# Aporte Redis-Kafka

**Caso de uso real – Kafka:**

Streaming de transacciones bancarias para auditoría antifraude: los microservicios publican eventos de transacción y un consumidor de riesgo correlaciona patrones en tiempo real para detectar anomalías.

**Caso de uso real – Redis:**

Cache de saldos y límites de cuenta con TTL para acelerar lecturas en consultas de balance y validar topes de operación.

**Caso hipotético en 3 microservicios:**

• ClientMS publica eventos de alta/actualización de clientes en Kafka.  
• AccountMS consume dichos eventos para sincronizar perfiles y emite eventos de apertura/cierre de cuentas.  
• TransactionMS consume eventos de cuentas para habilitar transacciones; cada transacción se publica en Kafka.  
• Redis se utiliza para cachear: (a) tokenización y límites por cliente, (b) últimos N movimientos por cuenta, (c) reglas de idempotencia para evitar duplicados.

**Beneficios:**

Desacoplamiento, escalabilidad horizontal, resiliencia ante picos; latencia sub-milisegundo en lecturas críticas con Redis.

**Riesgos y mitigaciones:**

Orden de eventos (usar claves por partición y semántica al menos una vez), consistencia eventual, y políticas de expiración en Redis.