Machine Translated by Google

ĐẠI HỌC BÁCH KHOA

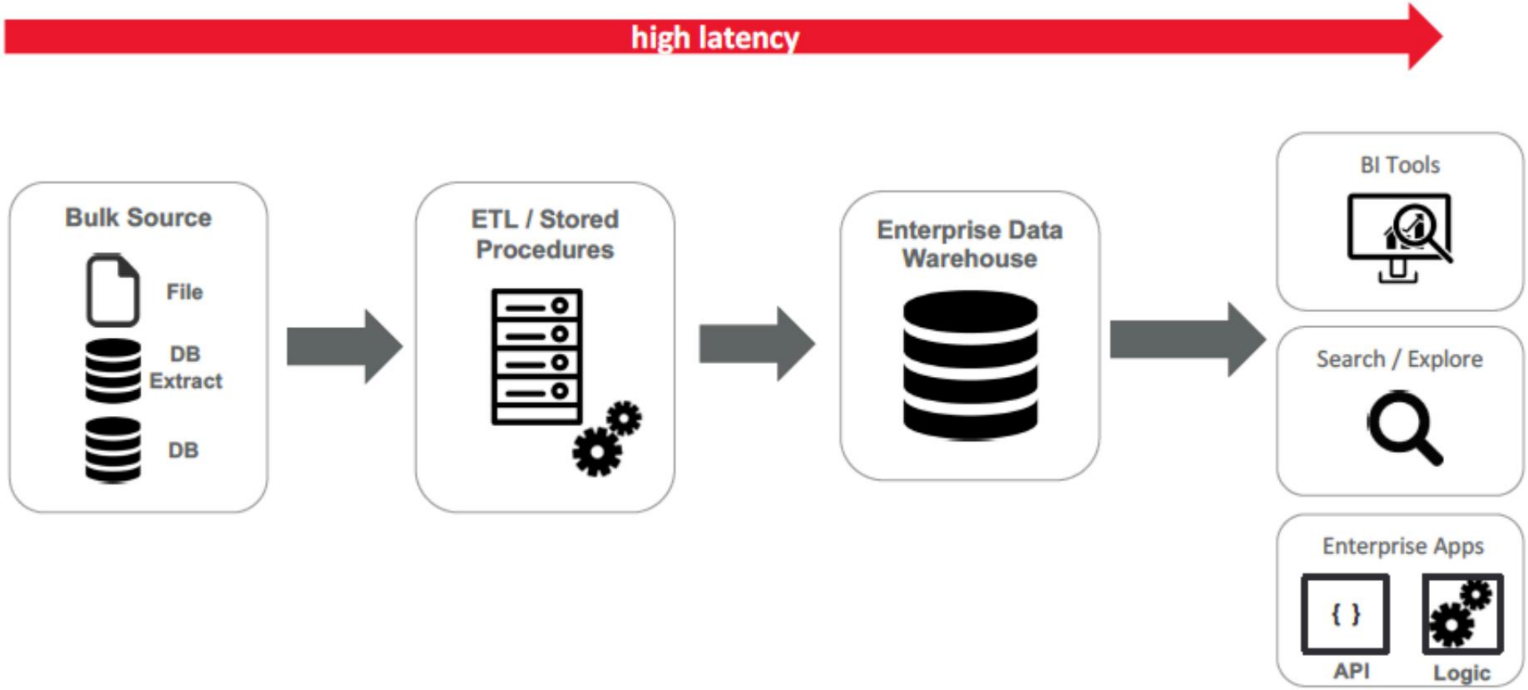
25 YEARS ANNIVERSARY SOICT

HA NOI UNIVERSITY OF SCIENCE AND TECHNOLOGY
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

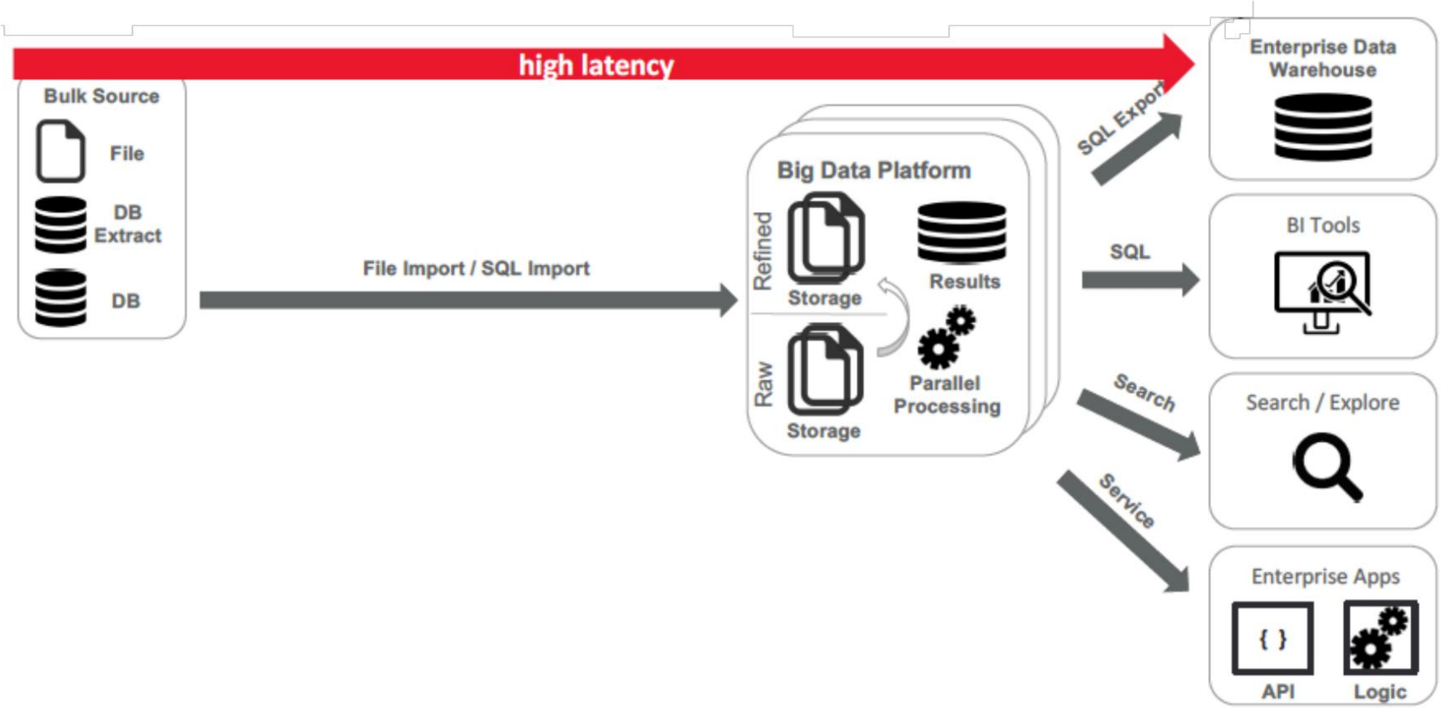
Machine Translated by Google

Chapter 8
Big data architecture

Traditional BI infrastructures



Hadoop solves Volume and Variety – not Velocity



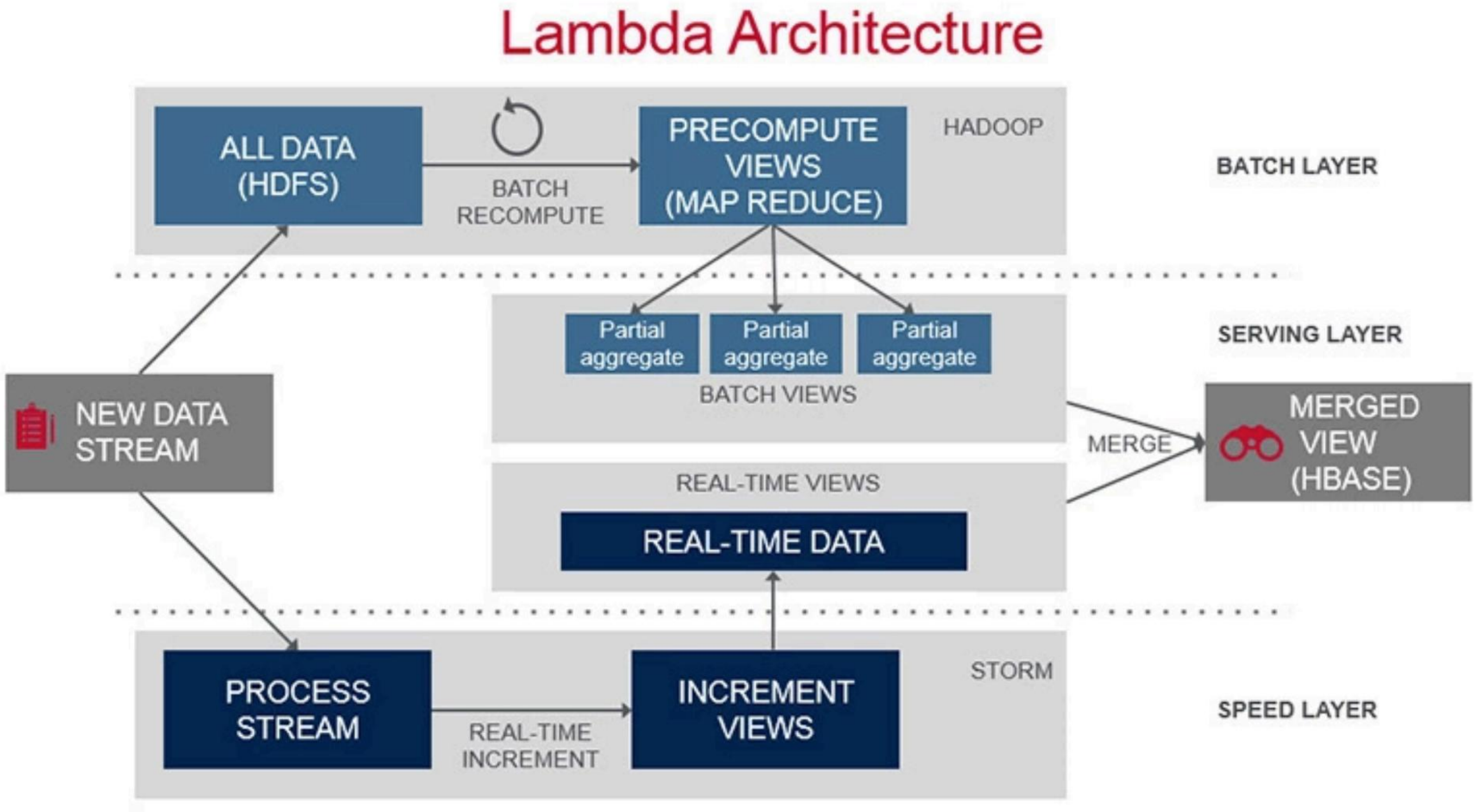
Machine Translated by Google

Lambda Architecture

- A data-processing architecture designed to handle massive quantities of data by taking advantage of both batch and stream processing methods.
- Spark is one of the few data processing frameworks that allows you to seamlessly integrate batch and stream processing
    - Of petabytes of data
    - In the same application

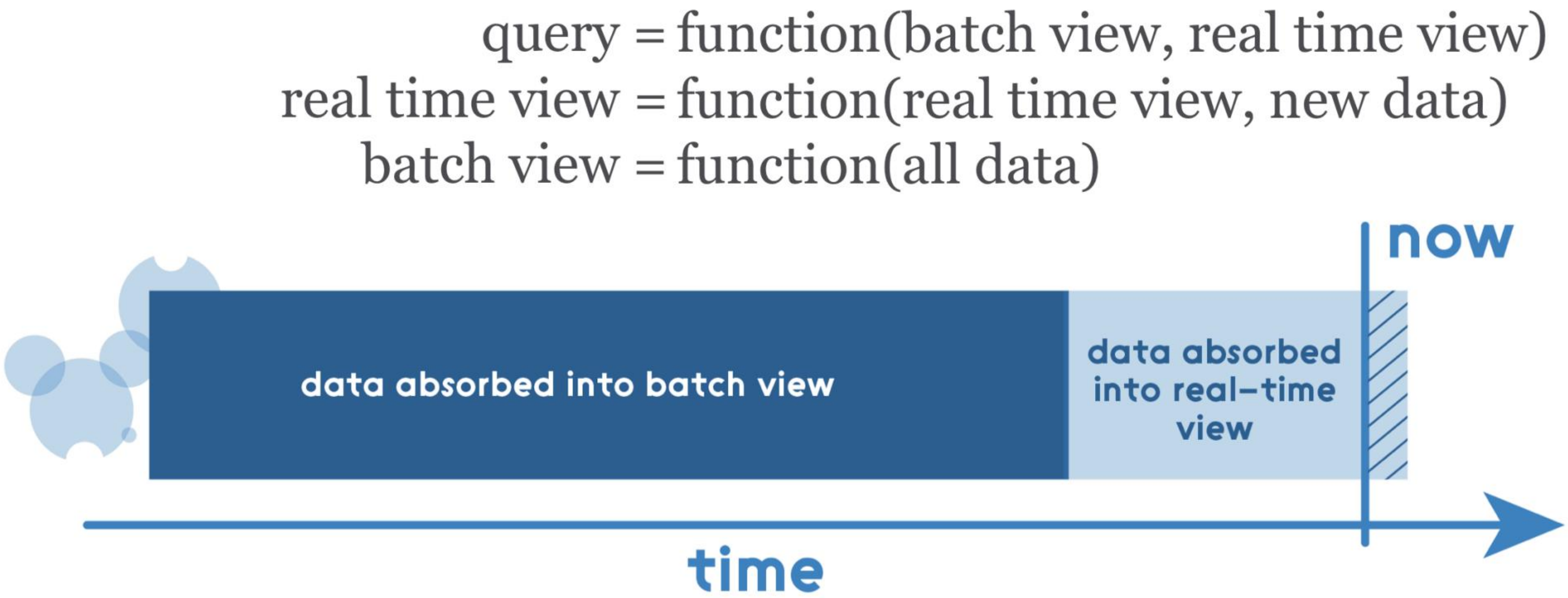
I need fast access
to historical data
on the fly for
predictive modeling
with real time data
from the stream

Lambda architecture



Machine Translated by Google

Relevance of data

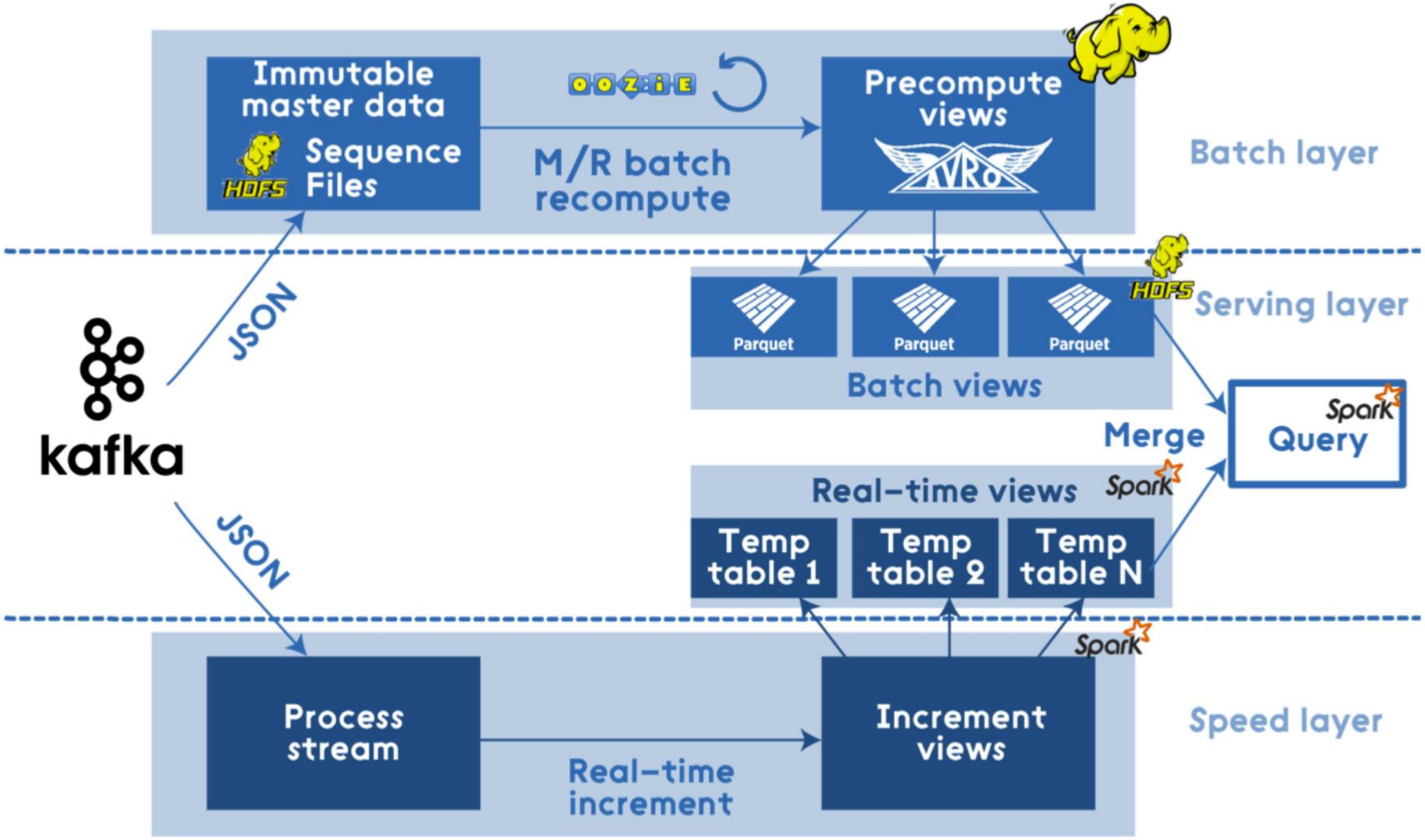


$$\text{query} = \text{function}(\text{batch view}, \text{real time view})$$
$$\text{real time view} = \text{function}(\text{real time view}, \text{new data})$$
$$\text{batch view} = \text{function}(\text{all data})$$

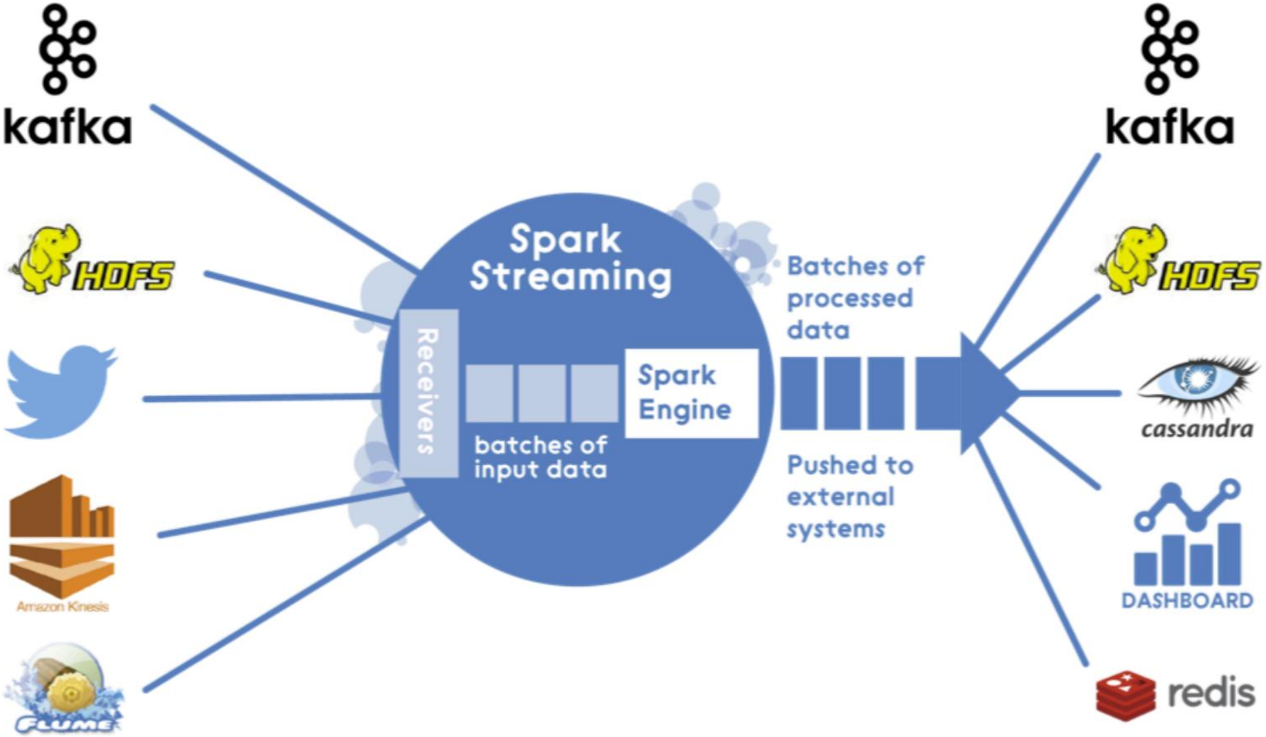
now
data absorbed into batch view
data absorbed into real–time view
time

Machine Translated by Google

Lambda architecture: one implementation



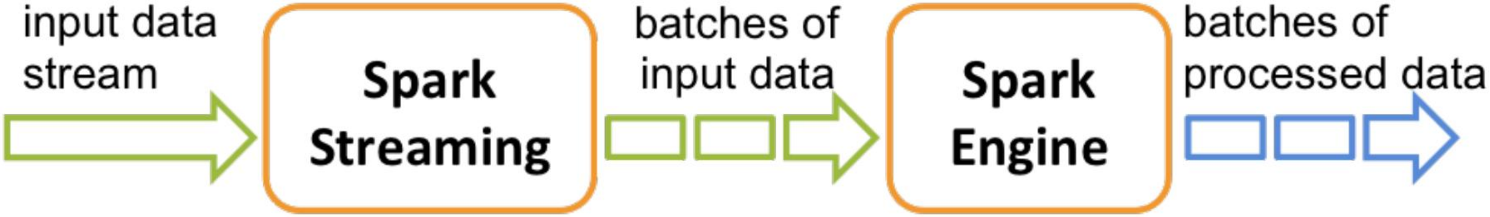
Spark streaming



Machine Translated by Google

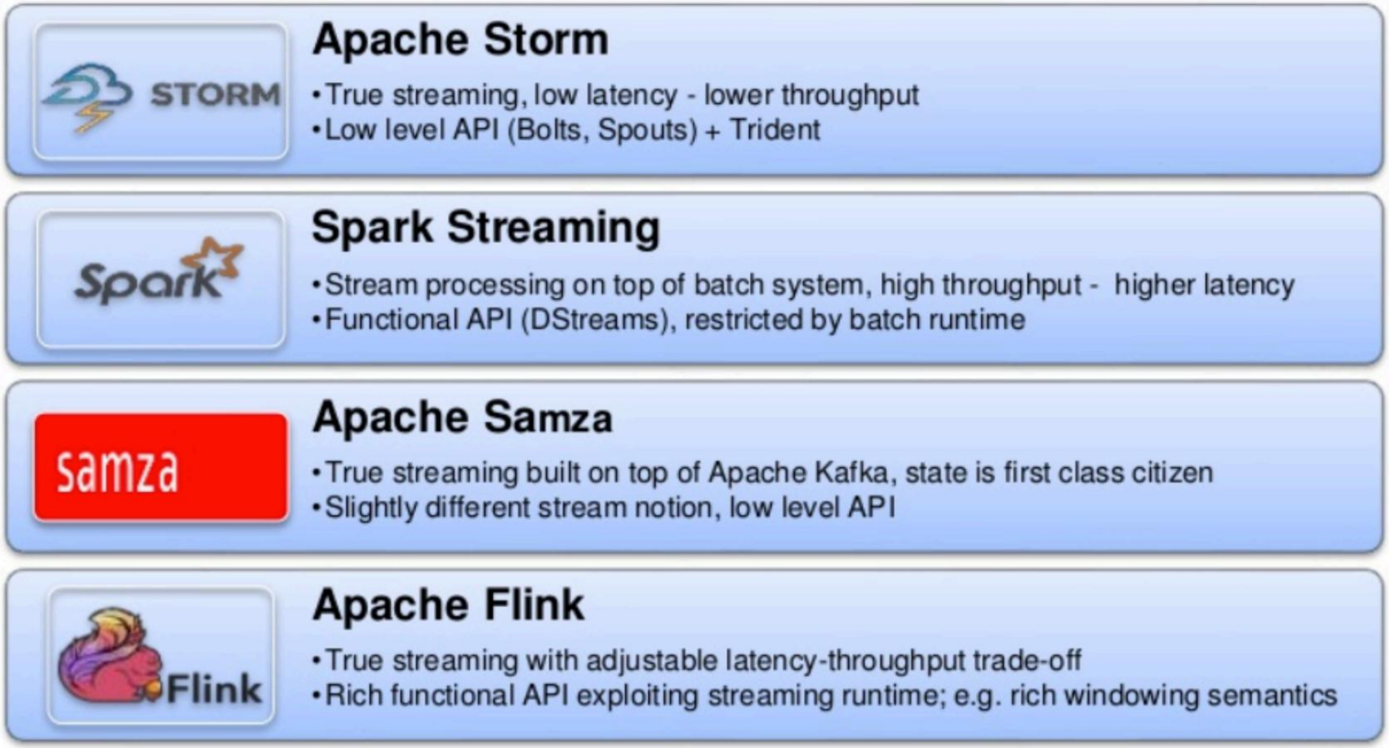
Spark streaming

- Scalable, fault-tolerance stream processing system
- a streaming computation as: a series of very small, deterministic batch jobs
  - Chop up the live stream into batches of X seconds
  - Spark treats each batch of data as RDDs and processes them using RDD operations
  - Finally, the processed results of the RDD operations are returned in batches



Machine Translated by Google

Streaming landscape



STORM
Spark
samza
Flink

Apache Storm

- True streaming, low latency - lower throughput
- Low level API (Bolts, Spouts) + Trident

Spark Streaming

- Stream processing on top of batch system, high throughput - higher latency
- Functional API (DStreams), restricted by batch runtime

Apache Samza

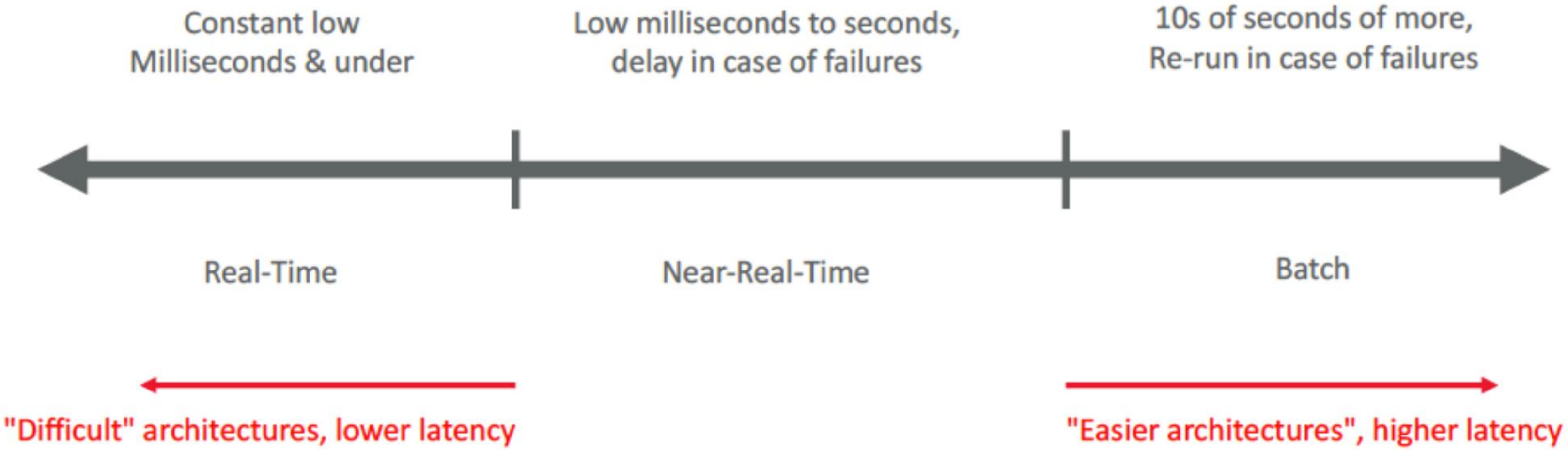
- True streaming built on top of Apache Kafka, state is first class citizen
- Slightly different stream notion, low level API

Apache Flink

- True streaming with adjustable latency-throughput trade-off
- Rich functional API exploiting streaming runtime; e.g. rich windowing semantics

Machine Translated by Google

Stream vs. Batch processing


Constant low
Milliseconds & under
Low milliseconds to seconds,
delay in case of failures
10s of seconds of more,
Re-run in case of failures
Real-Time
Near-Real-Time
Batch
"Difficult" architectures, lower latency
"Easier architectures", higher latency

Machine Translated by Google

References

- https://github.com/OryxProject/oryx
- https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/cosmos-db/lambda-architecture.md
- https://github.com/apssouza22/lambda-arch
- https://github.com/knoldus/Lambda-Arch-Spark

Machine Translated by Google

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Thank you

for your

attention!!!

soict.hust.edu.vn/ fb.com/groups/soict