Spring XD

Introduction

Data Ingestion and Spring XD

© Copyright 2014 Pivotal. All rights reserved.

Introduction

- What is Data Ingestion, why do we need it?
- What is Spring XD?
- When to use Spring XD
- Lab

© Copyright 2014 Pivotal. All rights reserved.

What is Data Ingestion?

- Data ingestion is the process of:
  - reading, cleansing, transforming, and potentially enriching data from one or more sources;
  - and storing the resulting processed data into a consolidated repository

© Copyright 2014 Pivotal. All rights reserved.

Why Do We Need Data Ingestion?

- Modern applications generate <u>massive</u> data amounts.
- Modern data repositories are optimized for the task at hand
  - Not optimized for data analysis
- Data stored in diverse repositories difficult to analyze.

- Ingesting data into a consolidated repository permits practical data analysis.

© Copyright 2014 Pivotal. All rights reserved.

Data Ingestion Tools

- Various tools exist to simplify the data ingestion process
- Batch based:
  - Apache Sqoop
  - Spring XD
  - Spark
- Stream based:
  - Apache Flume
  - Spring XD
  - Spark

spring Pivotal™
© Copyright 2014 Pivotal. All rights reserved.

Apache Spark

Apache Spark is a fast and general engine for large-scale data processing.

- http://spark.apache.org/

- **XD Streams**
  - Ability to microbatch based on event count via Reactor and RxJava APIs.
  - Ability to create data pipelines to process one event at a time.
  - Flexibility to specify hosts to dictate the location of data computations.
- **XD Jobs**
  - Provides REST-API and lifecycle management for Spark jobs.
  - Extensible to integrate with other batch systems.

© Copyright 2014 Pivotal. All rights reserved.

Apache Flume

- *A service for collecting, aggregating, and moving large amounts of log data.*

  - http://f lume.apache.org

- Spring XD uses an interactive shell and DSL for stream creation, while Flume uses property (key/value pair) files.
- Administration and monitoring via the admin UI.
- Granular controls to manifest batch job and step execution to create complex data driven workflows.
- Flexibility through a deployment manifest to declar- atively configure data partitioning strategy to route
- data to a specific consumer instance in the cluster.

© Copyright 2014 Pivotal. All rights reserved.

Apache Sqoop

Tool for transferring bulk data between Apache Hadoop and structured datastores (such as relational databases.)

- http://sqoop.apache.org/

• Orchestrate Pig, Hive, HBase, MapReduce or other batch systems.
• Extend and customize batch workflow infrastructure.
• High level configuration DSL to create, deploy and destroy batch workflows.
• Operationalize custom data pipelines through REST.
• Unified functional programming support to build reactive-style data pipelines.

© Copyright 2014 Pivotal. All rights reserved.

Apache Oozie

Oozie is a workflow scheduler engine to manage Hadoop workloads such as MapReduce or Pig jobs.

- http://sqoop.apache.org/

- Building upon Spring Batch, a JSR standardization (JSR-352) of batch workload data processing, Spring XD inherits workflow scheduling and execution functionalities.
- Provides out of the box batch jobs that support plain files, JDBC, HDFS, FTP, MongoDB, Spark and Sqoop.
- Ability to scale jobs without having to bring down the runtime.
- Provides bi-directionality between real-time streaming and batch workflows to accommodate complex data processing use cases.
- Ability to create, and launch jobs from the admin UI. • Ability to view historical snapshots of job executions from the admin UI.

© Copyright 2014 Pivotal. All rights reserved.

Spring XD

"Spring XD is a unified, distributed, and extensible service for data ingestion, real time analytics, batch processing, and data export."

- http://docs.spring.io/spring-xd/docs/1.1.0.RELEASE/reference/html/

© Copyright 2014 Pivotal. All rights reserved.

Spring XD

Problems	Spring XD Benefits
Batch and Streaming are often handled by multiple platforms	**Unified Approach - Stream Processing and Batch Jobs** - Hadoop Batch workflow orchestration - Analytics - Machine Learning Scoring - Eye on the big picture
Fragmented Big Data Ecosystem	**Runtime provides critical Non-functional requirements** - Scalable, Distributed, Fault-Tolerant - Portable on prem DIY cluster, YARN, EC2, (WIP for PCF) - Easy to use, extend and integrate other technologies
Not all data is Hadoop bound	**Proven** - Built on proven EAI and Batch spring projects (7years)

© Copyright 2014 Pivotal. All rights reserved.

Spring XD

- Project home: http://projects.spring.io/spring-xd/
- "XD": "Extreme Data"
- *Lambda* Architecture
- Support for both stream processing and batch jobs
- Scalable, distributed, and fault-tolerant
- Deploy to
  - On premise cluster
  - YARN
  - EC2

© Copyright 2014 Pivotal. All rights reserved.

Lambda Architecture

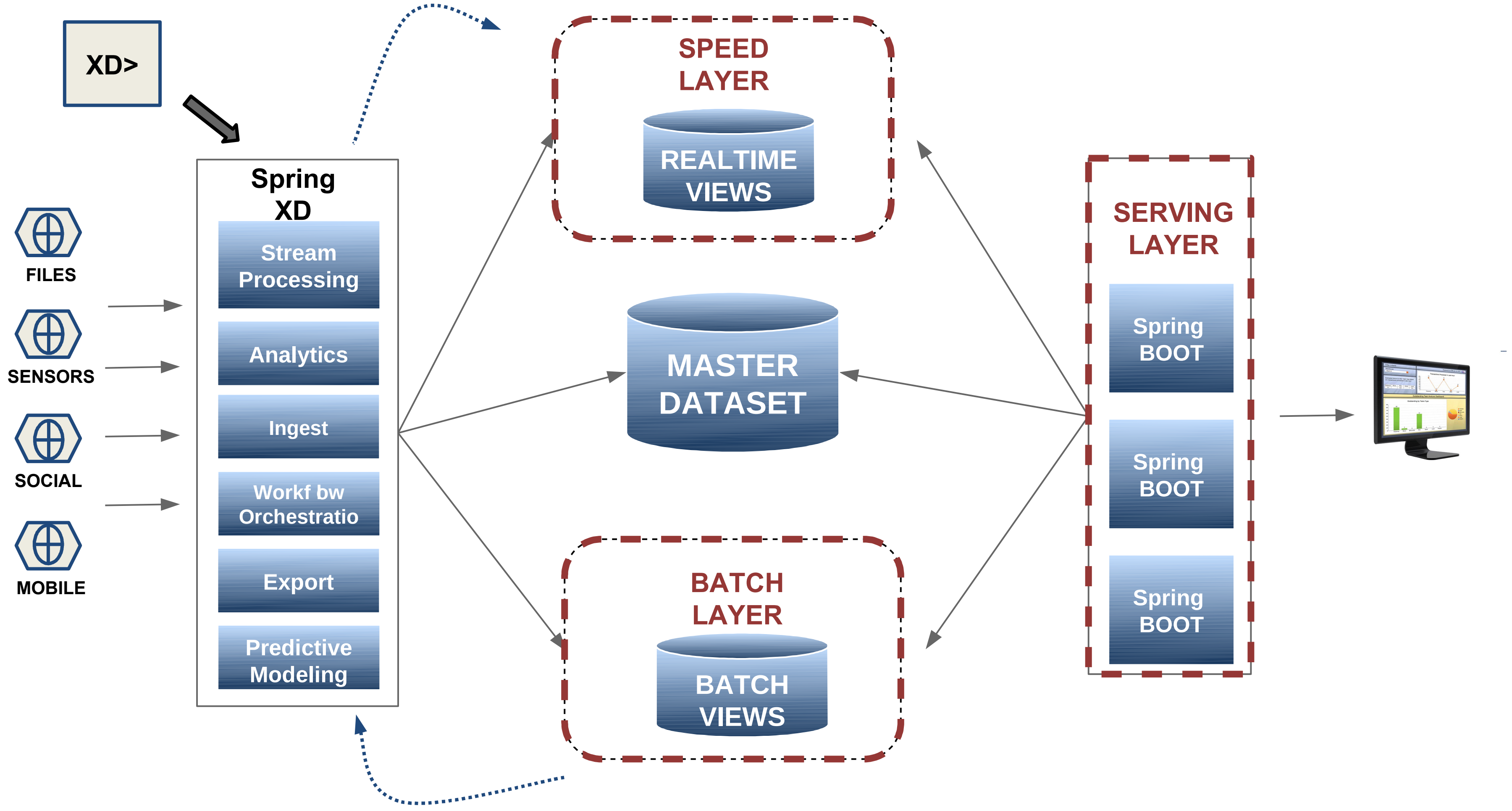
"Data processing architecture designed to handle massive quantities of data by taking advantage of both batch and stream processing methods."

- 3 layers
  - Batch: Pre-computes and stores results from all available data
  - Speed: Processes data streams in real time
  - Serving: Stores output from batch and speed layers, amalgamates results from both into views

- http://en.wikipedia.org/wiki/Lambda_architecture

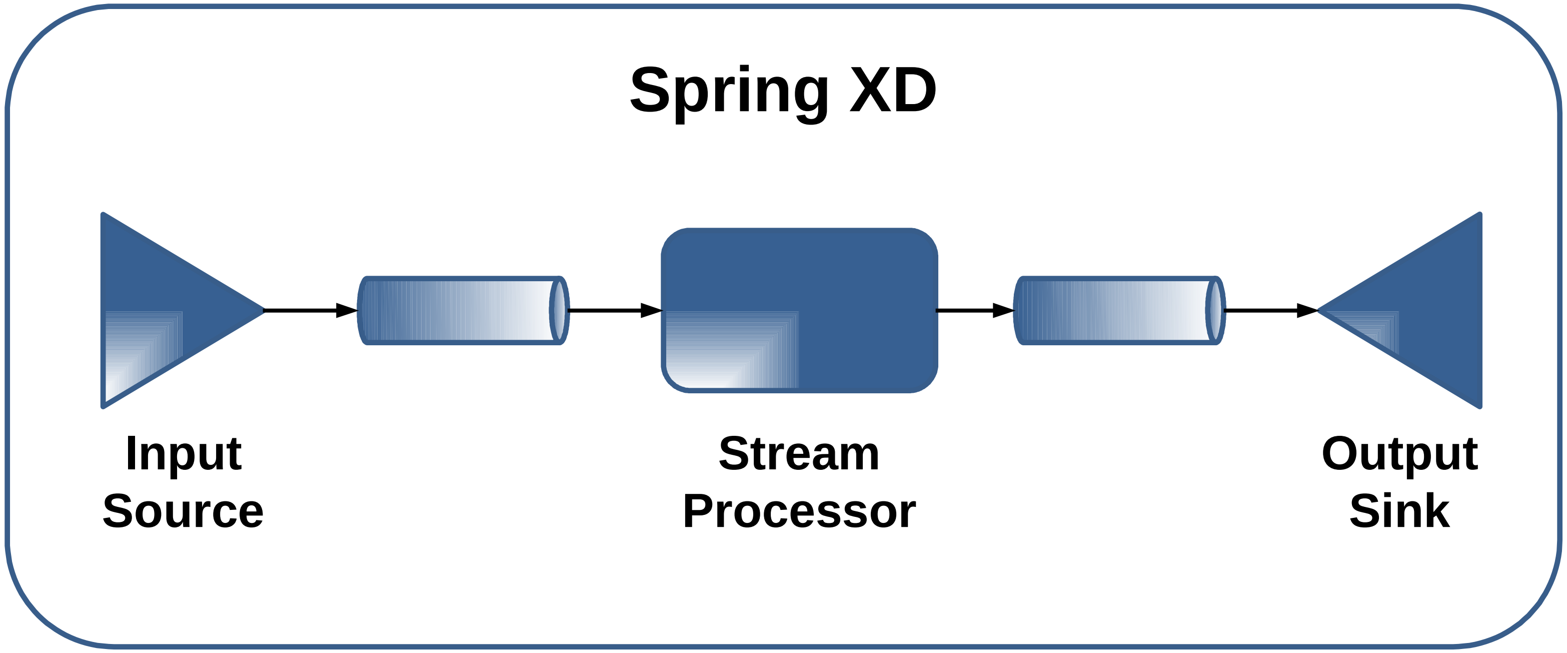
spring Pivotal™
© Copyright 2014 Pivotal. All rights reserved.

Lambda Architecture



spring Pivotal™
© Copyright 2014 Pivotal. All rights reserved.

Streams



© Copyright 2014 Pivotal. All rights reserved.

Input Sources

- HTTP
- Tail
- File
- Mail
- Twitter Search
- Twitter Stream
- Gemfire Source
- Gemfire Continuous Query
- Syslog

- TCP
- TCP Client
- Reactor IP
- RabbitMQ
- JMS
- Time
- MQTT
- Stdout Capture
- Kafka
- Jdbc

© Copyright 2014 Pivotal. All rights reserved.

Output Sinks

- Log
- File Sink
- Hadoop (HDFS)
- HDFS Dataset (Avro/ Parquet)
- JDBC
- TCP Sink
- Mongo
- Mail

- RabbitMQ
- GemFire Server
- Splunk Server
- MQTT Sink
- Dynamic Router
- Kafka
- Null
- Redis

© Copyright 2014 Pivotal. All rights reserved.

Stream Processing

- Filter
- Transform
- Script
- Splitter
- Aggregator
- HTTP Client
- JSON to Tuple
- Object to JSON

© Copyright 2014 Pivotal. All rights reserved.

Batch Jobs

- Support for batch processing includes:
  - Import CSV files to HDFS
  - Import CSV files to JDBC
  - Import JDBC to HDFS
  - Export HDFS to JDBC
  - Export HDFS to MongoDB
  - Spark Job

© Copyright 2014 Pivotal. All rights reserved.

When To Use Spring XD

- High throughput distributed data ingestion from various input sources into big data stores
- Real-time analytics at ingestion time
- Workflow management via batch jobs
- High throughput data export

© Copyright 2014 Pivotal. All rights reserved.

Use Cases

Telco service provider continuously **analyzing customer usage patterns** to up-sell products and services. Perform **analytics to predict churn** and take corrective actions to avoid customer acquisition costs.

Retail service provider delivering personalized mobile content including targeted advertisements and promotional offers based on geo-location.

Agriculture service provider creating batch workflows to govern pre, current, post and delivery phases of harvesting lifecycle.

e-Commerce service provider delivering customized, personalized and lucrative offers to online shoppers in real-time.

spring Pivotal™

© Copyright 2014 Pivotal. All rights reserved.

Getting Ready for the Labs

- Let's setup $LAB_HOME
  - $LAB_HOME == spring-xd-one-day directory
- Git
  - git clone https://github.com/cppwfs/spring-xd-one-day.git
- Thumb Drive
  - Copy the spring-xd-one-day directory to your local drive

spring Pivotal™
© Copyright 2014 Pivotal. All rights reserved.

LAB #1 Installation

1) Go to the instructions section of your $LABS_DIR.
2) Open either the lab/index.html or lab.pdf files
3) Follow the instructions on Chapter 1

- Git
  - git clone https://github.com/cppwfs/spring-xd-one-day.git

© Copyright 2014 Pivotal. All rights reserved.