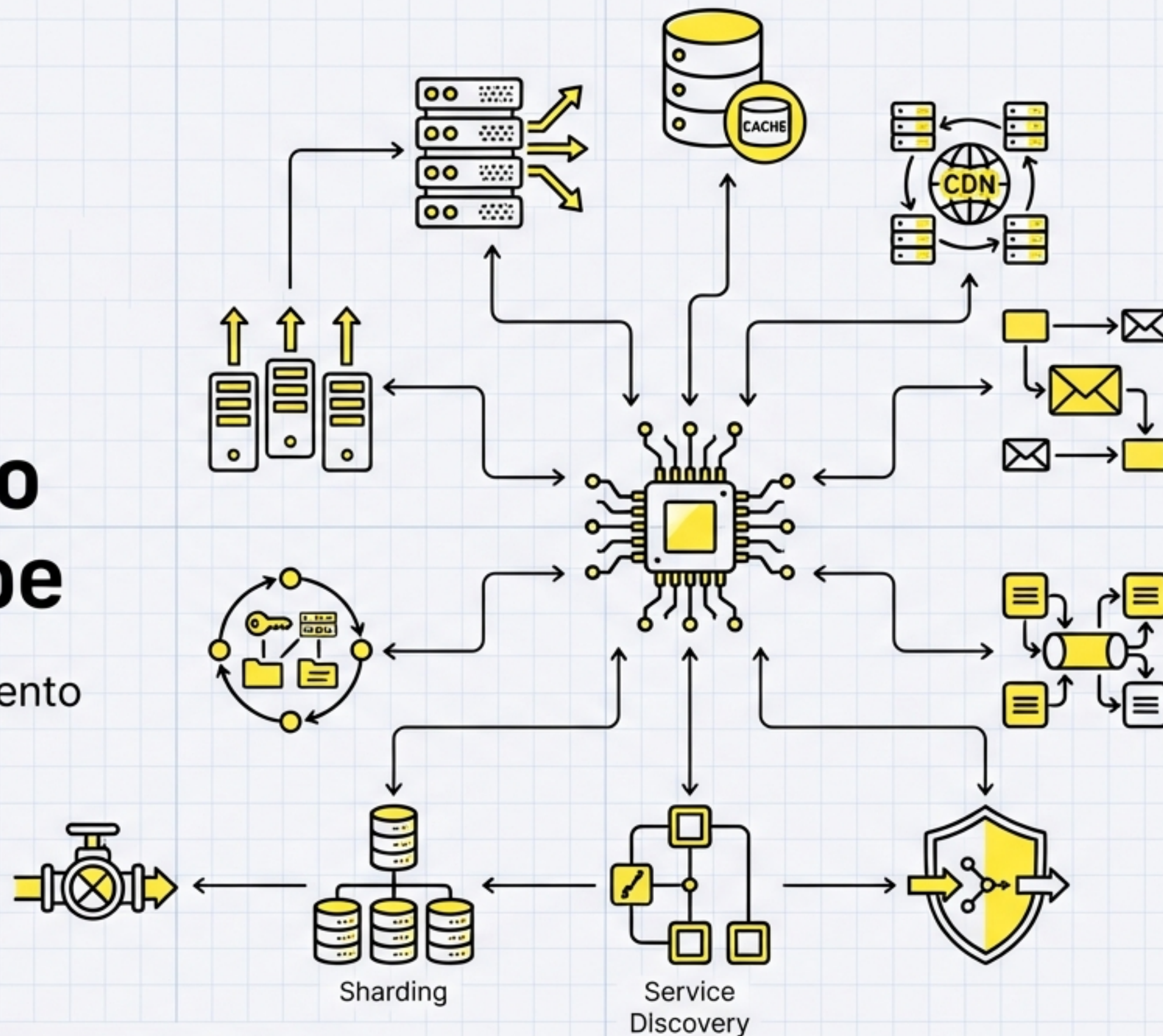


12 Conceptos de Arquitectura de Software que Todo Desarrollador Debe

Una guía visual para escalar tu conocimiento técnico de Junior a Arquitecto.



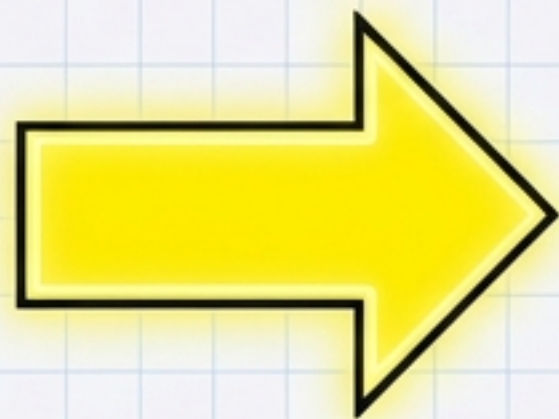
Basado en material de ByteByteGo

De Programador a Arquitecto

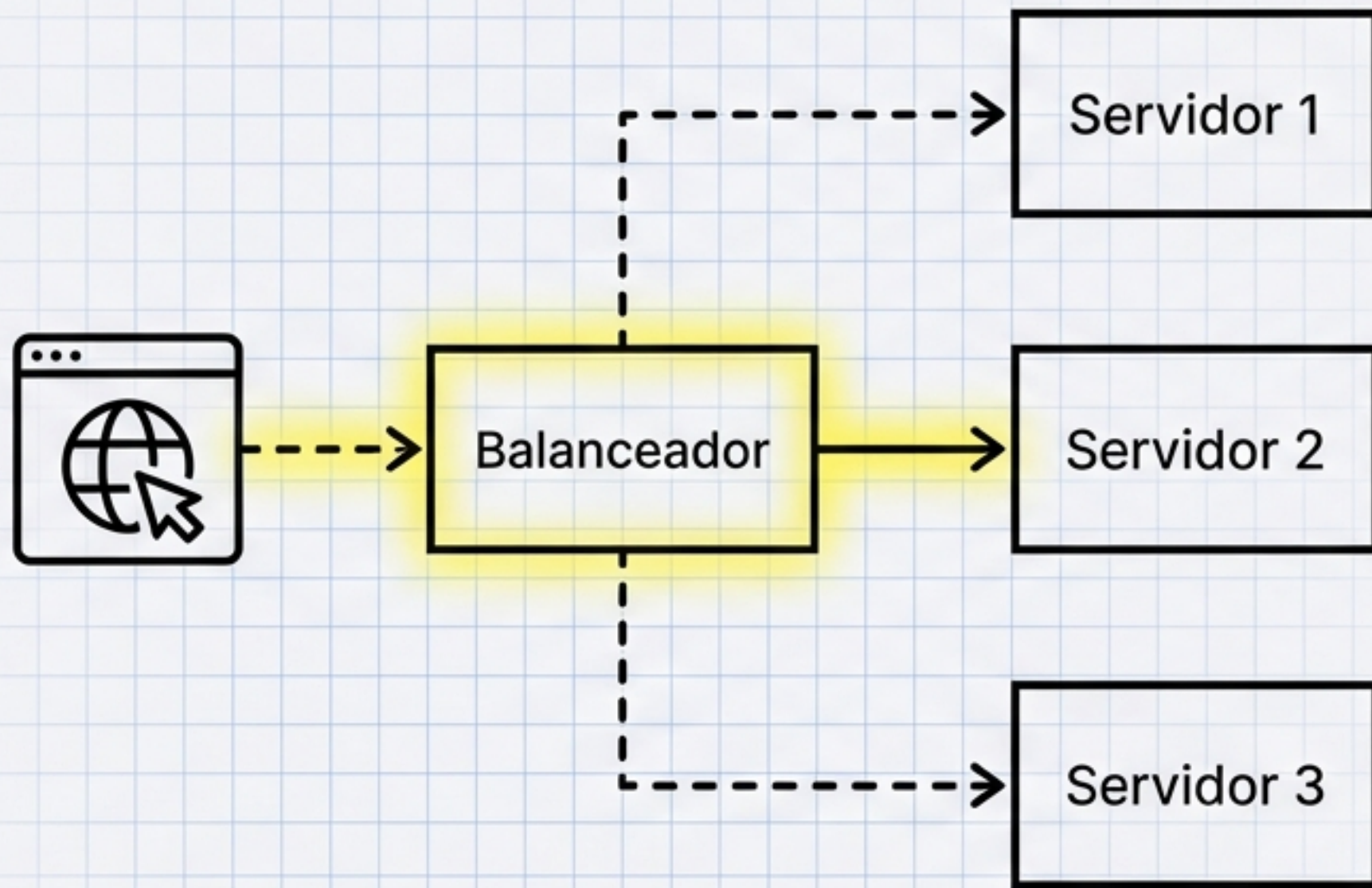
Escribir código es solo la mitad de la batalla. Diseñar sistemas que sean escalables, confiables y rápidos requiere nuevas herramientas mentales. Estos 12 patrones son el vocabulario fundamental del diseño de sistemas modernos.



Funcionalidad Local



Escalabilidad y Resiliencia



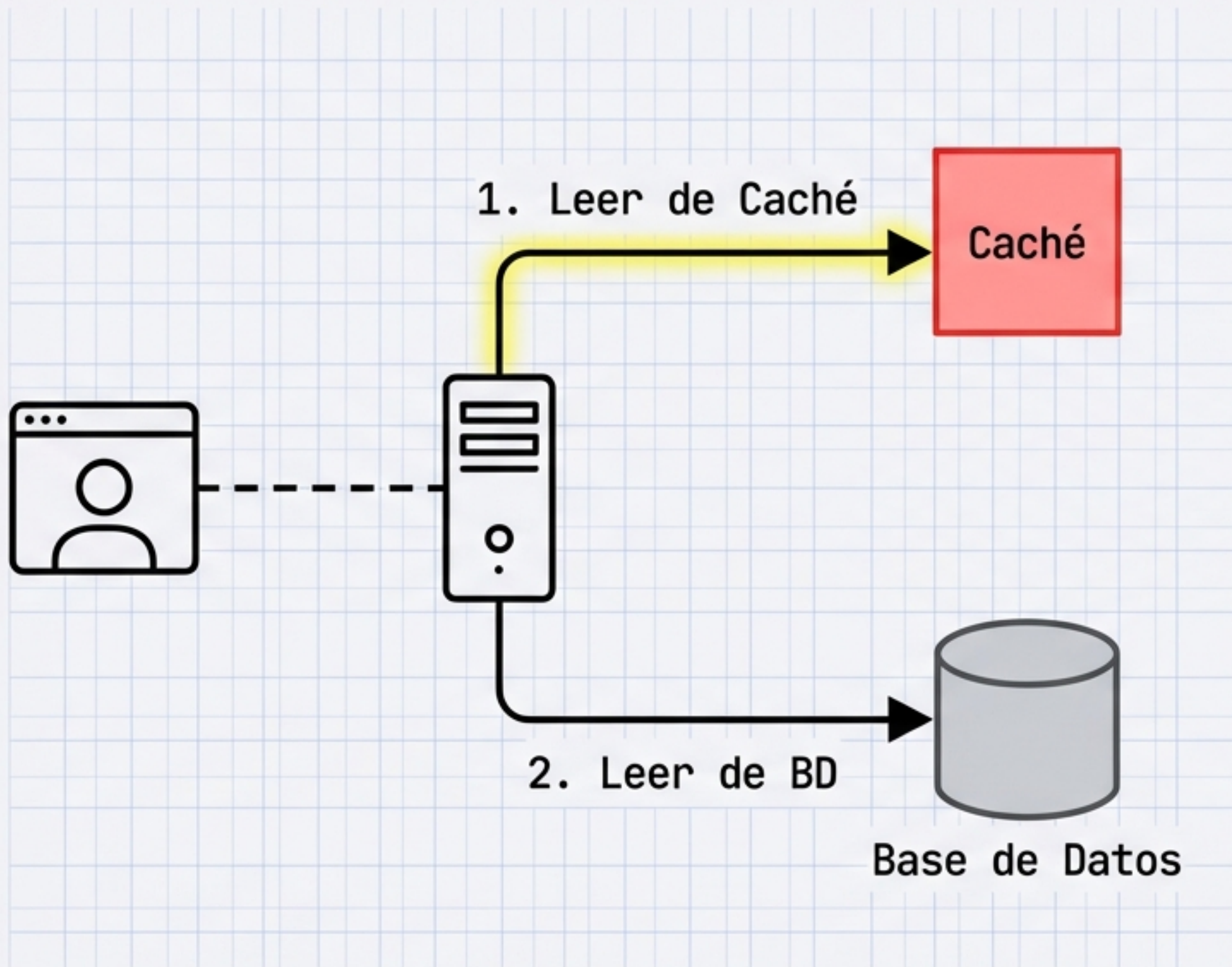
Balanceo de Carga

| Load Balancing

🛡️ Alta Disponibilidad

Distribuye el tráfico entrante a través de múltiples servidores para asegurar que ningún nodo individual se sature.

Previene caídas del sistema al evitar cuellos de botella en un solo punto.



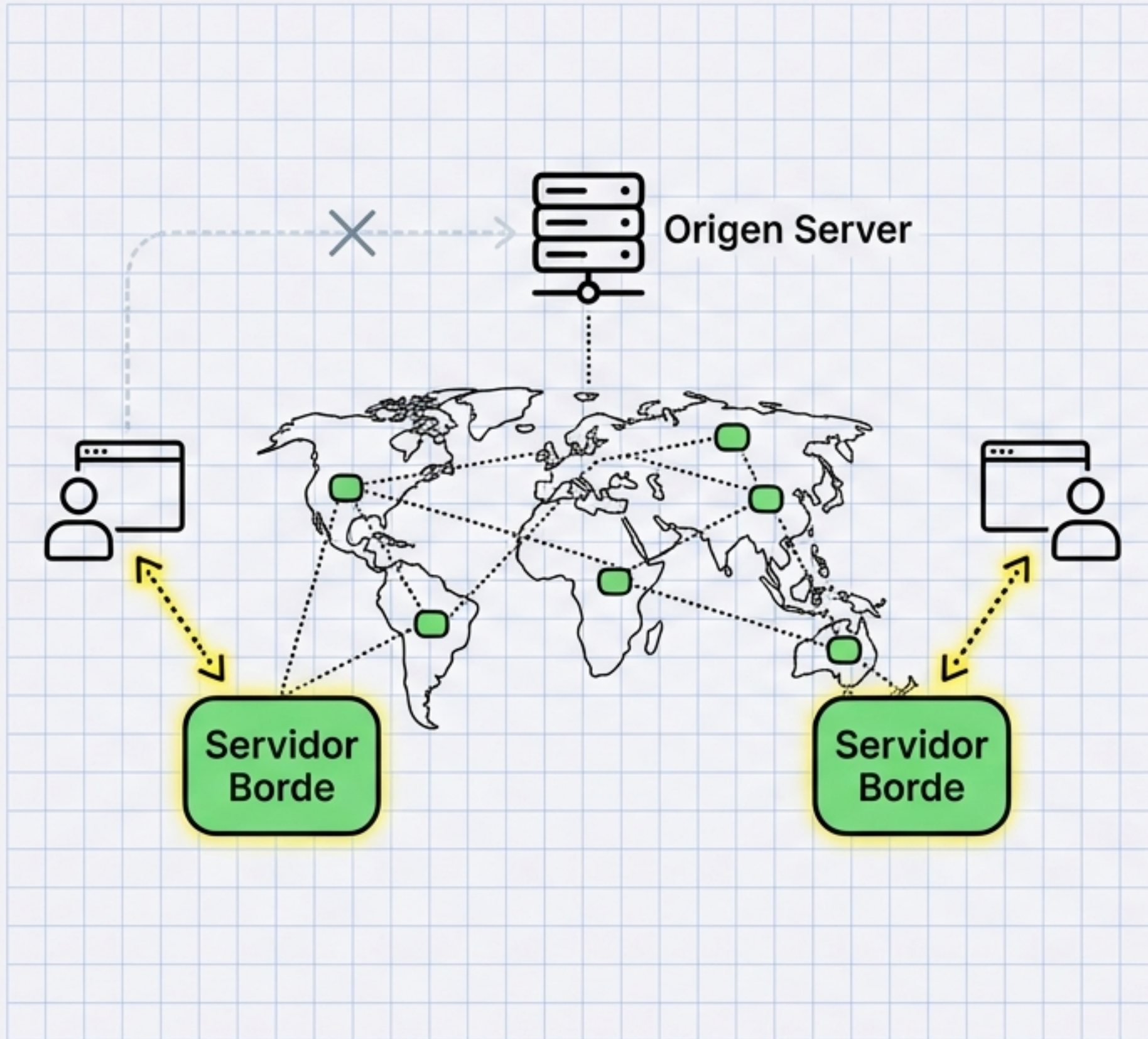
Caché

| Caching

⚡ Velocidad Extrema

Almacena datos consultados frecuentemente en la memoria RAM para reducir la latencia.

Intercambia consumo de memoria por velocidad de respuesta, reduciendo drásticamente la carga en la base de datos principal.



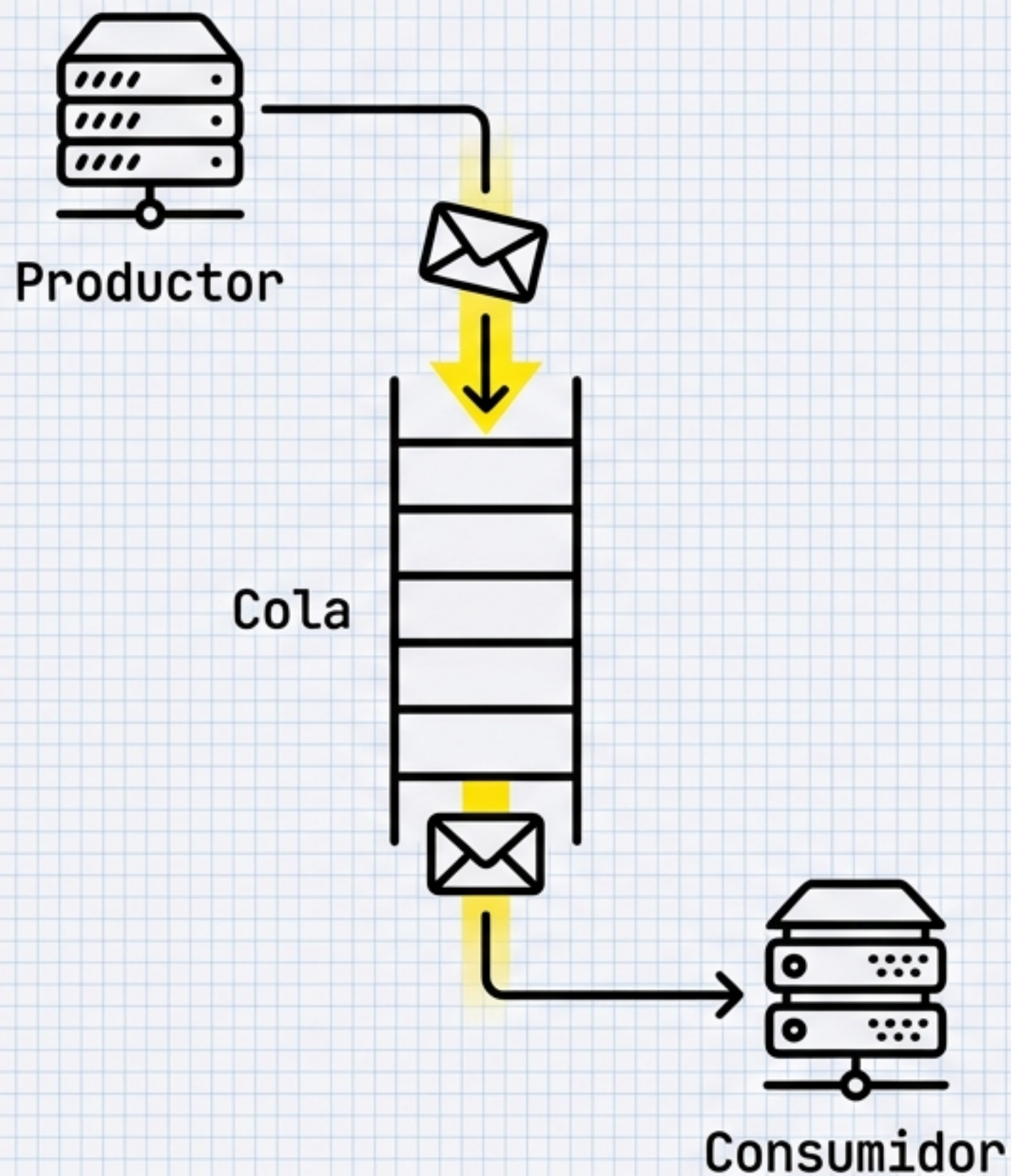
Red de Distribución de Contenido

| CDN

🌐 Menor Latencia

Almacena activos estáticos (imágenes, CSS) en servidores geográficamente distribuidos para que los usuarios **descarguen el contenido** desde la ubicación más cercana.

Acerca físicamente los datos al usuario, mejorando la experiencia de carga global.



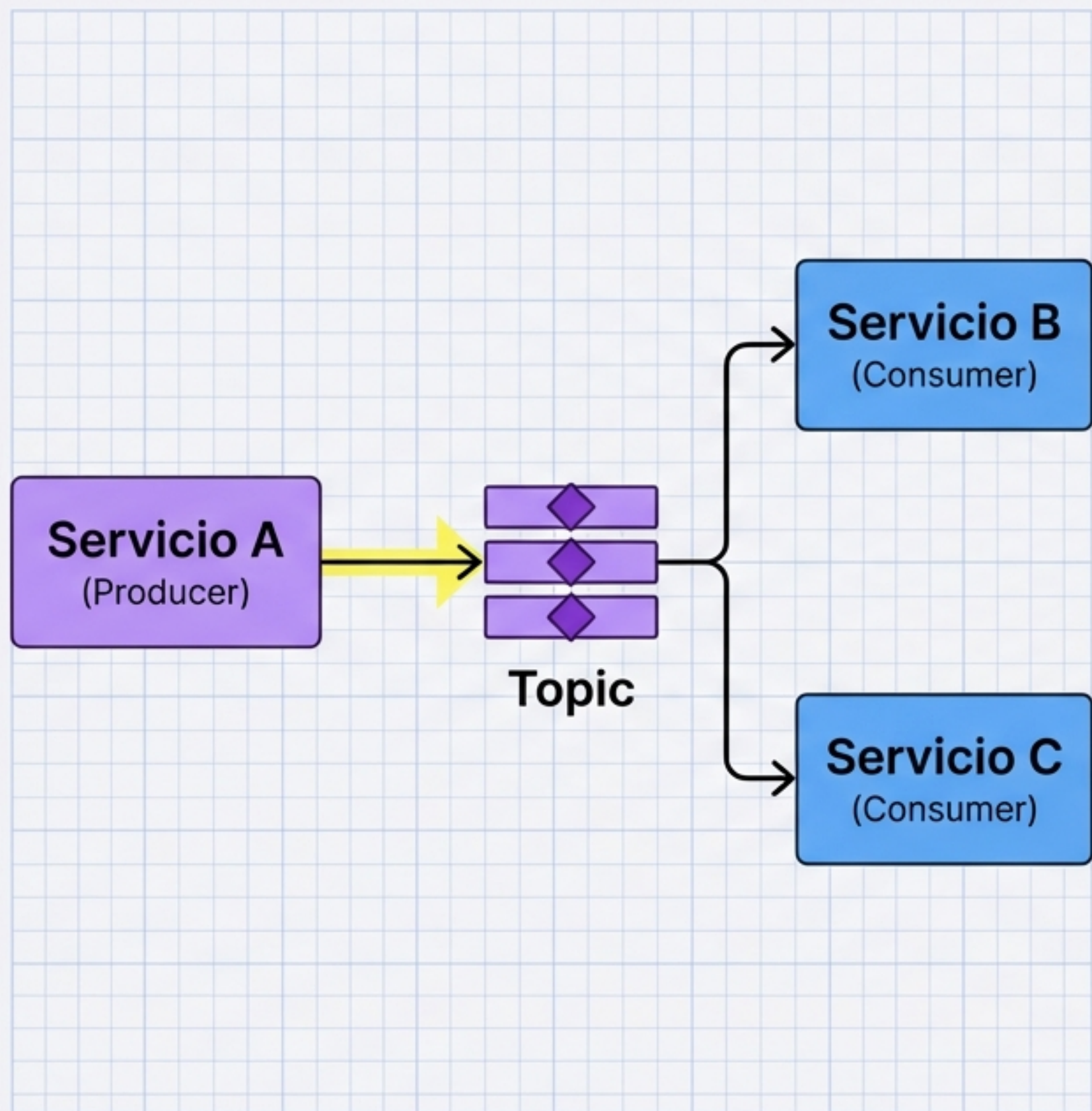
Colas de Mensajes JetBrains Mono

| Message Queue

Highlighter Yellow : 🔄 Desacoplamiento

Desacopla componentes permitiendo que los productores envíen mensajes que los consumidores procesarán de forma asíncrona.

Key Detail El productor no necesita esperar a que el consumidor termine; evita bloqueos en el sistema.



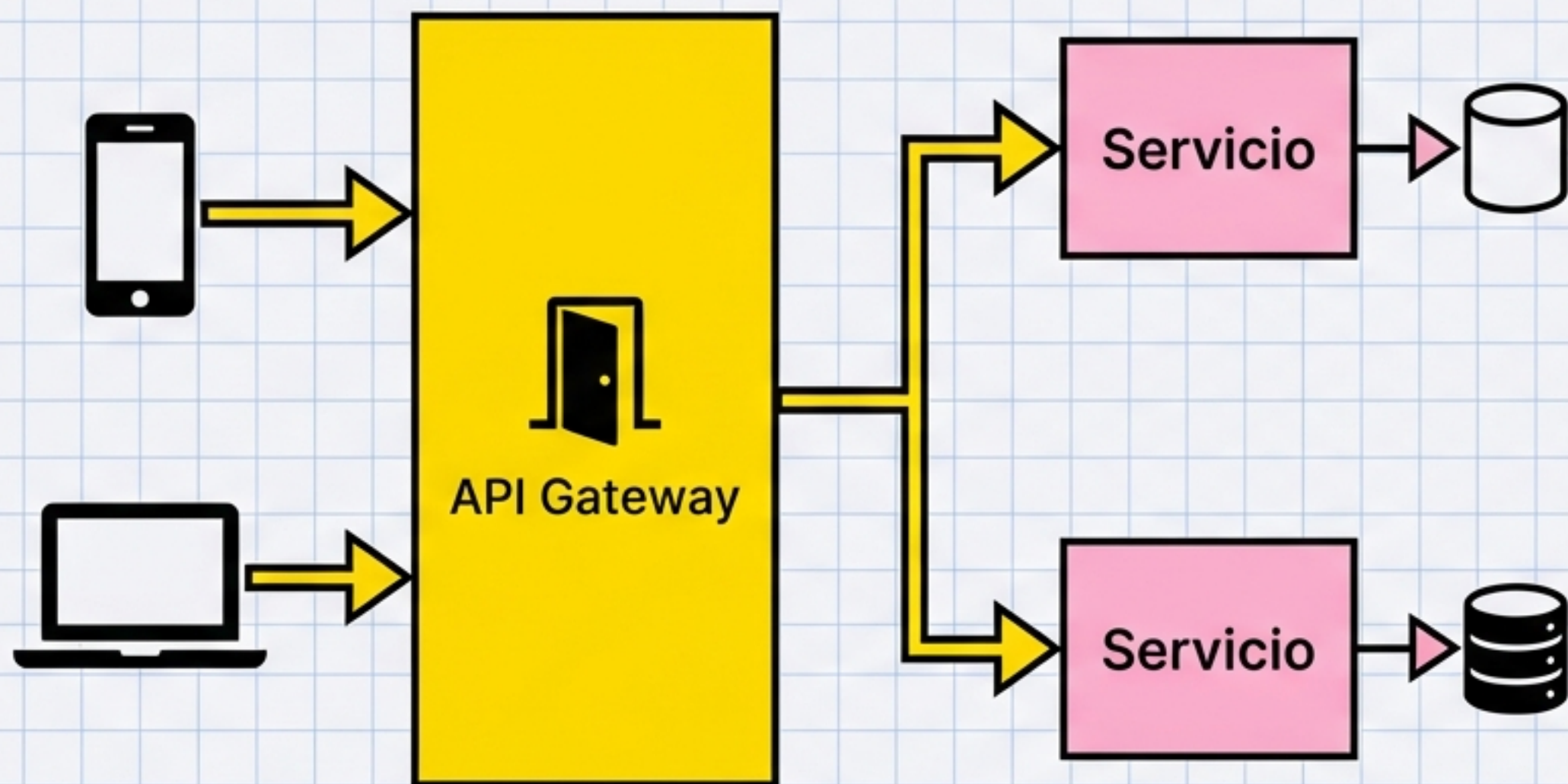
Publicar-Suscribir

| Pub-Sub

📢 Comunicación "Uno a Muchos"

Permite que múltiples consumidores reciban mensajes de un "tópico" o tema específico simultáneamente.

Un solo evento puede desencadenar múltiples acciones independientes en diferentes microservicios.



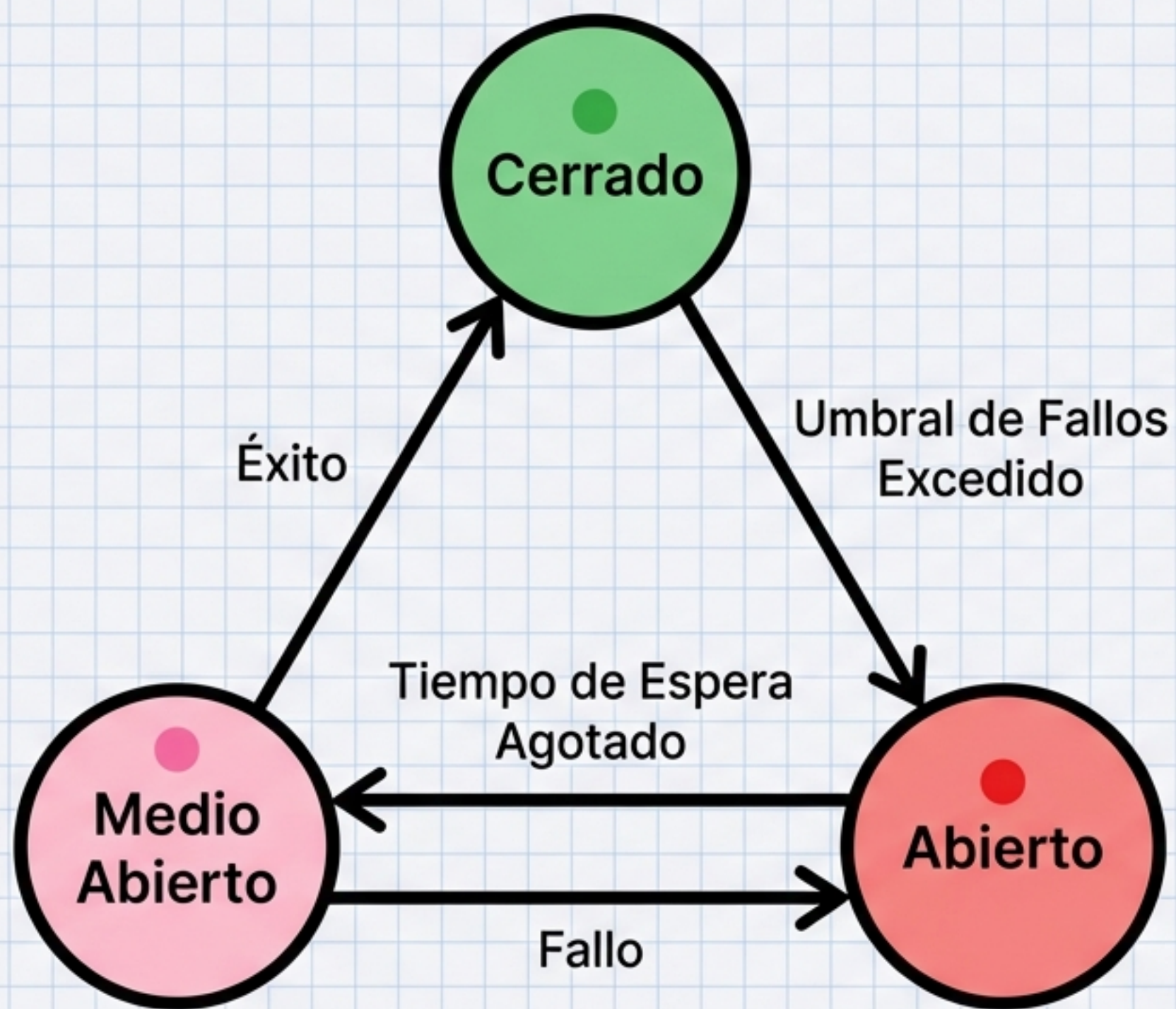
API Gateway

| API Gateway

Gestión Centralizada

Actúa como un punto de entrada único para las solicitudes del cliente.

Maneja **enrutamiento**, **autenticación**, **limitación de tasas** y **traducción de protocolos** en un solo lugar.



Circuit Breaker

| Interruptor de Circuito

 **Prevenir Fallos en Cascada**

Monitorea las llamadas a servicios fallidos y detiene los intentos cuando los fallos superan un umbral.

'Falla rápido' para permitir que el sistema se recupere en lugar de saturarlo con intentos inútiles.

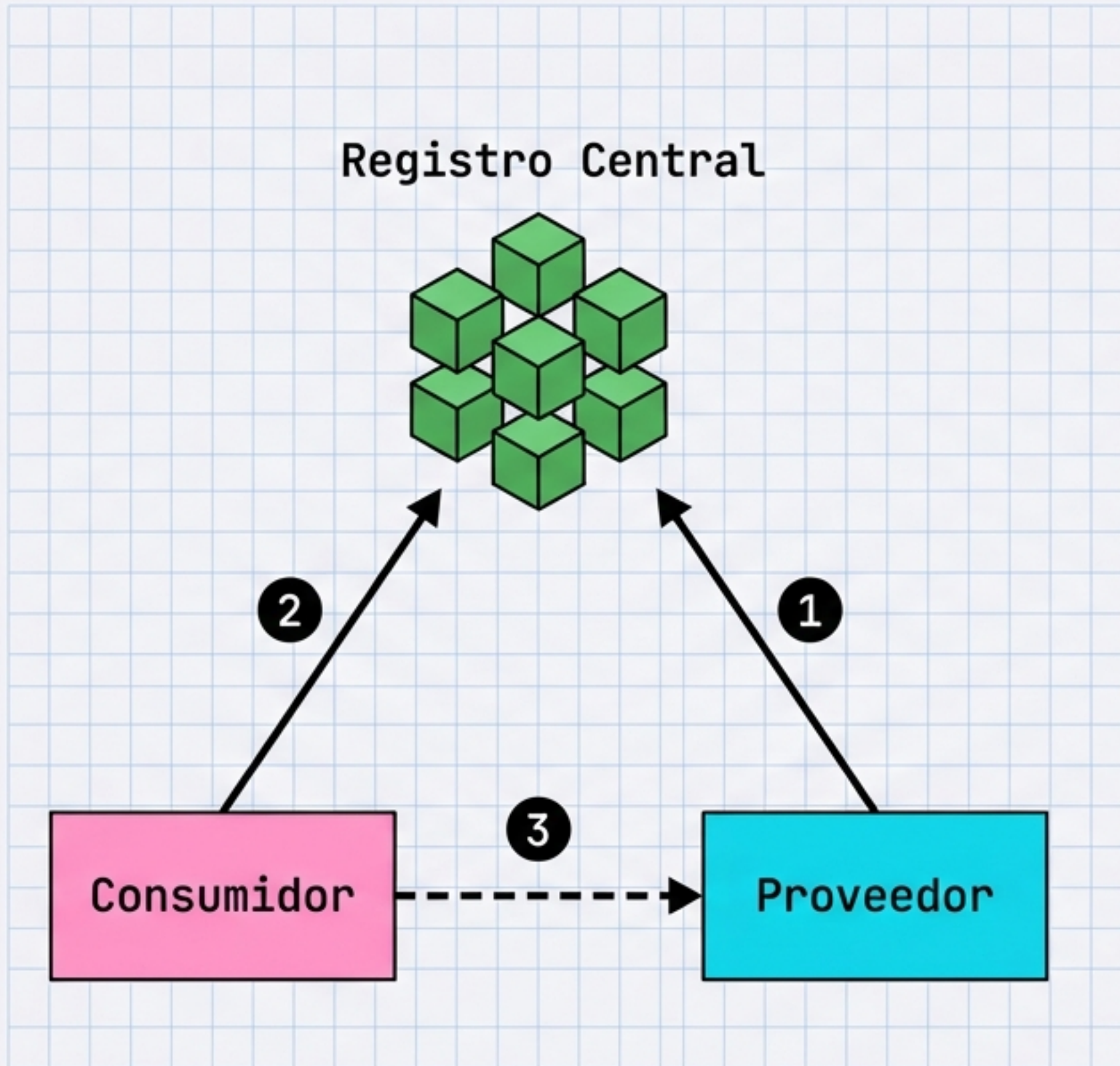
Descubrimiento de Servicios

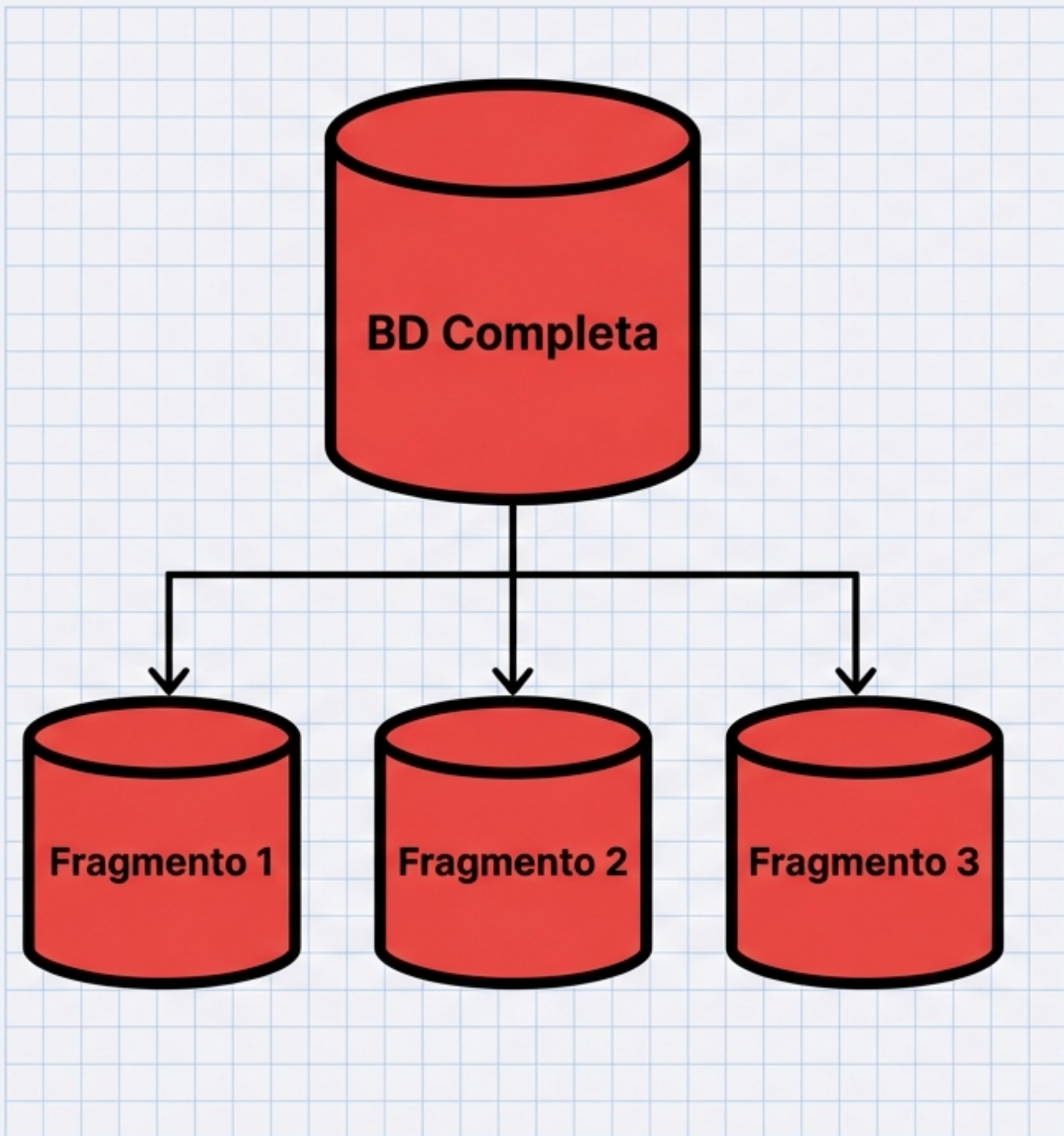
| Service Discovery

🔍 Localización Dinámica

Rastrea automáticamente las instancias de servicio disponibles para que los componentes puedan localizarse y comunicarse dinámicamente.

Esencial en la nube donde las direcciones IP de los servidores cambian constantemente.





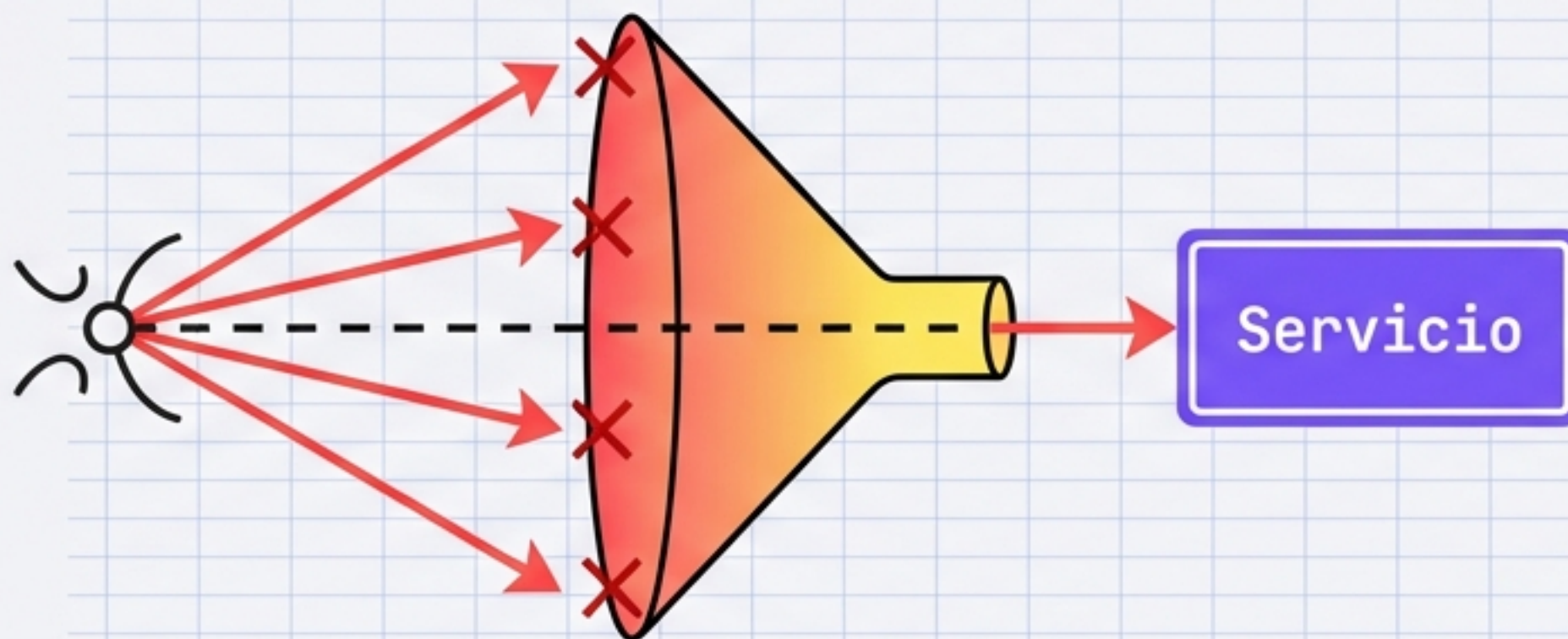
Fragmentación

| Sharding

Escalado Horizontal de Datos

Divide grandes conjuntos de datos en múltiples nodos basándose en una clave de fragmentación específica (shard key).

Permite manejar terabytes de datos distribuyendo la carga de lectura y escritura entre muchas máquinas.



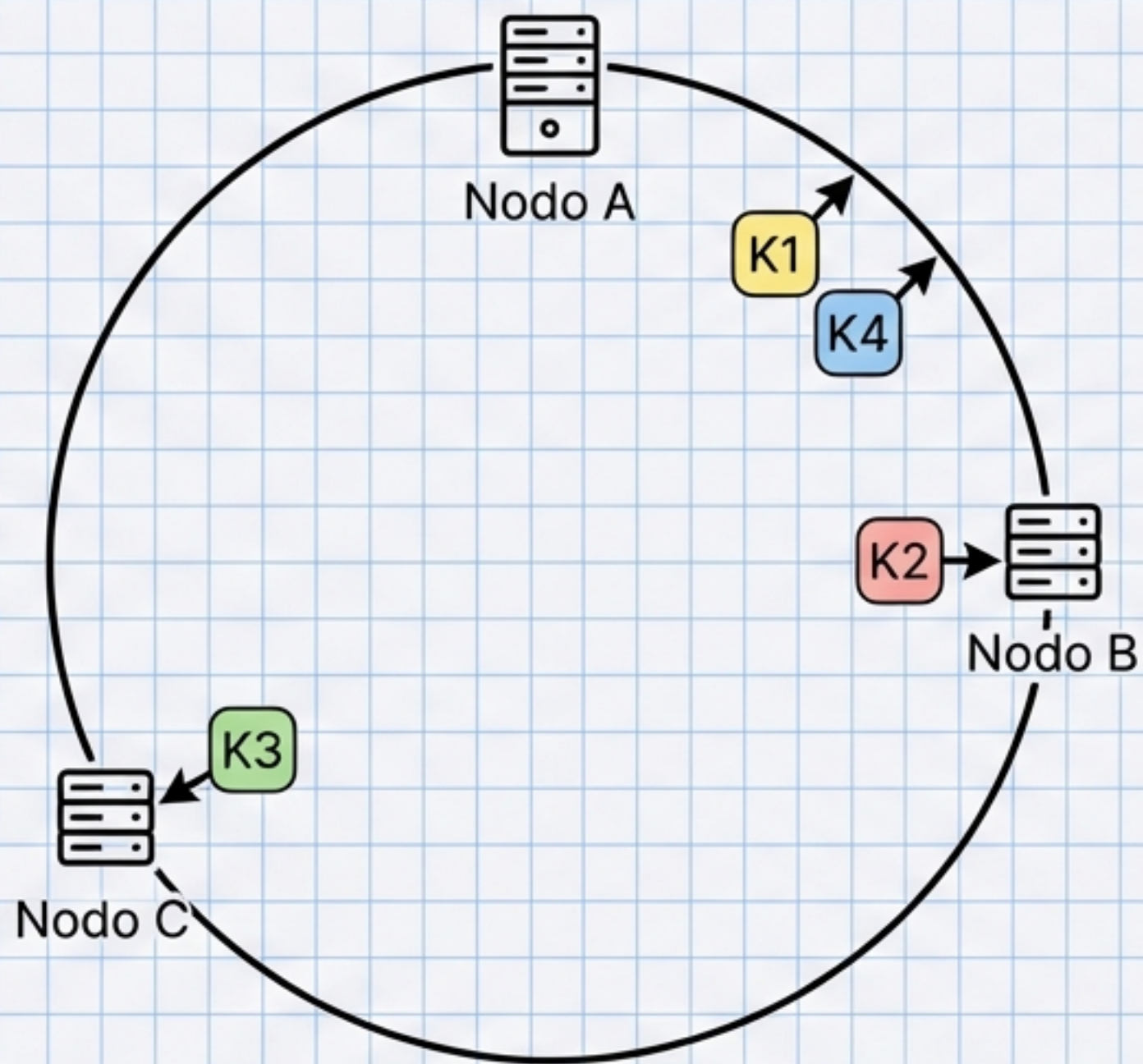
Limitación de Tasas

| Rate Limiting

🛡️ Protección contra Sobrecarga

Controla el número de solicitudes que un cliente puede realizar en una ventana de tiempo determinada.

Protege los servicios contra **ataques de fuerza bruta, DDoS o usuarios abusivos**.



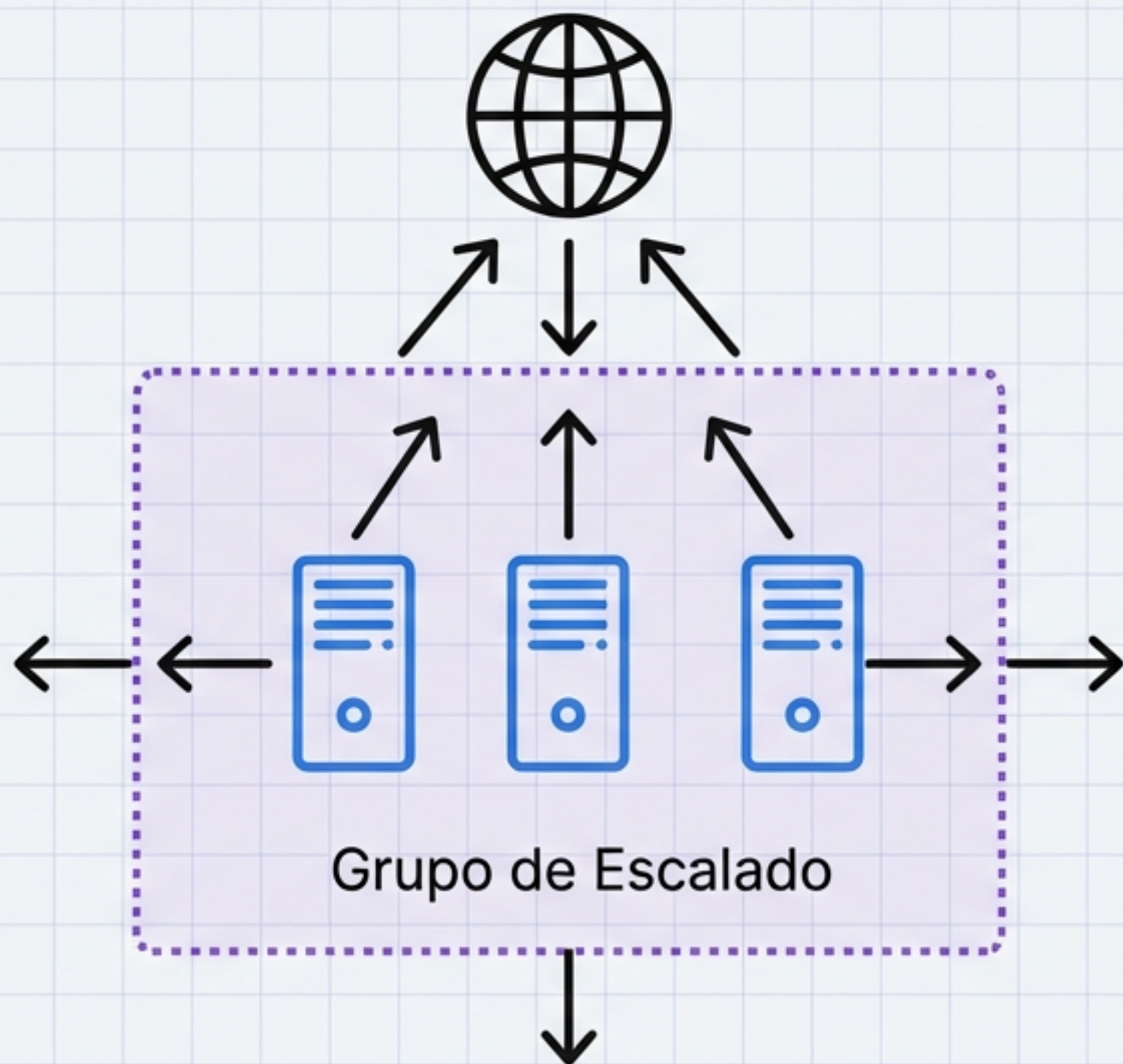
Hashing Consistente

| Consistent Hashing

⚖ Escalado sin Caos

Distribuye datos entre nodos de una manera que minimiza la reorganización cuando los nodos se unen o abandonan la red.

Solo una pequeña porción de las claves necesita moverse cuando la topología del servidor cambia.



Escalado Automático

| Auto Scaling

✓ Elasticidad y Ahorro

Agrega o elimina recursos informáticos automáticamente en función de métricas definidas (como uso de CPU o Memoria).

Key Detail

Asegura que el rendimiento se mantenga durante los picos de tráfico y reduce costos durante los valles.

JetBrains Mono

Tu Turno

La arquitectura de software es una disciplina en constante evolución. Dominar estos 12 conceptos es solo el comienzo.

Microservices

Event Sourcing

Event Sourcing

Serverless



¿Qué concepto arquitectónico añadirías tú a esta lista?