Managing and Monitoring Streaming Queries

Janani Ravi
CO-FOUNDER, LOONYCORN

www.loonycorn.com

Overview

Monitoring streaming queries

The Spark Web UI and the Spark History server

Apache Spark and the Apache Beam unified streaming model

Monitoring Streaming Queries

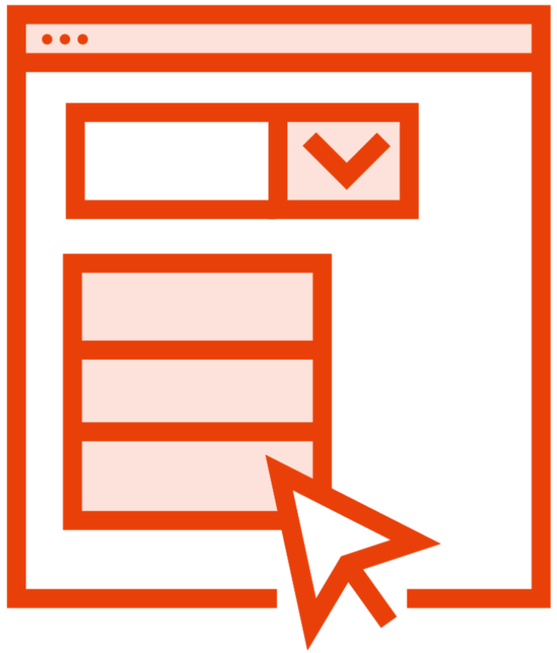
Monitoring Streaming Queries

Interactively

Programmatically
(Asynchronous APIs)

Programmatically
(Dropwizard library)

Interactive Monitoring

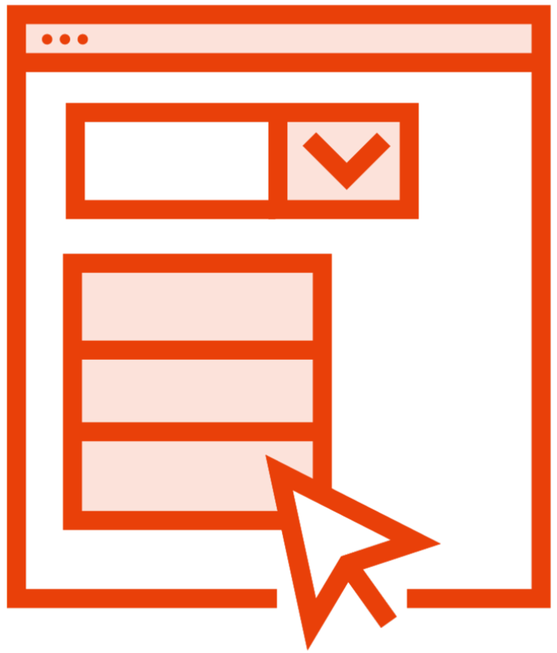


Directly get status and metrics of query

streamingQuery.lastProgress()

streamingQuery.status()

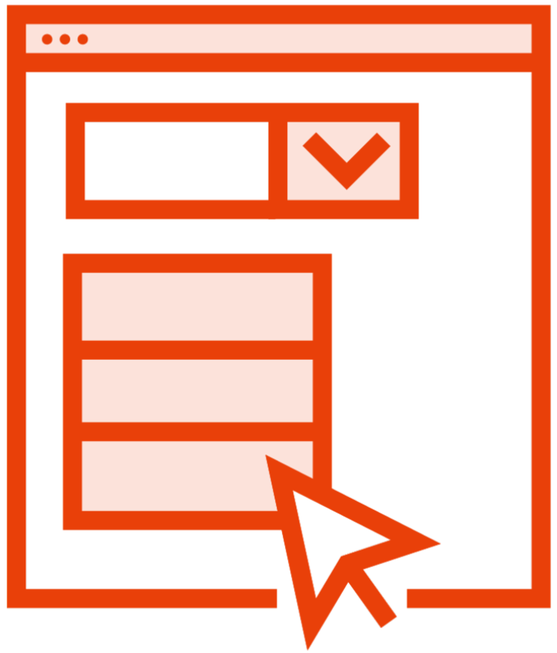
Asynchronous APIs



Attach StreamingQueryListener to the SparkSession object

Callbacks will be invoked when query is started/stopped or as progress occurs

Dropwizard Library



Dropwizard Metrics library is a third-party Java library

Used for profiling code in production environment

Instrument Spark code

Write metrics to various sinks

Demo

Monitoring interactive metrics in Spark

Demo

Exploring the Spark Web User Interface

Demo

Using the history server to monitor a completed application

Apache Beam Compatibility: Structured Streaming in Spark

Apache Beam

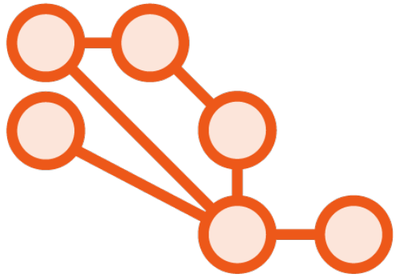
Open-source, unified model for defining both batch and streaming, data-parallel piplines.

Using Apache Beam

Write code for pipeline

Submit job for execution

Back-end assigns workers to execute



Pipeline parallelized and executed

Writing Code

Java

Python

Go

Scio - a Scala interface

Driver Program

Driver program utilizes Beam SDKs

Defines pipeline

Input, transforms, outputs

Execution options for pipeline

Driver program is executed on one of the Apache Beam back-ends

Available Runners

Apache Flink

Apache Spark

Google Cloud Dataflow

Apache Samza

Hazelcast Jet

Different back-end runners have very different capabilities and manner of stream processing

Runner Capabilities

What

Where

When

How

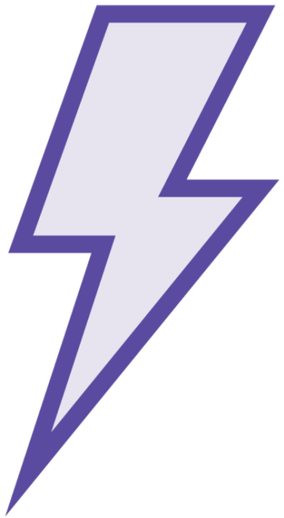
What

What is being computed?

Decides whether the result being computed

- Element-wise
- As an aggregate
- As a composite

Apache Spark 2



Most Beam operations only partially supported for streaming data

All Beam operations supported for batch data

Where

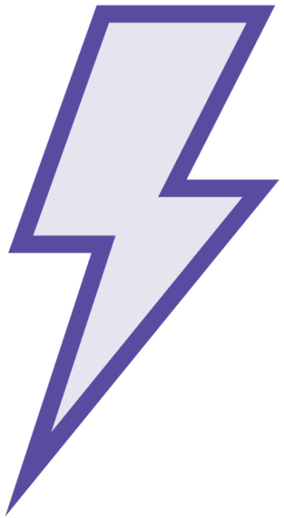
Where in event time is the result being computed?

Decides what type of windowing is being used

- Fixed
- Sliding
- Sessions

Most important for aggregation operations

Apache Spark 2



All Beam window types only partially supported for streaming data

All Beam window types supported for batch data

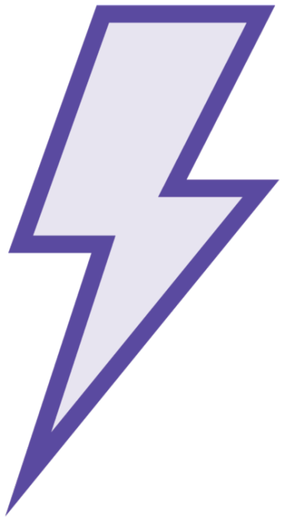
When

When in processing time is the result being computed?

Governs

- Type of Trigger
- Early and late firing

Apache Spark 2



All Beam triggers only partially supported for streaming data

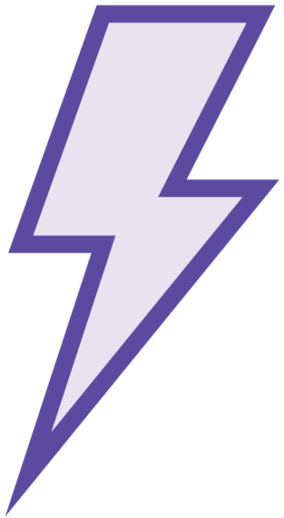
All Beam triggers supported for batch data

How

How do refinements relate?

- How should multiple outputs per window be reconciled?
- Accumulate
- Discard
- Accumulate and retract

Apache Spark 2



Only discard refinement supported

Accumulate and accumulate and retract not supported

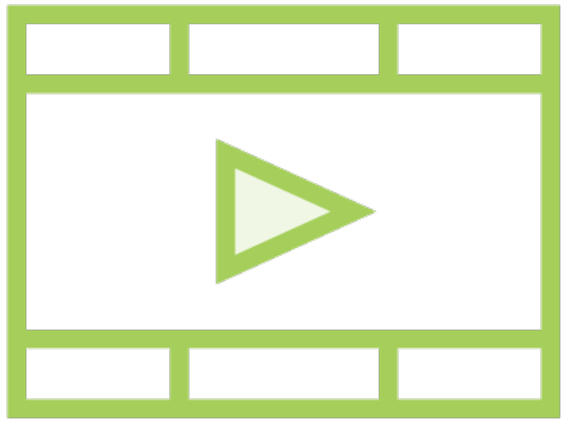
Summary

Monitoring streaming queries

The Spark Web UI and the Spark History server

Apache Spark and the Apache Beam unified streaming model

Related Courses



Modeling Streaming Data for Processing with Apache Beam

Exploring the Apache Beam SDK for Modeling Streaming Data for Processing