

OEC128

Kafka Confluent

Durata: 4 gg

Descrizione

Lo scopo del corso è presentare la piattaforma open source di stream processing "Kafka" col fine di fornire le nozioni utili per:

- Il design architetturale di un ecosistema software in cui introdurre Kafka
- Installazione, configurazione e amministrazione di Kafka
- Integrazione con Kafka con ambienti di sviluppo Java

Al termine del corso gli studenti saranno in grado di:

- Disegnare un'architettura software comprensiva di Kafka e relativi cluster
- Configurare e ottimizzare i sistemi operativi e kafka stesso al fine di ottenere il massimo delle prestazioni
- Sviluppare Consumer, Producer e altro in ambiente Java per l'integrazione con Kafka.

A chi è rivolto?

Il corso è rivolto alle seguenti figure professionali:

- Architect
- Sistemisti
- Sviluppatori Java

Prerequisiti

Per partecipare con profitto a questo corso è necessario che gli allievi possiedano i seguenti prerequisiti:

- Designer Architeturali: conoscenze base sullo stream processing
- Sistemisti: conoscenza di ambienti Linux
- Sviluppatori: conoscenza del linguaggio Java

Contenuti

Unit 1: Introduzione all'ecosistema software Kafka

- Il mondo Apache
- Concetti Base della Messagistica
- Principali modelli di Messagistica
- Protocolli
- Apache Kafka
- Architetture Distribuite
 - Evoluzione delle Architetture SOA
 - Evoluzione delle Architetture SOA a microservizi
- Kafka e SOA
 - Kafka con SOA
 - Kafka con SOA a Microservizi
 - Considerazioni Generali
- Kafka & Architetture Software in Cloud

- Kafka & Hadoop
- Campi di applicazione
 - Metriche
 - Logs
 - IoT
 - Stream
 - Varie
- Conclusioni su Kafka e Architetture Distribuite

Unit 2: Architettura di Kafka, Installazione, Configurazione e Tuning

- Installazione Ambiente di sviluppo
 - Installazione di Zookeeper
 - Installazione di Kafka
 - Installazione di altri moduli Confluent
 - Configurazione Java
 - IDE di sviluppo Java
 - Kafka's Command Line Tools
- Publish/Subscribe Messaging e Streaming
- Apache Kafka APIs, configuration and metrics
- Kafka Message & Message metadata
- Message "key selection" (scelte e fattori)
- Message schema management
- Struttura Kafka:
 - Topics, Partitions e Segments
 - Log Replication & Log Compaction
 - Consumers e Producers
 - Consumer Groups
 - Brokers & Cluster
- Zookeeper
 - Ruolo di Zookeeper e dei metadati
 - Cluster Zookeeper
- Approfondimenti
 - Disk-Based Retention
 - Scalabilità
 - Alte performance
 - Security
 - CPU, RAM, network, storage considerations
- Business Continuity / DR
- Data retention
- Managing, configuring, e optimizing un cluster in funzione delle performance
 - Startup sequence; component dependencies
 - How many partitions? Tradeoffs
 - Scalability factors
 - Sources and tools for monitoring; Display of metrics
 - InSyncReplicas (ISR); Fully and Under replicated, and offline
 - Consumer lag, Under/Over Consumption

- Broker failure, detection, and recovery
- Batching and its impacts/consequences
- Determining and solving data imbalance across brokers
- Impacts of average and maximum message sizes
- Quotas
- Tuning

Unit 3: Sviluppare Su Kafka

- Accedere a Kafka a livello di codice
- Le APIs Kafka
- Sviluppare un Producer
 - Sviluppo di un Producer
 - Configurazione e ottimizzazione del Producer
- Sviluppare un Consumer
 - Sviluppo di un Consumer
 - Configurazione e ottimizzazione del Consumer
- Sviluppo Avanzato
 - Dimensioni e durata dei messaggi
 - Abilitazione di Exactly Once Semantics (EOS)
 - Uso degli offset
 - Specificare gli Offsets
 - Commettere manualmente gli Offsets
 - Consumer Liveness & Rebalancing
- Schema Management In Kafka
 - Data Serialization
 - Introduzione ad Avro e agli Gli Avro Records
 - Avro Schemas & Schema Evolution
 - Usare lo Schema Registry
- Tuning

Unit 4: Kafka Connect e Kafka Streams

- Definire Data Pipelines con Kafka Connect
 - Kafka Connect
 - Tipi di Connectors
 - Kafka Connect Implementation
 - Standalone & Distributed Modes
 - Configurazione dei Connectors
- Sviluppare con Kafka Streams
 - Kafka Streams API
 - Kafka Streams Concepts
 - Creare una Kafka Streams Application
 - Kafka Streams by Example
 - Managing Kafka Streams Processing
- Sviluppare con con Confluent KSQL
 - KSQL for Apache Kafka
 - Scrivere KSQL Queries
- Aspetti Avanzati:

- Risoluzione problemi del Client
- Debugging
- Ordine dei messaggi e garanzie di consegna
- Serialization/Deserialization
- Producer partition selection
- Consumer offset management
- Consumer Groups, partition assignments, partition rebalances
- Data retention strategies and implications
- Topic co-partitioning

Unit 5: LAB: Esercitazioni.

- Esercitazione in aula.