AWS Kinesis

STREAMING DATA

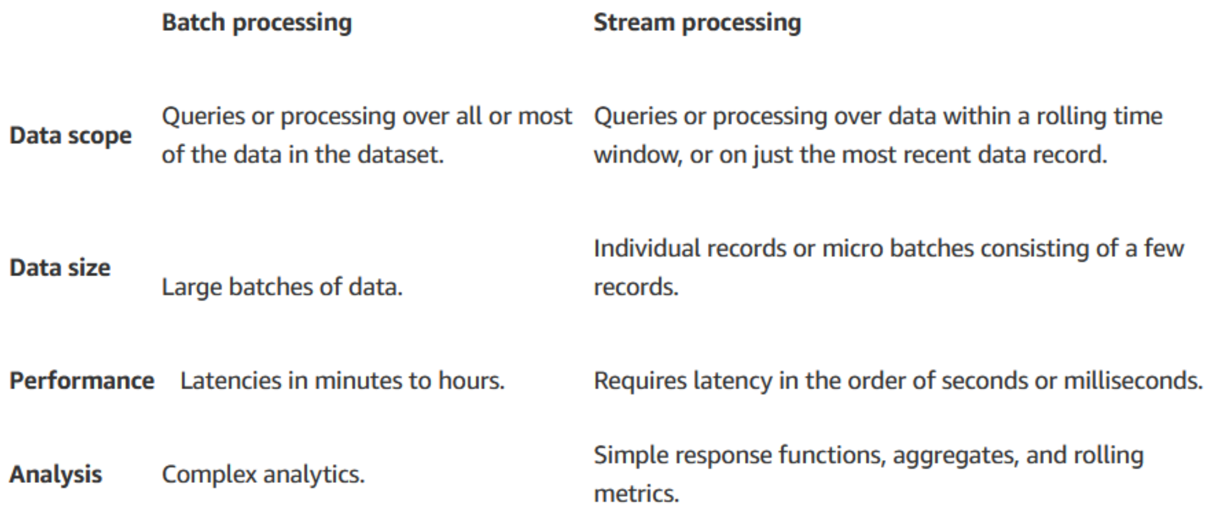
Data that is emitted at high volume in a continuous and incremental manner with the goal of low latency processing is called streaming data.

Used for: data sources that typically emit simultaneous messages and data

Features: > processes data with timestamps
> enables constant and continuous data collection

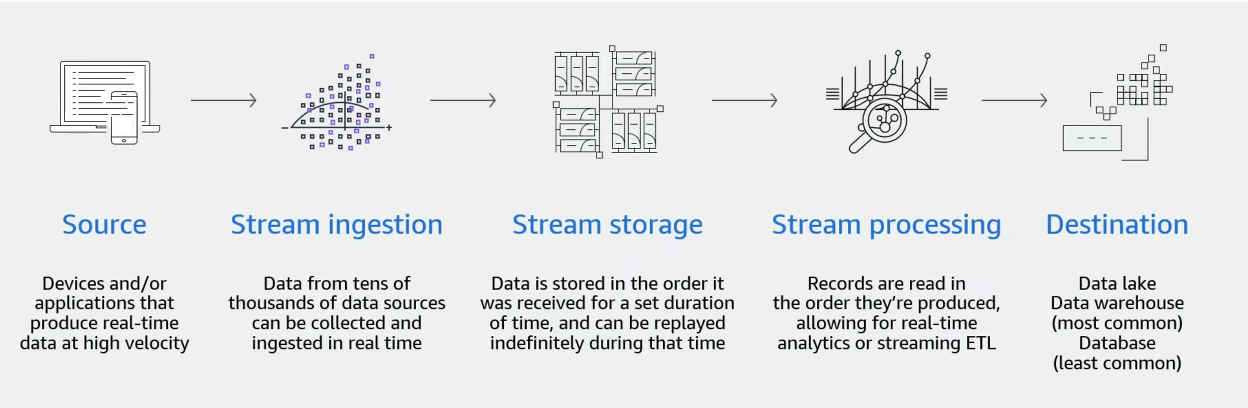
> supports multiple formats of data

Difference between batch processing and stream processing:



	Batch processing	Stream processing
Data scope	Queries or processing over all or most of the data in the dataset.	Queries or processing over data within a rolling time window, or on just the most recent data record.
Data size	Large batches of data.	Individual records or micro batches consisting of a few records.
Performance	Latencies in minutes to hours.	Requires latency in the order of seconds or milliseconds.
Analysis	Complex analytics.	Simple response functions, aggregates, and rolling metrics.

Components:



Source

Devices and/or applications that produce real-time data at high velocity

Stream ingestion

Data from tens of thousands of data sources can be collected and ingested in real time

Stream storage

Data is stored in the order it was received for a set duration of time, and can be replayed indefinitely during that time

Stream processing

Records are read in the order they're produced, allowing for real-time analytics or streaming ETL

Destination

Data lake
Data warehouse
(most common)
Database
(least common)

Stream Producers: Mobile devices, web applications etc are sources. Software components these apps and devices that collect data, and transmit records to stream processor are called stream producers. It usually contains a stream name, data value and sequence number, processor groups data records by stream name temporarily. It otherwise uses the sequence number to track the unique position of each record and process data chronologically.

Storage Layer: must support record ordering and strong consistency to enable fast processing of data streams

Processing Layer: responsible for consuming data, notifies the storage layer to delete data that is not needed.

Stream Consumers: Software components at the destination that process and analyse data streams buffered in the processor. Each consumer has analytical capabilities; each stream has multiple consumers. Consumers can send the changed data back to the processor to create new streams for other consumers.

Problem with Stream Data: availability, scalability and durability are an issue.

SOLVED BY KINESIS.

Kinesis makes it easy to load and analyze streaming data while also allowing you to build custom streaming data applications for specialized needs.

AWS KINESIS

It is a serverless, cloud native streaming data service that processes and stores data streams at any scale and provides analysis of real time data using AWS streaming services.

It is a massively scalable, low costing data service that enables continuous capturing of GBPS of data from multiple sources. It solves the high availability issue as the storage is already fine tuned and Kinesis has compute resources aligned for maximum throughput and low latency. It enables reaction to new information instantly.

Has 4 capabilities:

1. *AK Video Stream*

   Secure streaming of videos from devices connected to AWS for ML, analytics etc.
   Provisions and elastically scales all infra needed to ingest data from multiple devices automatically
   Durably encrypts, stores, indexes video data and allows access of data through easy to use APIs
   Uses Amazon S3 as the underlying data store

Has the HTTP Live Streaming capability (HLS)

2. *AK Data Streams*

Gathers together and processes huge streams of data records in real time
Use Kinesis Client Library for these ops and run as EC2 instances
Processed records can be sent to AWS dashboards and can be used to generate alerts, send data to other services and dynamically change advertisement and pricing strategy
Sensitive data encrypted within KDS so that it can be accessed only by Amazon Virtual Private Cloud(VPC)

3. *AK Data Firehose*

Fully managed service that delivers real time streaming data to services like S3, ES etc.
You do not get to write applications or manage resources
You configure data producers to send data to it and it automatically delivers data to the destination specified by me
Easily converts raw streaming data from data sources into formats like Parquet or ORC reqd by the data stores without having to build pipelines
Access to new data is sooner, and payment is only for the volume of data transmitted through the service and the data format conversion if needed

4. *AK Data Analytics*

A new ML feature to detect hot spots in streaming data.
Real time processing engine that lets you write and execute SQL queries to extract info from data
Supplies output to Data Streams

AMAZON KINESIS DATA STREAMS

Kinesis Data Streams provide accurate data feed intake because the data is not batched on the servers before intake
Works as the data is streaming in
Combines parallel processing with the value of real time data

Managed service aspect of Kinesis Data streams relieves the operational burden of creating and running a data intake pipeline
You can create streaming Map Reduce type applications
Their elasticity enables scaling up or down so that you never lose data records before they expire

Amazon Kinesis Client Library (KCL) delivers all records for a given partition key to the same record processor.
KCL makes it easier to build multiple applications that read from the same stream and enables fault tolerant consumption of data from streams.

All stream producers rely on Kinesis SDK to send data records into the stream.

Each data record consists of 2 parts: PARTITION KEY and a DATA BLOB(up to 1 MB)
Partition key: a Unicode string that determines the shard the record will be placed in.
When an app puts data into a stream it must specify a partition key
Data Blob: holds actual value

Data streams are divided into shards that handle the data load.

SHARD: uniquely identified sequence of data records in a stream
A stream has more than 1 more shard, each w fixed unit capacity
Each shard supports 5 transactions per second, max total data read rate of 2 MBPS
Multiple consumers per shard: multiple consumers can read the same data

Scaling of Amazon Kinesis Data Streams is manual and is altered by CAPACITY MODE: how capacity is managed and how you are charged for the usage of your data stream

On demand mode:
System adjusts capacity based on observed throughput peaks from the last 30 days
Each shard has default cap of 4 MBps
Pricing based on stream's hourly usage and data Input/Output Per GB
Provisioned Mode:
Manual selection of the number of shards and adjustment of them as needed using the API.
Each shard supports one megabyte per second or 1,000 records per second for ingestion.
Overall throughput per shard is two megabytes per second.

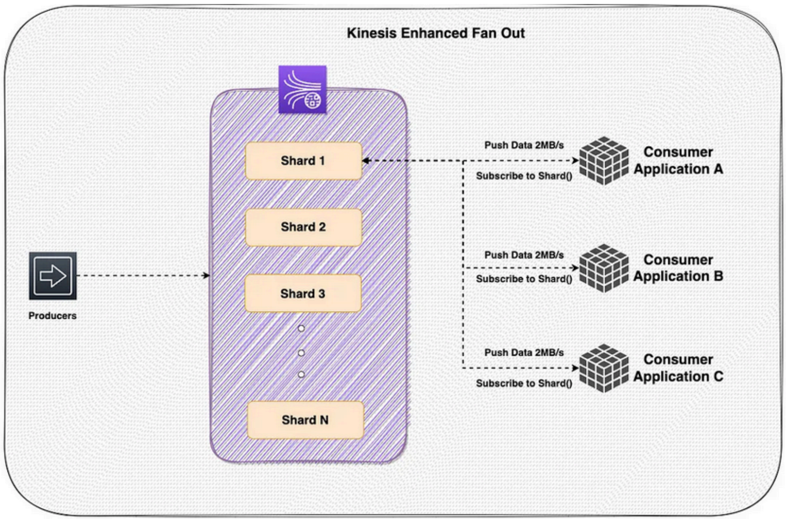
Pricing is based on the number of shards provisioned per hour.

No servers to be managed
On demand mode eliminates the need and you get automatic provisioning and scaling

RETENTION PERIOD
Streaming data is stored in the order received for a set duration of time, can be replayed indefinitely during that time.
Default retention period for stream data is 24 hours but that can be extended acc to a user's data replay and retrieval needs
"extended retention" is up to 7 days, "long term retention" is up to 1 year
Therefore, the benefits of using Amazon Kinesis Data Streams is:
>serverless scaling
>billion of events per day
>terabytes of data per day IN PROPER ORDER
>consistent performance
>concurrency at low latency
>provision of handsfree scaling
>you only pay what you use

EFO
To ensure high analytics performance for certain steam data, users can set up an enhanced fan out or EFO to avoid data read congestion

Each EFO consumer is given its own dedicated bandwidth within a shard to ensure consistent low latency



Transforms how data is consumed from Kinesis Data Streams:
>consumer receives 2 MBPS of data per shard
>reduced latency of 70ms
>effortless scaling

It is a consumer type
Std consumers are ideal for scenes w low consumers (1-3) where 200ms of latency can be tolerated, cost of these are included in kinesis
EFO: multiple consuming apps that require low latency
Has a default limit of 20 consumers

AMAZON KINESIS DATA FIREHOSE

Amazon Kinesis Data Firehose is an ETL service that reliably captures, transforms and delivers streaming data to data stores and analytics services
Eg data from Amazon s3 delivered to Amazon Redshift for real time analytics; no manual processing or management

https://medium.com/@reach2shristi.81/aws-kinesis-a-comparison-of-kinesis-data-streams-kinesis-firehose-and-kinesis-analytics-69c9f4847c2b

STREAMING ETL WITH APACHE FLINK AND AMAZON KINESIS DATA ANALYTICS

Amazon Kinesis Data Analytics enables you to run Flink applications in a fully managed env:
The service:
  >manages and provisions the reqd infra
  >scales the Flink application in response to changing traffic patterns
  >automatically recovers from infra and application failures

Robust ETL streaming pipelines and reduces the operational overhead of provisioning and operating infra

Architecture supports:
>private network connectivity VPC
>multiple sources and sinks
>data partitioning (of the ingestion into S3 based on info extracted from event payload)
>multiple elasticsearch and custom doc IDs- fan out from a single input stream to diff ES indexes and explicitly control the doc ID
>exactly once semantics- avoids duplicates when ingesting and delivering data

https://aws.amazon.com/blogs/big-data/streaming-etl-with-apache-flink-and-amazon-kinesis-data-analytics/