

Redis Pub/Sub

by

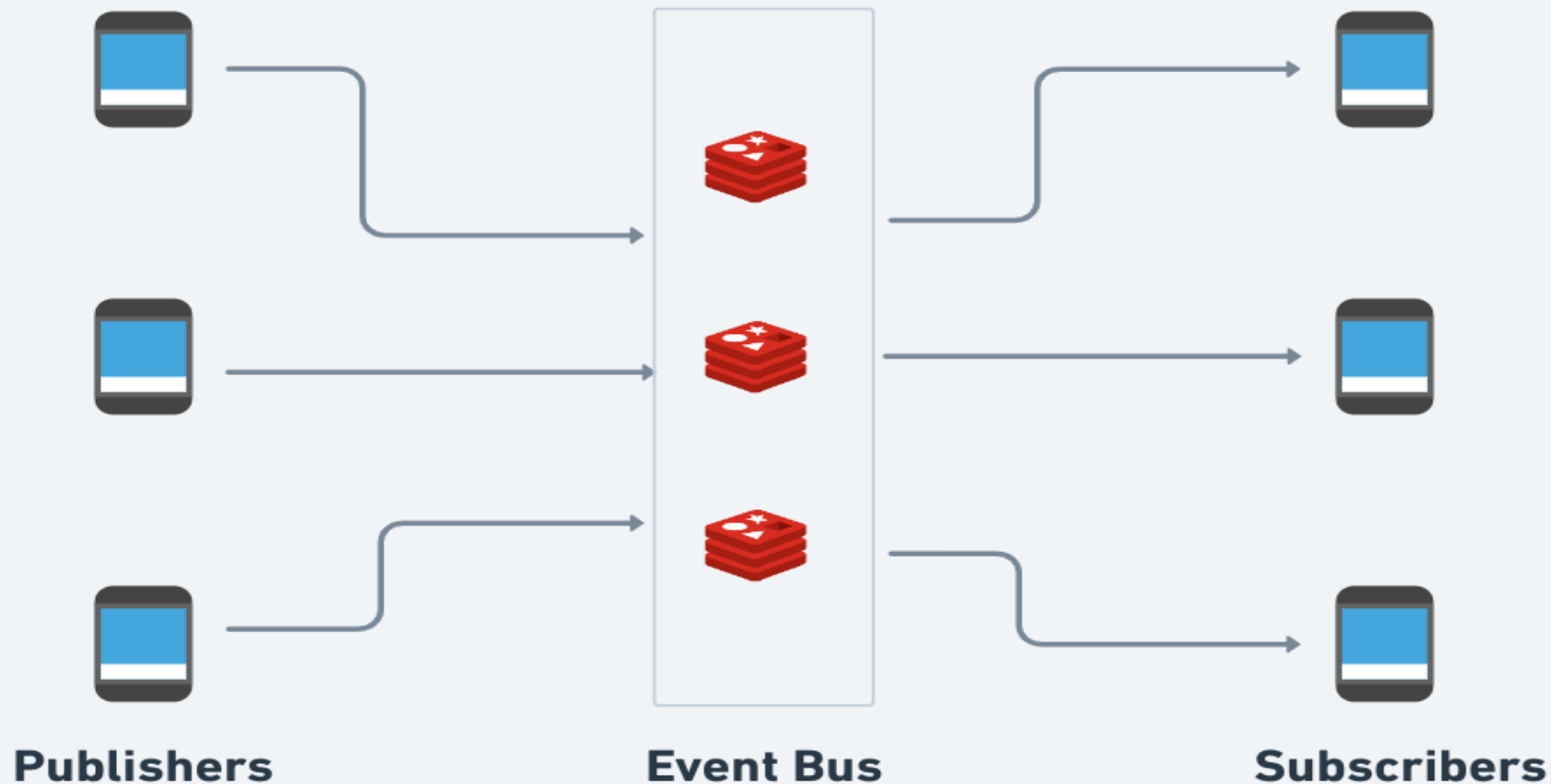


Infinity milesman tech



What is pub/sub model?

- Pub sub basically means, publish subscribe.
- Publish/subscribe messaging, or pub/sub messaging, is a form of asynchronous service-to-service communication used in serverless and microservices architectures.
- There is a channel or event bus or broker between publisher and consumer, you can take it as middleman, which takes data from all producers and gives data to all subscribers which has subscribed to that channel.
- Producers and consumers are unaware of each other.



Architecture of PUB/SUB

When should you use pub/sub?

1. Chat applications are a classic use case of the Pub/Sub pattern. In a chat application, participants can subscribe to chat rooms which have a designated Pub/Sub topic. When a user sends a message to a chat room, his/her chat app instance publishes the message on that chat room's topic. Subscribers of the topic receives the message.
2. Message Queue Service / Message Queuing or batch processing applications can also use the pub/sub pattern. Where all those who want a particular task done will publish the message to a queue and processing units who have subscribed to that queue will receive the message to process the job.

Main advantages of redis pub/sub?

- Enables better & faster integration
- Ensures smoother scalability
- Improves responsiveness and many more.

Commands

- **To subscribe to any channel**
 - SUBSCRIBE channel-name
 - ex. SUBSCRIBE infinity-milesman-tech
- **To subscribe to mutiple channels**
 - SUBSCRIBE channel1 channel2 channel3
- **To unsubscribe a channel**
 - UNSUBSCRIBE channel-name

- **To publish messages to any channel**
 - PUBLISH channel-name “message”
 - ex. PUBLISH infinity-milesman-tech “Welcome learners.”
- **To subscribe on the channel that matches the pattern**
 - PSUBSCRIBE redis*
 - It will consume all the messages from the channel that starts with redis.
- To unsubscribe from the pattern matching channel.
 - PUNSUBSCRIBE channel1 channel2

Note: Redis commands are case insensitive like sql or postgresql

- Apart from it, redis has newly introduced sharded features, it is basically to narrow down the publish subscribers a bit.
- It uses SSUBSCRIBE, SUNSUBSCRIBE and SPUBLISH commands paradigms to implement this feature.

Thank you

Happy Learning

From

 Infinity milesman tech