STREAMING BIG DATA

- Popular Options for Ingesting LIVE Data Streams -

**SPARK STREAMING**

A framework for scalable, fault-tolerant processing of live data streams.

**THE SETUP:**

- Spark Engine
- Directed Acyclic Graph
- Tasks are assigned to worker nodes by the DAG scheduler.

DSTREAM

Discretized chunks of data are created at a fixed time interval.

Operations performed on a DSTREAM are continuously applied to new microbatches of data as they are received.

**INDIVIDUAL OPERATIONS**

Spark Streaming enables you to perform computations over a sliding "window" of data:

1. **Batch Interval:**
   - Sampling frequency - the size of each batch (here: 1 second)
2. **Slide Interval:**
   - Computational frequency - how often are calculations refreshed? (here: 3 seconds)
3. **Window Interval:**
   - Total history size for each calculation (here: 6 seconds)

**3 KEY CONCEPTS**

- Batch Interval
- Slide Interval
- Window Interval

**EXAMPLE ARCHITECTURE:**

Data produced by the server

Data ingested by Flume

Avro-formatted data published on port XXXX

Spark Streaming listens on port XXXX & performs windowing operations on incoming minibatches.

**Real World Example:**

Average click through rate for the last hour (window interval), updated every 5 minutes (slide interval) with data collected every 30 seconds (batch interval).

**POPULAR ALTERNATIVES TO SPARK:**

**STORM**

Networks of Spouts & Bolts => graph of computation

**FLINK**

The youngest technology in the Streaming arena.

- Supports:
  - Real-time processing
  - Machine learning

**Example Architecture:**

Data produced by the server

Web Logs

Data ingested by Flume

Channel

Source

Sink

Flume Agent

"Push Mechanism"

"Spout" (sources)

"Bolt" (processing)