

# Bases de datos

## relacionales

- relaciones predefinidas entre los elementos
- tablas, columnas, filas
- primary key, foreign keys

## Relational Database OLTP

- SQL = Structured Query Language
- Data Integrity : completados, exactitud y consistencia
  - uso de primary keys, foreign keys, «Not Null», constraint, «Unique», «Default», «Check»
- Database transaction: ejecución de un set de instrucciones que forman una «unidad» → commit o rollback
- ACID
  - Atomicidad : ejecución de un todo o nada
  - Consistencia : que respete las reglas, restricciones, constrains, cascades y triggers
  - Isolation : independencia de las diferentes instrucciones
  - Durability : durabilidad de los cambios una vez la transacción se ha realizado correctamente

## AWS RDS

- SQL Server
- Oracle
- MySQL
- PostgreSQL
- Aurora
- MariaDB

## No relacionales

- schemas flexibles
- fácil desarrollo, funcionalidad y escalabilidad
- usado en gran cantidad de modelos de datos (documentos, gráficos, key-value, búsquedas)
- usadas en aplicaciones que requieren mucho volumen de datos, baja respuesta y modelos de datos flexibles, relajando restricciones de consistencia
- database:
  - collection = table
  - document = row
  - key-value = campos
- AWS DynanoDB

## Data Warehousing - OLAP

- repositorio central de información que se puede analizar para tomar decisiones bien informadas
- Amazon RedShift

## Elastic Cache

- Memcached o Redis
- recuperación rápida de memoria
- mejora en tiempos de respuesta y reducción del coste asociado en el escalado de aplicaciones web

## AWS RDS

- <https://aws.amazon.com/products/databases/>
- Amazon Aurora = RDB<sup>1)</sup>
- Amazon Neptune = BDD tipo grafo

## Laboratorio RDS

- Crear instancia MySQL RDS
- Crear EC2

user data

```
#!/bin/bash
yum install httpd php php-mysql -y
yum update -y
chkconfig httpd on
service httpd start
echo "<?php phpinfo();?>" > /var/www/html/index.php
cd /var/www/html
vi connect.php

<?php
$username = "";
$password = "";
$hostname = "";
$dbname = "";

//connection to the database
$dbhandle = mysql_connect($hostname, $username, $password) or die("Unable
to connect to MySQL");
echo "Connected to MySQL using username - $username, password -
$password, host - $hostname<br>";
$selectd = mysql_select_db("$dbname", $dbhandle) or die("Unable to
connect to MySQL DB - check the database name and try again.");
?>
```

## RDS : Backups, Multi AZ, Read Replicas

### backups

- automated
  - ...
  - preferred backup windows
  - durante la copia puede haber degradación de servicio
  - usar MultiAZ para tener replicas automáticas sincronizadas (y además se usa para hacer copias de seguridad sin degradación)
  - las copias se borran al borrar la BDD
- DB Snapshots
  - se guardan en S3
  - no se borran al borrar BDD

## encrypted

- soporta todo tipo de BDD
- si tenemos una BDD sin encriptar, podemos hacer una snapshot, encriptarlo y recuperarlo de ahí

## MultiAZ

- no mejora rendimiento
- redundancia
- no se accede a la «segunda» base de datos, solo vemos una
- está por si cae la principal
- replicación síncrona

## Read Replicas - Scaling

- mejorar lectura, NO ESCRITURA
- solo Aurora, MySQL, MariaDB, PostgreSQL
- usa el sistema de réplica propio del motor que se usa
- escenarios:
  - carga alta en lectura de datos
  - servir tráfico de lectura si cae la principal
- backup automático ha de estar activo
- Aurora hasta 15 réplicas
- MySQL, MariaDB, PostgreSQL hasta 5 réplicas
- replicación asíncrona
- permite réplicas entre regiones
- PostgreSQL no permite...
- Aurora, MySQL, MariaDB permite un segundo nivel de réplica de lectura (réplica de réplica)

## DynamoDB

- replica síncrona en las diferentes AZ de una región (alta disponibilidad y alta durabilidad)
- totalmente gestionada
- modos:
  - eventually consistent read : puede tardar 1 segundo en actualizar la información
  - strongly consistent reads : mucho más rápido (y más caro)
- get/put...
- reparte los datos entre diferentes datacenters (alta disponibilidad por defecto)
- discos SSD
- global tables → DynamoDB a través de regiones
- cada Dynamo tiene unos ratios de lectura/escritura (coste) y por el espacio usado)

- read capacity unit : 1 strongly consistent read o 2 eventually consistent read, hasta un máximo de 4KB
- write capacity unit : 1 escritura por segundo, hasta un máximo de 1KB

## Redshift

- warehouse
- SQL Standard
- a partir de \$0,25/h hasta \$1000/TB/año
- nodo simple, a partir de 160GB
- multi nodo, hasta 128
- performance
  - ...
- pricing:
  - por hora y nodo
  - almacenamiento
  - transferencia de datos
  - datos escaneados ← Redshift Spectrum
- por defecto lo encripta y permite:
  - AES256
  - HMS = Hardware Security Modules
  - KMS

## ElastiCache

- redis y memcached
- diferencias:

|  | Memcached | Redis |
|--|-----------|-------|
| Sub-millisecond latency                          | Yes       | Yes   |
| Developer ease of use                            | Yes       | Yes   |
| Data partitioning                                | Yes       | Yes   |
| Support for a broad set of programming languages | Yes       | Yes   |
| Advanced data structures                         | -         | Yes   |
| Multithreaded architecture                       | Yes       | -     |
| Snapshots  | -         | Yes   |
| Replication                                      | -         | Yes   |
| Transactions                                     | -         | Yes   |
| Pub/Sub  | -         | Yes   |
| Lua scripting                                    | -         | Yes   |
| Geospatial support                               | -         | Yes   |

## RDS Aurora

- optimizado y recomendado por Amazon
- versión SQL propia
- copias en 3 AZ
- ...

1)

Relational DataBase

From:

<https://miguelangel.torresegea.es/wiki/> - **miguel angel torres egea**

Permanent link:

<https://miguelangel.torresegea.es/wiki/info:cursos:itformacion:awsassociate:ddb>

Last update: **15/10/2018 09:47**

