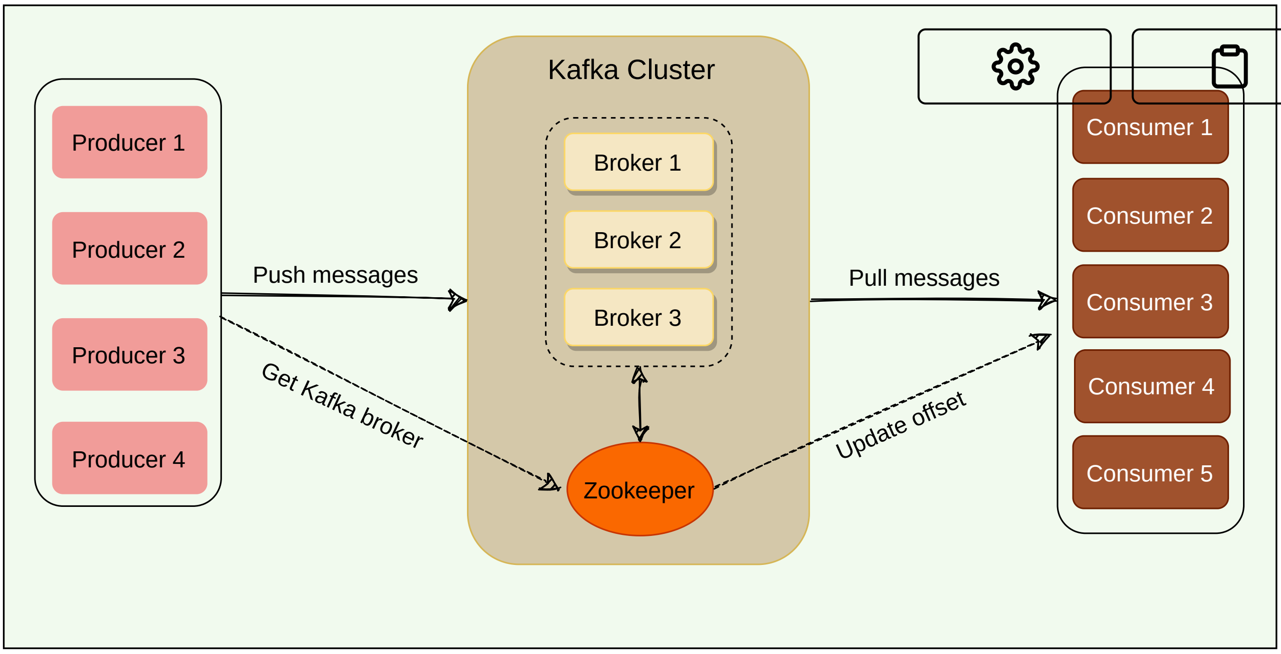
Role of ZooKeeper

Let's delve into how Kafka interacts with ZooKeeper.

What is ZooKeeper?#

A critical dependency of Apache Kafka is Apache ZooKeeper, which is a distributed configuration and synchronization service. ZooKeeper serves as the coordination interface between the Kafka brokers, producers, and consumers. Kafka stores basic metadata in ZooKeeper, such as information about brokers, topics, partitions, partition leader/followers, consumer offsets, etc.



ZooKeeper as the central coordinator#

As we know, Kafka brokers are stateless; they rely on ZooKeeper to maintain and coordinate brokers, such as notifying consumers and producers of the arrival of a new broker or failure of an existing broker, as well as routing all requests to partition leaders.

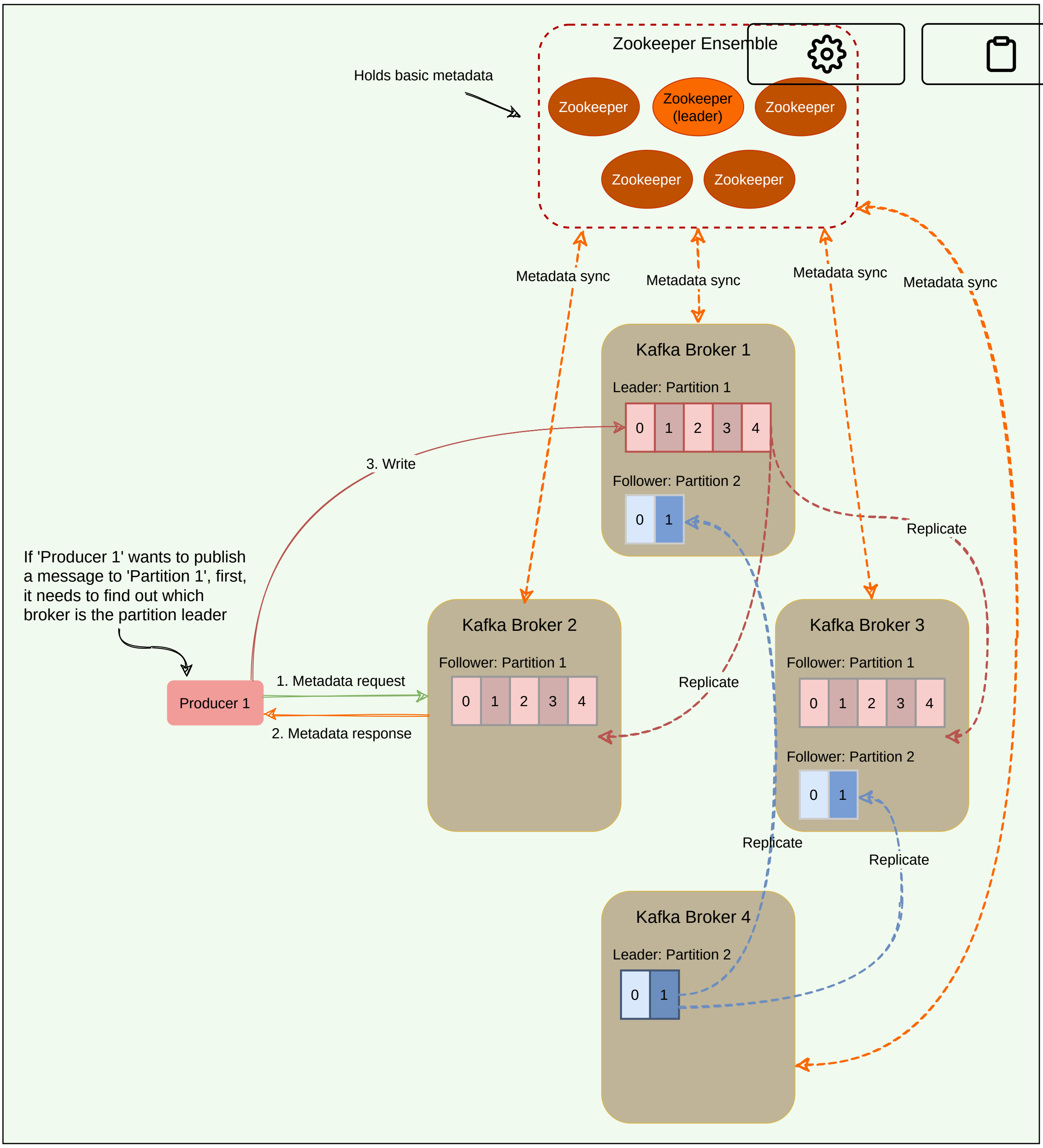
ZooKeeper is used for storing all sorts of metadata about the Kafka cluster:

- It maintains the last offset position of each consumer group per partition, so that consumers can quickly recover from the last position in case of a failure (although modern clients store offsets in a separate Kafka topic).
- It tracks the topics, number of partitions assigned to those topics, and leaders'/followers' location in each partition.
- It also manages the access control lists (ACLs) to different topics in the cluster. ACLs are used to enforce access or authorization.

How do producers or consumers find out who the leader of a partition is?#

In the older versions of Kafka, all clients (i.e., producers and consumers) used to directly talk to ZooKeeper to find the partition leader. Kafka has moved away from this coupling, and in Kafka’s latest releases, clients fetch metadata information from Kafka brokers directly; brokers talk to ZooKeeper to get the latest metadata. In the diagram below, the producer goes through the following steps before publishing a message:

1. The producer connects to any broker and asks for the leader of ‘Partition 1’.
2. The broker responds with the identification of the leader broker responsible for ‘Partition 1’.
3. The producer connects to the leader broker to publish the message.



Role of ZooKeeper in Kafka

All the critical information is stored in the ZooKeeper and ZooKeeper replicates this data across its cluster, therefore, failure of Kafka broker (or ZooKeeper itself) does not affect the state of the Kafka cluster. Upon ZooKeeper failure, Kafka will always be able to restore the state once the ZooKeeper restarts after failure. Zookeeper is also responsible for coordinating the partition leader election between the Kafka brokers in case of leader failure.

← Back

Kafka Workflow

Next

Controller Broker

☑ Mark as Completed

Report an Issue