

Migración de Bases de Datos: Conceptos y Principios

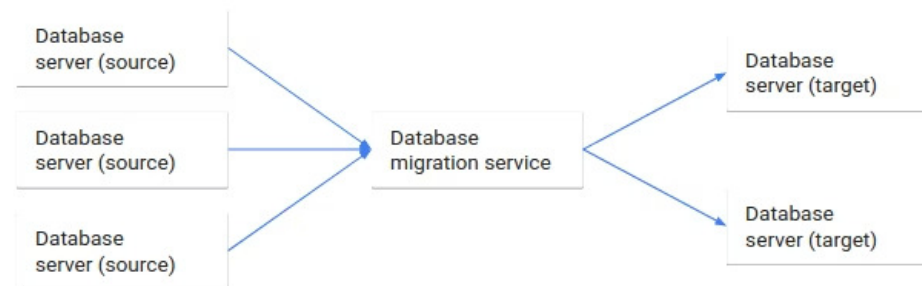
Una guía completa sobre arquitectura, terminología y principios fundamentales para migrar bases de datos segun la guia de Google Cloud.



¿Qué es la Migración de Bases de Datos?

La **migración de bases de datos** es el proceso de transferir datos desde una o más bases de datos de origen hacia bases de datos de destino mediante un servicio especializado.

Al finalizar, el conjunto de datos reside completo en las bases de datos de destino. Los clientes se redirigen al nuevo sistema y las bases de datos de origen se desactivan.



Servicios Involucrados

Tecnologías y herramientas que facilitan la transferencia de datos

Tipos de Migración

Homogénea vs. heterogénea según las tecnologías

Tiempo de Inactividad

Estrategias para minimizar la interrupción del servicio

Terminología Clave

Base de Datos de Origen

Contiene los datos que se migrarán hacia una o más bases de datos de destino.

Base de Datos de Destino

Recibe los datos migrados desde las bases de datos de origen.

Sistema de Migración

Software que conecta origen y destino, ejecutando la transferencia de datos.

Migracion vs Replicacion

En una **migración**, mueves datos de bases de datos de origen a bases de datos de destino. Una vez que los datos se migran por completo, borras las bases de datos de origen y redireccionas el acceso de los clientes a las bases de datos de destino.

Con la **replicación de base de datos**, transfieres datos de forma continua desde las bases de datos de origen hacia las bases de datos de destino, sin borrar las de origen. A veces, la replicación de bases de datos se conoce como transmisión de bases de datos.

Clasificación de Migraciones

Homogénea vs. Heterogénea

Homogénea: Misma tecnología (MySQL a MySQL). **Heterogénea:** Diferentes tecnologías (Oracle a Postgres).

Completa vs. Parcial

Transferencia de toda la base de datos o solo un subconjunto específico de datos.

Tiempo de Inactividad

Nulo (imposible) vs. casi nulo (minimizado mediante estrategias específicas).

Cardinalidad de Migración

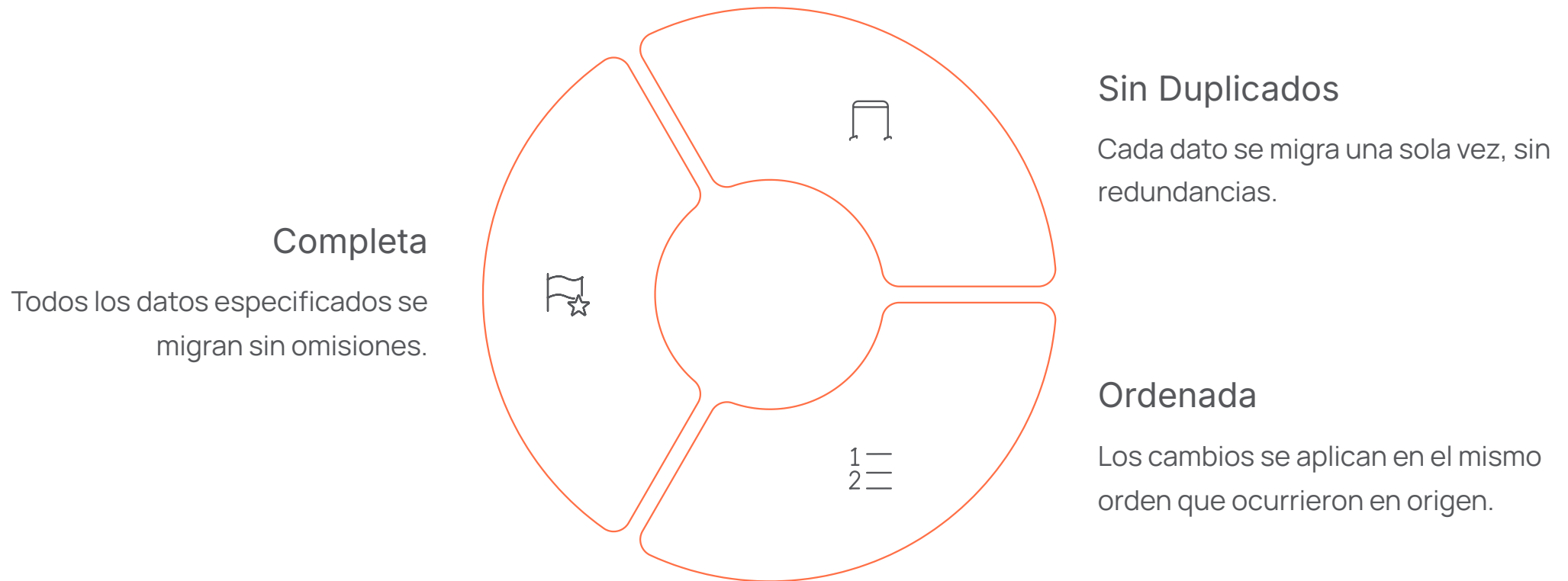
- **1:1** - Mapeo directo
- **n:1** - Consolidación
- **1:n** - Distribución
- **n:m** - Redistribución

Modos de Operación

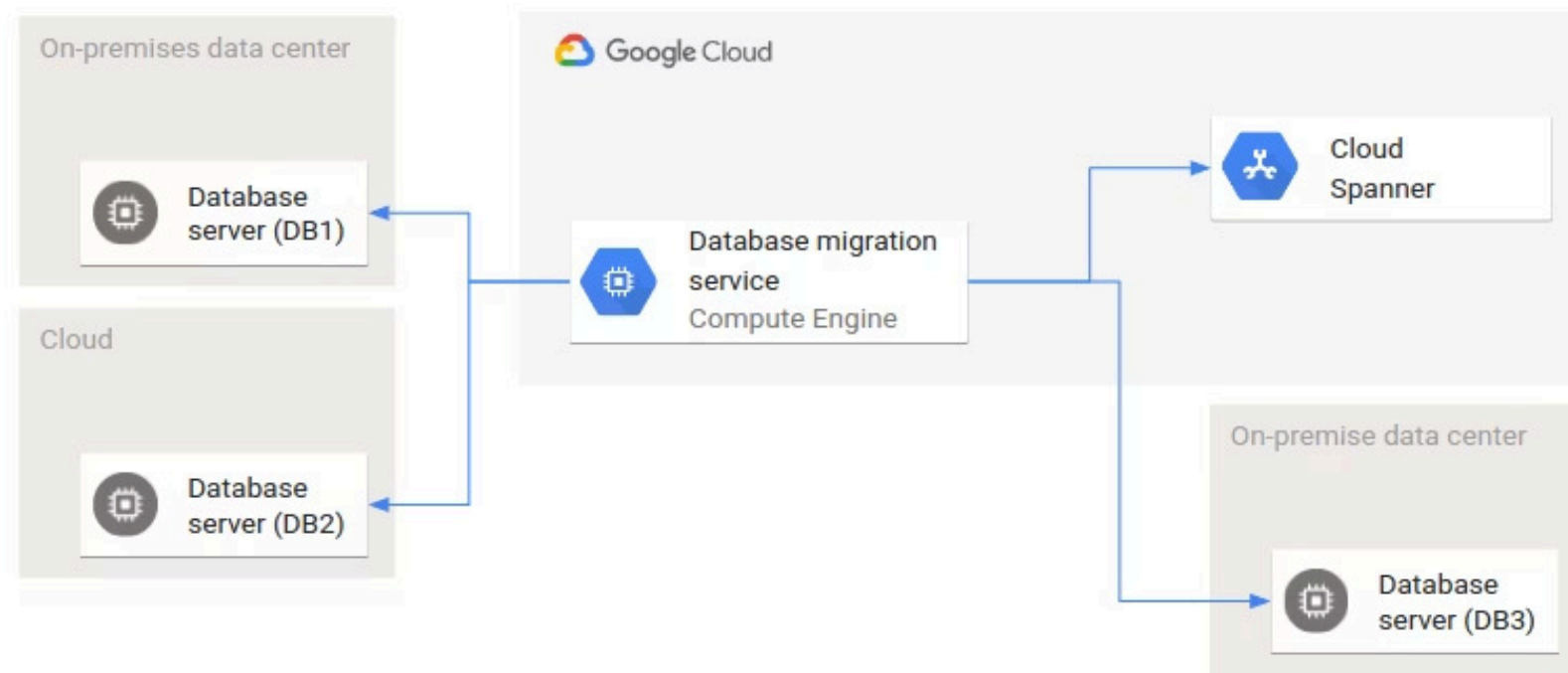
- **Activa-Pasiva:** Solo origen modificable
- **Activa-Activa:** Ambas modificables (requiere resolución de conflictos)

Coherencia en la Migración

Una migración exitosa debe garantizar que el estado de datos sea equivalente entre origen y destino.



Arquitectura de Implementación



La migración puede ocurrir entre bases de datos ubicadas en cualquier entorno: diferentes nubes o centros de datos locales.



Sistema de Migración de Bases de Datos

01

Extracción de Datos

CDC basada en registros de transacciones o consultas diferenciales con marcas de tiempo.

02

Modificación de Datos

Transformación de tipos, estructuras y valores; enriquecimiento y filtrado.

03

Direccionamiento

Selección basada en esquemas o enrutamiento basado en contenido.

04

Persistencia

Almacenamiento del estado para recuperación ante fallas.

Ejemplos de sistemas: Database Migration Service, Striim, Debezium, tcVision, Cloud Data Fusion



Modificación de Datos Durante la Migración



Transformación de Datos

Conversión de tipos, reestructuración de esquemas y modificación de valores.



Correlación y Enriquecimiento

Combinación de datos de múltiples fuentes y agregación de datos de referencia.



Reducción y Filtrado

Eliminación de atributos innecesarios o elementos de datos completos.

Ejemplos de Transformación

Tipo de Datos

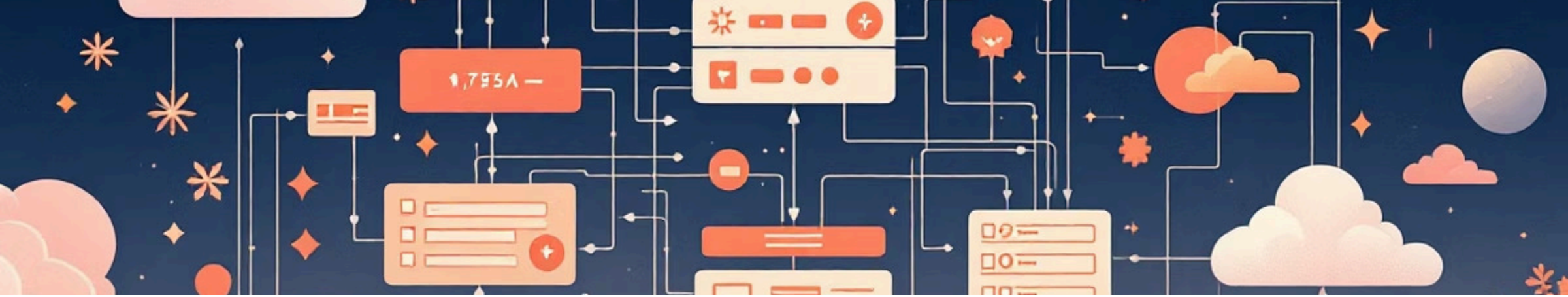
Marca de tiempo → String

Estructura

Tabla única → Dos tablas relacionadas

Valor

Zona horaria local → UTC



Consideraciones Adicionales



Manejo de Errores

Las fallas no deben causar pérdida de datos ni procesamiento desordenado. La integridad debe preservarse ante cualquier tipo de error.



Escalabilidad

La velocidad de transferencia debe superar la velocidad de actualizaciones del origen para completar la migración en tiempo razonable.



Alta Disponibilidad

El sistema debe reconectarse a bases de datos promovidas tras fallas zonales y recuperar desde el último punto coherente.



Recuperación ante Desastres

Conmutación por error a regiones diferentes con continuidad desde la última transferencia coherente de datos.

Errores Comunes a Evitar



Incumplimiento del Orden

El procesamiento paralelo puede alterar el orden de transacciones. Debe mantenerse el orden original de las confirmaciones.



Incumplimiento de Coherencia

Cambios con la misma marca de tiempo deben insertarse en una sola transacción para evitar estados incoherentes temporales.



Datos Faltantes o Duplicados

La conmutación por error puede causar flashback. El sistema debe reconocer y recuperarse de esta situación.



Transacciones Locales

Escribir directamente en ambas bases sin coordinación puede generar incoherencias por fallas parciales o desorden.

Fases de la Migración de Bases de Datos

Una migración exitosa sigue un ciclo de vida bien definido, desde la planificación inicial hasta la optimización post-migración.



Planificación y Evaluación

Análisis de requisitos, identificación de riesgos y diseño de la estrategia de migración.



Preparación e Implementación

Preparar la base de datos de origen, ejecutar la transferencia y transformación de datos.



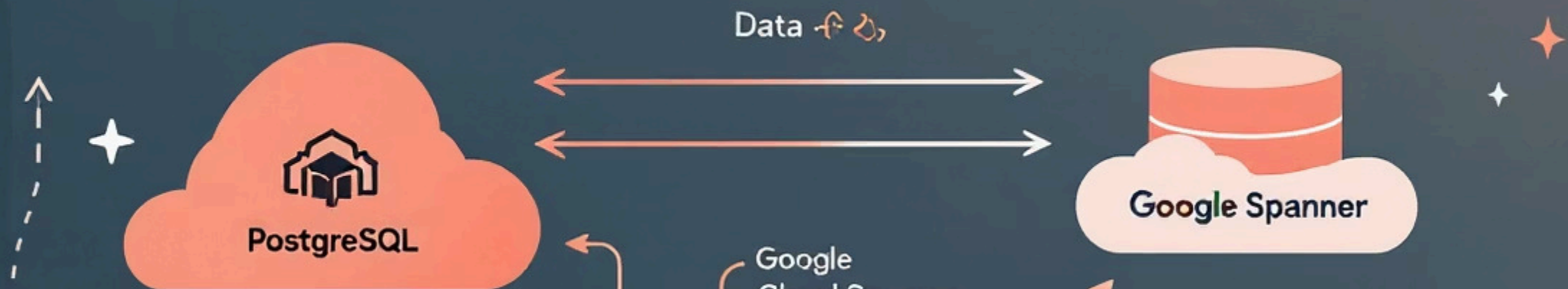
Pruebas y Validación

Verificación exhaustiva de la integridad de los datos, rendimiento y funcionalidad.



Corte y Optimización

Redireccionamiento del tráfico de la aplicación y ajustes post-migración para el rendimiento.



Ejemplo: Migración de PostgreSQL a Spanner

Migrar de PostgreSQL a Spanner permite a las aplicaciones aprovechar una base de datos distribuida globalmente con alta disponibilidad y consistencia fuerte, ideal para cargas de trabajo de escala empresarial.



Origen: PostgreSQL

Base de datos relacional de código abierto, robusta y madura, pero que puede enfrentar desafíos de escalabilidad horizontal extrema.



Destino: Cloud Spanner

Base de datos distribuida globalmente con consistencia fuerte, escalabilidad ilimitada y alta disponibilidad para aplicaciones críticas.



Consideraciones

Adaptación de esquema, mapeo de tipos de datos, reescritura de SQL y planificación de la transferencia de datos (inicial y continua).

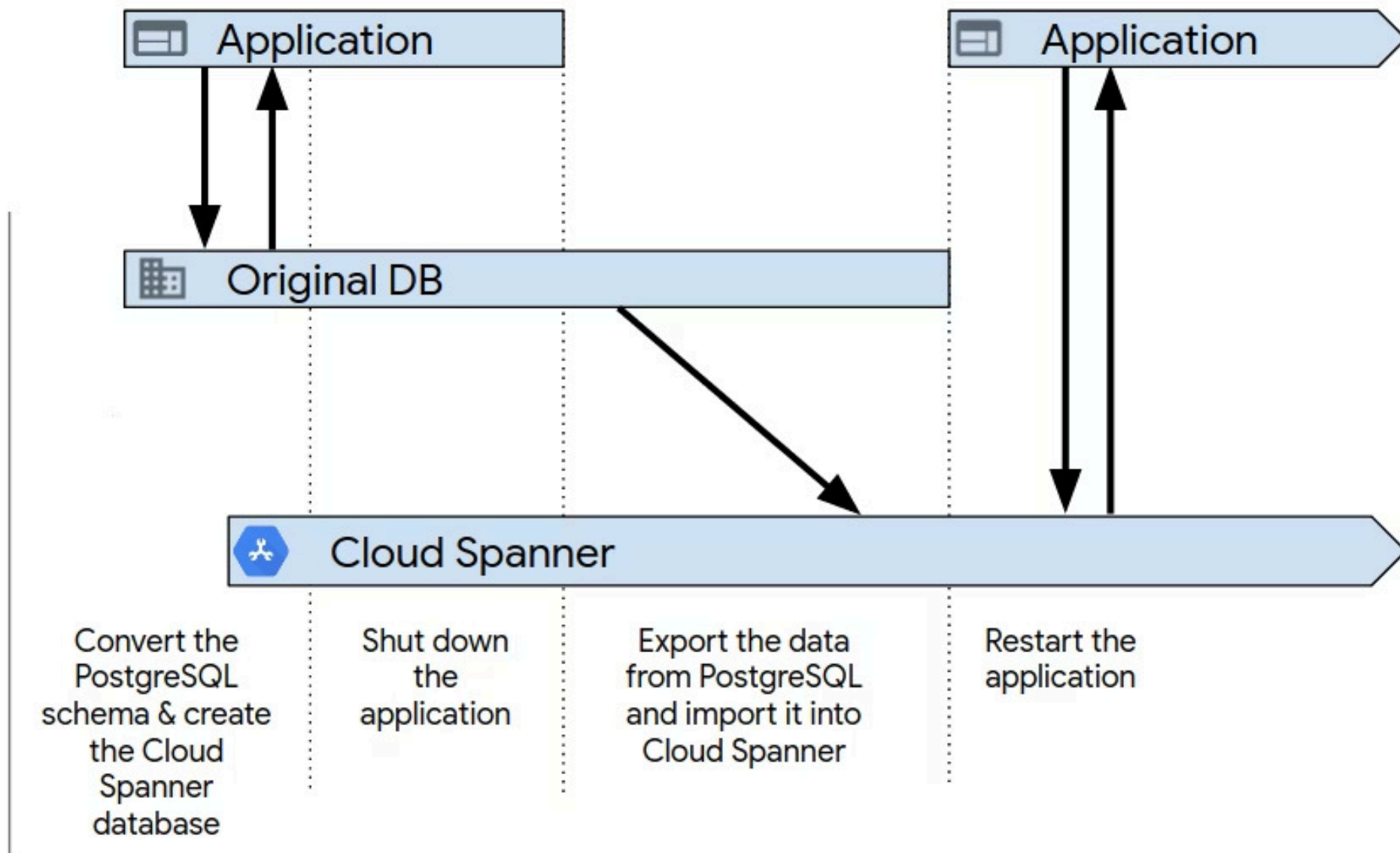


Beneficios

Escalabilidad automática, baja latencia para usuarios globales, continuidad del negocio y operaciones simplificadas a gran escala.

Ejemplo: Migración de PostgreSQL a Spanner

Timeline of Offline Migration Process



Bibliografía:

Documentacion Oficial Google Cloud



 Google Cloud



Migración de bases de datos: Conceptos y principios (parte 1) | Cloud Ar...

Presenta conceptos, principios, terminología y arquitectura de la migración de bases de datos con tiempo de inactividad casi nulo desde las instalaciones locales o desde...



 Google Cloud



Migración de bases de datos: Conceptos y principios (parte 2) | Cloud Ar...

En este documento, se analiza la configuración y la ejecución del proceso de migración de la base de datos, incluidas las situaciones de errores. Este documento...



 Google Cloud



Migra de PostgreSQL a Spanner (dialecto de GoogleSQL) | Google Cloud

En esta página, se proporcionan instrucciones para migrar una base de datos de PostgreSQL de código abierto a Spanner.