

Tarea 2

Sistemas Distribuidos

Jorge Contreras 201573547-6 jorge.contreras@sansano.usm.cl

César Quiroz 201573578-6 cesar.quiroz@sansano.usm.cl

En la presente tarea se trabajó con dos arquitecturas distintas Remote procedure call (utilizando gRPC) y Asynchronous Messaging (usando RabbitMQ) para implementar un servicio de mensajería compuesto por un servidor y N clientes. A continuación, se describirán ambas arquitecturas, sus diferencias y se indicará cual es más apropiada para el servicio.

- gRPC:
gRPC es un framework iniciado por Google basado en HTTP/2 y Protocol Buffers, además de ser un servicio remote procedure call (RPC).

Su funcionamiento se basa en mensajes mediante un stubs, para la comunicación entre dos partes además de usar threadpool para tener n clientes haciendo peticiones a un servidor.

Finalmente, gRPC tiene la ventaja de soportar distintos tipos de lenguajes (Python, java, etc) interactuando entre si.

- RabbitMQ:

RabbitMQ es un software desarrollado por Rabbit, utiliza protocolos como HTTP y XMPP, además de ser un servicio del tipo Asynchronous Messaging.

Su funcionamiento se basa en tres componentes, servidor, cliente y mensaje Broken, donde se pueden encolar los mensajes.

.

El mensaje broken cuenta con librerías para la recuperación de mensajes y de conexiones.

Diferencias:

Existen varias diferencias entre gRPC y RabbitMQ:

- RabbitMQ posee una cola de mensajes produciendo un Delay con muchas conexiones, mientras que gRPC es una conexión directa entre ambas partes.

- Gracias a la cola de RabbitMQ es posible que un mensaje llega a destino a pesar de que el receptor no esta levantado, por otro lado gRPC, necesita que ambas partes esten levantadas, para la llegada de un mensaje.
- gRPC requiere que se defina la estructura de los mensajes antes de ser enviados mientras que RabbitMQ solo necesita colas.

Recomendación:

No se puede decir si RabbitMQ o gRPC son mejores, ya que, aunque apunte a resolver el mismo problema (mesajería), ambas tienen ventajas y desventajas, sin embargo, dependiendo el escenario se puede dar una ganadora.

Para aplicaciones con poco tráfico de mensajes y se requiera que la información llegue al destino es mejor usar gRPC, mientras que para aplicaciones con una gran demanda de mensajes es mejor RabbitMQ además de ofrecer servicios de recuperación y un monitoreo de mensajes.