Kafka 생태계 들여다보기

이동진 | dongjin@apache.org

개요

- "Kafka가 뭐 하는 기술인지 모르겠어요"
  - RabbitMQ하고 같은 거 아닌가요?
- "Kafka 말고 이것저것 많던데 이게 뭐 하는 것들인지 모르겠어요"
  - KSQL? Kafka Connect? 얘네는 어디에 쓰는 거예요?
- "Kafka 이거 그래서 어디어디에 쓸 수 있는 거예요?"
  - 예제 좀...?

What is Kafka? (1)

- “At a very high level, Kafka is a fault tolerant, distributed publish-subscribe messaging system that is designed for speed and the ability to handle hundreds of thousands of messages.” - Bill bejeck
- "분산된 형태로 돌아가는 publish-subscribe 방식 메시지 교환 시스템"
- "수십만 개의 메시지를 처리할 수 있을 정도로 '빠른 속도'에 중점을 맞춰서 설계되었음."

What is Kafka? (2)

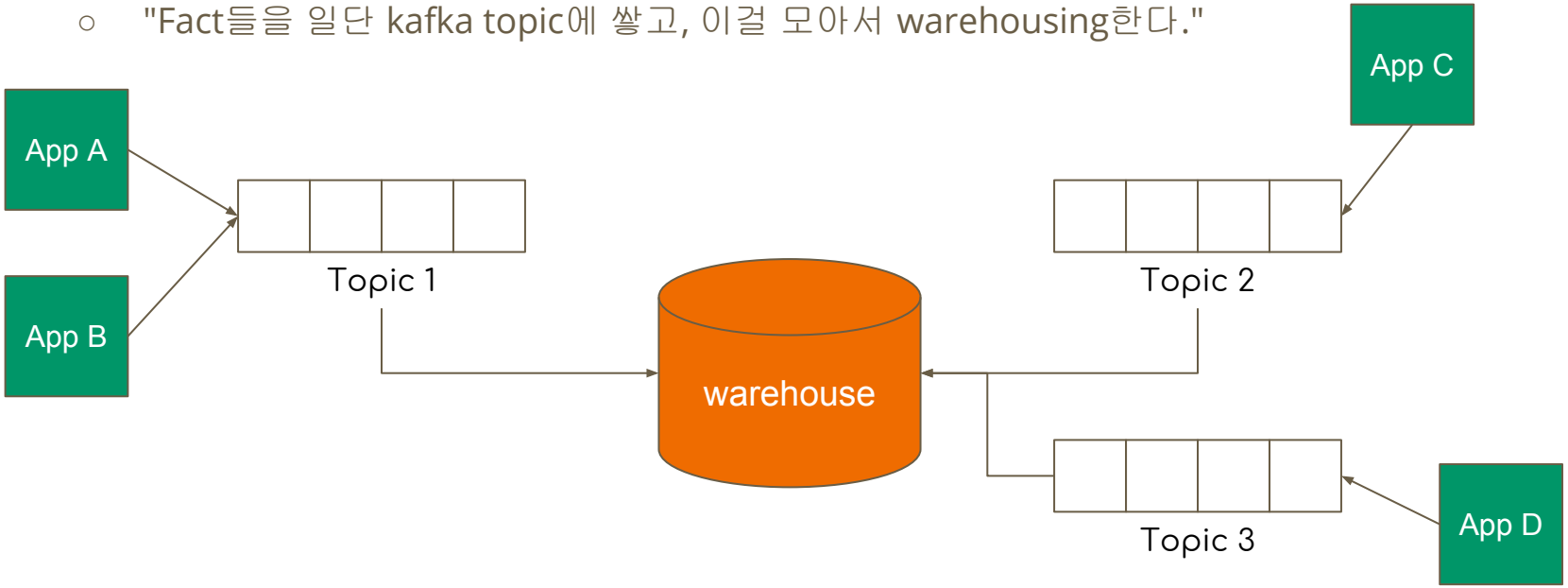
- "그럼 RabbitMQ 같은 Messaging Queue 아닌가요?"
  - "아닙니다. (단호)" - Jay Kreps
- Kafka vs. Messaging System
  - Message를 받았다고 해서 삭제되지 않음.
  - Memory 크기 넘어가도 멀쩡하게 돌아감.
  - Replication 잘 됨.
  - **So: Data 저장해도 됨. (결정적인 차이)**
- cf) 언제 RabbitMQ를 써야 하고 언제 Kafka를 써야 하는가? [#]

What is Kafka? (3)

- 한 줄 요약: "분산된 형태로 동작하는 changelog" [#]
  - Key 값에 할당된 Value 값들에 대한 변경 내역을 (분산된 형태로) 저장한다.
  - 저장된 Key-Value pair들을 (분산된 형태로) 분배한다.
- 설계 원칙
  - 동일한 key값을 가진 record들은 반드시 동일한 partition에 저장된다.
  - 임의의 partition 안에서 동일한 key를 갖는 record의 순서가 뒤집어지는 일은 없다.
- Log compaction
  - "더이상 의미 없는 변경 내역을 삭제한다."

Kafka 활용하기 (1)

- Data Warehousing
  - "Fact들을 일단 kafka topic에 쌓고, 이걸 모아서 warehousing한다."

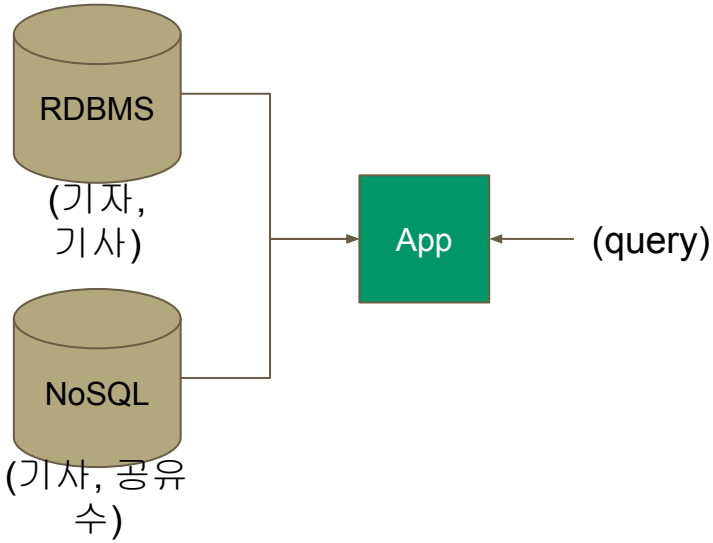


Kafka 활용하기 (2)

- Real-time system
  - Topic에 record가 들어오는 대로 처리해서 결과를 다른 topic으로 내보낸다.
    - 사용자에게 notification 보내기 (예제)
  - 응용: 실시간 machine learning service
    - Kafka topic에 들어온 record를 사용해서 tensorflow로 예측 수행 (예제)

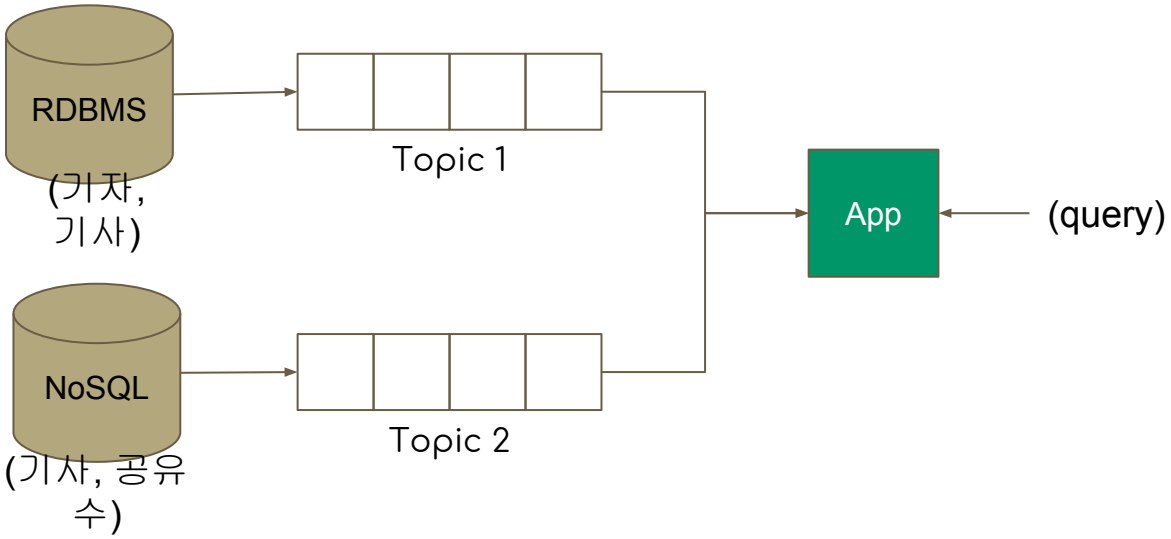
Kafka 활용하기 (3)

- Materialized View
  - "관계형 Database에서 query의 결과물을 table처럼 사용할 수 있도록 해주는 것."
  - 서로 다른 Storage에 대해서도 할 수 있음.
  - Newyork Times가 해서 유명해졌음.
- 예) "기자별 기사가 공유된 횟수를 보여 주기"
  - 요청 들어올 때마다 query?
  - cache?



Kafka 활용하기 (4)

- 해법: storage 변경 사항을 kafka topic에 저장하고, 이걸 cache해서 service.



Kafka Ecosystem (1)

- Kafka Connect
  - Kafka Topic에 들어온 것을 Cold Storage에 밀어넣거나 (changelog → state)
  - Cold Storage의 변경점을 Kafka Topic으로 밀어넣거나 (state → changelog)
- 사용법
  - connect cluster를 설치한다.
  - 사용할 plugin jar 파일을 connect cluster에 업로드하고 job을 실행시킨다.
    - Pre-built plugin
    - Custom plugin
  - "어떤 값들을 어디로 내보낼 것인지에만 집중하면 된다."
    - "어떻게" 말고.

Kafka Ecosystem (2)

- Schema Registry
  - Record의 value에 저장되는 값의 schema를 정의하고, 관리할 수 있는 시스템.
  - Subproject가 아님. (Confluent 프로젝트)

```
import io.confluent.kafka.serializers.KafkaAvroSerializer;
import io.confluent.kafka.serializers.AbstractKafkaAvroSerDeConfig;

...
props.put(ProducerConfig.VALUE_SERIALIZER_CLASS_CONFIG, KafkaAvroSerializer.class);
props.put(AbstractKafkaAvroSerDeConfig.SCHEMA_REGISTRY_URL_CONFIG, schemaRegistryUrl);
...
KafkaProducer<String, Payment> producer = new KafkaProducer<String, Payment>(props));
final Payment payment = new Payment(orderId, 1000.00d);
final ProducerRecord<String, Payment> record = new ProducerRecord<String, Payment>(TOPIC, payment.getId().toString(),
payment);
producer.send(record);
...
```

Kafka Ecosystem (3)

- Kafka Streams
  - Kafka topic을 읽어와서 처리할 수 있는 시스템을 개발하기 위한 라이브러리.
- KSQL
  - SQL 형태의 언어를 사용해서 Kafka topic을 처리하는 logic을 정의할 수 있도록 해 주는 툴.
  - Kafka Streams를 기반으로 개발.
  - 역시 Confluent 프로젝트.

Kafka Ecosystem (4)

- Kafka Log Appender
  - Log4j log를 kafka topic으로 보내 주는 appender. [#]
  - Logback용도 있음.
  - ~~솔직히 submodule 중에서 제일 안 쓰는 거 같음~~

정리 & 요약 (1)

- Kafka
  - 분산된 changelog (Distributed changelog).
- 활용
  - Data Warehousing
  - Real-time system
  - Materialized View

정리 & 요약 (2)

- Kafka Connect
  - RDBMS 등 다양한 Storage와 Kafka topic 간의 연결 제공. (state ↔ changelog)
- Kafka Streams
  - Kafka Topic을 활용하는 Application을 만들기 위한 library. (cf. Spark Structured Streaming)
- Kafka Log Appender
  - Log4j log → Kafka Topic
- 기타: KSQL, Schema Registry, ...

감사합니다!

- 슬라이드
  - https://speakerdeck.com/dongjin/kafka-ecosystem-explained
- Kafka 한국 사용자 모임
  - https://www.facebook.com/groups/kafkakorea/