

Middleware Architectures - [IS3108 / SCS3203] - Assignment 01

Task 4: Enhance the Architecture to Gain Improvement in Availability and Reliability

To improve the availability and reliability of the Pub/Sub system, we can design a distributed architecture that eliminates the single point of failure associated with a single server. This can be achieved using multiple servers (broker nodes) that work together to manage the distribution of messages among publishers and subscribers.

Distributed Pub/Sub Architecture

Components

1. Broker Nodes
 - Multiple broker nodes that handle the distribution of messages.
 - Each broker node is responsible for a subset of topics.
 - Broker nodes can replicate topics to ensure reliability and availability.
2. Load Balancer
 - Distributes incoming client connections (publishers and subscribers) across multiple broker nodes.
 - Ensures even load distribution and redirects traffic if a broker node fails.
3. ZooKeeper (or equivalent coordination service):
 - Maintains configuration information, naming, and synchronization for distributed systems.
 - Keeps track of broker nodes and their responsibilities.
 - Handles leader election for broker nodes to manage topic partitions.
4. Publishers
 - Connect to the load balancer to send messages to the appropriate broker node.
 - Can handle reconnections if redirected by the load balancer due to node failure.
5. Subscribers
 - Connect to the load balancer to receive messages from the appropriate broker node.
 - Can handle reconnections if redirected by the load balancer due to node failure.

Description

- Each broker node is responsible for managing a set of topics. Publishers send messages to these topics, and broker nodes distribute these messages to the appropriate subscribers. Topics can be replicated across multiple broker nodes to ensure that if one node fails, another can take over without losing messages. As the number of topics or clients grows, additional broker nodes can be added to handle the increased load.
- The Load Balancer acts as the entry point for all publishers and subscribers, distributing connections to broker nodes. It monitors the health of broker nodes and redirects traffic if a node fails, ensuring continuous availability.

- The Zookeeper (Coordination Service) manages the configuration and coordination of the distributed system. In case of node failure, ZooKeeper can elect a new leader for managing topic partitions, ensuring high availability. It also keeps track of which broker nodes are responsible for which topics and informs the load balancer and clients of any changes.
- Publishers send messages to the load balancer, which forwards them to the appropriate broker node. If a broker node fails, the load balancer redirects the publisher to a new node. Subscribers connect to the load balancer to receive messages from the appropriate broker node. If a broker node fails, the load balancer redirects the subscriber to a new node.

Architecture Diagram

