



UNIVERZITET U NIŠU  
ELEKTRONSKI FAKULTET  
Katedra za računarstvo



# APACHE CASSANDRA – CLUSTER REŠENJA

-SISTEMI ZA UPRAVLJANJE BAZAMA PODATAKA-

---

SEMINARSKI RAD III

Student:

Marijana Cvetković, br. ind. 1431

Mentor:

Doc. dr Aleksandar Stanimirović

# Uvod

---

- Klaster baze podataka – funkcionisanje arhitekture klastera
- Klaster rešenja kod Apache Cassandra baze podataka
  - Inicijalizacija klastera sa više čvorove (jedan/više centra podataka)
  - Dodavanje čvorova
  - Zamenja čvorova
  - Uklanjanje čvorova
  - Komunikacija među čvorovima – Gossip protokol
- Praktična primena
- Zaključak

# Klaster baze podataka

---

Klaster baze podataka je kolekcija baza podataka kojima upravlja jedna instanca pokrenutog servera baze podataka.

Grupisanje baza podataka je proces kombinovanja više od jednog servera ili instanci koje povezuju jednu bazu podataka. Ponekad jedan server možda nije adekvatan za upravljanje količinom podataka ili brojem zahteva, tada je potreban klaster podataka.

Glavni razlozi za klasterisanje baze podataka su prednosti koje server dobija, redundantnost podataka, balansiranje opterećenja, visoka dostupnost, i na kraju, nadzor i automatizacija:

- Redudansa podataka
- Balansiranje opterećenja
- Visoka dostupnost
- Monitoring i automatizacija

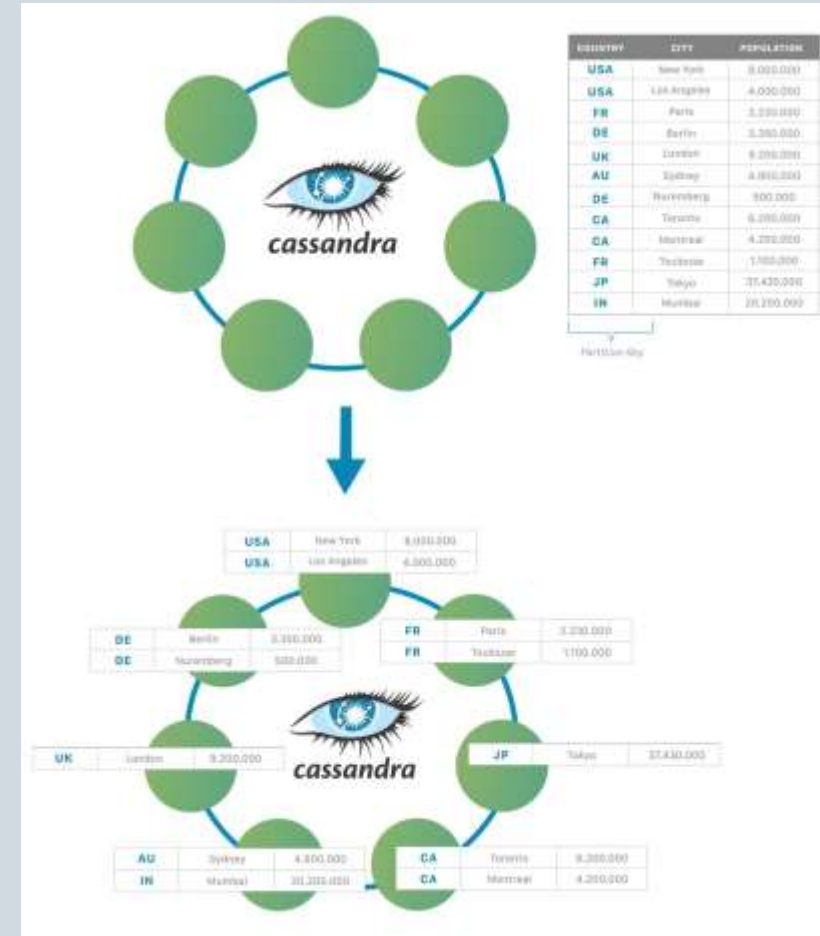
# Tipovi klastera baze podataka

---

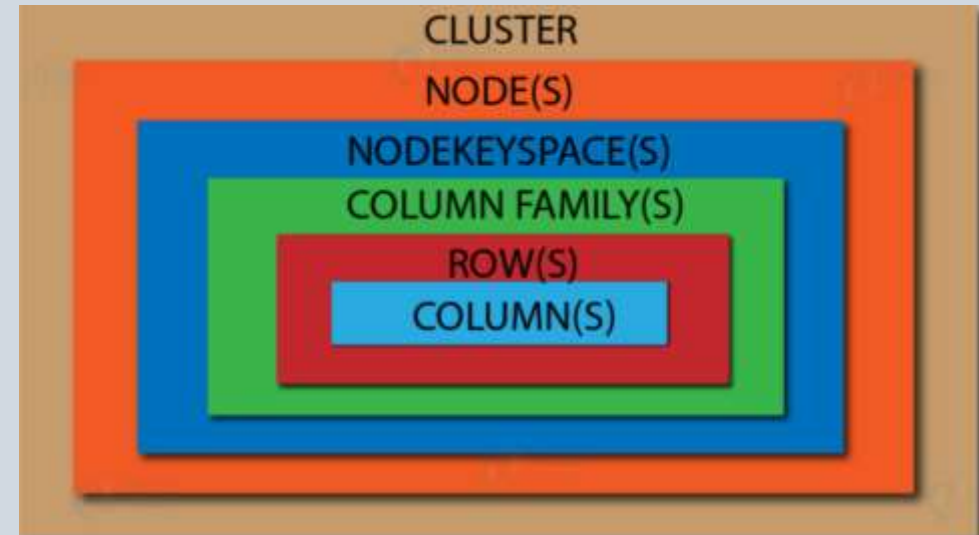
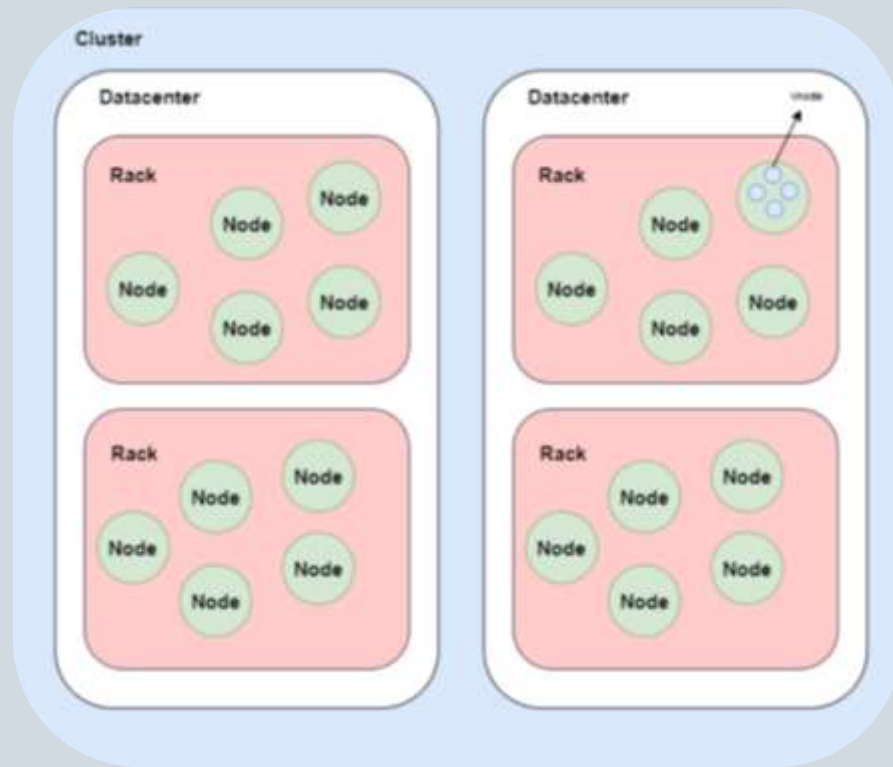
- **Klasteri za grešku/visoku dostupnost** - Ove vrste klastera su profitabilne za one korisnike koji u potpunosti zavise od svojih računarskih sistema. Na primer, e-trgovina, veb stranice itd.
- **Klasteri visokih performansi** - proizvodnja računarskih sistema visokih performansi. Oni upravljaju ko-proširujućim programima koji su potrebni za dugotrajne proračune. Ovakvu raznolikost klastera obično preferiraju naučne industrije. Osnovni cilj je inteligentno podeliti opterećenje.
- **Klasteri za balansiranje opterećenja** - služe za distribuciju opterećenja između različitih servera. Oni nastoje da obezbede povećan kapacitet mreže, konačno povećavajući performanse. Sistemi u ovoj mreži integrišu svoje čvorove, uz pomoć kojih se zahtevi korisnika podjednako dele na čvorove koji učestvuju. Sistem ne radi zajedno, već preusmerava zahteve podjednako kako se pojave.

# Klaster rešenja kod Apache Cassandra baze podataka

- NoSQL distribuirani sistem za upravljanje bazom podataka
- Rukovanje velikom količinom struktuiranih podataka
- Visoka dostupnost
- Arhitektura prstenastog tipa
- Najamjnja logička jedinica je čvor



- Klaster je komponenta koja sadrži jedan ili više cenatara podataka
  - Najspoljni kontejner za skladištenje u bazi podataka
  - Čvor – nezavistan (CPU, RAM, skladište)
- 
- Virtualni čvor



# Inicijalizacija klastera – jedan/više centar podataka

---

## **PREDUSLOVI:**

- Dobro razumevanje kako Cassandra radi.
- Instaliranje Cassandre na svakom čvoru.
- Odabir imena za klaster.
- Dobijanje IP adrese svakog čvora.
- Određivanje koji čvorovi će biti semenski čvorovi. Ne prave se svi čvorovi semenskim čvorovima.
- Određivanje strategije dojavljivanja i replikacije. Preporučeni su GossipingPropertyFileSnitch i NetworkTopologyStrategy.
- Određivanje konvencije imenovanja za svaki stalak. Na primer, dobra imena su RAC1, RAC2 ili R101, R102.

Konfiguraciona datoteka *cassandra.yaml* i datoteke svojstava kao što je *cassandra-rackdc.properties* daju više opcija za konfiguraciju.

# Inicijalizacija klastera – jedan centar podataka

- Ip adrese čvorova
- Otvaranje portova zaštitnog zida
- Isto ime klastera za sve čvorove
- Podešavanje svojstava u cassandra.yaml datoteci
- Dodeljivanje naziva centara podataka i rack-a
- Pokretanje početnih čvorova
- Pokretanje čvorova

```
node0 110.82.155.0 (seed1)
node1 110.82.155.1
node2 110.82.155.2
node3 110.82.156.3 (seed2)
node4 110.82.156.4
node5 110.82.156.5
```

```
cluster_name: 'MyCassandraCluster'

num_tokens: 256

seed_provider:
  - class_name:
    org.apache.cassandra.locator.SimpleSeedProvider
    parameters:
      - seeds: "110.82.155.0,110.82.155.3"

listen_address:

rpc_address: 0.0.0.0

endpoint_snitch: GossipingPropertyFileSnitch
```

```
paul@ubuntu:~/cassandra-2.1.0$ bin/nodetool status
```

```
Datacenter: datacenter1
```

```
=====
```

```
Status=Up/Down
```

```
|| State=Normal/Leaving/Joining/Moving
```

--	Address	Load	Tokens	Owns	Host ID	Rack
UN	10.194.171.160	53.98 KB	256	0.8%	a9fa31c7-f3c0-44d1-b8e7-a2628867840c	rack1
UN	10.196.14.48	93.62 KB	256	9.9%	f5bb146c-db51-475c-a44f-9facf2f1ad6e	rack1
UN	10.196.14.239	83.98 KB	256	8.2%	b8e6748f-ec11-410d-c94f-b8e7d88a28e7	rack1
...						



# Inicijalizacija klastera – više centara podataka

- Ip adrese čvorova
- Otvaranje portova zaštitnog zida
- Isto ime klastera za sve čvorove
- Podešavanje svojstava u cassandra.yaml datoteci
- Dodeljivanje naziva centara podataka i rack-a
- Pokretanje početnih čvorova, a onda i ostalih

```
node0 10.168.66.41 (seed1)
node1 10.176.43.66
node2 10.168.247.41
node3 10.176.170.59 (seed2)
node4 10.169.61.170
node5 10.169.30.138
```

```
cluster_name: 'MyCassandraCluster'
num_tokens: 256
seed_provider:
  - class_name: org.apache.cassandra.locator.SimpleSeedProvider
    parameters:
      - seeds: "10.168.66.41,10.176.170.59"
listen_address:
endpoint_snitch: GossipingPropertyFileSnitch
```

```
paul@ubuntu:~/cassandra-2.1.0$ bin/nodetool status
Datacenter: datacenter1
=====
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address           Load           Tokens      Owns    Host ID                               Rack
UN  10.194.171.160     53.98 KB      256        0.8%    a9fa31c7-f3c0-44d1-b8e7-a2628867840c rack1
UN  10.196.14.48        93.62 KB      256        9.9%    f5bb146c-db51-475c-a44f-9facf2f1ad6e rack1
UN  10.196.14.239       83.98 KB      256        8.2%    b8e6748f-ec11-410d-c94f-b8e7d88a28e7 rack1
...
```

# Operacija – dodavanje čvora u postojeći klaster

---

- Instaliranje Cassandre na svim čvorovima, bez pokretanja.
- Podešavanja svojstva u datoteci `cassandra-topology.properties` ili `cassandra-rackdc.properties`.
- Podešavanja svojstva u datoteci `cassandra.yaml` (`auto_bootstrap`, `cluster_name`, `listen_address/broadcast_address`, `endpoint_snitch`, `num_tokens`, `allocate_tokens_for_local_replication_factor`, `seed_provider`)
- Pokretanje bootstrap čvorova
- Provera da li je čvor potpuno pokrenut uz pomoć `nodetool`
- Pokretanje `nodetool cleanup` na svakom od čvorova da bi se uklonili ključevi koji više ne pripadaju tim čvorovima

# Operacija – zamena čvora (aktivnog čvora)

---

**Kod dodavanja čvora, a onda povlačenja starog,** mora se prvo priprmiti i pokrenuti zamenski čvor, integrirati ga u klaster, a zatim poništiti stari čvor. Procedura je sledeća:

- (1) Priprema i pokretanje zamenskog čvora.
- (2) Potvrda da je zamenski čvor živ (pokretanje nodetool ring ako se ne koristi vnodes ili pokretanje status nodetool ako se koristi vnodes) i status treba da pokaže nodetool ring:UP ili status nodetool:UN.
- (3) Treba znati ID domaćina originalnog čvora, za potrebe sledećeg koraka.
- (4) Korišćenjem ID hosta originalnog čvora, poništava se rad originalnog čvora iz klastera pomoću komande nodetool za dekomicioniranje.
- (5) Pokretanje nodetool čišćenje na svim ostalim čvorovima u istom centru podataka.

**Kod korišćenja nodetool-a za zamenu pokrenutog čvora,** procedura je:

- (1) Zaustavljanje Cassandre a čvoru koji se treba zameniti.
- (2) Praćenje uputstva za zamenu mrtvog čvora koristeći IP adresu starog čvora za – Dcassandra.replace\_address.
- (3) Uveriti se da se nivo doslednosti ONE ne koristi na ovom čvoru.
- (4) Uklanjanje čvora

# Operacija – zamena čvora (mrtvog čvora)

---

```
paul@ubuntu:~/cassandra-2.1.0$ bin/nodetool status
Datacenter: datacenter1
=====
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address          Load          Tokens      Owns    Host ID                               Rack
UN 10.194.171.160    53.98 KB      256         0.8%    a9fa31c7-f3c0-44d1-b8e7-a2628867840c rack1
UN 10.196.14.48      93.62 KB      256         9.9%    f5bb146c-db51-475c-a44f-9facf2f1ad6e rack1
DN 10.196.14.239      ?             256         8.2%    null
```

- Beleženje podešavanja centra podataka
- Dodavanje zamenskog čvora u mrežu i snimanje njegove IP adrese
- Ako je mrtvi čvor bio početni, menja se konfiguracija početnog čvora klastera na svakom čvoru
- Uveravanje da novi čvor ispunjava sve preduslove i zatim se instalira Cassandra na novi čvor
- Cassandra.yaml fajl
- Pokretanje novog čvora
- Provera da li je uspešno pokrenut

# Operacija – uklanjanje čvora

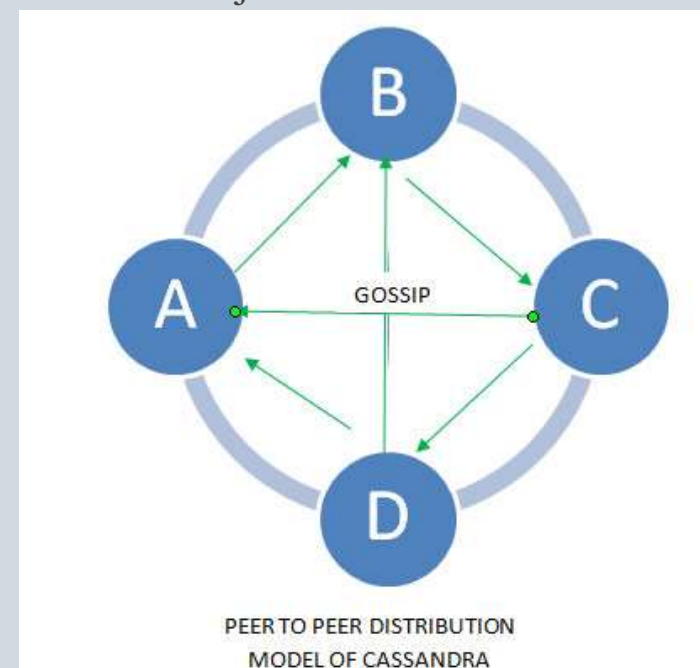
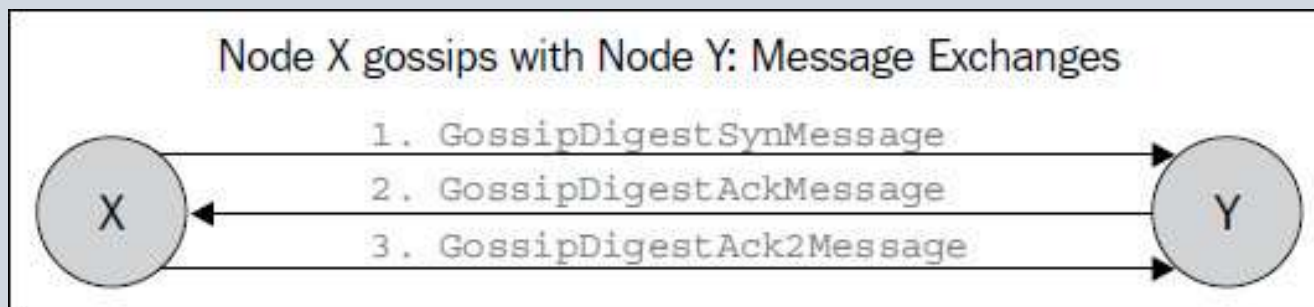
---

```
paul@ubuntu:~/cassandra-1.2.0$ bin/nodetool status
Datacenter: datacenter1
=====
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address            Load          Tokens   Owns    Host ID                               Rack
UN  10.194.171.160      53.98 KB      256      0.8%    a9fa31c7-f3c0-44d1-b8e7-a2628867840c rack1
UN  10.196.14.48         93.62 KB      256      9.9%    f5bb146c-db51-475c-a44f-9facf2f1ad6e rack1
DN  10.196.14.239        ?             256      8.2%    null
```

- (1) Ako je čvor podignut, pokreće se `nodetool decommission`. Ovo dodeljuje opsege za koje je čvor bio odgovoran drugim čvorovima i na odgovarajući način replicira podatke.
- (2) Ako čvor ne radi, bira se odgovarajuća opcija:
  - Ako klaster koristi `vnode`, uklanja se čvor pomoću naredbe `nodetool removemode`.
  - Ako klaster ne koristi `vnode`, pre pokretanja komande `nodetool removemode`, prilagođavaju se tokeni da ravnomerno rasporede podatke na preostale čvorove kako bi se izbeglo stvaranje vruće tačke.
- (3) Ako `removemode` ne uspe, pokreće se `nodetool assassinate`. To je prisilno uklanjanje mrtvih čvorova bez ponovnog repliciranja podataka. To je poslednje sredstvo ako se ne može uspešno ukloniti `nodetool removemode`

# Komunikacija čvorova u klasteru – Gossip protokol

- Tehnika interne komunikacije za čvorove u klasteru da razgovaraju jedni sa drugima.
- Efikasan, lagan, pouzdan inter-nodalni protokol emitovanja za širenje podataka.
- Decantralizovan je 'epidemijski', tolerantan je na greške i protokol za međusobnu komunikaciju.



# Otkrivanje kvarova i oporavak

---

- Detekcija kvara je metod za lokalno određivanje na osnovu stanja ogovaranja i istorije da li je drugi čvor u sistemu u kvaru ili se vratio.
- Proces ogovaranja prati stanje od drugih čvorova i direktno (čvorovi koji ga ogovraju direktno) i indirektno (čvorovi o kojima komuniciraju iz druge ruke, iz treće ruke i tako dalje).
- Kvarovi čvorova mogu biti posledica različitih uzroka kao što su kvarovi na hardveru i prekidi mreže. Prekidi čvorova su često prolazni, ali mogu trajati duže vreme.
- Mehanizmi za popravku postoje za oporavak propuštenih podataka, kao što su nagoveštena primopredaja i ručna popravka sa popravkom nodetool-a. Dužina prekida će odrediti koji mehanizam popravke se koristi da bi podaci bili dosledni.

# Praktični deo

```
endpoint_snitch: SimpleSnitch
- seeds: "seeds: 192.168.0.17,"
- listen address: 192.168.253.1
```

```
endpoint_snitch: SimpleSnitch
- seeds: "seeds: 192.168.0.17"
- listen address: 192.168.0.17
```

```

Kamanda linx
Subnet Mask . . . . . 255.255.255.0
Default Gateway . . . . .
Ethernet adapter VirtualBox Host-Only Network:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . fe80::1c1:99fc:b23b:f976%
IPv4 Address. . . . . 192.168.56.1
Subnet Mask . . . . . 255.255.255.0
Default Gateway . . . . .

Wireless LAN adapter Local Area Connection* 1:

Media State . . . . . Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 2:

Media State . . . . . Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . fe80::248c:47fe:5ab6:f3d2%5
IPv4 Address. . . . . 192.168.0.17
Subnet Mask . . . . . 255.255.255.0
Default Gateway . . . . . 192.168.0.1

C:\Users\Users>

```

```
Datacenter: datacenter1
=====
```

Status=Up/Down

```
| / State=Normal/Leaving/Joining/Moving
```

--	Address	Load	Tokens	Owns (effective)	Host ID	Rack
UN	192.168.0.17	235.79 KiB	256	100.0%	fb1d89bb-cbe2-488f-b2e7-da145bd2dde7	rack1
UN	192.168.253.1	176.01 KiB	256	100.0%	472fd4f0-9bb3-48a3-a933-9c9b07f7a9f6	rack1



# Praktični deo

---

```
CREATE KEYSPACE "OnlineMusicConcert"
WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication_factor' : 3 };
CREATE TABLE "Koncert" (
    "koncertID" text,
    ime text,
    opis text,
    organizator text,
    sponsor text tip text,
    PRIMARY KEY ("koncertID")
);
```

```
INSERT INTO "Koncert" ("koncertID",ime,opis,organizator,sponsor,tip)
VALUES('122','Moderna','Dobrodosli','Grad Nis','Bmw','20-te');
INSERT INTO "Koncert" ("koncertID",ime,opis,organizator,sponsor,tip)
VALUES('133','UvekZabava','Dobar provod','Firma Anoo','Stark','Pop');
```

```
select * from koncert;
```

koncertID	ime	opis	organizator	sponzor	tip
122	Moderna	Dobrodosli	Grad Nis	Bmw	20-te
133	UvekZabava	Dobar provod	Firma Ano	Stark	Pop
144	Nikad bolje	Dobrodosli!	Srednjoskolci	Restoran In	Razno

# Zaključak

---

- Redudantnost podataka, balansiranje opterećenja, visoka dostupnost, nadzor i automatizacija
- Inicijalizacija čvorova
- Dodavanje čvorova, zamena čvorova (aktivnog/mrtvog), brisanje čvorova
- Komunikacija čvorova – Gossip protokol
- Otkrivanje kvarova i grešaka
- Dobro poznavanje Cassandra baze podataka i njenih koncepata

# HVALA NA PAŽNJI!

---

Marijana Cvetković, br. ind. 1431