Course Outline – Kafka – Confluent and NiFi

Kafka Introduction

- Architecture
- Overview of key concepts
- Overview of ZooKeeper
- Cluster, Nodes, Kafka Brokers
- Consumers, Producers, Logs, Partitions, Records, Keys
- Partitions for write throughput
- Partitions for Consumer parallelism (multi-threaded consumers)
- Replicas, Followers, Leaders
- How to scale writes
- Disaster recovery
- Performance profile of Kafka
- Consumer Groups, “High Water Mark”, what do consumers see
- Consumer load balancing and fail-over
- Working with Partitions for parallel processing and resiliency
- Brief Overview of Kafka Streams, Kafka Connectors

Lab Kafka Setup single node, single ZooKeeper

- Create a topic
- Produce and consume messages from the command line

Lab Set up Confluent Kafka multi-broker cluster

- Configure and set up three servers
- Setup Confluent Control Centre
- Create a topic with replication and partitions
- Produce and consume messages from the command line

Writing Kafka Producers Basics

- Introduction to Producer Java API and basic configuration

Lab Write Kafka Java Producer using Java

- Create topic from command line
- View topic layout of partitions topology from command line
- View log details
- Use ./kafka-replica-verification.sh to verify replication is correct

Writing Kafka Consumers Basics

- Introduction to Consumer Java API and basic configuration
- **Lab Write Java Consumer using Java an**
- View how far behind the consumer is from the command line
- Force failover and verify new leaders are chosen

Low-level Kafka Architecture

- Motivation Focus on high-throughput
- Embrace file system / OS caches and how this impacts OS setup and usage
- File structure on disk and how data is written
- Kafka Producer load balancing details

- Producer Record batching by size and time
- Producer async commit and commit (flush, close)
- Pull vs poll and backpressure
- Compressions via message batches (unified compression to server, disk and consumer)
- Consumer poll batching, long poll
- Consumer Trade-offs of requesting larger batches
- Consumer Liveness and fail over redux
- Managing consumer position (auto-commit, async commit and sync commit)
- Messaging At most once, At least once, Exactly once
- Performance trade-offs message delivery semantics
- Performance trade-offs of poll size
- Replication, Quorums, ISRs, committed records
- Failover and leadership election
- Log compaction by key
- Failure scenarios

Writing Advanced Kafka Producers

- Using batching (time/size)
- Using compression
- Async producers and sync producers
- Commit and async commit
- Default partitioning (round robin no key, partition on key if key)
- Controlling which partition records are written to (custom partitioning)
- Message routing to a particular partition (use cases for this)
- Advanced Producer configuration

Lab 1: Write Kafka Advanced Producer using Java

- Use message batching and compression

Lab 2: Use round-robin partition

Lab 3: Use a custom message routing scheme

Writing Advanced Kafka Consumers

- Adjusting poll read size
- Implementing at most once message semantics using Java API
- Implementing at least once message semantics using Java API
- Implementing as close as we can get to exactly once Java API
- Re-consume messages that are already consumed
- Using ConsumerRebalanceListener to start consuming from a certain offset (consumer.seek*)
- Assigning a consumer a specific partition (use cases for this)

Lab 1 Write Java Advanced Consumer

Lab 2 Adjusting poll read size

Lab 3 Implementing at most once message semantics using Java API

Lab 4 Implementing at least once message semantics using Java API

Lab 5 Implementing as close as we can get to exactly once Java API

Kafka Security

- SSL for Encrypting transport and Authentication
- Setting up keys
- Using SSL for authentication instead of username/password
- Setup keystore for transport encryption
- Setup truststore for authentication
- Producer to server encryption
- Consumer to server encryption

Kafka Schema Registry and REST Proxy

- AVRO File Format Introduction
- Kafka Schema Registry
- Kafka REST Proxy
- Ingesting data using Kafka REST Proxy

Lab : Setting up Schema Registry and REST Proxy

Lab : Ingesting and Validating the data using Schema Registry and REST Proxy

Kafka Connect

- Kafka Connect Introduction
- Components of Kafka Connect
- File Source and File Sink
- A Deeper Look at Connect

Lab : Setting up of Kafka Connect

Lab : Kafka Connect from RDBMS source

Lab : Kafka Connect using File Source
Lab : Kafka Connect HDFS Sink and source

Kafka Streaming and KSQL

- Components of Kafka Streaming
- Overview of Kafka Streams
- Kafka Streams Fundamentals
- Kafka Streams Application
- Working with low-level Streams
- Working with Kafka Streams DSL
- Lab : Demonstrating the real-time event partitions using Kafka
- Components of KSQL
- Using KSQL
- KSQL - Data Manipulation
- KSQL - Aggregations
- Lab : Exercises using KSQL

Introduction to NiFi and Data Flows

- Introduction to Enterprise Data Flow
- Introduction to Apache Nifi
- Apache Nifi Architecture
- NiFI Pre-requisites
- Install and Configure NiFi Single Node with Hands-on
- NiFi UI – UI Summary and History with Hands-on
- Introduction to NiFI FlowFIle
- Introduction to NiFi Processor with Hands-on
- Introduction to NiFi Connector with Hands-on
- NiFi Controller services and Reporting Tasks

NiFI Repositories, Templates, Process Groups and Registry

- NiFi Data Flows with Hands-on
- Performing ETL Data Flow using NiFi with Hands-on
- NiFI Repositories
- NiFI Templates
- Introduction to NiFi Process Group with Hands-on
- Introduction to NiFi Remote Process Group
- FlowFile Topology - Content and Attributes
- Remote process Group Transmission
- NiFI Flow Creation – Hands-on : PutFIle to FlowFIle
- NiFI Registry – Hands-on

NiFI Expression Language, attributes and cluster

- Function and Purpose of NiFi Expression Language with Hands-on
- Structure of a NiFi Expression Language
- Using NiFi Expression Language Editor with hands-on
- Performing If/Then/Else in NiFi Expression Language with Hands-on
- NiFi Attributes and Properties with Hands-on
- Create, Manage and Instantiate NiFi Templates with Hands-on
- Optimizing NiFi Data Flows
- Introduction to NiFi Data Provenance and Defining Data Provenance Events
- Event Search and APIs
- NiFi Cluster and State Management
- NiFi Cluster setup and Management using NiFi UI with Hands-on
- NiFi Monitoring with Hands-on

Advanced NiFi

- Big Data Ingestion using NiFi with Hands-on
- Performing Kafka Ingestion using NiFi with Hands-on
- NiFI Best Practices