

RDS DYNAMODB S3

RAČUNARSTVO U OBLAKU

FAKULTET TEHNIČKIH NAUKA

UNIVERZITET U NOVOM SADU



Šta je RDS?

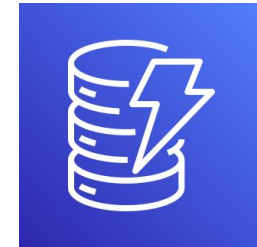
- RDS = Relational Database Service
 - Skalabilna
 - Visoko dostupna
 - **Relaciona baza podataka**
- **Platforma = PaaS**
- Baza podataka pod upravom AWS-a
 - Biramo provajdera i konfiguracije
 - MySQL, PostgreSQL, Oracle, SQL Server i MariaDB
 - AWS podigne EC2 instancu i podesi sve za nas
 - OS, instalacija, održavanje, updates, backups



Amazon Relational Database Service
(Amazon RDS)

Šta je DynamoDB?

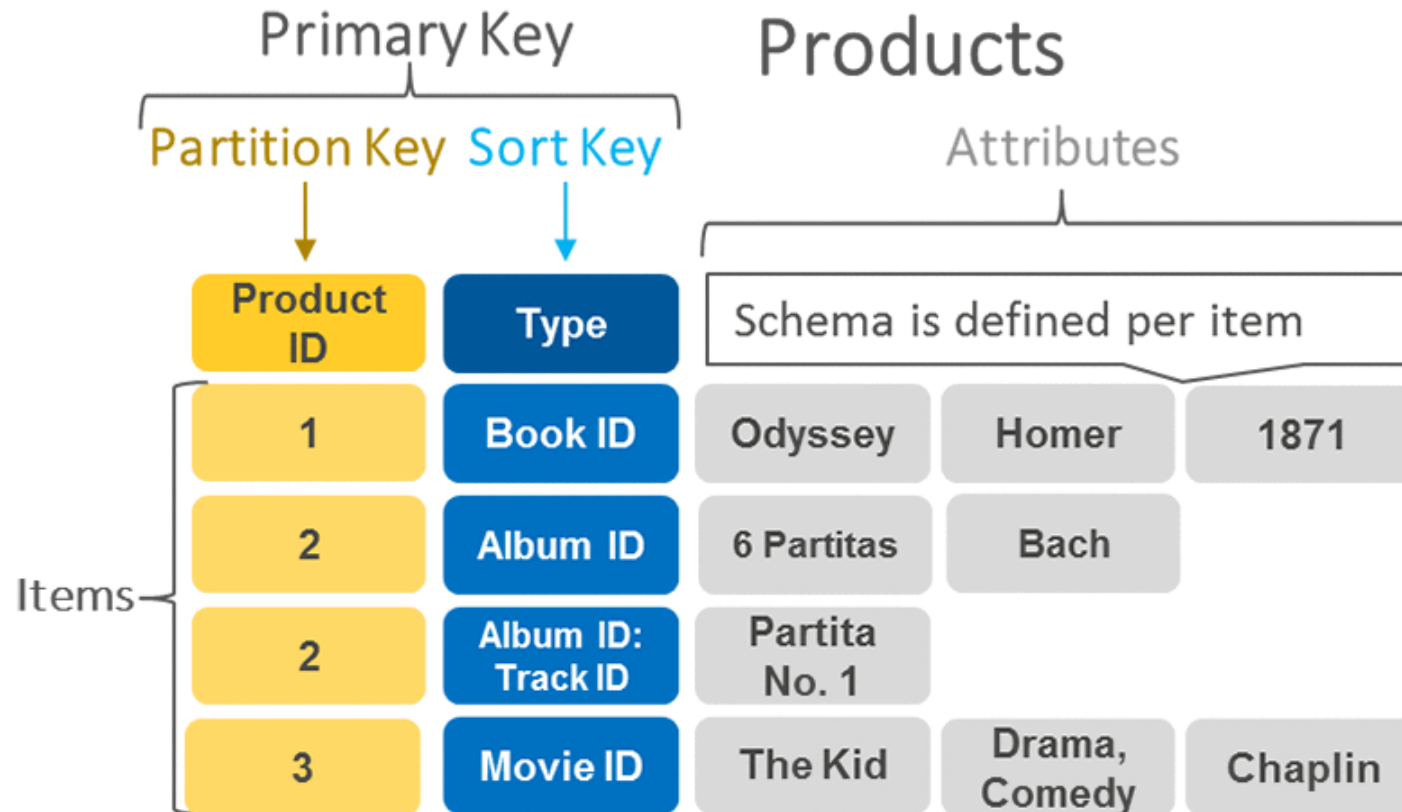
- DynamoDB
 - Skalabilna
 - Visoko dostupna
 - **NoSQL baza podataka**
- **Platforma = PaaS**
- NoSQL baza podataka pod upravom AWS-a
 - Razvijena od strane AWS-a
 - Dizajnirana za horizontalno skaliranje
 - Key-value baza podataka
 - Fleksibilan model podataka uz podršku raznih tipova podataka



Amazon DynamoDB

DynamoDB model podataka

- Table ~ tabela
- Item ~ torka
- Attribute ~ polje
 - Ključ particije i ključ sortiranja
 - Sekundarni indeks
 - Query VS Scan operacija



DynamoDB model podataka

- **Tabela** – okuplja vrednosti zajedničkog tipa
 - Slično relacionim bazama
 - Ali ne definiše strukturu, *schemaless*
- **Item** – stavka, torka
 - Slično torci u relacionim bazama
 - Polja u okviru jedne torke mogu varirati u okviru tabele
- **Attribute** – obeležje
 - Ključ-vrednost, oznaka i vrednost obeležja
 - Skalarni tipovi (numbe, string, boolean, binary, null), dokument tipovi (lista i mapa), setovi; *detaljnije na [linku](#)*
- **Primarni ključ**
 - Minimalno se sastoji od ključa particije, ključ za sortiranje je opcioni
 - Ključ particije ne mora biti jedinstven ako se koristi ključ za soritranje
 - Distribuirana baza podataka -> ključ particije definiše na kojoj „mašini“ će se nalaziti torka
- **Sekundarni indeks**
 - Dodatan indeks za poboljšavanje performansi
 - Ubrzava pretragu i pristup po obeležjima koja nisu deo primarnog ključa
- **Query VS Scan**
 - Query pretražuje samo na osnovu primarnog ključa i sekundarnih indeksa
 - Scan prolazi kroz celu tabelu kako bi našao sve torke koje sadrže određenu vrednost obeležja
 - Dosta sporije

RDS VS DynamoDB

RDS

- PaaS
- Relaciona baza podataka
- Vertikalno skaliranje
- Solidne performanse
- Složen model podataka sa velikim brojem relacija
- Jako definisan model podataka
- Fleksibilni i složeni upiti
- „Šta se skladišti?“ – akcenat na podacima pri modelovanju

DynamoDB

- PaaS
- NoSQL key-value baza podataka
- Horizontalno skaliranje
- Izuzetno visoke performanse
- Jednostavniji model podataka bez relacija
- Fleksibilan model podataka
- Ograničeni i jednostavni upiti
- „Kako ćemo pristupiti podacima?“ – akcenat na akcijama pri modelovanju

Šta je S3?

- S3 = Simple Storage Service
 - Skalabilno
 - Visoko dostupno i dugotrajno
 - **Skladište podataka**
- **Skladište = IaaS**
- Osnovne odlike:
 - Skladište objekata
 - PDF, docx, audio, video, fotografije, binarni podaci...
 - Šta god do 5TB veličine
 - Enkripcija
 - Upravljanje životnim ciklusom skladištenih objekata
 - Verzionisanje



Amazon Simple Storage Service
(Amazon S3)

*<https://www.youtube.com/watch?v=77IMCiiMilo>

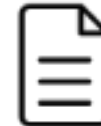
S3 struktura skladišta

- Bucket
 - Kontejner
 - Sve se nalazi u bucket-u
 - Max 100 po nalogu
- Objekat
 - Jedinstveno identifikuje objekat u bucket-u
 - Prefiksima ključa se simulira struktura foldera
- Verzija objekta
 - Opciona
- Metapodaci
 - Tip objekta, veličina, vreme nastanka,...

example-bucket



abc:



Version23

demo/abc:



Version1

demo/a12:



Version2

demo/g1/a:



Version1

a12:



Version11



Amazon RDS



Amazon DynamoDB



Amazon S3



RDS, DynamoDB, S3

Zadaci

1. Konfigurisati pristup RDS-u i deployovati SpringBoot kao Java aplikaciju

- Adresa baze: demo-postgres-db.ccrtaaucml9m.eu-central-1.rds.amazonaws.com
- Ime baze: clouddb
- Kredencijali: postgres/postgres

*materijali

2. (DynamoDB) Upotrebom LocalStack-a:

1. Kreirati tabelu
2. Upisati podatak u tabelu
3. Pročitati podatak
4. Obrisati podatak

*[AWS Docs](#), [AWS CLI Docs](#), [LocalStack Docs](#)

3. (S3) Upotrebom LocalStack-a:

1. Kreirati bucket
2. Upisati objekat
3. Pročitati objekat
4. Obrisati objekat

Domaći: Upotrebom AWS SDK-a po izboru, rešiti drugi i treći zadatak
(preporuka: Python, TypeScript, Ruby su zgodni za pisanje Lambdi)