Apache Beam in the Data Analytics Life Cycle

Griselda Cuevas

Product Manager - Google Cloud

http://linkedin.com/in/griscz

Data industry trends

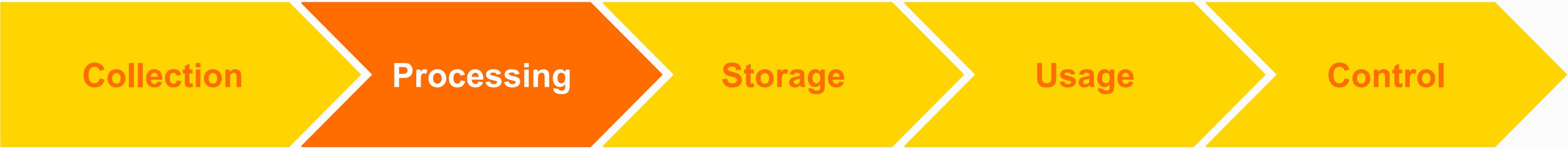
Happening Now

- Migration to the Cloud
- Massive amounts of (raw) data
- Emergence of new regulations
- Need to reduce time to insights

Emerging Trends

- Data reliability
- Real-time analytics
- Governed data democratization
- AI/ML operationalization

Data analytics & data processing



The data analytics flow

Data analytics is an overarching practice that encompasses the complete life cycle of insight generation, from collection to quality and access control.

Data processing is a component of the Data Analytics practice. It transforms raw data into valuable insights and information.

Data processing is done in three phases:

Ingestion

Processing

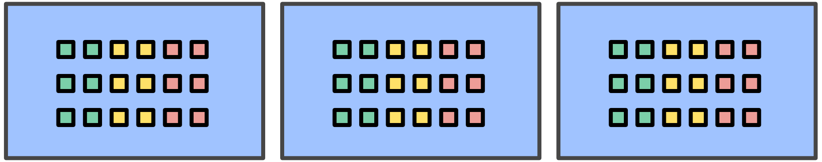
Writing

There are two types of data processing

Batch

Data is collected and processed in chunks.
It is Used for large amounts of data.

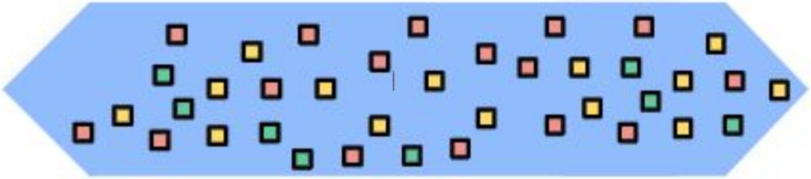
E.g.: payroll systems, preventive manufacturing maintenance, insurance billing, etc.



Streaming (Real-Time)

It is the continuous processing of data that aims to derive insights or new information shortly after a data point enters a system for the first time.

E.g.: experience personalization, anomaly detection, malfunction aletring system, etc.



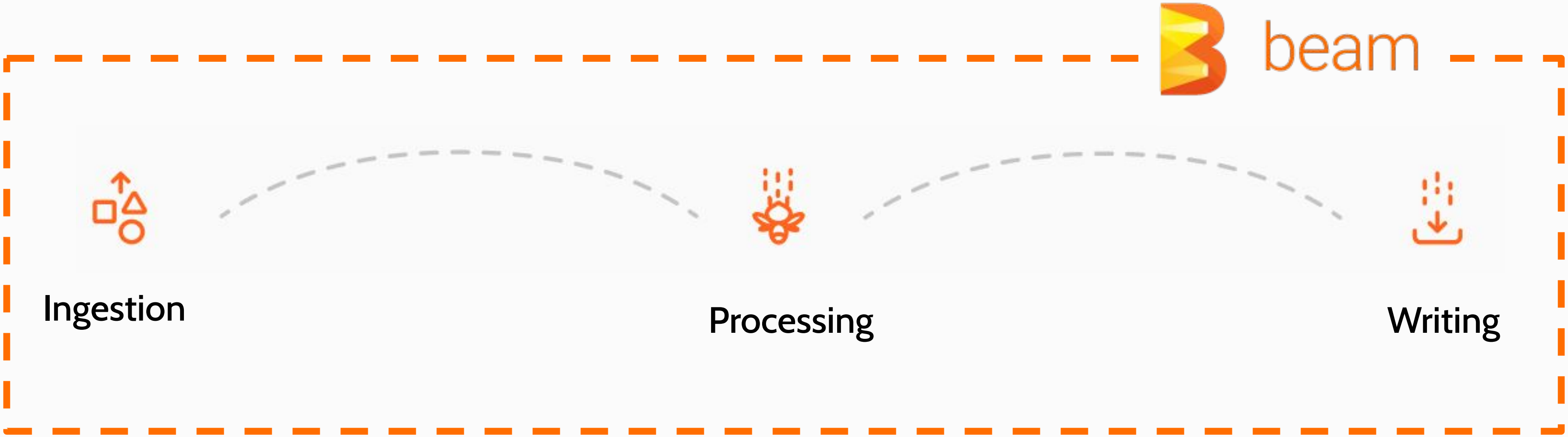
Where does Apache Beam fits in?

A common misconception...

Apache Beam is a substitu pache Spark or Apache Flink

Truth is...

Apache Beam is a programming model
to build batch and streaming data processing pipelines



You can build data processing in 3 steps:

1. Choose your preferred runner
2. Write your pipeline in your favorite programming language
3. Use Apache Beam transformations to process your data

Step 1. Choose your runner, Apache Beam is portable!

You can run Apache Beam pipelines in any supported runner

including Apache Spark, Apache Flink and Dataflow

Step 2. Choose your favorite language, Apache Beam is multi-language

You can develop Apache Beam pipelines in your language of choice: Java, Python, SQL and Go

Step 3. Use the 3 Apache Beam components to define your pipelines

Input Connectors

- File systems
- Messaging services
- Databases
- Caches
- Others

Operations

- GroupByKey
- CoGroupByKey
- Combine
- Flatten
- Partition
- ...

Output Connectors

- File systems
- Messaging services
- Databases
- Caches
- Others

In today’s module

- Apache Beam in action
- Apache Beam Overview
- Defining a directed acyclic graph
- Runner specific overview: Architecture, management and autotuning
- Putting it all together with a Python demo

Thank You!

BEAM

COLLEGE