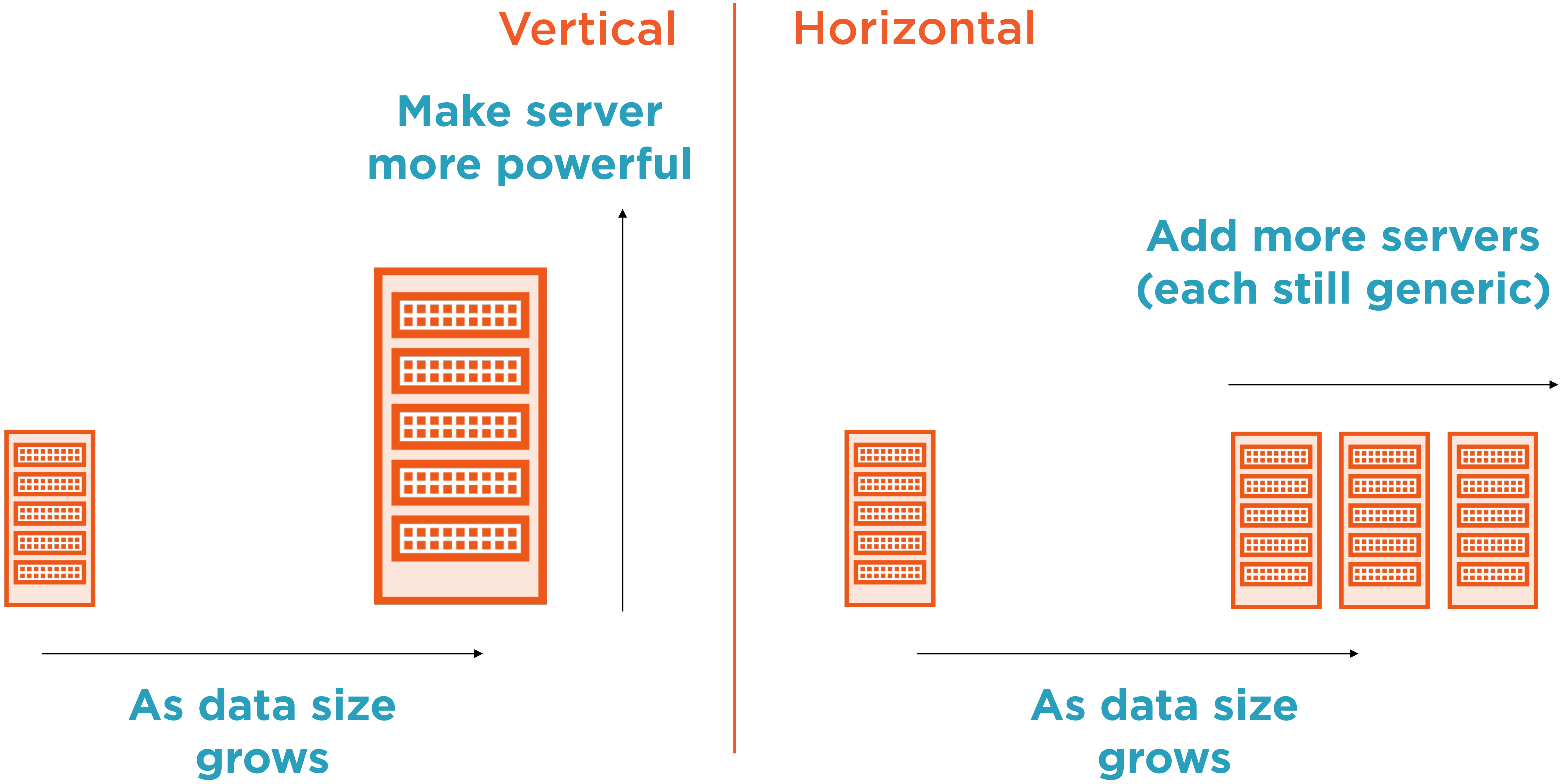
3 Vs of Big Data

Volume: Amount of data

Variety: Number and type of sources

Velocity: Batch and streaming

Two Types of Scaling



Two Types of Scaling

Vertical	Horizontal
Monolithic architecture	Distributed architecture
No need for orchestration	Orchestration required
No need to shard data	Need to shard data
No need to replicate data	Need to replicate data
No replication delay	Incur replication delay

Two Types of Scaling

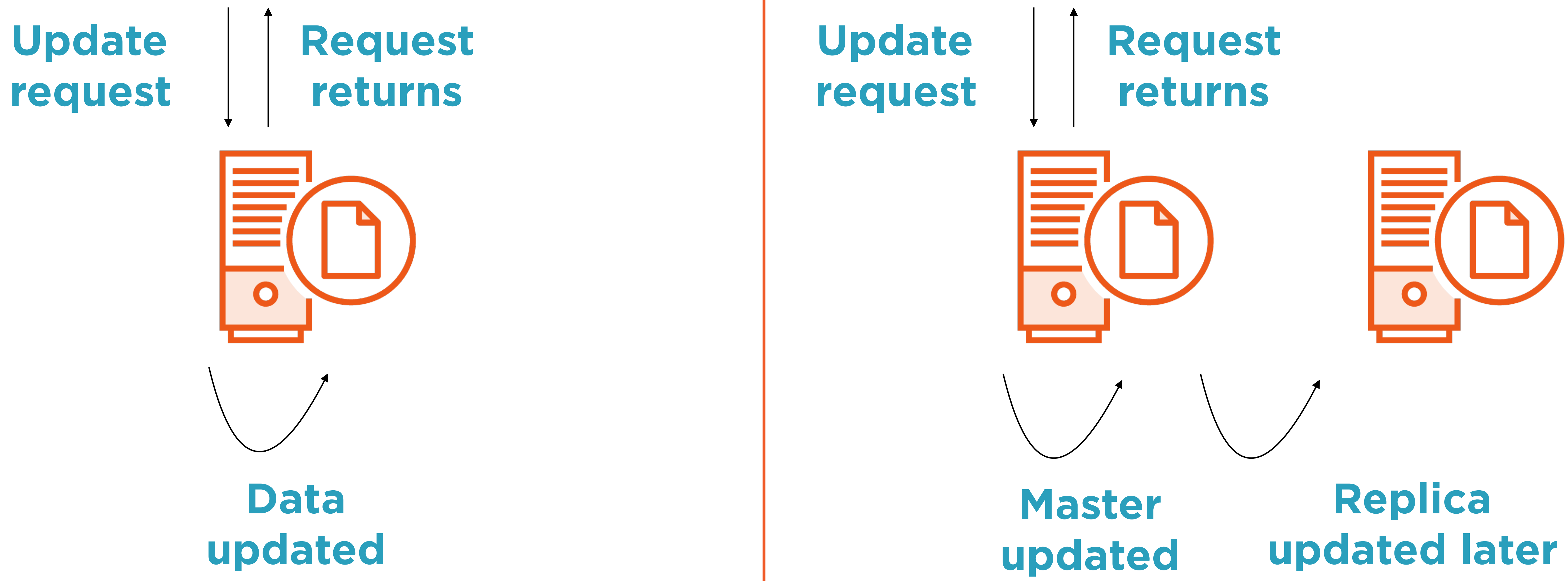
Vertical	Horizontal
Easy to ensure consistency	Hard to ensure consistency
Can offer strong consistency fairly easily	Usually offer only eventual consistency
ACID support easy to provide	ACID support hard to provide
So, usually used in OLTP applications	So, usually used in OLAP applications

Two Types of Scaling

Vertical	Horizontal
Easy to ensure consistency	Hard to ensure consistency
Can offer **strong consistency** fairly easily	Usually offer only **eventual consistency**
ACID support easy to provide	ACID support hard to provide
So, usually used in OLTP applications	So, usually used in OLAP applications

Consistency Guarantees

Strong | **Eventual**



Batch and Stream Processing

Bounded datasets are processed in **batches**

Unbounded datasets are processed as **streams**

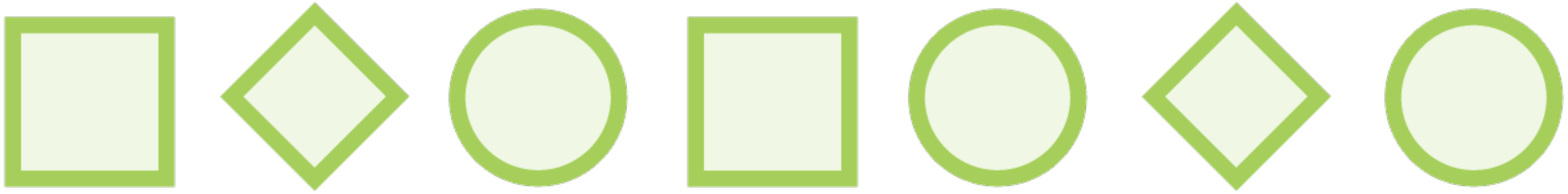
Batch vs. Stream Processing

Batch	Stream
Bounded, finite datasets	Unbounded, infinite datasets
Slow pipeline from data ingestion to analysis	Processing immediate, as data is received
Periodic updates as jobs complete	Continuous updates as jobs run constantly

Batch vs. Stream Processing

Batch	Stream
Order of data received unimportant	Order important, out of order arrival tracked
Single global state of the world at any point in time	No global state, only history of events received

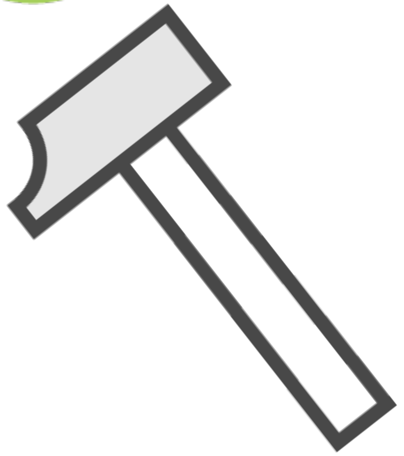
Stream Processing



Data is received as a stream

- Log messages
- Tweets
- Climate sensor data

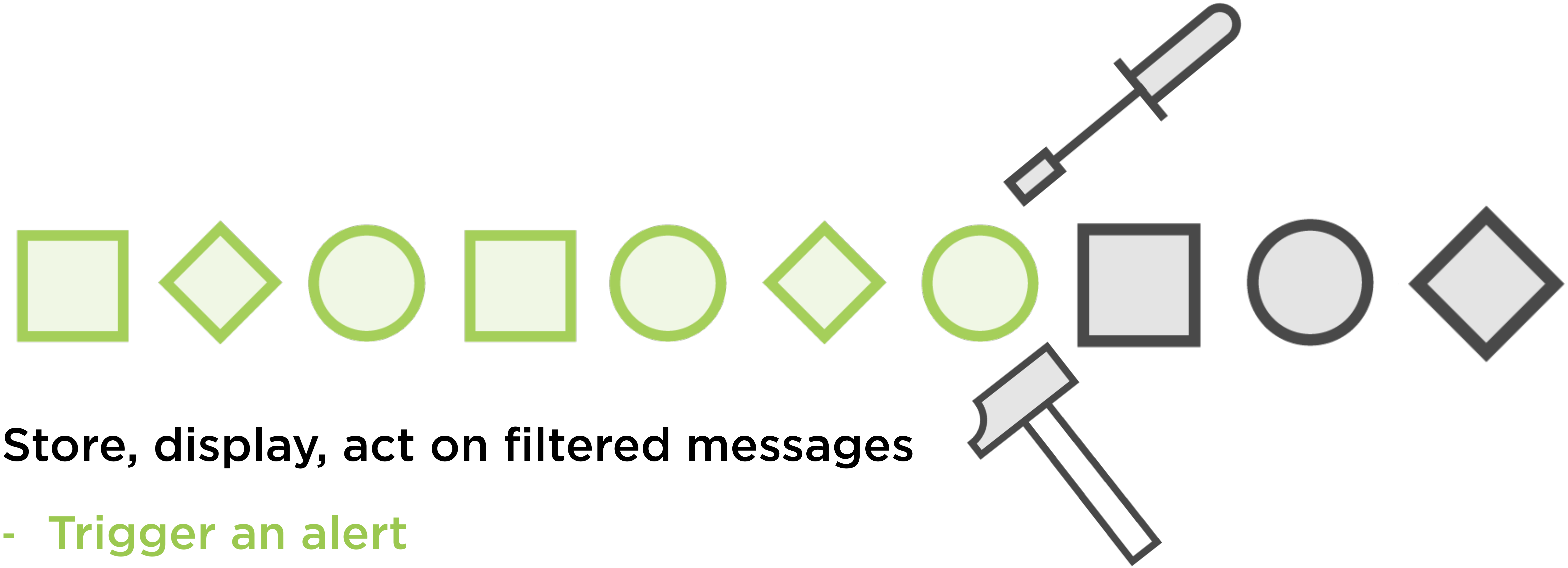
Stream Processing



Process the data one entity at a time

- Filter error messages
- Find references to the latest movies
- Track weather patterns

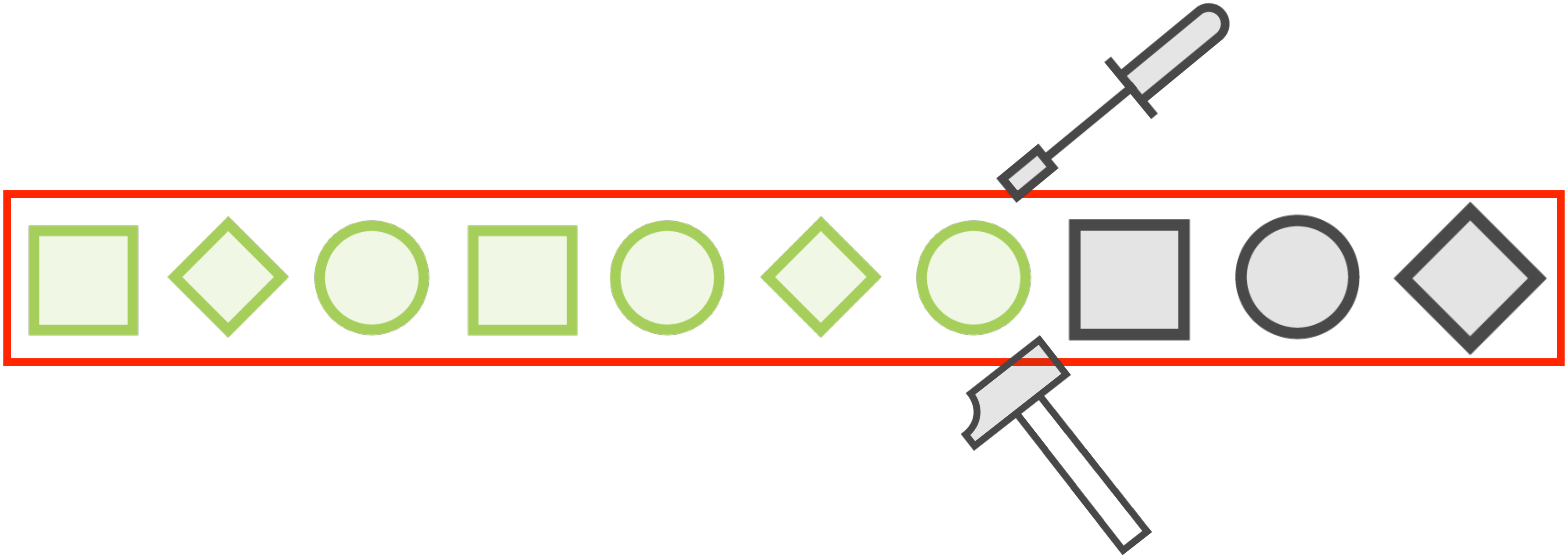
Stream Processing



Store, display, act on filtered messages

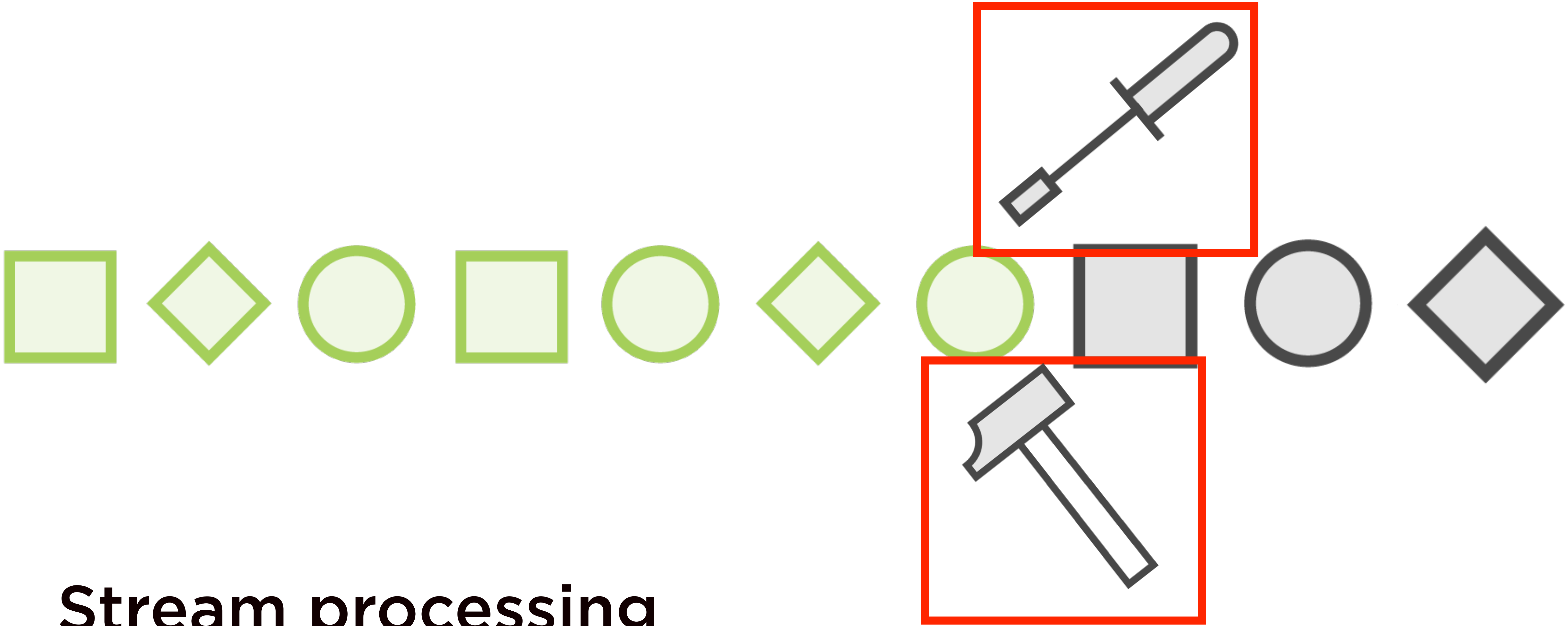
- Trigger an alert
- Show trending graphs
- Warn of sudden squalls

Stream Processing



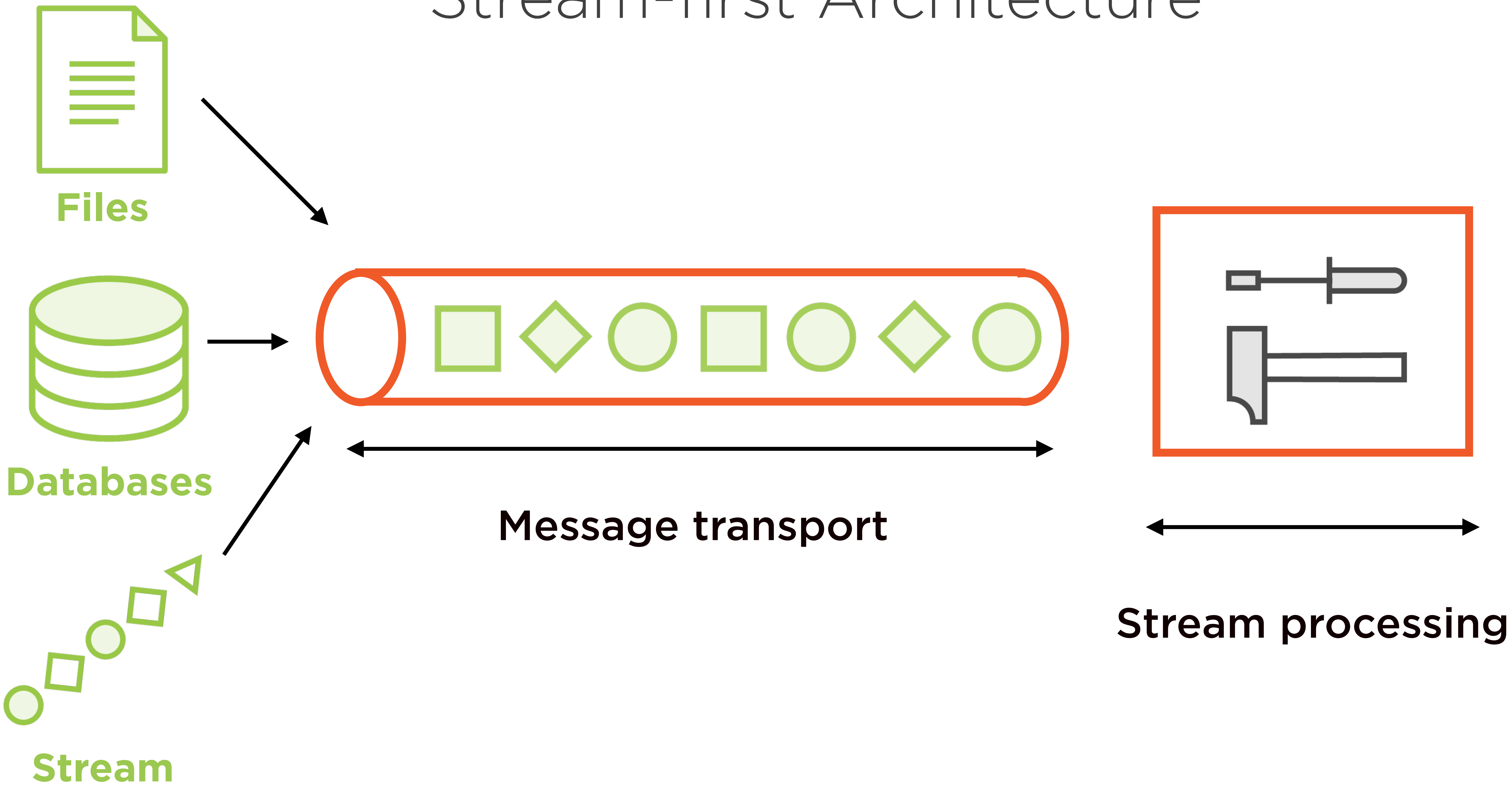
Streaming data

Stream Processing

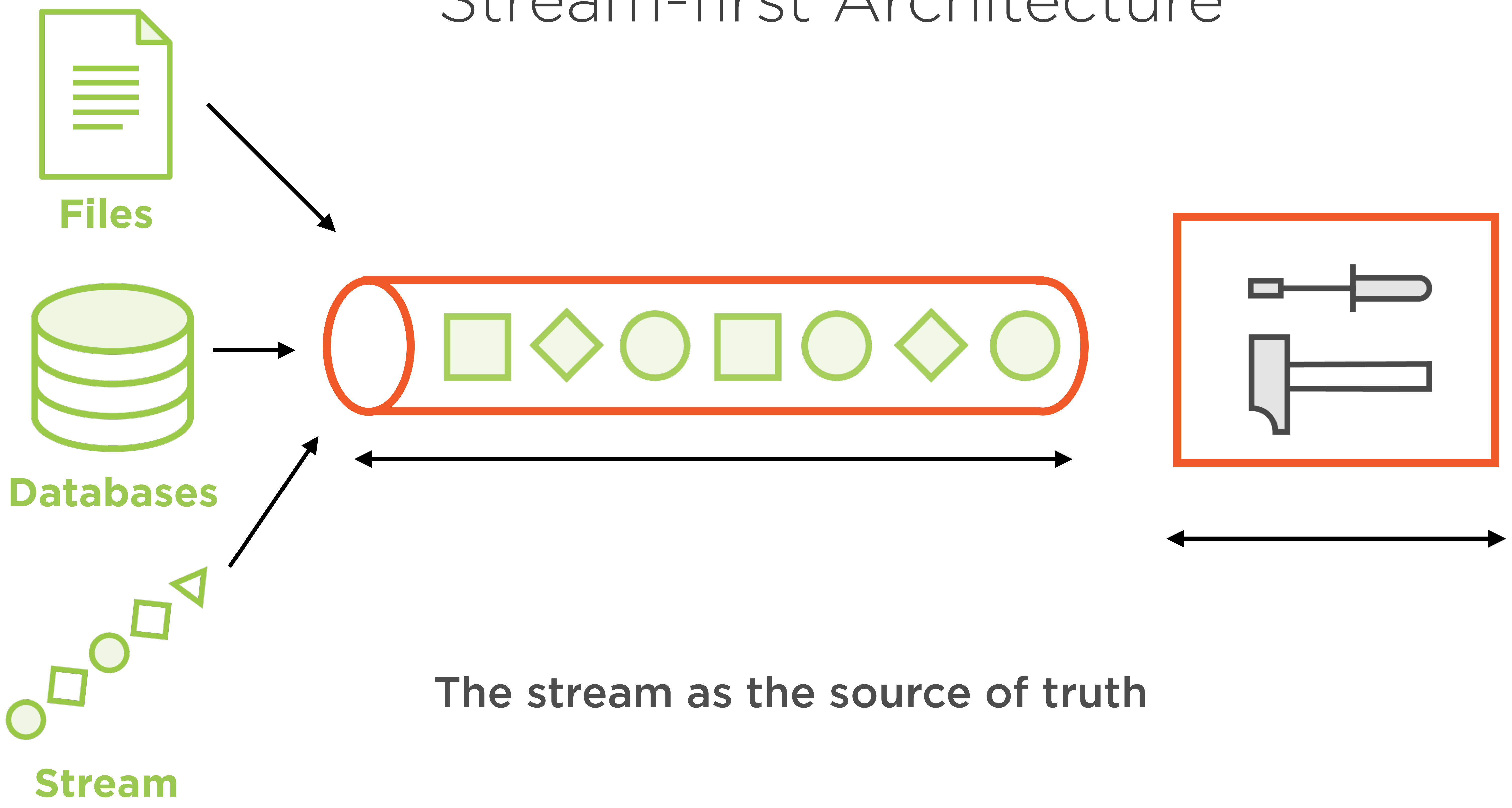


Stream processing

Stream-first Architecture



Stream-first Architecture

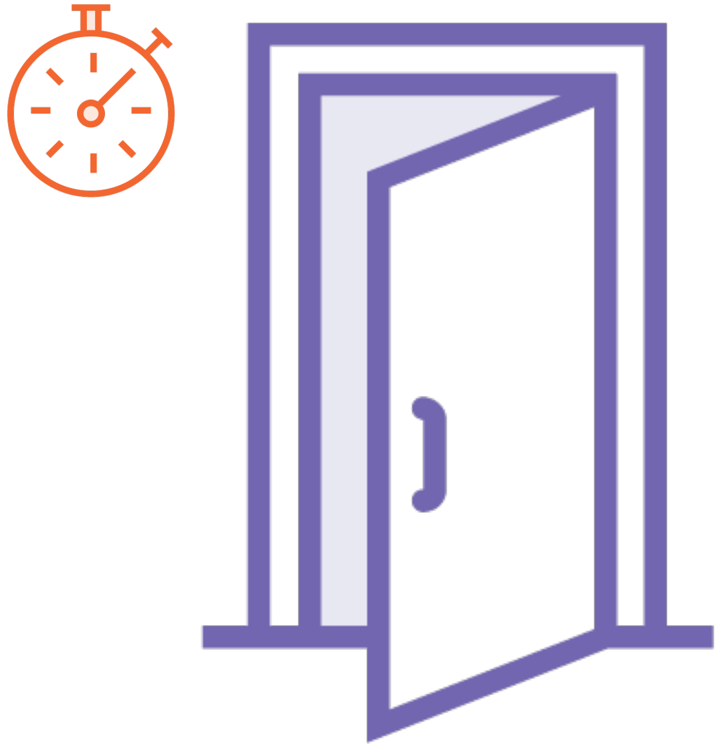


The stream as the source of truth

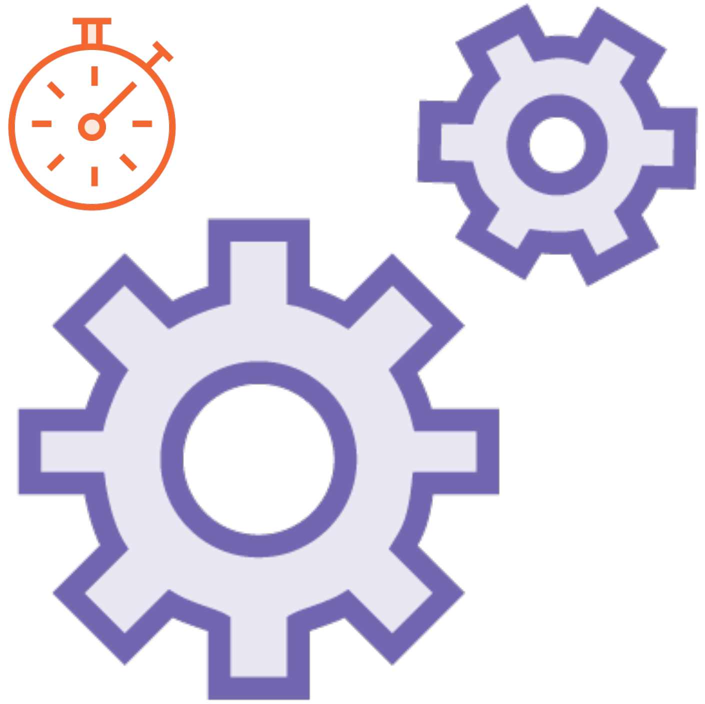
Event Time and Processing Time

Time

Event Time



Ingestion Time



Processing Time

Event Time

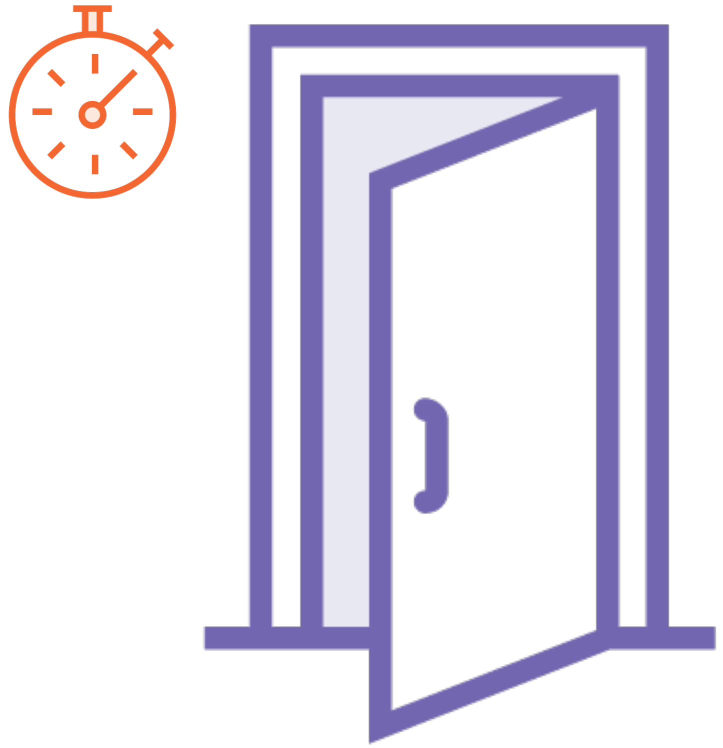
The time at which the event occurred at its original source

- Mobile phone, sensor, website

Usually embedded within records

Gives correct results in case of out of order or late events

Ingestion Time

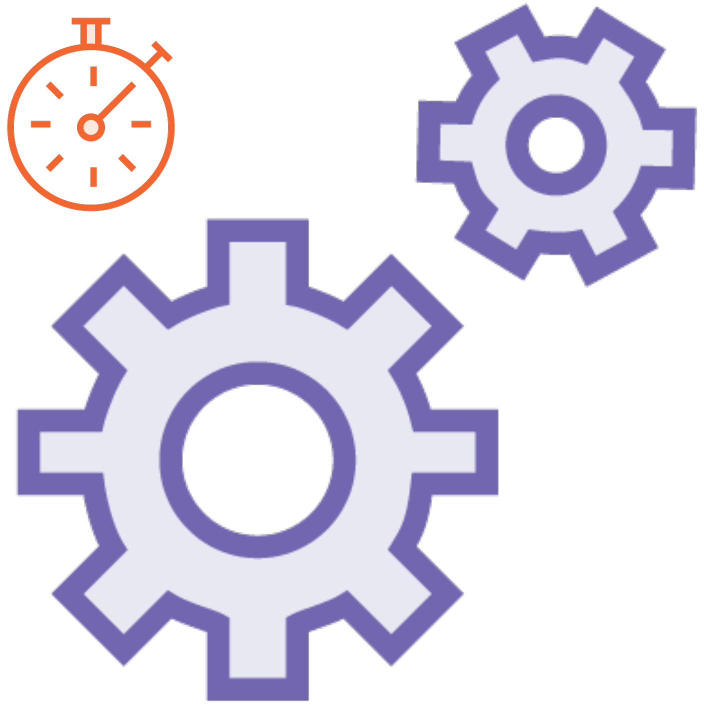


The time at which the event enters the processing system via a source

Timestamp given by system chronologically after the event time

Cannot handle out of order events

Processing Time



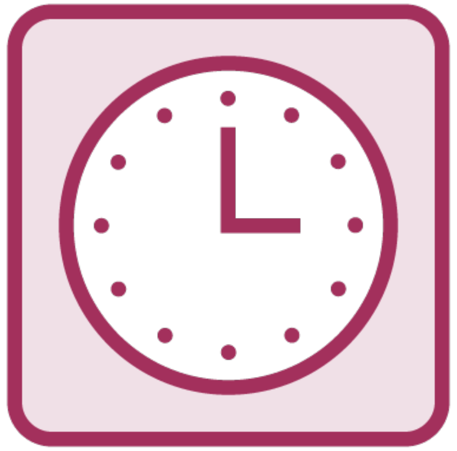
The system time of the machine processing entities

Chronologically after event time and ingestion time

Non-deterministic, depends on when data arrives, how long operations take

Simple, no coordination between streams and processors

How Late is Late?



Class At 9 am

Class starts when clock strikes 9

Is 9:01 Late?

Realistically, at least some folks are going to be a minute late

Is 10:10 late?

A student is an hour late - allow in or send back?

Late Data

The professor “knows” what lateness is reasonable

Students entering within this reasonable lateness are late but OK

Students entering after this reasonable lateness are too late

“Allowed Lateness”

Watermarks and Late Data

The system “knows” what lateness is reasonable

Data entering within this reasonable lateness is late but OK

Data entering after this reasonable lateness is too late

Watermarks and Late Data

Watermark

Threshold of allowed lateness (event time)

Late Data

Data within watermark is processed

Dropped Data

Data outside watermark is dropped

Summary

Standardization and normalization

Binning and sampling

Big data

Batch vs. streaming data

Event time and processing time