

# No SQL baze podataka

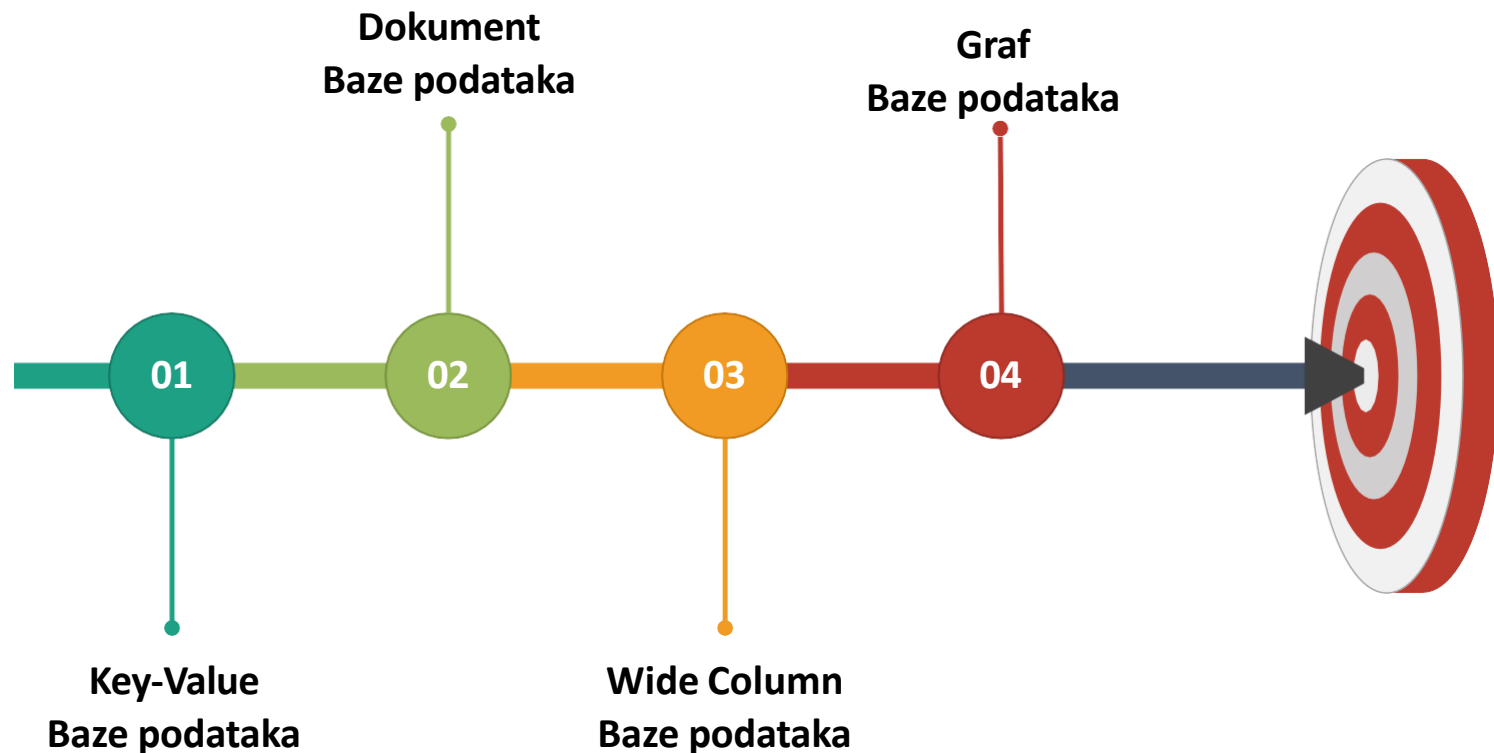
---

Predmet: Administriranje baze podataka

Predavač: dr Dušan Stefanović

# Vrste NoSQL baze podataka

- Pod NoSQL terminom podrazumevaju se različiti modeli baza podataka
- Svaka od njih je specijalizovana za određeni use case



# Key Value baza podataka

- Najjednostavniji tip NoSQL baze podataka
  - Jednostavan data model
- Čuva podatke kao kolekcija **key-value** para
  - **Key 1** → **Value 1**
  - Key 2 → Value 2
  - Key 3 → Value 3
- **Nema tabela, nema atributa** povezanih sa tabelama
- Svaka vrednost u key-value paru može da sadrži potpuno drugačiji tip polja
  - Key 1 → **Akademija strukovnih studija**
  - Key 2 → **062-111-222**
  - Key 3 → **C:\Direktorijum\_Slike\Slika01.jpg**



# Key Value baza podataka

## ■ Operacije

- **Put**(key, value)
- **Get**(key)
  - Jednostavan model → brzi upiti → “daj ključ i dobićeš vrednost”
  - Ne postoji opcija za složene upite (npr. filter po vrednostima)
- **Delete** (key)

## ■ Primena

- **Brz pristup** za veliku količinu **jednostavnih podataka**
- **Novi tip podataka** se jednostavno dodaje kroz key-value par
- Pretraga se zasniva na ključu ili opsegu ključeva
- Nije optimizovana za upite na osnovu vrednosti
  - WHERE name=“vts”

## ■ Use Case

- Najčešće se implementira za **keširanje** (in-memory + key value)

# Key Value baza podataka

- **Website kupovina (shopping cart)**
  - Korisnik se loguje i dodaje proizvode u korpu
  - Radi se uglavnom o jednostavnim podacima
  - High volume – milioni klijenata se igraju sa korpom
  - Key-value baza podataka
    - Key = Klijent ID
    - Values = Lista proizvoda dodata u kolica
- **Key-Value baza podataka je rešenje**



Azure Cosmos DB



# Key Value Šema

## Student

Sid	Ime	Telefon
123	Ana	062123123
321	Marko	064111222



- **Relacija student se u key-value bazi podataka predstavlja šemom**

**Ključ**

**Vrednost**

Student:123:ime = "Ana"

Student:123:Telefon = 062123123

Naziv tabele: Primarni ključ : Naziv kolone = Vrednost

# Key Value - Primeri

## Telefonski imenik

Ključ	Vrednost
Milan	018111222
Jelena	065123432



## ARP tabela

Ključ	Vrednost
172.16.12.3	00:a1:89:cd:1c:07
202.45.123.4	00:25:33:da:4c:01

## Berza

Ključ	Vrednost
123456789	APPL, Buy, 100, 84.47
234567890	CERN, Sell, 50, 52.78

# Dokument baza podataka

- **Dokument Baza podataka**
  - Grupisanje **key-values para** u objekte koji se zovu **dokumenti**
  - Svaki dokument je fajl kodiran u nekom od formata
    - **JSON**, XML, or YAML
  - Svakom dokumentu je dodeljen jedinstven ID, ključ kojim se čita kompletan dokument
- Dokument u **JSON** formatu

```
{  
  "Ime": "Ana",  
  "Telefon": "111-222",  
  "Adresa": "Episkopska 10"  
}
```

JSON dokument je skup key-value parova koji su predstavljeni kao običan tekst





# Dokument baza podataka



- XML i JSON dokumenti koji mogu da se čuvaju u dokument bazi podataka

```
<artist>
  <artistname>Iron Maiden</artistname>
  <albums>
    <album>
      <albumname>The Book of Souls</albumname>
      <datereleased>2015</datereleased>
      <genre>Hard Rock</genre>
    </album>
    <album>
      <albumname>Killers</albumname>
      <datereleased>1981</datereleased>
      <genre>Hard Rock</genre>
    </album>
    <album>
      <albumname>Powerslave</albumname>
      <datereleased>1984</datereleased>
      <genre>Hard Rock</genre>
    </album>
    <album>
      <albumname>Somewhere in Time</albumname>
      <datereleased>1986</datereleased>
      <genre>Hard Rock</genre>
    </album>
  </albums>
</artist>
```

```
{
  '_id' : 1,
  'artistName' : { 'Iron Maiden' },
  'albums' : [
    {
      'albumname' : 'The Book of Souls',
      'datereleased' : 2015,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Killers',
      'datereleased' : 1981,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Powerslave',
      'datereleased' : 1984,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Somewhere in Time',
      'datereleased' : 1986,
      'genre' : 'Hard Rock'
    }
  ]
}
```

# Dokument baza podataka



Način čuvanja istih podataka u **dokument** i **relacionim** bazama podataka

```
{
  '_id' : 1,
  'artistName' : { 'Iron Maiden' },
  'albums' : [
    {
      'albumname' : 'The Book of Souls',
      'datereleased' : 2015,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Killers',
      'datereleased' : 1981,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Powerslave',
      'datereleased' : 1984,
      'genre' : 'Hard Rock'
    }, {
      'albumname' : 'Somewhere in Time',
      'datereleased' : 1986,
      'genre' : 'Hard Rock'
    }
  ]
}
```

ArtistId	ArtistName
1	Iron Maiden
2	Devin Townsend
3	The Wiggles

GenreId	Genre
1	Country
2	Blues
3	Hard Rock

AlbumId	AlbumName	DateReleased	ArtistId	GenreId
1	The Book of Souls	2015	1	3
2	Killers	1981	1	3
3	Powerslave	1984	1	3
4	Somewhere in Time	1986	1	1
5	Ziltoid the Omniscient	2007	2	2

# Dokument baza podataka

- Aplikacija uzima dokument na osnovu **ključa za taj dokument**
  - Nema praznih polja u dokumentu (slučaj kod relacionih baza)
  - Dokument baza podataka **nema fiksnu šemu**
  - Svaki dokument je skup **jedinstvenih polja**
  - Add/remove podataka iz određenih dokumenata **ne utiče** na druge dokumente
- 
- **USE CASE**
    - Sistem za upravljanje sadržajem (Content Management System)
    - Blog platforme
    - Analitika u realnom vremenu
    - E-commerce aplikacije



# Dokument baza podataka

## ■ USE CASE - Katalog proizvoda

{  
  “Proizvod ID”: “1”,  
  “Proizvod Ime”: “Igračka”,  
  “Proizvod Opis”: “Daljinsko upravljanje”,  
  “Uzrast”: “5-10”,  
  “Domet”: “25m”,  
  “Snabdevač-ID”: “200”  
  “Snabdevač-ID”: “Dexico”  
}



{  
  “Proizvod ID”: “222”,  
  “Proizvod Ime”: “Bežična bušilica”,  
  “Proizvod Opis”: “Bušilica za eksperte”,  
  “Voltaža”: “20”,  
  “Broj Obrtaja”: “750”,  
  “Snabdevač-ID”: “300”  
  “Snabdevač-ID”: “Black & Decker”  
}



# Dokument baza podataka - Šema

## Student

Sid	Ime	Telefon
123	Ana	062123123
321	Marko	064111222

- **Relacija student se u dokument bazi podataka predstavlja šemom**
  - Jedan red u relacionoj bazi podataka predstavlja jedan dokument

```
{  
  _id : 123,  
  Ime : "Ana",  
  Telefon: 0621233123,  
}
```

**Student\_123** dokument

```
{  
  _id : 321,  
  Ime : "Marko",  
  Telefon: 064111222,  
}
```

**Student\_321** dokument

# Dokument baza podataka

---



Amazon DynamoDB



**Couchbase**



Azure Cosmos DB



Firestore



CouchDB

# Wide Column baza podataka

- **Wide Column Baza podataka**

- Zove se još i:
  - *Column-oriented* baza podataka
  - *Columnar* database
  - *Column-family* baza podataka
- “**row**” je lista vrednosti povezana sa istom **kolonom**
- Dizajnirana je da podrži veliki broj kolona (atributa)

- Relacione baze podataka su **row-oriented**



- Svaki red ima ključ **row key**
- Svako polje (**field**) unutar reda čuva se zajedno u tabeli
- Jednostavno dodavanje novih **redova (rows)** u tabeli
- Kolona predstavlja atribut ili polje unutar reda

	<b>Id</b>	<b>Ime</b>	<b>Prezime</b>	<b>Telefon</b>	
	1	Ana	Mitić	111-222	
	2	Ognjen	Ilić	333-444	
	3	Petar	Petrović	555-666	

# Wide Column baza podataka

- Relacione baze podataka su **row-oriented** (*način čuvanja podataka*)
  - **Način skladištenja (Storage Level)**
    - Podaci su jednodimenzionalni, jedna linija
      - 1, Ana, Mitić, 111-222, 2, Ognjen, Ilić, 333-444, 3, Petar, Petrović, 555-666
    - Kada se vrši upit nad dve kolone – **Ime i Prezime**
      - Skeniraju se **svi redovi i sve kolone**
        - 1, Ana, Mitić, 111-222, 2, Ognjen, Ilić, 333-444, 3, Petar, Petrović, 555-666
      - Bez obzira **koja kolona** se traži

## ■ Primer

- 1 x Red = 10 Kbytes, 1M redova → 10 GB  Aplikacija treba da pročita 10GB podataka na tako jednostavnom upitu
- 2 Kolone = 2 Kbytes, 1M redova → 2 GB  Dve kolone koje želimo da pročitamo su 2KB po svakom redu  
**Kako njih samo da pročitamo?**



# Wide Column baza podataka

- **Wide Column Database**

- Column-oriented db\columnar db\column-family db
- **Rotira se storage sistem (promenjena je struktura podataka)**
  - Svaki red predstavlja drugi atribut
- **Red** je lista vrednosti povezana sa istom kolonom (Sadržaj kolone se čuva u redu)

**Row-oriented**

Id	Ime	Prezime	Telefon
1	Ana	Mitić	111-222
2	Ognjen	Ilić	333-444
3	Petar	Petrović	555-666



**Column-oriented**

Id	1	2	3
Ime	Ana	Ognjen	Petar
Prezime	Mitić	Ilić	Petrović
Telefon	111-222	333-444	555-666

# Wide Column baza podataka

---

- **Wide Column Baza podataka**

- **Struktura**

- Jednodimenzionalna, jedna linija

- 1, 2, 3, Ana, Ognjen, Petar, Mitić, Ilić, Petrović, 111-222, 333-444, 555-666

- Upit nad dve kolone – Ime i Prezime

- 1, 2, 3, Ana, Ognjen, Petar, Mitić, Ilić, Petrović, 111-222, 333-444, 555-666

- Kolone su organizovane u grupe (**familije**)

# Wide Column baza podataka

## Prednosti

- **Smanjuje količinu podataka** koji se čitaju i učitavaju sa diska
- Bolje performanse i vreme izvršenja upita
  - Bitan faktor za analizu podataka (big data)
- Dobro je prilagođena OLAP baze podataka
- **Bolja kompresija podataka**
  - Podaci u jednoj koloni su uvek istog tipa
- **Skalabilnost**
  - Dizajnirana je za horizontalno skaliranje "out" kroz distribuirani klaster



## Nedostaci

- **Upis podataka** u *column-oriented* bazu podataka je **spor proces**
- **Row-oriented** baze podataka su znatno bolje rešenje za OLTP (Transakcije)

# Wide Column baza podataka

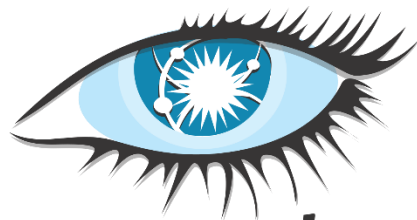
---



Google Cloud  
Bigtable



Azure Cosmos DB

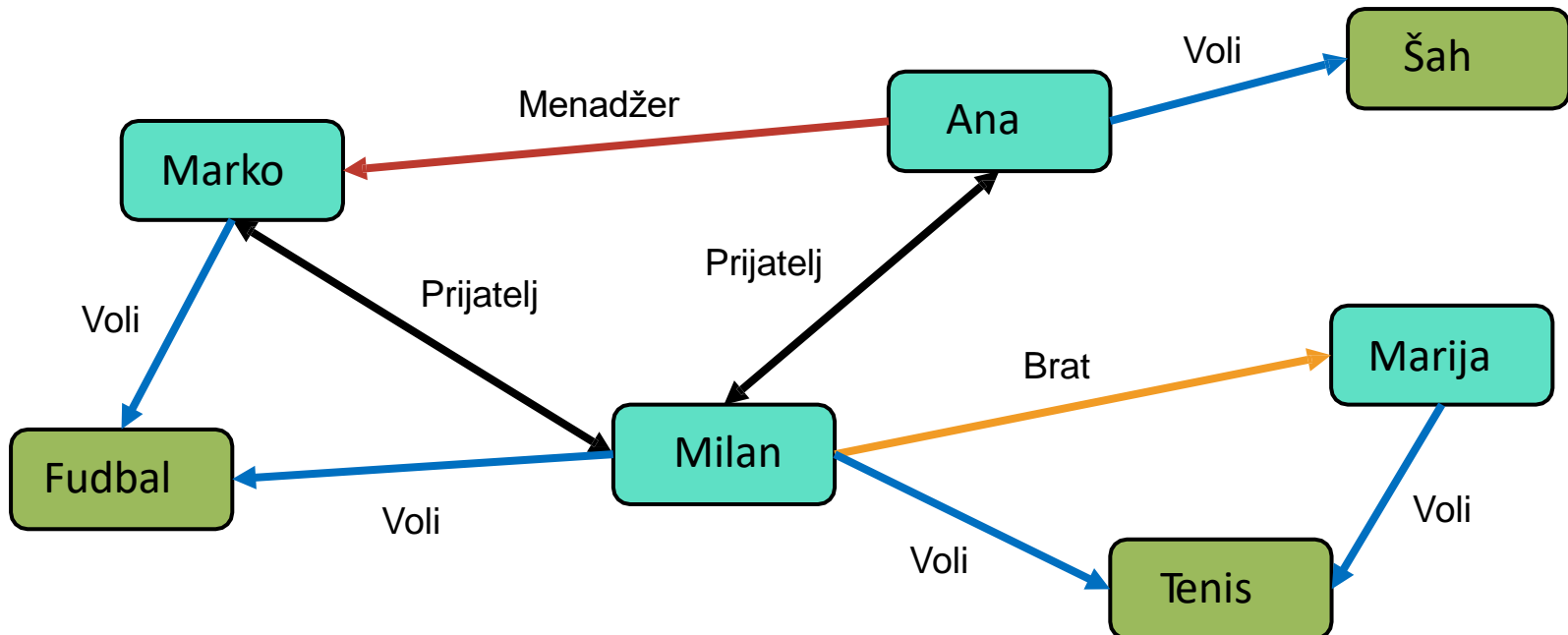


*cassandra*



# Graf baza podataka

- Graf baze podataka mogu da se razumeju i bez poznavanja **teorije grafova**
- **Podaci se predstavljaju formi Grafa**
  - Veze između stavki podataka su jednako važne kao i same stavke podataka
  - Dva tipa informacija
    - **Čvor (Nodes)** – predstavlja entitet (korisnik, proizvod,...)
    - **Veze (Edges)** – opisuje način povezivanja dva čvora



# Graf baza podataka

## Performanse:

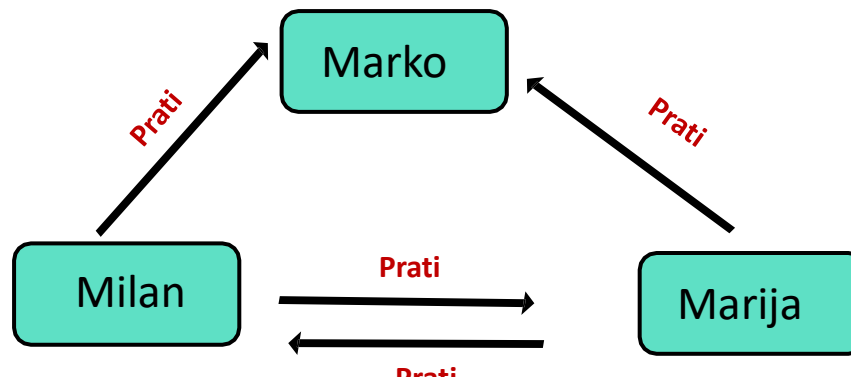
- Količina podataka se vremenom povećava
- Veze (connections) između podataka se uvećavaju znatno brže (eksponencijalno)
- Problem sa performansama relacione baze podataka imaju sa dubinom i brojem konekcija
- Performanse graf baze podataka ostaju konstatne i sa porastom podataka

## Fleksibilnost:

- Struktura i šema graf baze podataka je fleksibilna i prilagodljiva aplikaciji koja se razvija sa promenom strukture u skladu sa modernim tehnologijama
- RDBMS model diktira prilagođavanje tabularnom prikazu podataka

## Use Case:

Društvene mreže (Twitter koji povezuje 330 miliona aktivnih korisnika)



# Graf baza podataka

## Veze (Relationship):

- U graf bazama podataka su veze jednako ili bitnije od samih podataka
- Veze se ne izračunavaju na osnovu stranog ključa već su direktno prisutne

**Student**

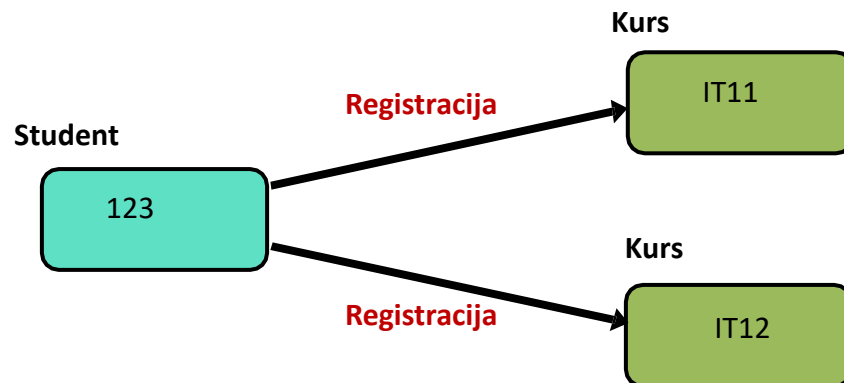
Sid	Sime
123	Ana
124	Dejan

**Kursevi**

Kid	Knaziv
IT11	Baze podataka
IT12	Administriranje BP

**Registracija**

Sid	Kid	Datum
123	IT11	01/05/2020
123	IT12	22/01/2021



# Uporedni prikaz relacionog dokument i graf modela

## RELACIONI MODEL

### Student

Sid	Sime
123	Ana
124	Dejan

### Kursevi

Kid	Knaziv
IT11	Baze podataka
IT12	Administriranje BP

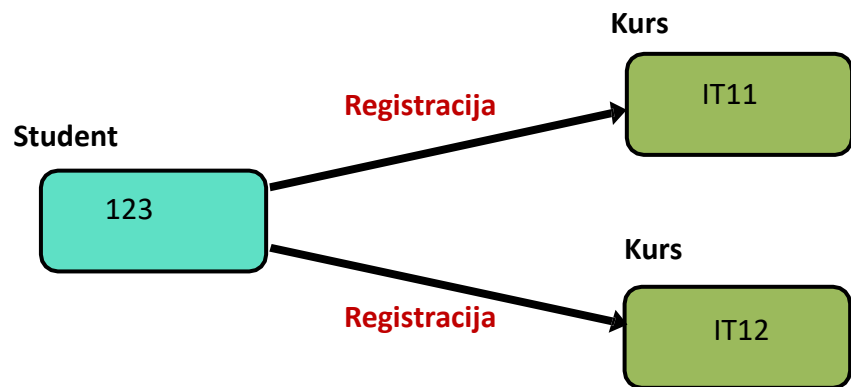
### Registracija

Sid	Kid	Datum
123	IT11	01/05/2020
123	IT12	22/01/2021

## DOKUMENT MODEL

```
{  
  Sid: 123,  
  Sime: "Ana"  
  Kurs: [ {  
    Kid: "IT11",  
    Knaziv: "Baze podataka",  
    Datum: "01/05/2020"  
  },  
  {  
    Kid: "IT12",  
    Knaziv: "Administriranje BP",  
    Datum: "22/01/2021"  
  }  
}
```

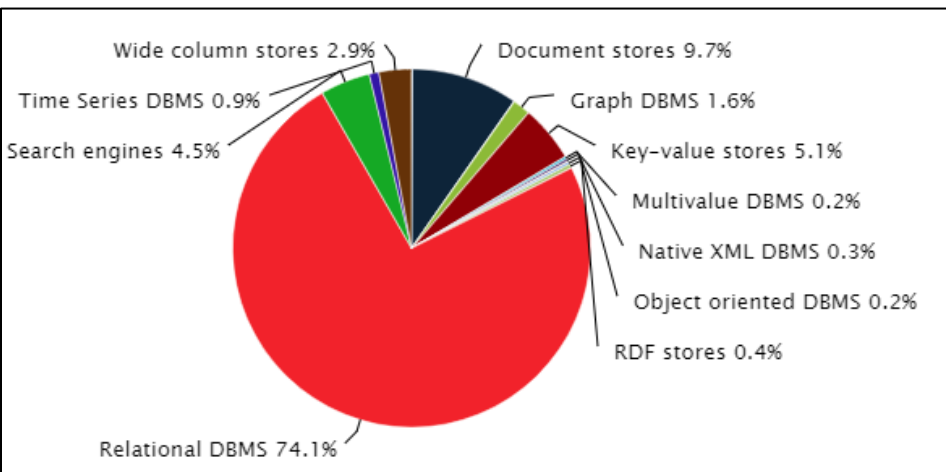
## GRAF MODEL



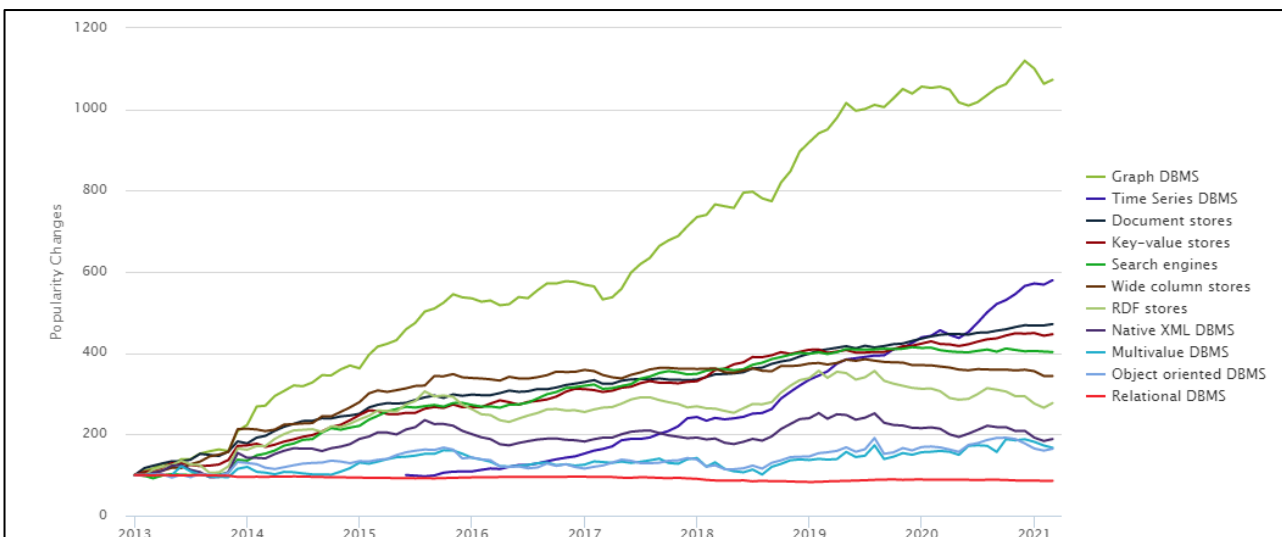


# Rangiranje baza podataka

<https://db-engines.com/en/ranking>



## RANGIRANJE BAZA PO POPULARNOSTI MART 2021



## TREND KORIŠĆENJA BAZA OD 2013.