MISA
TIN CẬY - TIỆN ÍCH - TẬN TÌNH

KAFKA Seminar

Trình bày: Lê Việt Hoàng

KHÁCH THÍCH - KHÁCH YÊU - KHÁCH CHIA SẺ

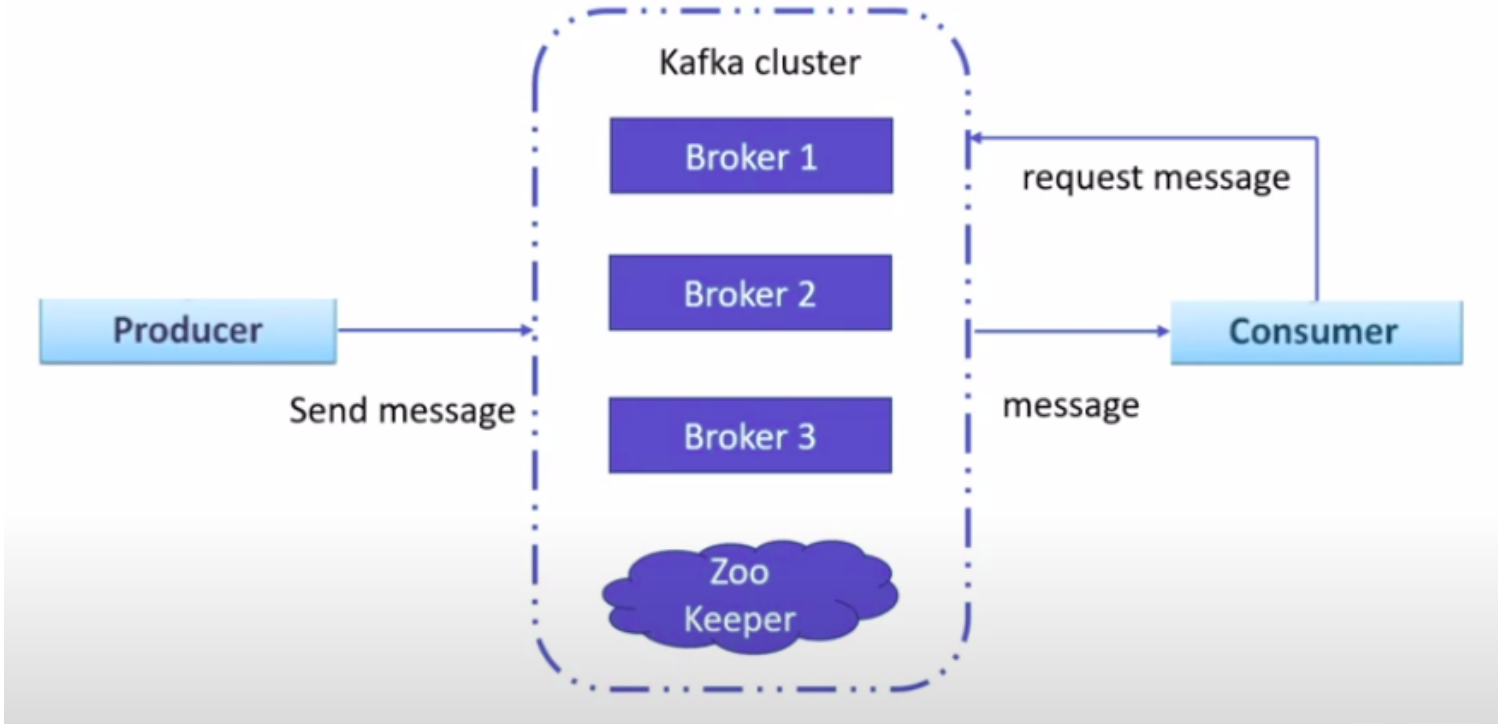
Nội dung

1. Kafka là gì?
2. Các khái niệm cần hiểu rõ để sử dụng được Kafka
3. Luồng chạy của hệ thống Kafka
4. Ứng dụng vào dự án Mintax

1. Kafka là gì ?

- Kafka là một hệ thống xử lý dữ liệu dựa trên luồng (stream processing) và cũng là một hệ thống hàng đợi (message queue).

- Nó được phát triển bởi Apache Software Foundation và được thiết kế để xử lý, lưu trữ và truyền dữ liệu theo thời gian thực.



1. Kafka là gì ?

***Một số đặc điểm quan trọng của Kafka**:*

- **Khả năng chịu tải cao**: Kafka có thể xử lý hàng triệu tin nhắn và đảm bảo độ tin cậy cao trong việc xử lý dữ liệ

- **Bảo đảm độ tin cậy**: Kafka được thiết kế để đảm bảo rằng dữ liệu không bị mất đi và có thể được xử lý lại trong trường hợp có sự cố xảy ra.

- **Khả năng mở rộng dễ dàng**: Kafka có thể mở rộng theo qui mô và cho phép thêm các producer và consumer một cách linh hoạt.

- **Xử lý dữ liệu theo thời gian thực**: Kafka cho phép xử lý dữ liệu theo thời gian thực và hỗ trợ xử lý luồng dữ liệu (stream processing).

2. Các khái niệm cần hiểu rõ để sử dụng được Kafka

Producer: là thành phần tạo ra và gửi các message. Mặc định, producer sử dụng cơ chế round-robin để chọn partition

lưu message khi gửi message cho topic. (Các ứng dụng phát sinh ra dữ liệu và cần xử lý – VD: MintaxAPI,...)

Consumer: là thành phần nhận và đọc các message (Thường là các Worker chuyên dụng cho 1 loại công việc – VD: SendEmailWorker hoặc có thể là chính các ServiceApp VD: MintaxAPI, PrintAPI,...)

Consumer group: Một nhóm các consumer được nhóm lại với nhau để thực hiện 1 nghiệp vụ hoặc 1 nhiệm vụ. Mỗi consumer-group có 1 id-group đơn nhất. Mỗi consumer-group có thể truy cập 1 hoặc nhiều topic.

Broker: Một máy chủ Kafka được gọi là broker. Broker nhận đc message từ producers , gán địa chỉ cho chúng và lưu trữ chúng trên ổ đĩa. Comsumers kết nối broker, truy cập các message thông qua địa chỉ và đọc chúng. Message được lưu trong broker trong 1 thời gian(thiết đặt được) hoặc trong giới hạn dung dượng (thiết đặt được); nếu vượt qua giới hạn, các message sẽ bị xóa để nhường chỗ cho các message mới.

Cluster: là 1 tập các broker. Số broker trong cluster nên là số lẻ. Message trong broker được nhân bản và sao lưu trên các broker khác nhằm đảm bảo nếu 1 broker bị lỗi, dữ liệu luôn sẵn sàng phục vụ từ 1 broker khác. ZooKeeper được kafka cluster sử dụng để giám sát, quản lý điều phối các broker nhằm cung cấp một cơ chế đồng bộ linh hoạt và mạnh mẽ giữa các broker.

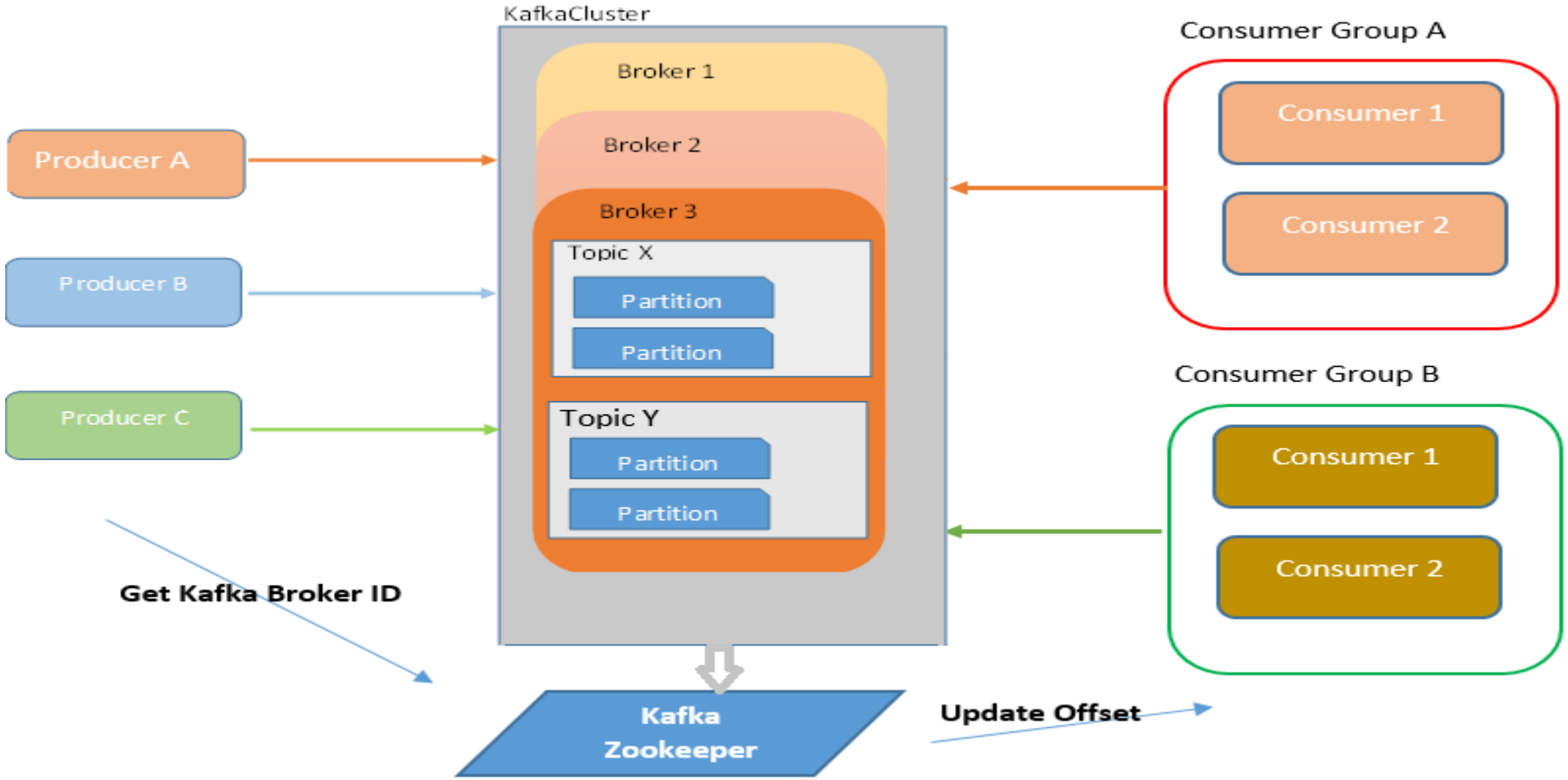
2. Các khái niệm cần hiểu rõ để sử dụng được Kafka

Topic: là 1 một chuỗi(stream) message được producer gửi tới broker, từ đó consumer nhận được message. Tên topic là duy nhất trong toàn hệ thống.

Partition: là 1 phân vùng của topic lưu dưới dạng log, chứa 1 chuỗi tin nhắn có thứ tự và bất biến. Topic có thể bao gồm nhiều partition, mặc định là 1. Số lượng partition trong 1 topic có thể thiết lập tùy chỉnh được.

Offset: Là id của message ứng với số message đã được đọc ở 1 partition, nó định vị tin nhắn trong partition. Offset chỉ lưu giá trị commit gần nhất(đánh dấu dữ liệu đã được thay đổi trạng thái, đồng bộ trạng thái đó cho các broker, set giá trị mới cho offset). Khi consumer đọc(hoặc producer ghi) dữ liệu thành công, nó phải được commit. Nếu consumer chưa xử lý dữ liệu thành công, chưa commit giao dịch đọc thì consumer đó có thể thử đọc lại hoặc 1 consumer khác có thể đọc tiếp từ vị trí dữ liệu đó.

3. Luồng chạy của hệ thống Kafka



Source: TutorialsPedia.com

MISA®
TIN CẬY - TIỆN ÍCH - TẬN TÌNH

KHÁCH THÍCH - KHÁCH YÊU - KHÁCH CHIA SẺ