



SISTEMAS DISTRIBUTIVOS
COMUNICACIÓN EN
ARQUITECTURAS DISTRIBUIDAS

COMUNICACIÓN EN ARQUITECTURAS DISTRIBUIDAS

La comunicación en arquitecturas distribuidas se refiere al conjunto de mecanismos, protocolos y estrategias mediante los cuales los componentes de un sistema distribuyen datos, coordinan acciones y mantienen coherencia, a pesar de estar ubicados en distintos nodos y redes. Esta comunicación es la columna vertebral de los sistemas distribuidos, dado que permite que sus elementos funcionen de forma coordinada para ofrecer servicios unificados al usuario (Muñoz Escoí, 2013).

Importancia de la comunicación

En un sistema distribuido, cada componente realiza tareas específicas, pero requiere información de otros para completar procesos. Sin una comunicación efectiva:

- Se pierden datos en tránsito.
- Se generan inconsistencias en el sistema.
- Se afectan la escalabilidad y la disponibilidad.

Por ello, se emplean estrategias robustas de comunicación para garantizar integridad y eficiencia.

Modelos de comunicación

Los sistemas distribuidos emplean dos modelos de comunicación principales:

Comunicación síncrona

El emisor envía un mensaje y espera la respuesta antes de continuar con otras tareas.



Ejemplo. Una aplicación cliente que consulta a un servidor la disponibilidad de un producto antes de continuar con el proceso de compra.

Comunicación asíncrona

El emisor envía un mensaje y continúa con sus tareas sin esperar la respuesta inmediata (Muñoz Escoí, 2013).



Ejemplo. Un sistema de pedidos genera un evento de “pedido realizado” y el sistema de inventario lo procesa en segundo plano.

Protocolos utilizados

Entre los protocolos de comunicación en sistemas distribuidos, se destacan:

- **HTTP/HTTPS.** Usado en arquitecturas cliente-servidor y microservicios, facilitando la interoperabilidad entre aplicaciones web.
- **gRPC.** Permite llamadas a procedimientos remotos de manera eficiente, ideal para microservicios.

- **MQTT y AMQP.** Protocolos orientados a mensajes, útiles en arquitecturas basadas en eventos.
- **Sockets.** Permiten la transmisión de datos en tiempo real, común en aplicaciones de mensajería instantánea.

Mecanismos de comunicación

Existen diversas formas de implementar la comunicación en sistemas distribuidos:

- **Invocación de métodos remotos (RMI).** Permite llamar funciones en otro nodo como si fueran locales.
- **Intercambio de mensajes.** Se utilizan colas de mensajes o brokers para enviar y recibir datos.
- **Flujos de datos continuos.** Para sistemas que requieren transmisión constante de información, como sensores IoT o streaming de video.

Ejemplos prácticos

- **Sistema de reservas de vuelos.** Cuando un usuario reserva un vuelo, se comunica con el servicio de asientos, el servicio de pagos y el servicio de notificaciones de forma distribuida, intercambiando información para completar la reserva.
- **Aplicaciones de mensajería (WhatsApp, Telegram).** Utilizan comunicación asíncrona y en tiempo real, donde los mensajes se envían a servidores que los distribuyen a los receptores mientras se mantiene la sincronización entre dispositivos.
- **Sistemas IoT.** Los dispositivos (sensores) envían constantemente datos a un servidor central mediante protocolos MQTT, manteniendo comunicación eficiente con bajo consumo de recursos.

Retos en la comunicación distribuida

- **Latencia de red.** Afecta la rapidez con que se transmiten los datos.
- **Fallas de comunicación.** Pueden provocar pérdida de información.
- **Consistencia de datos.** Es un desafío cuando se intercambian datos de manera concurrente, entre múltiples nodos.
- **Seguridad.** Es necesario cifrar y autenticar las comunicaciones para proteger la información distribuida.

Para mitigar estos retos, se implementan mecanismos como reintentos automáticos, protocolos tolerantes a fallos, sistemas de mensajería con confirmación de entrega y cifrado de datos en tránsito.