

# 10-Características avanzadas de bases de datos relacionales



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA  
BARCELONATECH



## Secciones

1. Necesidad de arquitectura
2. Consideraciones de la capa de base de datos
3. Amazon RDS
4. Amazon DynamoDB
5. Controles de seguridad de la base de datos
6. Migración de datos a las bases de datos de AWS

## Demostración

- Réplicas de lectura y copias de seguridad automatizadas de Amazon RDS

## Laboratorios

- Laboratorio guiado: Creación de una base de datos de Amazon RDS
- Laboratorio de desafíos: Migración de una base de datos a Amazon RDS



**Evaluación de conocimientos**

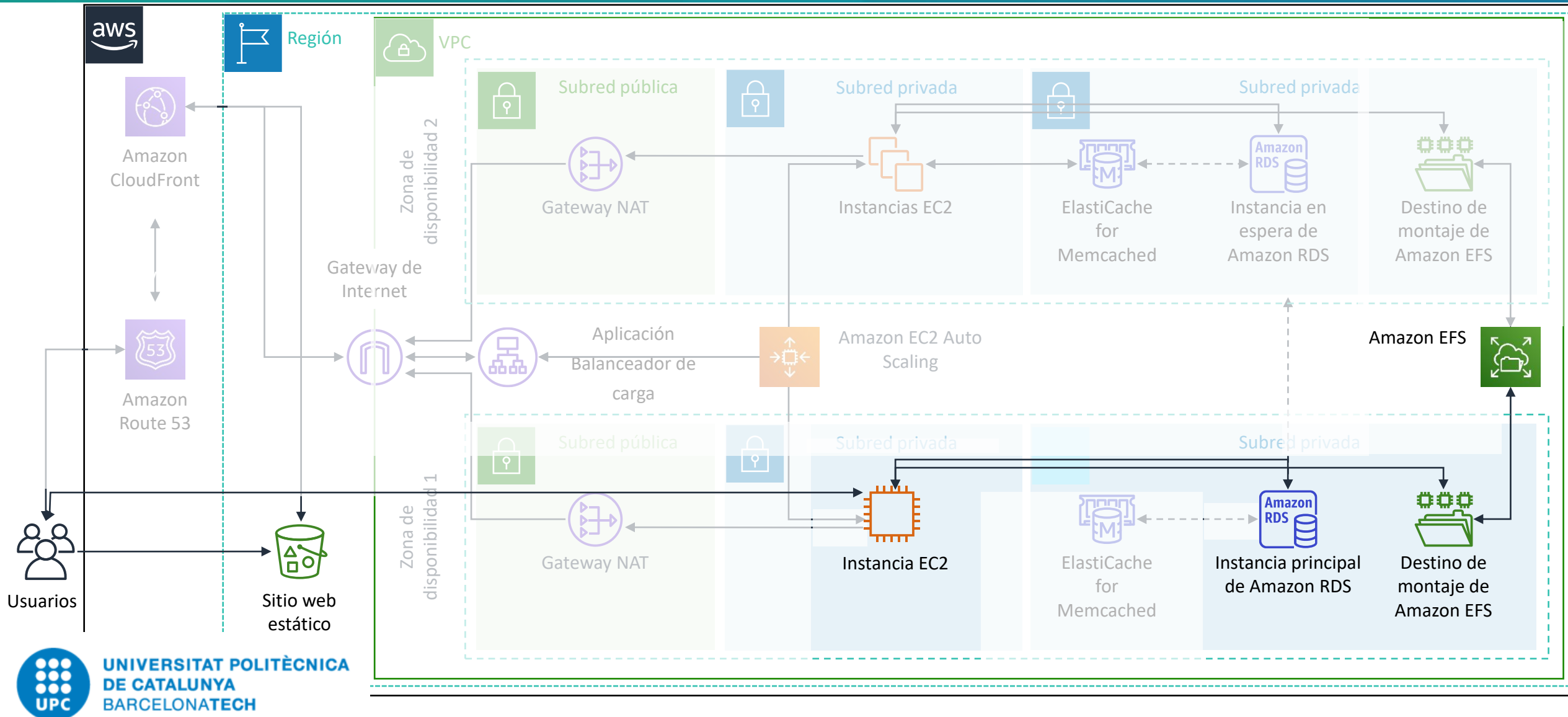
Una vez finalizado este módulo, debería ser capaz de lo siguiente:

- Comparar tipos de bases de datos
- Diferenciar entre servicios administrados y no administrados
- Explicar cuándo usar Amazon Relational Database Service (Amazon RDS)
- Explicar cuándo usar Amazon DynamoDB
- Describir los controles de seguridad de la base de datos
- Describir cómo migrar datos a bases de datos de Amazon Web Services (AWS)
- Implementar un servidor de base de datos

Incorporación de una capa de base de datos

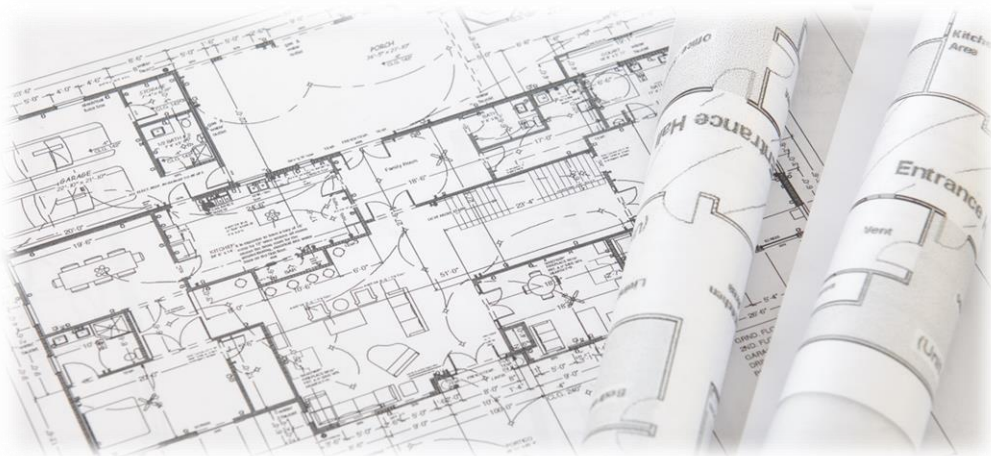
# Sección 1: Necesidad de arquitectura

# Bases de datos como parte de una arquitectura mayor



# Requisito empresarial de la cafetería

La cafetería necesita una solución de base de datos que sea más fácil de mantener y que ofrezca características esenciales como durabilidad, escalabilidad y alto rendimiento.



Incorporación de una capa de base de datos

## Sección 2: Consideraciones de la capa de base de datos

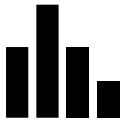
# Consideraciones de la base de datos: escalabilidad



## Escalabilidad



Requisitos de almacenamiento totales



Tamaño y tipo de objeto



Durabilidad

¿Cuánto rendimiento se necesita?

¿La solución elegida será capaz de escalar en forma ascendente más adelante si resulta necesario?





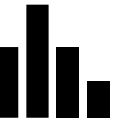
# Consideraciones de la base de datos: requisitos de almacenamiento



Escalabilidad



**Requisitos de almacenamiento totales**



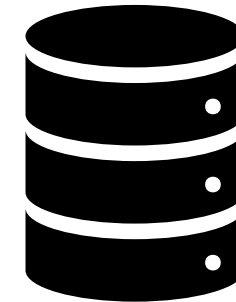
Tamaño y tipo de objeto



Durabilidad

¿Qué tan grande debe ser la base de datos?

¿Necesitará almacenar GB,  
TB o petabytes de datos?



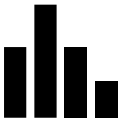
# Consideraciones de la base de datos: tamaño y tipo de objetos



Escalabilidad



Requisitos de almacenamiento totales

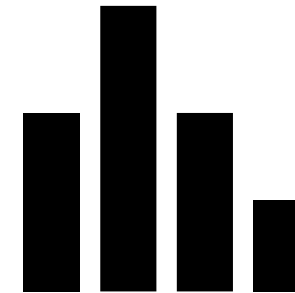


**Tamaño y tipo de objeto**



Durabilidad

¿Es necesario almacenar estructuras de  
datos simples, objetos de datos grandes o  
ambos?



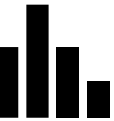
# Consideraciones de la base de datos: durabilidad



Escalabilidad



Requisitos de almacenamiento totales



Tamaño y tipo de objeto



**Durabilidad**

¿Qué nivel de **durabilidad, disponibilidad y capacidad de recuperación de datos** se requiere?

¿Se aplican obligaciones normativas?



Ahora que revisó las consideraciones clave, considere las dos categorías de opciones de base de datos disponibles:

## Relacional

Ejemplos tradicionales

Microsoft SQL Server  
Base de datos de Oracle  
MySQL

## No relacional

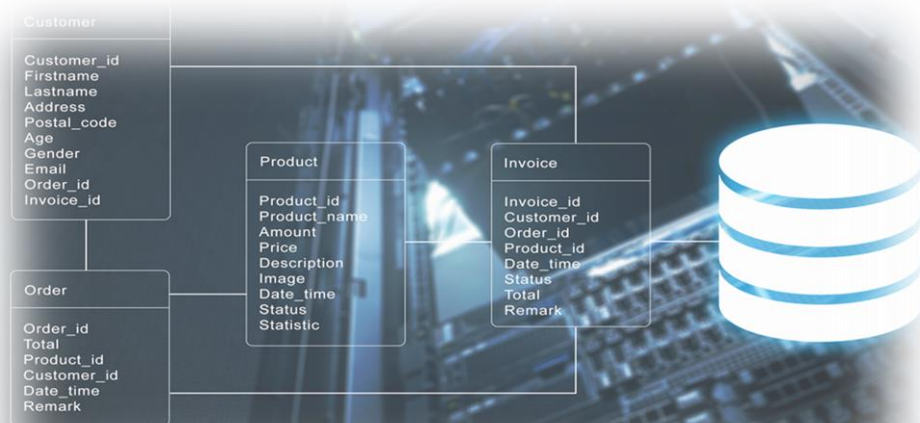
Ejemplos tradicionales

MongoDB  
Cassandra  
Redis

# Tipo de base de datos relacional

## Beneficios:

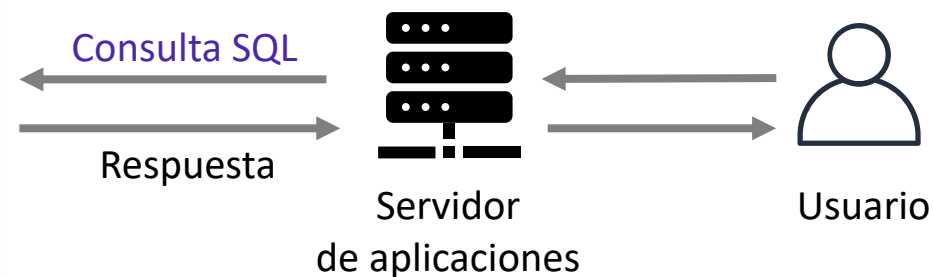
- Facilidad de uso
- Integridad de los datos
- Almacenamiento de datos reducido
- Lenguaje común (lenguaje de consulta estructurada o SQL)



Sistema de administración de bases de datos relacionales (RDBMS)

## La base de datos relacional es ideal cuando sucede lo siguiente:

- Necesita reglas estrictas de esquema, conformidad con ACID y cumplimiento de la calidad de los datos.
- No necesita una capacidad extrema de lectura o escritura.
- No necesita un rendimiento extremo.
  - Un RDBMS puede ser la mejor solución de menor esfuerzo



# Tipo de base de datos no relacional

## Beneficios

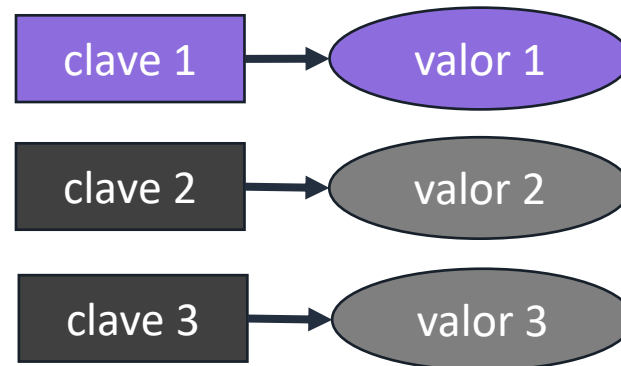
- Flexibilidad
- Escalabilidad
- Alto rendimiento
- API altamente funcionales

La base de datos no relacional es ideal cuando sucede lo siguiente:

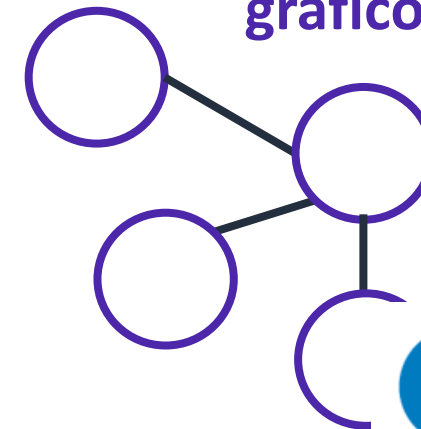
- La base de datos debe escalarse **de forma horizontal** para manejar un volumen de datos masivo.
- Los datos no se ajustan bien a los esquemas tradicionales.
- Las tasas de lectura o escritura superan las que se pueden sustentar económicamente por medio de un RDBMS tradicional.

Modelos de ejemplo

## Clave-valor



## gráfico



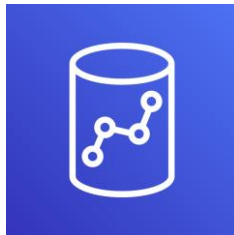
# Opciones de base de datos de Amazon

*Existen más opciones de bases de datos; estas opciones son ejemplos comunes*

## Bases de datos relacionales



Amazon  
RDS

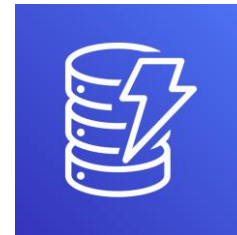


Amazon  
Redshift



Amazon  
Aurora

## Bases de datos no relacionales



Amazon  
DynamoDB



Amazon  
ElastiCache



Amazon  
Neptune

*Céntrese en este módulo*

# Aprendizajes clave de la Sección 2



- Cuando elija una base de datos, tenga en cuenta la **escalabilidad**, los **requisitos de almacenamiento**, el **tipo y tamaño de los objetos** que se almacenarán y los **requisitos de durabilidad**.
- **Las bases de datos relacionales** tienen reglas estrictas de esquema, proporcionan integridad de los datos y admiten SQL.
- **Las bases de datos no relacionales** se escalan de forma horizontal, proporcionan mayor escalabilidad y flexibilidad, y funcionan bien para datos semiestructurados y no estructurados.



Incorporación de una capa de base de datos

## Sección 3: Amazon RDS

Relacional

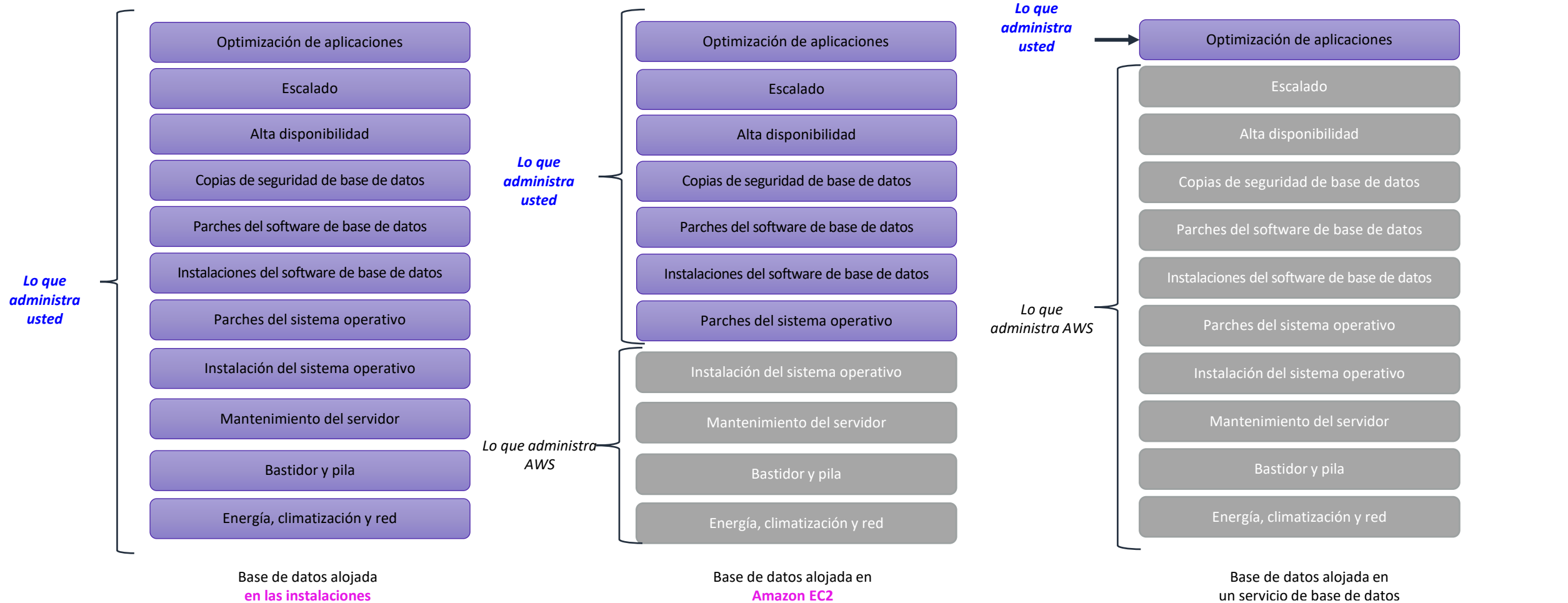


Amazon  
RDS

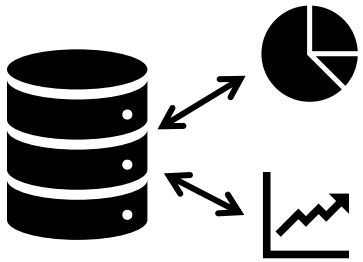
Amazon RDS es un **servicio**  
de base de datos relacional  
**completamente administrada.**



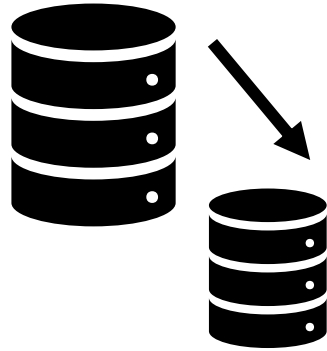
# Ventaja de los servicios de bases de datos administradas de AWS



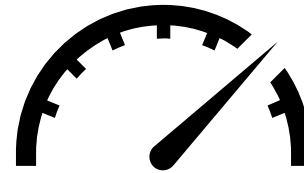
# Características de Amazon RDS



Patrón de acceso  
Transaccional,  
análisis ligero



Volumen de datos  
Rango de pocos TB



Eficacia  
Rendimiento de medio  
a alto, baja latencia



Casos de uso empresariales  
Transaccional  
OLAP

# Amazon RDS: usos y tipos de base de datos



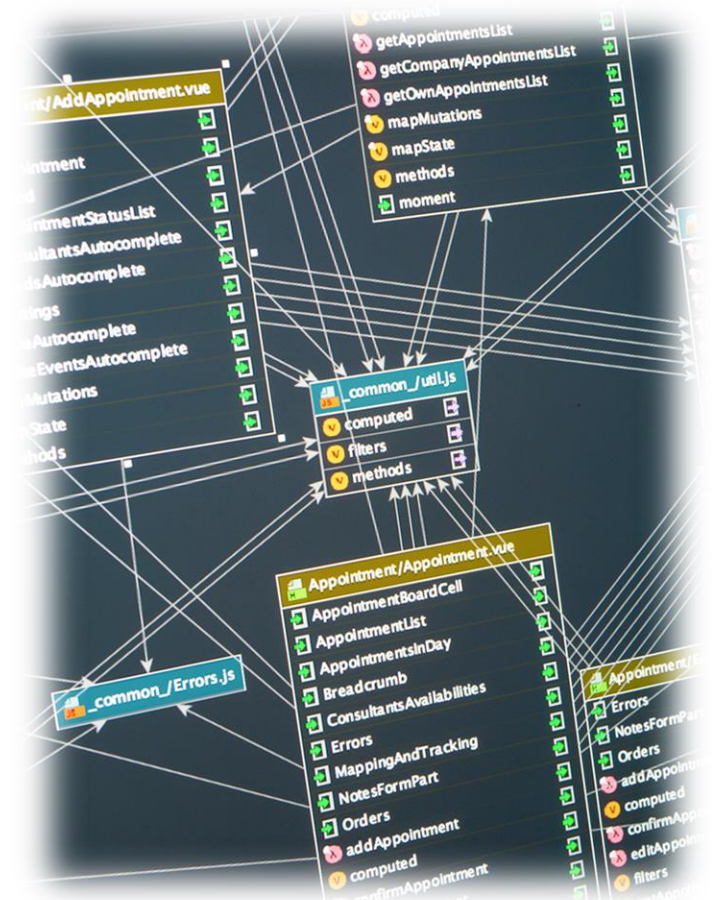
Amazon  
RDS

Funciona bien para aplicaciones que cumplen con lo siguiente:

- Tienen datos más complejos
- Necesitan combinar y unir conjuntos de datos.
- Necesitan reglas de sintaxis aplicadas.

Se admiten seis tipos de bases de datos:

- |                        |              |
|------------------------|--------------|
| • Microsoft SQL Server | • PostgreSQL |
| • Oracle               | • Aurora     |
| • MySQL                | • MariaDB    |

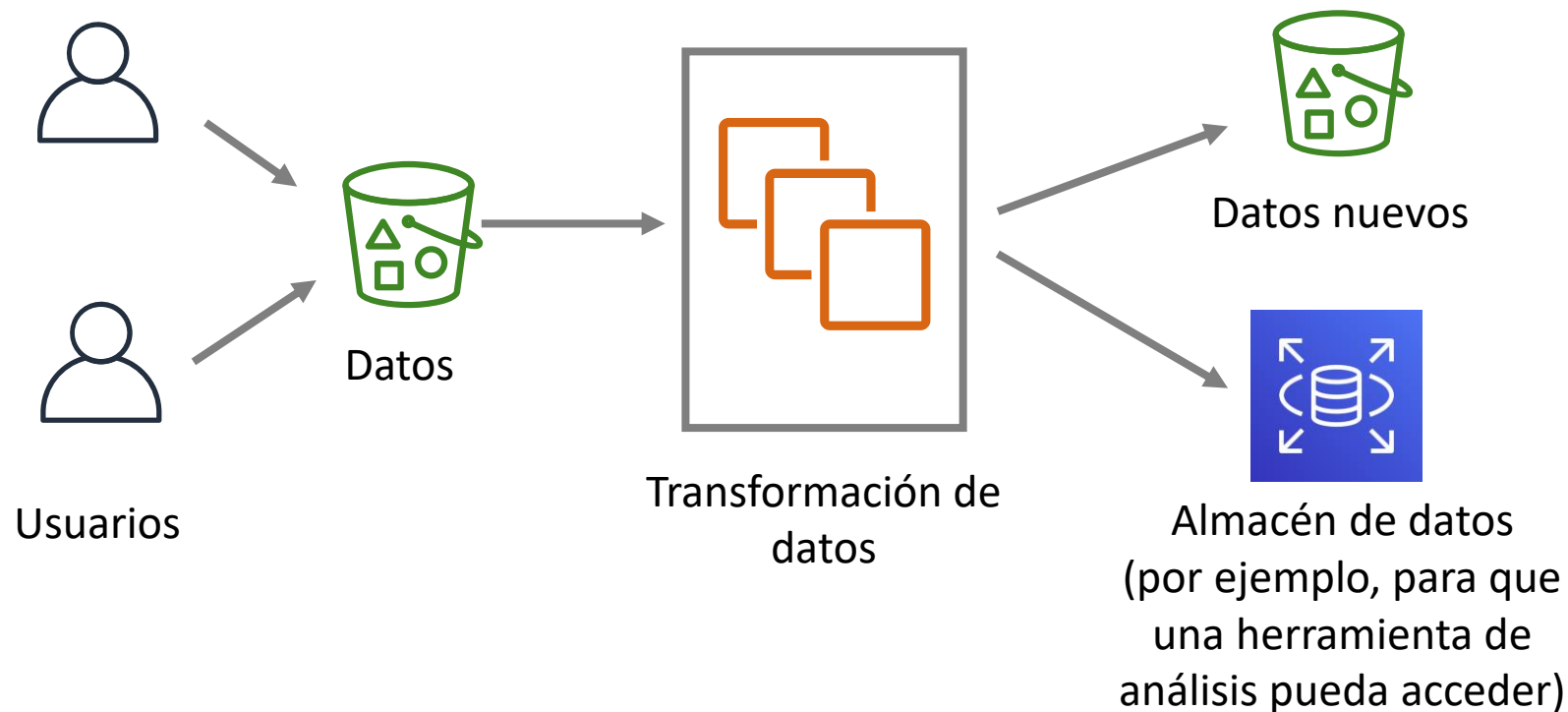


# Tamaño de la instancia de base de datos

	Familia T	Familia M	Familia R
Tipo	Instancias ampliables	Instancias de uso general	Instancias con optimización de memoria
Tamaño	1 vCPU/1 GB de RAM a 8 vCPU/32 GB de RAM	2 vCPU/8 GB de RAM a 96 vCPU/384 GB de RAM	2 vCPU/16 GB de RAM a 96 vCPU/768 GB de RAM
Redes	Rendimiento moderado	Alto rendimiento	Alto rendimiento
Carga de trabajo ideal	Menor o variable	Mucho consumo de CPU	Recuentos altos de conexiones con uso intensivo de consultas
Aspectos destacados	T3 puede ampliarse por encima de la referencia por un cargo adicional	M5 ofrece hasta 96 vCPU	R5 ofrece hasta 96 vCPU/768 GiB de RAM

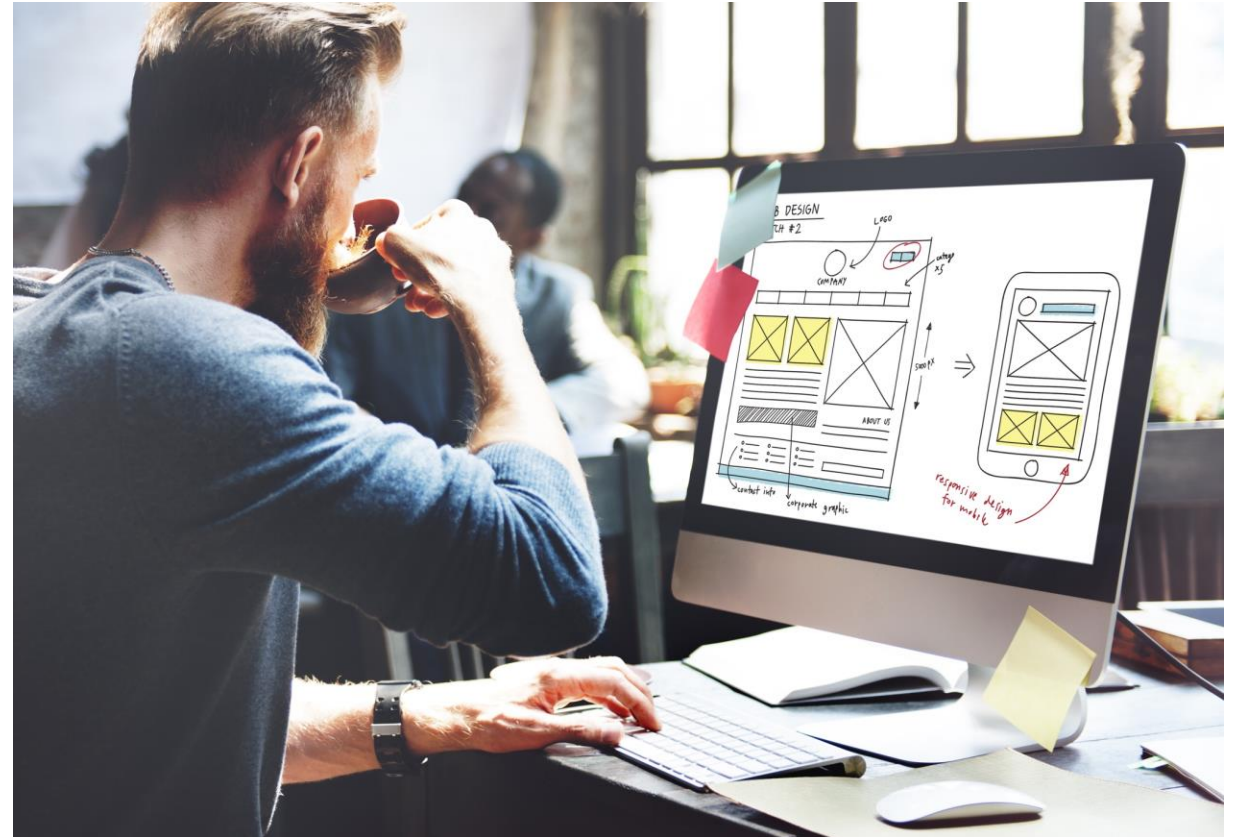
# Amazon RDS: ejemplo de caso de uso

## Análisis





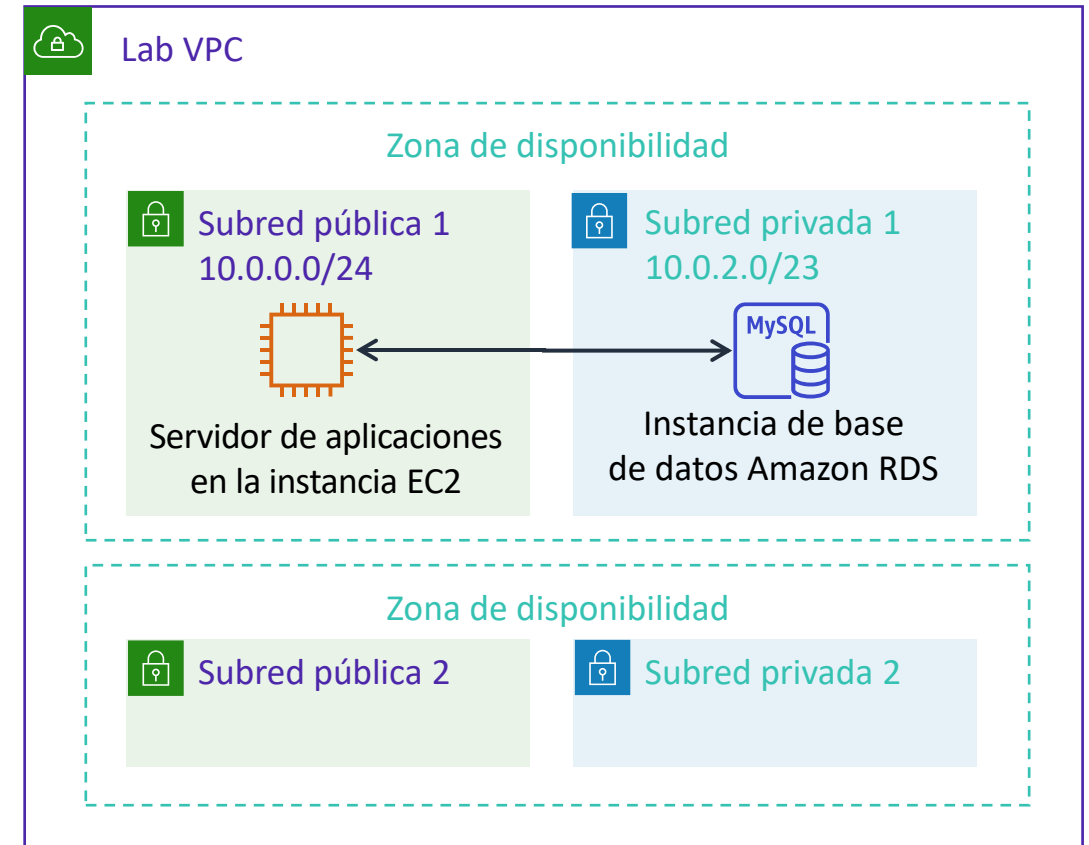
# Módulo 5 - Laboratorio guiado: Creación de una base de datos de Amazon RDS






# Laboratorio guiado: Tareas

1. Creación de una base de datos de Amazon RDS
2. Configuración de la comunicación de aplicaciones web con una instancia de base de datos





## Comenzar Módulo 5 - Laboratorio guiado: Creación de una base de datos de Amazon RDS



# Análisis del laboratorio guiado: aprendizajes clave



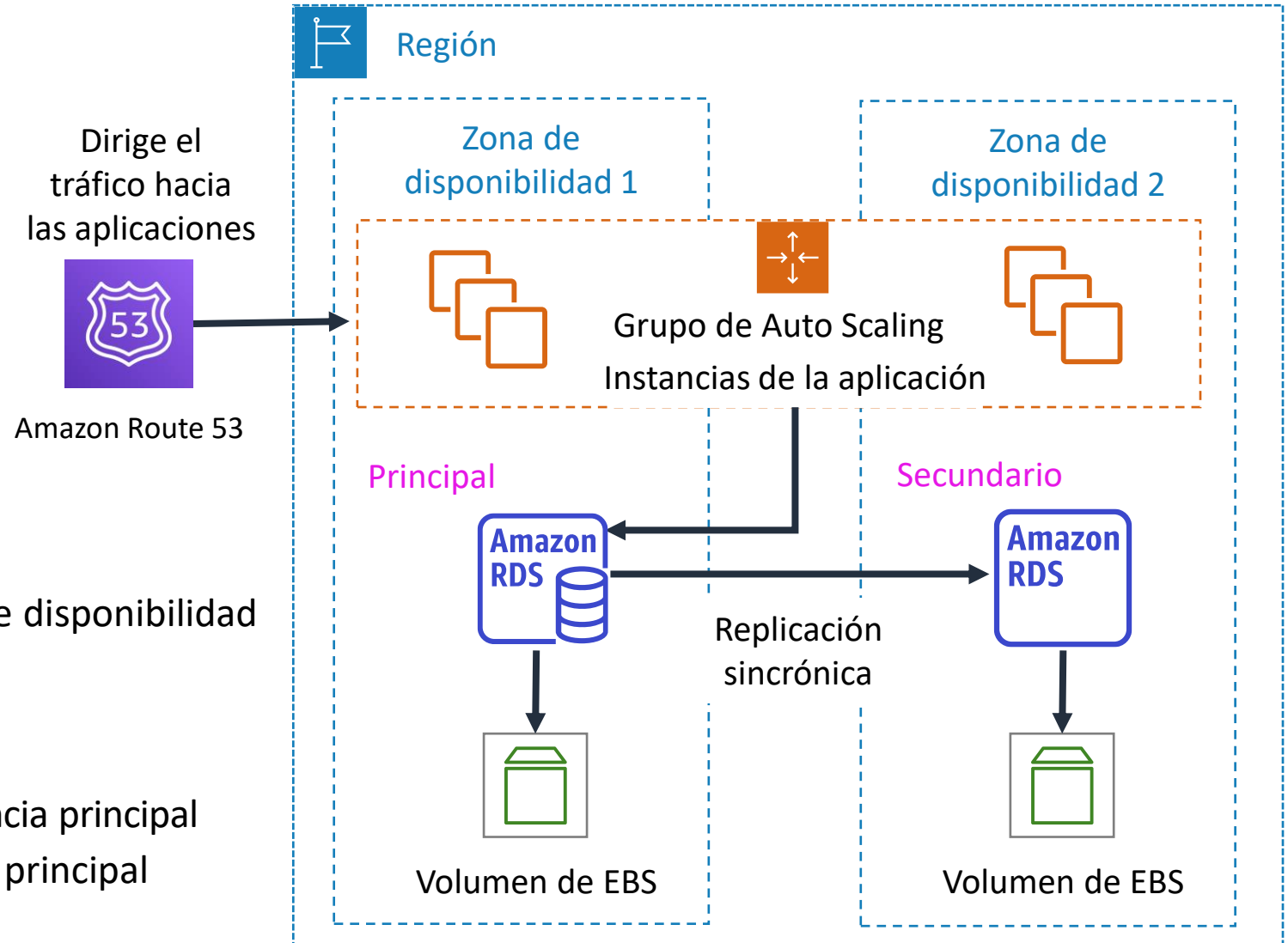
# Implementación Multi-AZ para alta disponibilidad

## Beneficios

- Durabilidad mejorada
- Mayor disponibilidad
- La conmutación por error al modo de espera se produce automáticamente

## Condiciones de la conmutación por error automatizada

- Pérdida de disponibilidad en la zona de disponibilidad principal
- Pérdida de conectividad de red con la instancia principal
- Error de unidad informática en la instancia principal
- Error de almacenamiento en instancia principal



# Réplicas de lectura para el rendimiento

## Beneficios

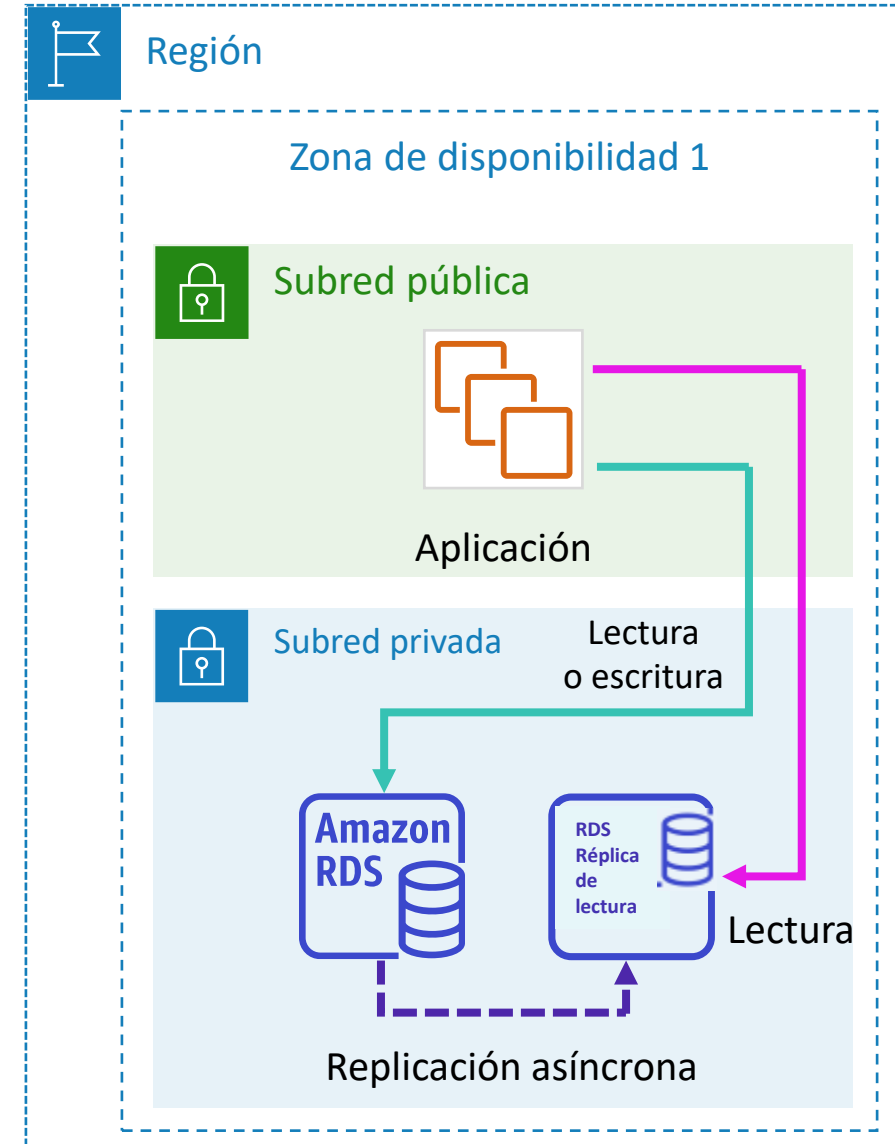
- Rendimiento mejorado
- Mayor disponibilidad
- Diseñado para garantizar la seguridad

## Compatible con

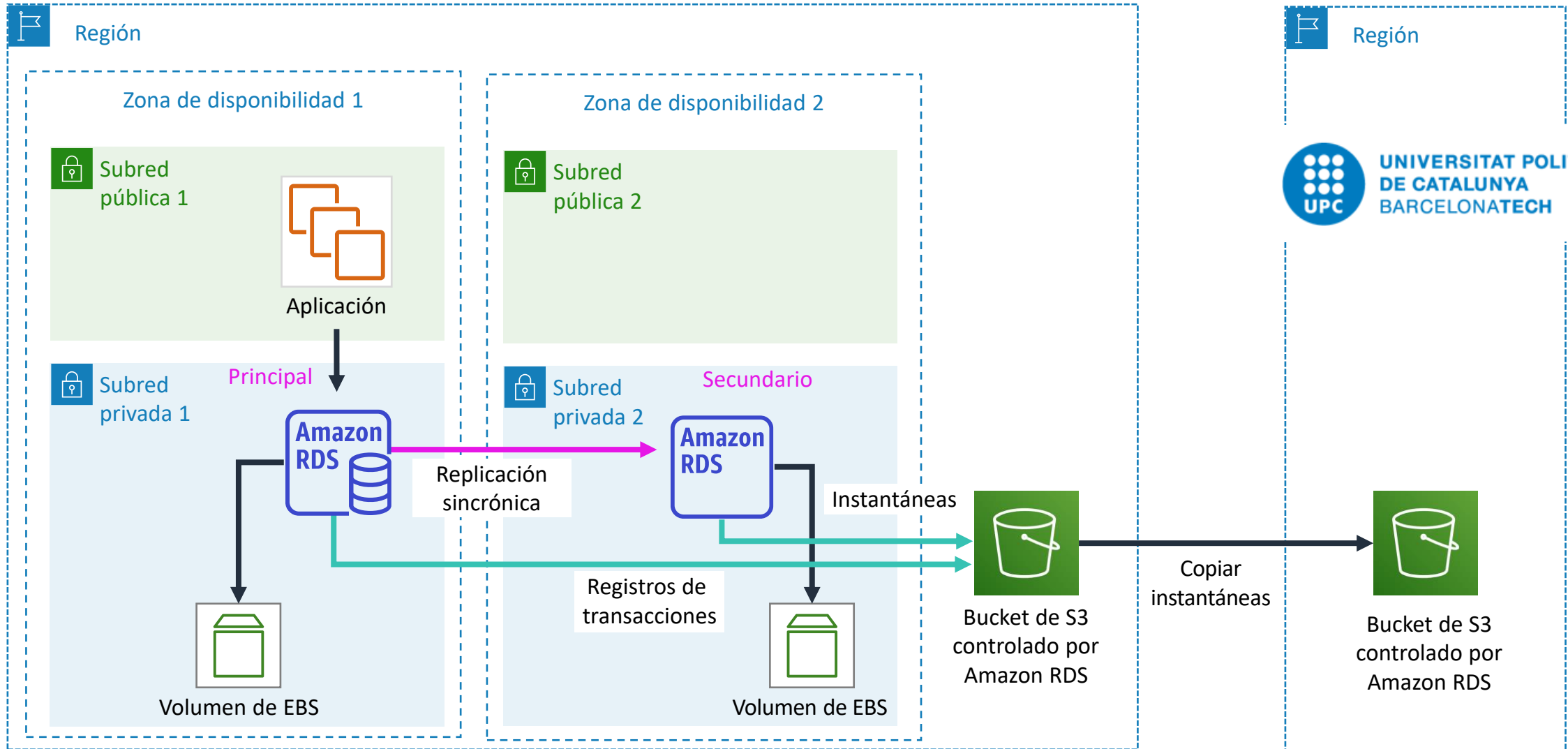
- MySQL
- MariaDB
- PostgreSQL
- Oracle

## Límites

- Cinco réplicas de lectura por instancia principal
- Para obtener una consistencia estricta de lectura después de escritura, lea desde la instancia principal



# Solución de copia de seguridad de Amazon RDS





# Demostración: Réplicas de lectura y copias de seguridad automatizadas de Amazon RDS





Amazon  
Aurora

Amazon Aurora es un motor de base de datos relacional completamente administrado compatible con MySQL y PostgreSQL.

- Se usa para el procesamiento de transacciones en línea (OLTP).
- Ofrece hasta cinco veces más rendimiento que MySQL\*.
- Ofrece hasta tres veces más rendimiento que PostgreSQL\*.
- Replica datos de seis maneras en tres zonas de disponibilidad.
- Requiere pocos cambios en su aplicación existente.

\* Los detalles del punto de referencia están disponibles para [MySQL](#) y [PostgreSQL](#).





Amazon  
Redshift

Amazon Redshift es un servicio de **almacenamiento de datos**.

- Se utiliza para el procesamiento de análisis en línea (OLAP).
- Almacena conjuntos muy grandes de datos.
  - Almacena datos muy estructurados y de acceso frecuente en Amazon Redshift.
  - También puede almacenar exabytes de datos estructurados, semiestructurados y no estructurados en Amazon S3.

# Aprendizajes clave de la Sección 3



- Los servicios de bases de datos administradas de AWS manejan las tareas de administración para que pueda centrarse en sus aplicaciones.
- Amazon RDS admite [Microsoft SQL Server](#), [Oracle](#), [MySQL](#), [PostgreSQL](#), [Aurora](#) y [MariaDB](#).
- Las [implementaciones Multi-AZ de Amazon RDS](#) ofrecen alta disponibilidad con conmutación por error automática.
- Puede tener hasta cinco [réplicas de lectura](#) por base de datos principal para mejorar el [rendimiento](#) de Amazon RDS.
- [Amazon Aurora](#) es un motor de base de datos relacional completamente administrada compatible con MySQL y PostgreSQL.
- Amazon Redshift es una oferta de base de datos relacional para [el almacenamiento de datos](#).

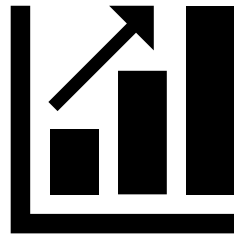
Incorporación de una capa de base de datos

## Sección 4: Amazon DynamoDB



Amazon  
DynamoDB

Un servicio de base de datos **no relacional** completamente administrada **de clave-valor y documentos**.



Eficacia  
a cualquier escala

Capacidad extrema de  
escalado horizontal



Sin servidor

Programación dirigida  
por eventos  
(informática  
sin servidor)



Lista para el uso  
empresarial

Cifrado, controles de  
acceso, copias de  
seguridad

# Características de Amazon DynamoDB



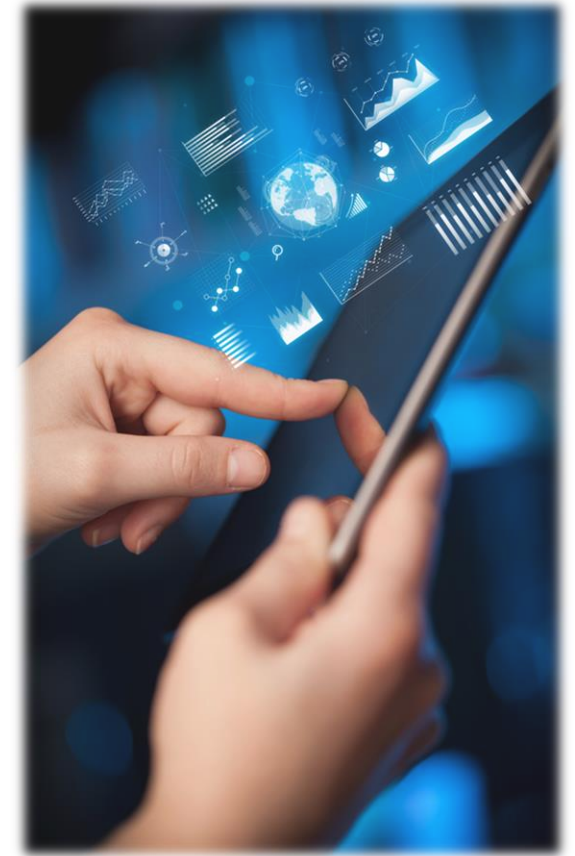
Amazon  
DynamoDB

Funciona bien para aplicaciones que cumplen con lo siguiente:

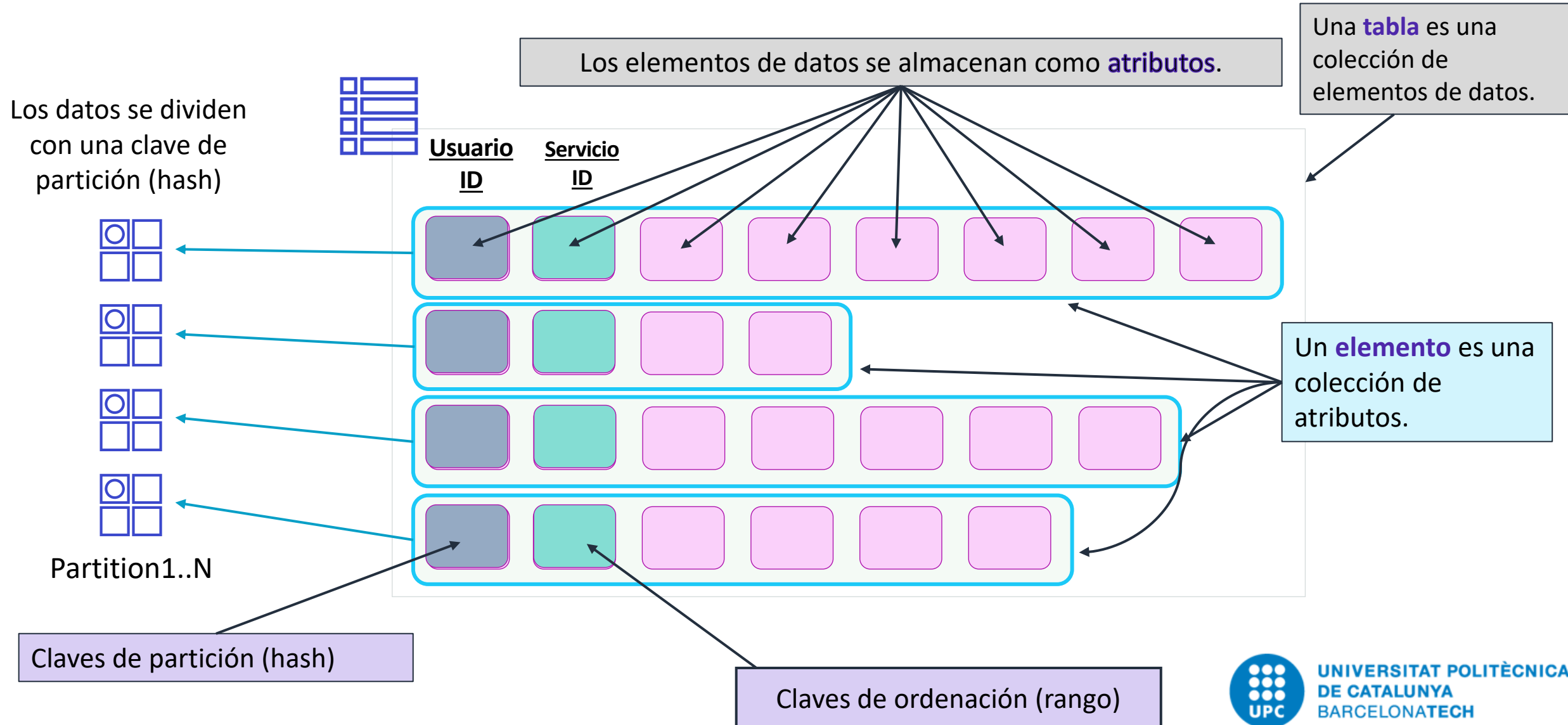
- Tienen grandes volúmenes de datos simples (rango alto de TB).
- Deben escalarse rápidamente.
- No necesitan combinaciones complejas.
- Requieren un rendimiento ultraalto y baja latencia.

Características clave

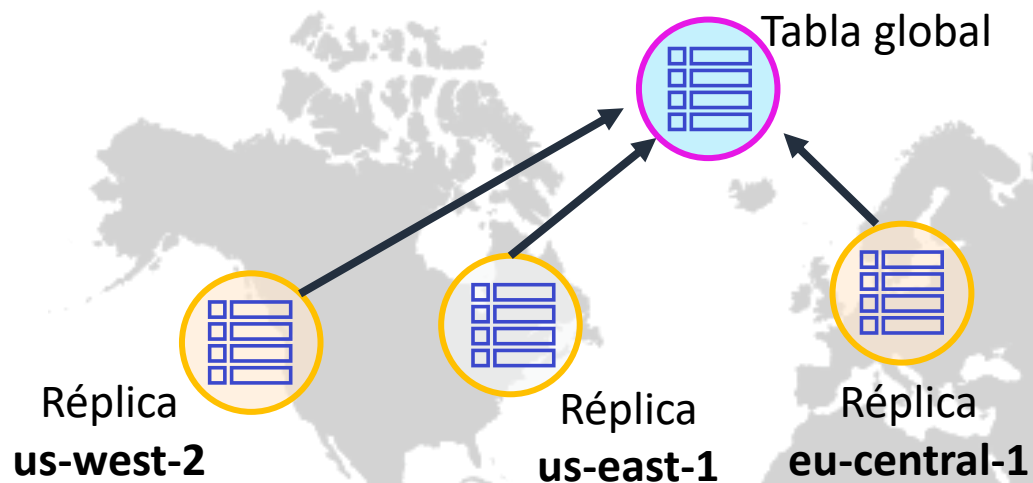
- Tablas NoSQL
- Los elementos pueden tener atributos diferentes
- Almacenamiento de caché en memoria
- Compatibilidad con picos de más de 20 millones de solicitudes por segundo



# Modelo de datos de Amazon DynamoDB



# Tablas globales de Amazon DynamoDB



Rápido rendimiento de lectura o escritura local para aplicaciones globales.

Las **tablas globales** proporcionan una base de datos con varios nodos maestros y de varias regiones.



## Marcadores y puntuaciones

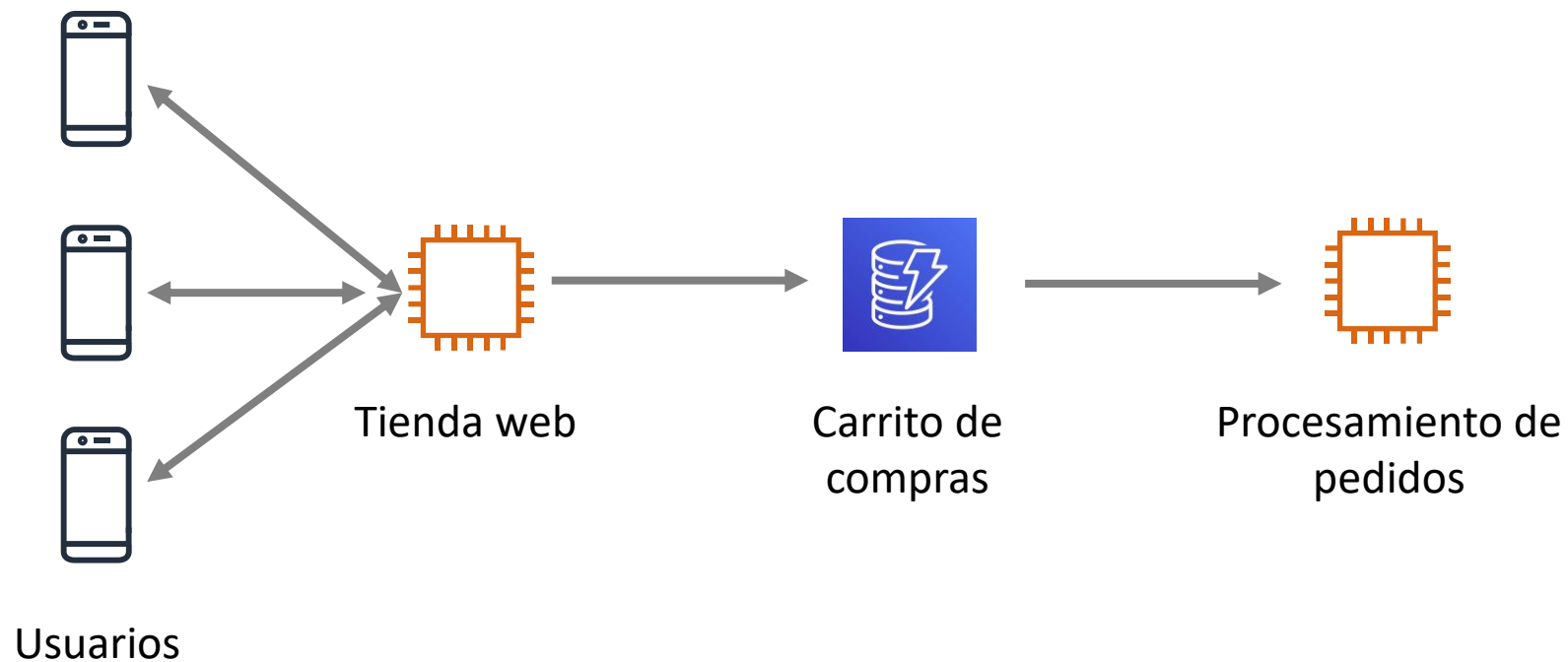


GameScores

UserId	GameTitle	TopScore	TopScoreDataTime	Victorias	Derrotas	
"101"	"Galaxy Invaders"	5842	"2015-09-15:17:24:31"	21	72	...
"101"	"Meteor Blasters"	1000	"2015-10-22:23:18:01"	12	3	...
"101"	"Starship X"	24	"2015-08-31:13:14:21"	4	9	...
"102"	"Alien Adventure"	192	"2015-07-12:11:07:56"	32	192	...
"102"	"Galaxy Invaders"	0	"2015-09-18:07:33:42"	0	5	...
"103"	"Attack Ships"	3	"2015-10-19:01:13:24"	1	8	...
"103"	"Galaxy Invaders"	2317	"2015-09-11:06:53:00"	40	3	...
"103"	"Meteor Blasters"	723	"2015-10-19:01:13:24"	22	12	...
"103"	"Starship X"	42	"2015-07-11:06:53:00"	4	19	...
...	...	...	...	...	...	...

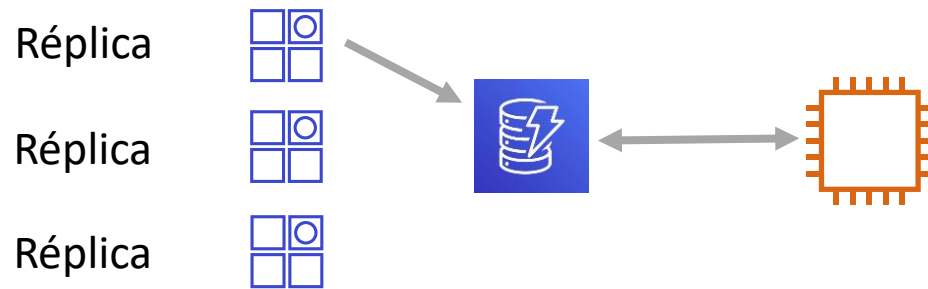


## Datos temporales (carrito en línea)



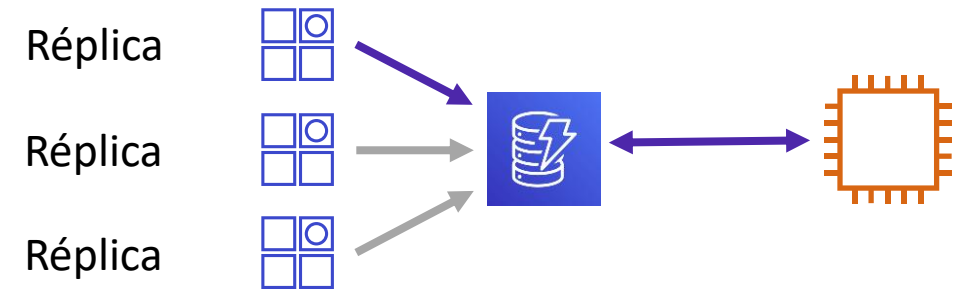
# Opciones de consistencia en Amazon DynamoDB

## Consistencia **final**



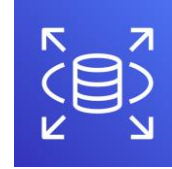
La configuración **predeterminada**. En general, todas las copias de los datos alcanzan la consistencia en **un segundo**.

## Consistencia **alta**

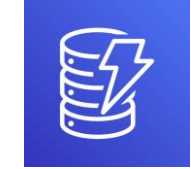


Esta característica es opcional. Utilícela para aplicaciones que requieren que todas las lecturas devuelvan un resultado que refleje todas las escrituras antes de la lectura.

Debate de la clase:  
¿Qué base de datos  
debería usar  
la cafetería?



Amazon  
RDS



Amazon  
DynamoDB



# Aprendizajes clave de la sección 4



- Amazon DynamoDB es un servicio de base de datos **NoSQL** no relacional completamente administrada **de clave-valor** y documentos.
- DynamoDB **no tiene servidor**, proporciona **escalado horizontal** extremo y **baja latencia**.
- **Las tablas globales** de DynamoDB garantizan que los datos se repliquen en varias regiones.
- DynamoDB proporciona **consistencia final** de forma predeterminada (en general, es totalmente consistente para lecturas un segundo después de la escritura). **La consistencia alta** también es una opción.

Incorporación de una capa de base de datos

## Sección 5: Controles de seguridad de la base de datos

# Protección de las bases de datos de Amazon RDS

## Recomendaciones

- Ejecute la instancia de RDS en una **nube privada virtual (VPC)**.
  - Proporciona aislamiento de servicios y protección de firewall de IP.
- Utilice las políticas de **AWS Identity and Access Management (IAM)** para la autenticación y el acceso.
  - Los permisos determinan quién puede administrar los recursos de Amazon RDS.
- Utilice **grupos de seguridad** para controlar qué direcciones IP o instancias EC2 pueden conectarse a sus bases de datos.
  - De forma predeterminada, el acceso a la red está desactivado.
- Utilice la **capa de conexión segura (SSL)** para el cifrado en tránsito.
- Utilice el **cifrado** de Amazon RDS en instancias de base de datos e instantáneas para proteger los datos en reposo.
- Utilice las **características de seguridad del motor de base de datos** para controlar quién puede iniciar sesión en las bases de datos de una instancia de base de datos.
- Configure notificaciones de eventos para que se le informe cuando se produzcan eventos importantes de Amazon RDS.



Amazon RDS

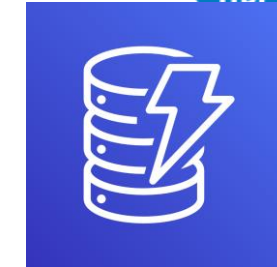


## Recomendaciones

- Utilice los **roles de IAM** para autenticar el acceso
- Utilice las **políticas de IAM**
  - Para definir permisos de acceso detallados para usar API de DynamoDB
  - Defina el acceso al nivel de tablas, elementos o atributos
  - Siga el **principio de concesión de privilegios mínimos**
- Configure **puntos de enlace de la VPC**
  - Evita que el tráfico de conexión atraviese la Internet abierta
  - Las políticas del punto de enlace de la VPC le permiten controlar y limitar el acceso de las API a una tabla de DynamoDB.
- Considere usar **cifrado del lado del cliente**.
  - Cifre datos lo más cerca posible de su origen.



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA  
BARCELONATECH



Amazon  
DynamoDB

Seguridad proporcionada de forma predeterminada.

- **Cifrado en reposo** de todos los datos del usuario almacenados en tablas, índices, secuencias y copias de seguridad
- **Cifrado en tránsito**: todas las comunicaciones hacia y desde DynamoDB y otros recursos de AWS utilizan HTTPS



Incorporación de una capa de base de datos

## Sección 6: Migración de datos a las bases de datos de AWS



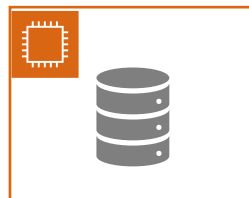
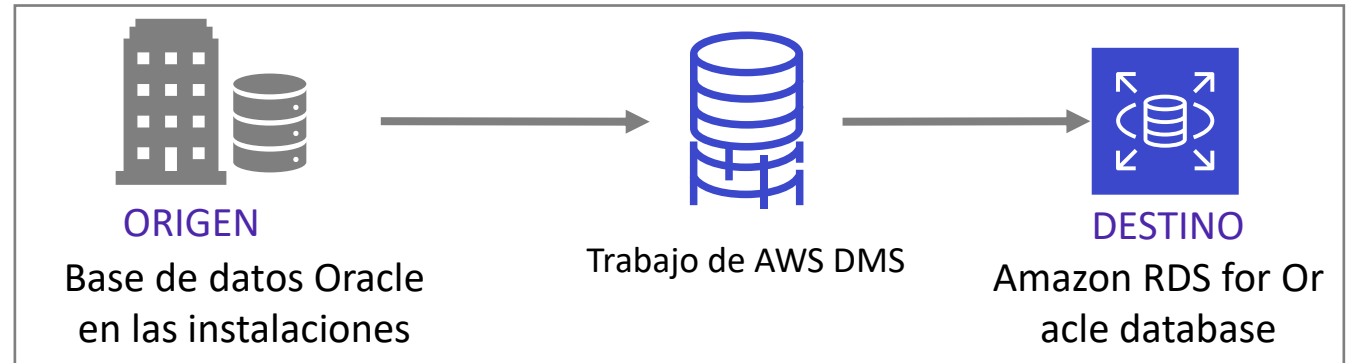
# AWS Database Migration Service



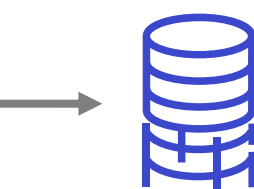
AWS Database  
Migration  
Service (AWS DMS)

- Se utiliza para migrar desde y hacia la mayoría de las bases de datos comerciales y de código abierto.
- Se puede migrar entre bases de datos en Amazon EC2, Amazon RDS, Amazon S3 y en las instalaciones.

Ejemplo de  
migración  
homogénea



Base de datos MySQL  
en instancia EC2



Trabajo de AWS DMS



Amazon Aurora

Ejemplo de migración  
heterogénea

# Características clave de AWS DMS

- Realice migraciones por única vez.
- O bien, realice replicación continua de datos.
  - Ejemplo: configure la replicación continua de datos de una base de datos en las instalaciones a una instancia de RDS
- **AWS Schema Conversion Tool (AWS SCT)** admite cambiar el motor de base de datos entre el origen y el destino.
- Pasos principales típicos de la migración:
  1. Crear una base de datos de destino
  2. Migrar el esquema de base de datos
  3. Configurar el proceso de replicación de datos
  4. Iniciar la transferencia de datos y confirmar la finalización
  5. Cambiar la producción a la nueva base de datos (para migraciones únicas)



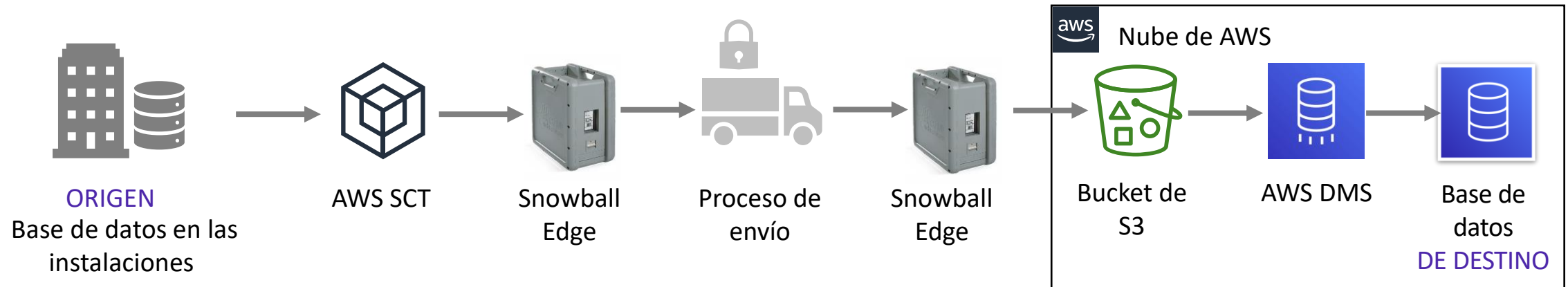
# Uso de AWS Snowball Edge con AWS DMS

Migrar datos no es práctico en los siguientes casos:

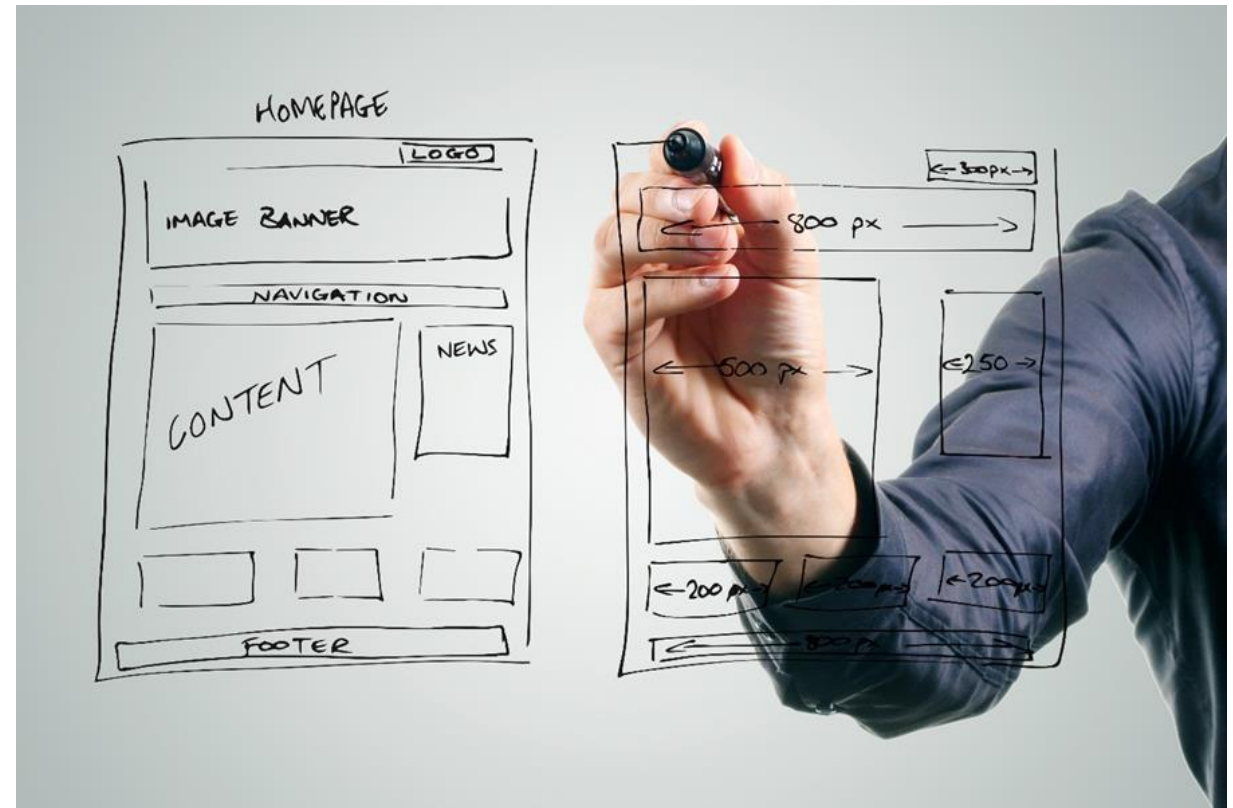
- cuando la base de datos es demasiado grande
- cuando la conexión es demasiado lenta
- cuando hay problemas de privacidad y seguridad

Utilice **AWS Snowball Edge**

- Transferencia de varios terabytes sin usar Internet



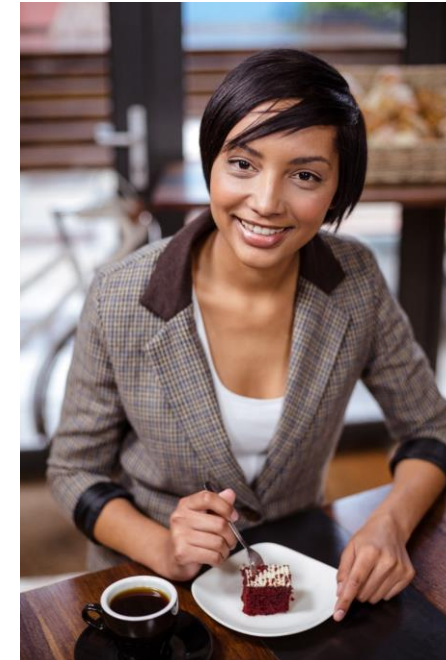
## Módulo 5 - Laboratorio de desafíos: Migrar una base de datos a Amazon RDS





# Necesidad empresarial: una base de datos administrada

La base de datos que se ejecuta en la instancia EC2 se está volviendo difícil de mantener para Sofía y Nikhil.



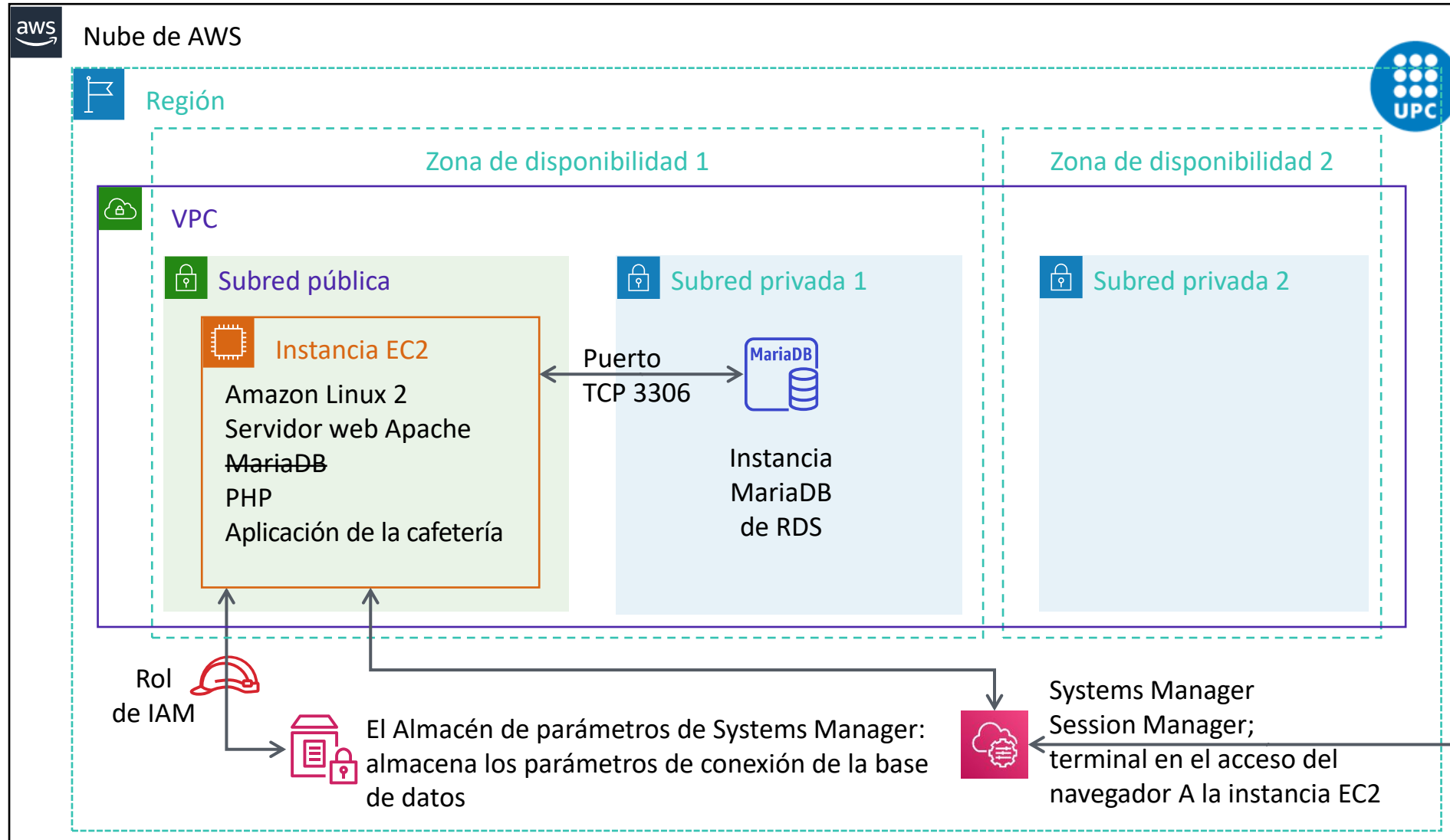
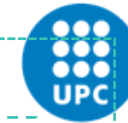
Cuando Olivia visitó la cafetería recientemente, les contó sobre las características de Amazon RDS.


Sofía y Nikhil decidieron migrar la base de datos de la cafetería a Amazon RDS.

1. Creación de una instancia de RDS
2. Análisis de la implementación de las aplicaciones existentes de la cafetería
3. Trabajo con la base de datos en la instancia EC2
4. Trabajo con la base de datos de RDS
5. Importación de los datos a la instancia de la base de datos de RDS
6. Conexión de la aplicación de la cafetería a la nueva base de datos



# Laboratorio de desafíos: Producto final





Comenzar con  
el Módulo 5 -  
Laboratorio de  
desafíos: Migrar  
una base de datos  
a Amazon RDS



# Análisis del laboratorio de desafíos: aprendizajes clave



Incorporación de una capa de base de datos

# Conclusión del módulo

A modo de resumen, en este módulo, aprendió a hacer lo siguiente:

- Comparar tipos de bases de datos
- Diferenciar entre servicios administrados y no administrados
- Explicar cuándo usar Amazon Relational Database Service (Amazon RDS)
- Explicar cuándo usar Amazon DynamoDB
- Describir los controles de seguridad de la base de datos
- Describir cómo migrar datos a bases de datos de Amazon Web Services (AWS)
- Implementar un servidor de base de datos



# Completar la evaluación de conocimientos





# Pregunta del examen de muestra

Una aplicación requiere una base de datos relacional de alta disponibilidad con una capacidad de almacenamiento inicial de 8 TB. La base de datos crecerá 8 GB todos los días. Para admitir el tráfico esperado, se requerirán al menos ocho réplicas de lectura para manejar las lecturas de la base de datos.

¿Qué opción cumplirá estos requisitos?

- A. DynamoDB
- B. Amazon S3
- C. Amazon Aurora
- D. Amazon Redshift

- [Bases de datos de AWS – Página de recursos](#)
- [Guía de introducción de Amazon RDS](#)
- [Prácticas recomendadas de Amazon RDS](#)
- [Preguntas frecuentes sobre Amazon RDS](#)
- [Guía para desarrolladores de Amazon DynamoDB](#)
- [Preguntas frecuentes de Amazon DynamoDB](#)



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA  
BARCELONATECH

# Gracias

© 2020, Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados. Este contenido no puede reproducirse ni redistribuirse, total ni parcialmente, sin el permiso previo por escrito de Amazon Web Services, Inc. Queda prohibida la copia, el préstamo o la venta de carácter comercial. Envíenos sus correcciones o comentarios relacionados con el curso a: [aws-course-feedback@amazon.com](mailto:aws-course-feedback@amazon.com). Si tiene cualquier otra duda, contacte con nosotros en: <https://aws.amazon.com/contact-us/aws-training/>. Todas las marcas comerciales pertenecen a sus propietarios.

