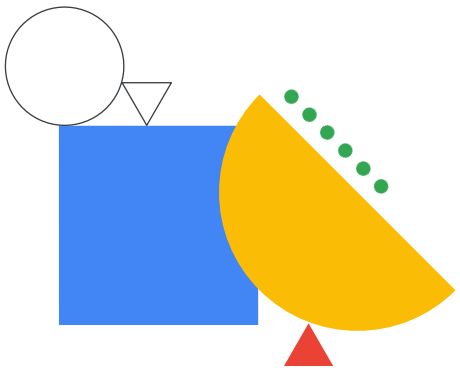
Google Cloud

Building Resilient Streaming Analytics Systems on Google Cloud



Data Engineering learning path

1. Modernizing Data Lakes and Data Warehouses with Google Cloud
2. Building Batch Data Pipelines on Google Cloud
3. Building Resilient Streaming Analytics Systems on Google Cloud
4. Smart Analytics, Machine Learning and AI on Google Cloud

Building Resilient Streaming Analytics Systems on Google Cloud is the third course of the Data Engineering Learning Path and it talks all about building resilient streaming analytics systems. Those systems allow enterprises to take accurate and timely decisions as data points are generated in real time. This course discusses what streaming data processing is, how it fits in your overall big data architecture, when streaming data processing makes sense, and what Google Cloud technologies and products you can choose from to build your own resilient streaming analytics solutions.

Course agenda

Introduction to Processing Streaming Data

Serverless Messaging with Pub/Sub

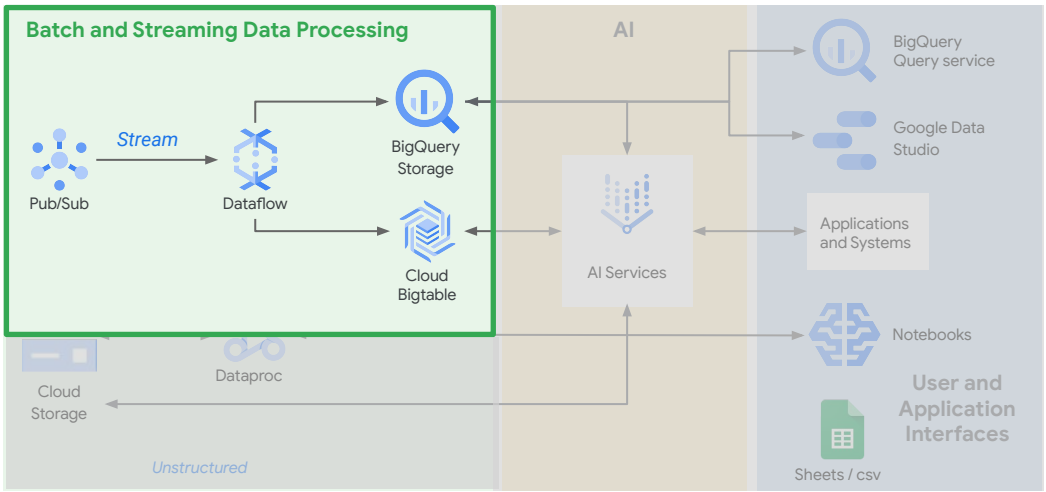
Dataflow Streaming Features

High-Throughput BigQuery and Bigtable Streaming Features

Advanced BigQuery Functionality and Performance

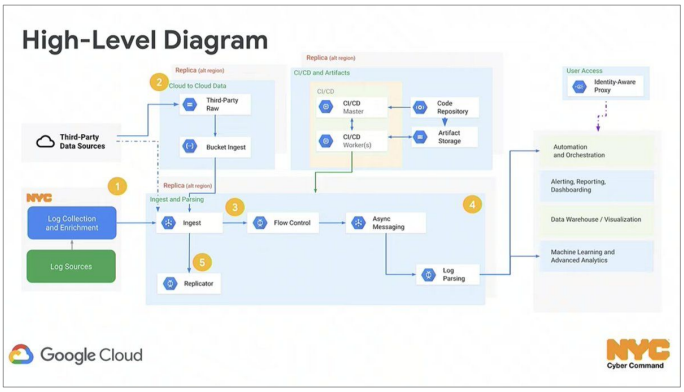
Here's how the course is broken down. We start off with what streaming data is and the challenges associated with processing streaming data. Like variable volumes and latency. Next we look at using Pub/Sub, Dataflow, and BigQuery to help us ingest, process, and derive insights from data as it streams in. We dive into each product and learn about its Streaming capabilities. Then, we'll also look at Bigtable when higher throughput is a requirement. Finally, we review some of BigQuery's advanced analysis capabilities like GIS Functions and ways to improve query performance.

Streaming Data Processing



This module is all about streaming and we will be discussing the part of the reference architecture. Data typically comes in through the Pub/Sub, then that data goes through aggregation and transformation in dataflow. Then, use BigQuery or Cloud Bigtable depending on whether you are trying to write aggregates or individual records coming in from streaming sources.

Many enterprises want to enable their analysts to be able to make decisions in real-time; NYC3 did it



"Real time is king, and that's the only data valuable to us,"

-- Noam Dorogoyer, New York City Cyber Command

Article in GCN: https://gcn.com/articles/2019/08/01/nyc-cloud-cyber-pipeline.aspx
Talk at NEXT 2019: https://www.youtube.com/watch?v=x4yQY8yhVJY

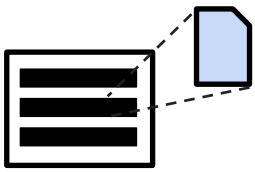
Let's look at streaming ideas first. Why do we stream? Streaming enables us to get real-time information in a dashboard or another means to see the state of your business.

New Cyber Command, Noam Dorogoyer says, "We have data coming from external vendors, and all this data is ingested through Pub/Sub, and Pub/Sub pushes it through to Dataflow, which can parse or enrich the data,"
If data comes in late, especially when it comes to cybersecurity, it's no longer valuable, especially during an emergency. So, from a data engineering standpoint, the way we constructed the pipeline is to minimize latency at every single step. If it's maybe a Dataflow job, we designed it so that as many elements as possible are happening in parallel so at no point is there a step that's waiting for a previous one."

The amount of data flowing through the command varies each day. On weekdays during peak times, it could be 5 or 6 terabytes, Dorogoyer said. On weekends, that can drop to 2 to 3 terabytes. As NYC Cyber Command increases visibility across agencies, it will deal with petabytes of data.

Security analysts can access the data in several ways, said Anthony Bocekci, Community Emergency Response Team specialist. They can run queries in BigQuery or use other tools that will provide visualizations of the data, such as Data Studio, a reporting solution.

Streaming is data processing for unbounded data sets



Bounded Data (Batch)

- Finite data set
- Usually complete
- Time of elements is usually disregarded
- Typically at rest
- Held in durable storage

Unbounded Data (Stream)

- Infinite data set
- Never complete
- Time of elements is usually significant
- Typically in motion
- Held in temporary storage

Streaming is data processing on unbounded data. Bounded data is data at rest. Stream processing is how you deal with unbounded data.

A streaming processing engine provides: low latency, speculative or partial results, the ability to flexibly reason about time, controls for correctness, and the power to perform complex analysis.

Stream analytics has many applications

Data integration
(10 sec - 10 min)

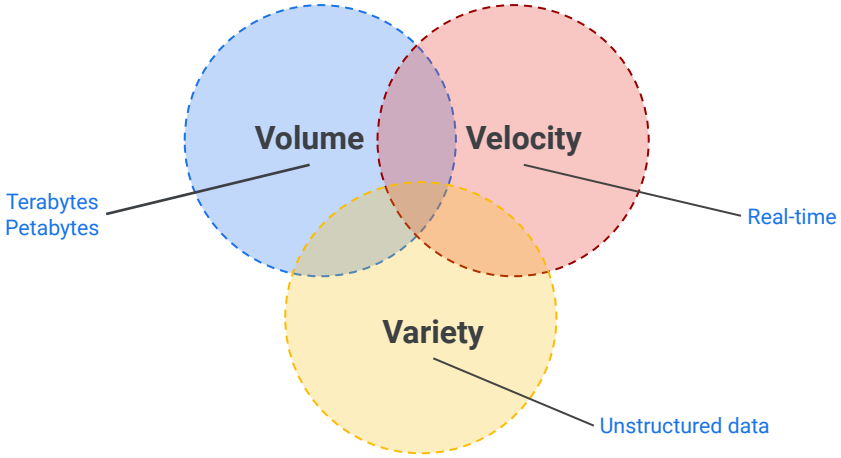
- Data warehouses becomes real-time
- Take load off source databases with change data capture (CDC)
- Microservices require databases and caches

Online decisions
(100 ms - 10 sec)

- Real-time recommendations
- Fraud detection
- Gaming events
- Finance back office apps

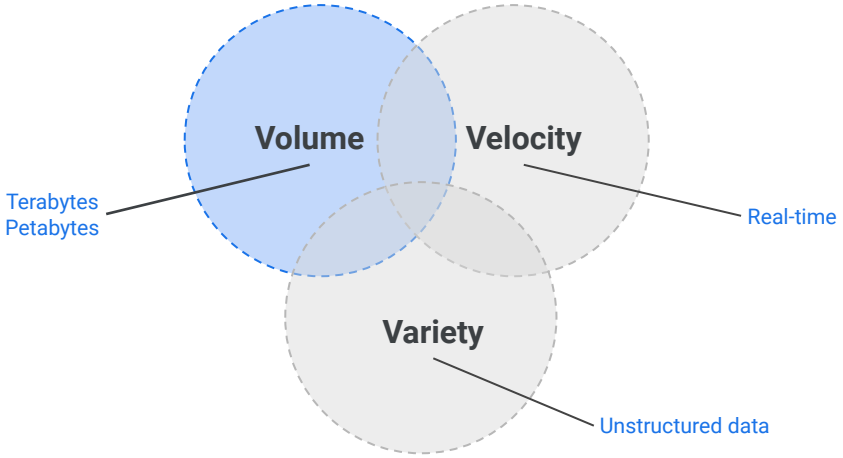
You can actually use streaming to get real time data warehouse, getting a dashboard of real-time information. For example, you could see in real-time the positive versus negative tweets about your company's product, use it to detect fraud, use for gaming events, or for finance back office apps, such as stock trading, anything dealing with markets, etc.

How to handle data volume, velocity, and variety?



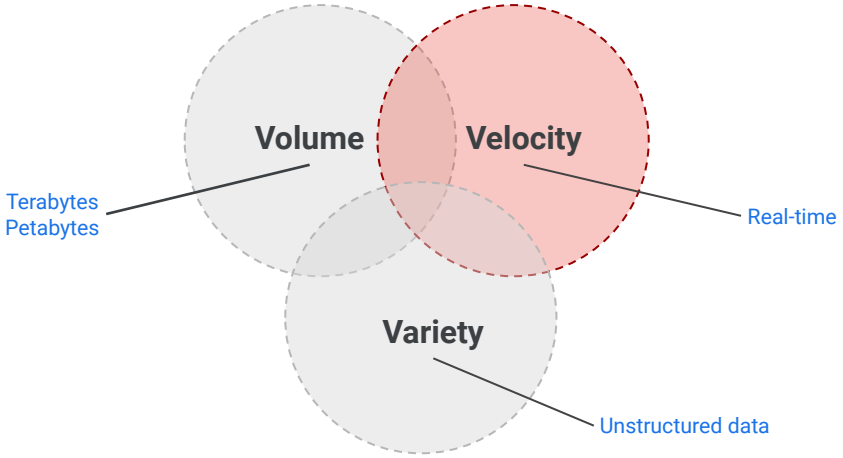
So, when we look at the challenges associated with streaming applications, we are talking about the three V's, Volume, Velocity, and Variety of data.

How to handle data volume, velocity, and variety?



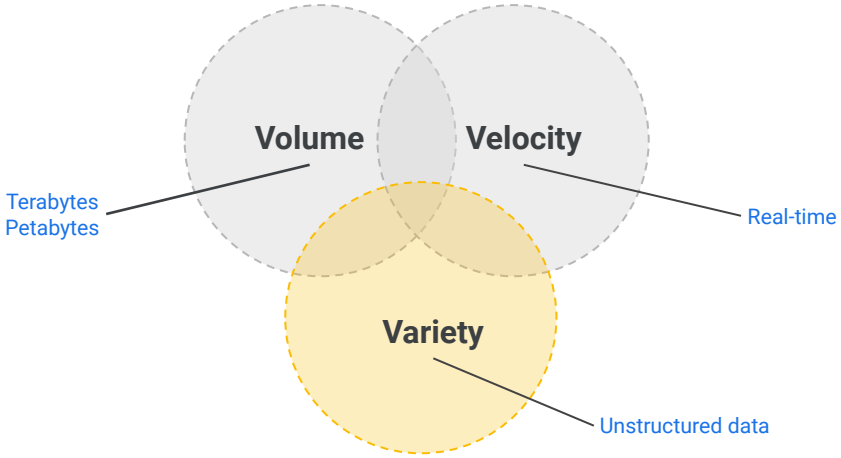
Volume is a challenge because the data never stops coming and quickly grows.

How to handle data volume, velocity, and variety?



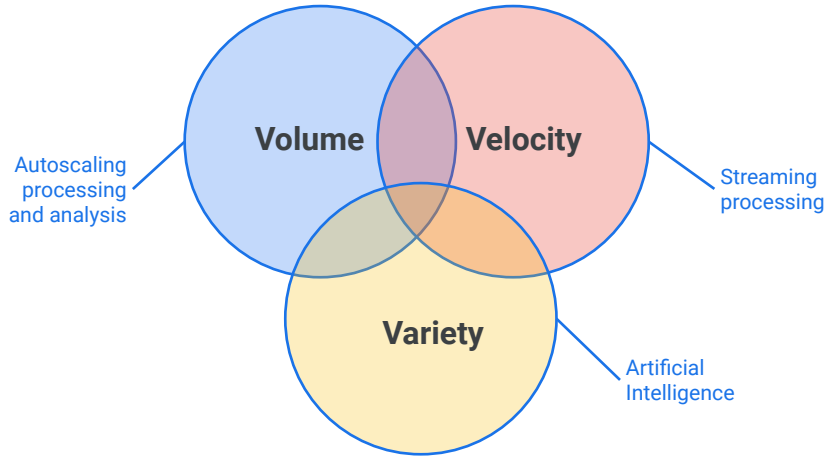
Velocity, depending what you are doing, trading stocks, tracking financial information, opening subway gates, you can have tens of thousands of records per second being transferred. Velocity can be very variable as well. For example, if you are a retailer designing your point of sales system nationwide, you are probably going to carry along at a reasonably steady volume all year until you get to Black Friday. Then, sales and data being transferred go through the roof. So, it is important to design systems that can handle that extra load.

How to handle data volume, velocity, and variety?



Variety of data is the third challenge. If we are just using structured data, data coming from a mobile app, that is easy enough to handle, but what if we have unstructured data, like voice data or images? These are streaming records that might have to use a null to deal with that type of unstructured data.

Autoscaling, Machine Learning, and Streaming



So, we are going to look at how streaming in the cloud can help us here. On the volume side, we will look at a tool to assist in autoscaling processing and analysis so that the system can handle the volume. On the velocity side, we will look at a tool that can handle the variability of the streaming process. And on the variety side, we will look at how artificial intelligence can help us with unstructured data.

Google Cloud products help you address key challenges in stream data processing and analytics

Pub/Sub

Changing and variable volumes of data

Dataflow

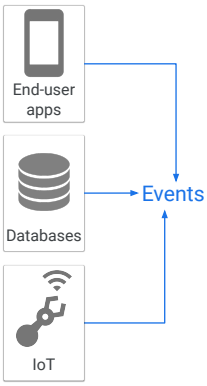
Process data without undue delays

BigQuery

Need ad-hoc analysis and immediate insights

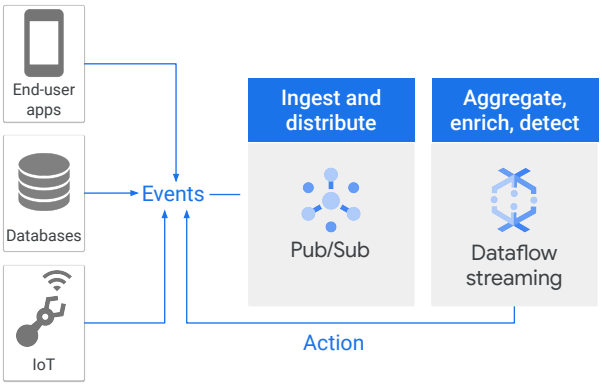
The three big things we are going to examine here are Pub/Sub, which will allow us to handle changing and variable volumes of data, Dataflow, which can assist in processing data without undue delays, and BigQuery, which we will use for our ad-hoc reporting, even on streaming data.

Stream analytics includes some common steps



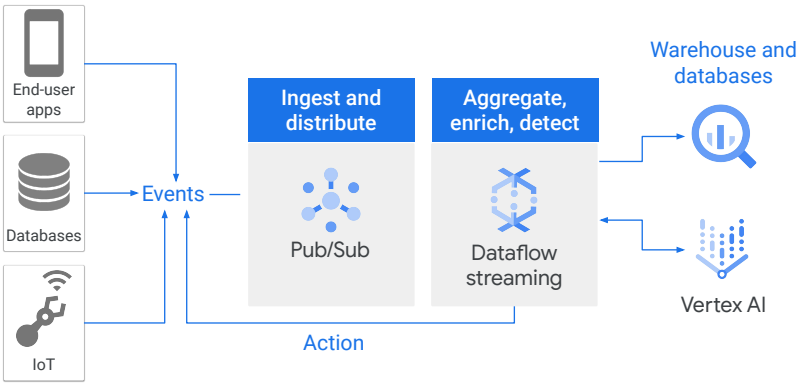
Let’s take a look at the steps that happen. First, some sort of data is coming in, possible from an app, a database, or an Internet of Things, or IoT. These are generating events.

Stream analytics includes some common steps



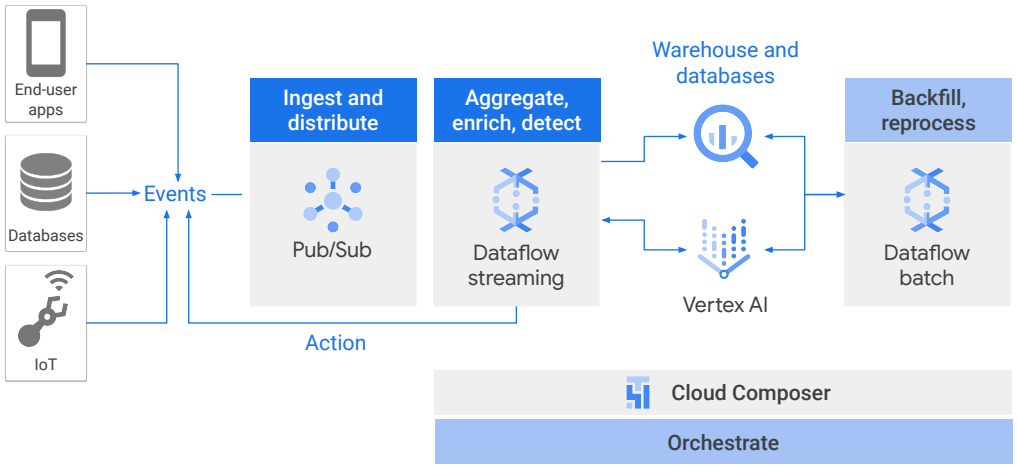
Then, an action takes place. We are going to ingest those and distribute those with Pub/Sub. This will ensure that the messages are reliable. This will give us buffering. Dataflow, then is what aggregates, enriches, and detects the data.

Stream analytics includes some common steps



Next, we will write into a warehouse of some kind, BigQuery or BigTable, or maybe run things through a Machine Learning (ML) model. For example, we might use this streaming data as it is coming in to train a model in Vertex AI.

Stream analytics includes some common steps



Then, finally, Dataflow or Dataproc could be used for batch processing, backfilling, etc.

So, this is a pretty common way to put things together in Google Cloud.