

NoSQL Baze podataka

Sanja Mijalković 1061/2013

Matematički fakultet, Studentski trg 16
11000 Belgrade, Serbia
sanja.mijalkovic@yahoo.com

Sadržaj

1	Uvod	3
2	Ime i poreklo	3
3	Motivacija i osobine	3
4	Pregled postojećih NoSQL rešenja	4
4.1	Ključ-vrednost skladista	4
	Oracle NoSQL	5
	Riak	5
4.2	Dokumentima orijentisane baze podataka	6
	MongoDB	6
5	Zaključak	7

Abstract. Ovaj rad se bavi novijim načinom pristupa baratanju podacima – **NoSQL-om**. Obradene su osobine baza podataka koje se mogu svrstati u ovu grupu kao i razlike između nerelacionih baza podataka i konvencionalnih. Posvećena je pažnja pojedinim vrstama koje se danas najviše koriste i u sklopu tog dela su prikazani i neki primeri koji odražavaju način organizacije baze kao i tip podataka koji se u njima može čuvati.

1. Uvod

U poslednjih par decenija tehnologija se razvija velikom brzinom a kao posledica se javlja problem da relacione baze podataka nisu uvek odgovarajuće za skladištenje i obradu podataka i u tome leži motivacija nastanka NoSQL baza. Nema jedinstvene definicije, ali osnovne razlike a ujedno i prednosti u odnosu na konvencionalne modele je da NoSQL baze ne počivaju na relacionom modelu podataka ili ga se bar ne drže čvrsto, lako se distribuiraju, horizontalno su skalabilne, struktura vrednosti obično nije strogo predefinisana.

2. Ime i poreklo

Originalno značenje termina ne odgovara njegovom današnjem načinu upotrebe. Sam termin NoSQL (eng. *Not only SQL*) je uveden 1998. godine i odnosio se na relacione baze podataka koje nisu imale podršku SQL-a. Uveo ga je Carlo Strozzi i tako nazvao svoj projekat koji je koristio bazu koja nije podržavala standardni SQL interfejs. Ovaj projekat se dugo razvijao, pod GNU licencom je i kao takav pripada slobodnom softveru (eng. *Open source*). Termin ponovo uvodi Eric Evans 2009. godine kada je organizovao okupljanje radi diskusije o pojavi sve većeg broja skladišta podataka koja ne vode računa o očuvanju atomicnosti, konzistentnosti, izolaciji i trajnosti (tzv. ACID¹ osobine), koje su ključne osobine relacionih modela. NoSQL ne predstavlja jedan proizvod ili tehnologiju već klasu proizvoda i kolekciju različitih, ponekad povezanih koncepata o načinima skladištenja i manipulacije podacima. [8] , [1]

3. Motivacija i osobine

Poslednjih godina su se razvili noviji sistemi koji obezbeđuju veliku horizontalnu skalabilnost za jednostavne operacije čitanja i pisanja nad podacima distribuiranim na više servera izuzetno velikom brzinom dok relacioni modeli baza podataka ne omogućavaju ili omogućavanju jako malu horizontalnu skalabilnost. Horizontalna skalabilnost znači da se lako može dodati novi čvor (server) kada je to potrebno i to ne utiče na rad sistema. Mnogi od ovakvih sistema su svrstani grupu NoSQL i za sve modele skladišta podataka koji su svrstani u ovu grupu vaze sledeće ključne osobine:

- Sposobnost ponavljanja i distribucije podataka na više servera.

¹ ACID predstavlja akronim za osobine transakcija u relacionim bazama podataka = atomičnost (eng. *Atomicity*), konzistentnost (eng. *Consistency*), izolaciju (eng. *Isolation*), i trajnost (eng. *Durability*).

- Obezbeđena horizontalna skalabilnost i jednostavne operacije između više servera.
- Jednostavan spoljašnji interfejs ili protokol (za razliku od SQL-a)
- Uglavnom podržavaju slabiji konkurentni model od ACID osobina transakcija (koje su zastupljene u SQL-u).
- Sposobnost da dinamički dodaju nove atribute.
- Efikasno korišćenje distribuiranih indeksa.

ACID osobine uglavnom nisu obezbeđene u ovakvim modelima. Ideja je da se zaobidju ograničenja ACID osobina jer se na taj način zaobilazi nepotrebna kompleksnost, podižu se performanse i omogućava skalabilnost. Obično se garantuje da važi skup principa BASE². Stanje sistema se menja tokom vremena, može doći do problema pri dodavanju novog čvora u sistem jer se tada sadržaj mora umnožiti i ne postoji garancija da će svi čvorovi sadržati identične kopije podataka u svakom trenutku, pa se teži tački u kojoj će to biti ispunjeno i svi podaci konzistentni. [4], [10]

4. Pregled postojećih NoSQL rešenja

Od 2009. godine razvilo se mnoštvo različitih NoSQL rešenja koji imaju različite pristupe organizaciji podataka koji se u njima čuvaju ali svi imaju zajedničke karakteristike koje ih vidno odvajaju od relacionih baza podataka. U ovom radu biće predstavljene ključ-vrednost skladišta i dokumentima orijentisane baze podataka, kao i neki od njihovih predstavnika.

4.1. Ključ-vrednost skladišta

Ovakva skladišta spadaju u najjednostavnija skladišta podataka. Podaci su organizovani u asocijativne nizove ili mape koje imaju jedinstven indeks po ključu za sve podatke. Zahvaljujući osobinama heš mape ovakva skladišta su izuzetno popularna jer je pretraga i pristup podacima vremenske složenosti $O(1)$. Ovakvi sistemi su skalabilni i omogućuju obradu velike količine podataka. Podaci se sastoje od ključa, koji je najčešće niska (eng. *string*), mada može biti i nekog drugog tipa i vrednosti, koja predstavlja stvarne podatke koji se čuvaju i mogu biti različitog tipa. Ovakva organizacija pruža bitne funkcionalnosti kao što su:

- Replikacija
- Pravljenje različitih verzija
- Zaključavanje
- Transakcije
- Sortiranje
- Mogu se koristiti B-stabla za organizaciju

Nedostatak ovakvih sistema je što ne podržavaju intervalne upite i ne pružaju mogućnost postavljanja indeksa drugog nivoa (eng. *secondary index*). Podaci koji bi mogli da se nađu u bazi predstavljeni su sledećim primerom:

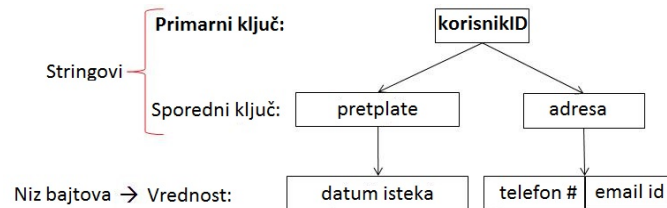
² BASE iz Engleskog i predstavlja akronim za *Basically Available, Soft state, Eventually consistent*.

- korisnik1-boja crvena
- korisnik1-godiste 1991
- korisnik2-godiste 1994
- korisnik1-visina 180
- korisnik2-boja ljubičasta
- korisnik5-visina 173
- korisnik1-ime Miloš Stanojević

Predstavnici ove grupe su Oracle NoSQL, Riak i mnogi drugi. [7]

Oracle NoSQL U najjednostavnijem obliku, ovakva baza podataka implementira se pomoću heš mape gde korisnik definiše ključeve koji moraju biti niske karaktera. Aplikacije ne treba da vode računa o uskladjivanju nekomaptibilnih verzija jer postoji jedinstven master čvor koji vrši replikaciju i on uvek ima najnoviju verziju ključeva, dok ostale kopije koji služe samo za čitanje podataka može da se desi da imaju malo starije verzije. Aplikacija može koristiti i brojeve verzija podataka kako bi obezbedila konzistenstnost operacija čitanja, pisanja i menjanja podataka.

U ovakvoj organizaciji ključevi igraju veliku ulogu jer se na osnovu njih pronalazi lokacija podataka. Ključevi mogu da imaju glavne i sporedne komponente. Vrednosti se čuvaju kao nizovi bitova i Oracle NoSQL baze nemaju interne pretpostavke o strukturi podataka smeštenim u niz bajtova. Preslikavanje niza bajtova u strukture podataka je prepušteno aplikaciji.



Slika 1. Primer organizacije Oracle NoSQL baze

Riak je NoSQL baza podataka, razvijena od strane Basho Technologies kompanije i sredinom 2009.-e godine je proglašena za "*open source*". Za razvoj su korišćeni Erlang, C, C++ programski jezici dok je za neke delove iskorišćen i JavaScript. Riak je pod Apache 2.0 licencom. Riak nije tipičan predstavnik ove grupe. Neki ga svrstavaju i u dokumentima orjentisane baze, jer ima dve bitne karakteristike zbog kojih je diskutabilno u koju grupu bi ga trebalo svrstati:

- * Riak objekti se mogu čuvati u JSON formatu i na taj način mogu imati više polja (kao i dokumenti). Mogu se grupisati u kolekcije (eng. *buckets*) i za svaku kolekciju je moguće definisati obavezna i opciona polja.

- * Bez obzira na složenu strukturu, nije moguće postavljanje indeksa nad poljima osim ključa. Pretraga se vrši samo po ključu tako da nije moguće pretraživanje po ostalim poljima objekta već samo njihovo skladištenje i preuzimanje.

Riak arhitektura je simetrična i jednostavna. Koristi dosledno heširanje³ (eng. consistent hashing). Jedinsvena karakteristika Riaka je da može da sačuva i veze između objekata. Ovim vezama se zaobilazi potreba za sekundarnim indeksima. [9]

Primer Riak objekta u JSON formatu:

```
{
  "kolekcija": "kupac",
  "kljuc": "12345",
  "objekat": {
    "ime": "Gospodin Markovic",
    "brojTelefona": "063-88-90-887"
  }
  "veze": [
    ["prodaja", "Gospodin Stankovic", "salesrep"],
    ["kupacPorudzbina", "12345", "porudzbine"]
  ]
  "poslednjaModifikacija": "Mon, 03 Aug 2009 18:49:42 GMT"
}
```

Važno je primetiti da je kljuc izdvojen dok su ostali atributi u okviru sekcije objekat. [3]

4.2. Dokumentima orijentisane baze podataka

Kao što samo ime govori, centralni koncept ovakvih skladišta je dokument. Ovakva skladišta podržavaju složenije podatke od ključ-vrednost skladišta. Dokumenti o kojima je reč predstavljaju kodirane podatke u JSON-u, XML-u, BSON-u ili u nekom drugom formatu.

Svaki JSON dokument predstavlja zaseban objekat koji aplikacija može da koristi. U jednom JSON dokumentu mogu da se čuvaju agregirani podaci za koje bi u relacionim modelima bilo potrebno spajanje mnogo tabela. Ovakve baze donose veliki napredak jer struktura dokumenata nije rigidna, vrlo je prilagodljiva i zahvalna za upotrebu u određenim situacijama. Primer se može videti na slici 2. [6]

MongoDB je jedan od najpoznatijih predstavnika. To je *"open-source"* implementirana baza podataka, kodirana u programskom jeziku C++. Upitni jezik malo podseća na SQL mada je napravljen u JavaScript-u. U MongoDB dokumenti su objekti koji se lako pridružuju tipovima podataka iz programskih jezika. Dokumenti imaju dinamičku shemu, što znači da dokumenti koji se nalaze u istoj kolekciji ne moraju da imaju ista polja ili istu strukturu, kao i da zajednička polja u dokumentima u kolekciji mogu čuvati različite tipove podataka. [2] [3]

Važne karakteristike su:

³ Dosledno heširanje se koristi za distribuciju podataka kroz prsten čvorova. Podaci proslednjeni K-tom čvoru se prosledjuju K+1-om, K+2-om.. K+n-tom gde je n željeni broj kopija.



Slika 2. Primer upotrebe dokumentima orjentisane baze podataka

- Podržava automatsku distribuciju dokumenata među serverima.
- Ne obezbeđuje globalnu konzistentnost, samo lokalnu.
- Podržava dinamičke upite sa automatskim korišćenjem indeksa.
- Obezbeđuje automičke operacije direktno nad poljima u dokumentima.

5. Zaključak

U trenutku nastajanja relacione baze podataka su zadovoljavale tadašnje potrebe skladišta podataka. Međutim, vremenom su se potrebe i očekivanja promenila. U prošlosti su prioriteti bili

- težnja smanjenju učestalosti padova sistema
- dinamički upiti
- rigidna struktura podataka
- centralizovana instalacija.

Danas su predviđeni padovi sistema, često se koriste predefinisani upiti, uglavnom statički. Zahteva se slobodna struktura podataka i distribuirana instalacija a sve to podržavaju NoSQL baze. One imaju svoje nedostatke, ali imaju i prostora za unapredjenje jer su novija tehnologija koja je još u fazi razvoja. Upravo zbog toga mogu se smatrati sledećom generacijom baza podataka, pripadaju slobodnom softveru i pružaju mogućnost efikasne obrade ogromnih količina podataka. Može se samo nagađati šta će dalje biti sa njihovim napretkom tako da će vreme pokazati da li su zaista i trajno bolje rešenje od konvencionalnih baza podataka.

Literatura

1. List of nosql databases (2013), <http://nosql-database.org/>
2. Agile and scalable mongodb (2014), <http://www.mongodb.org/>
3. Cattell, R.: Scalable sql and nosql data stores (December 2011), <http://cattell.net/datastores/Datastores.pdf>

4. Makarić, M.: Pregled komercijalnih i open source nosql baza podataka (Maj 2013), http://poincare.matf.bg.ac.rs/~vladaf/Courses/Matf%20MNSR/Prezentacije%20Individualne/Makaric_Pregled_komercijalnih_i_open_source_NoSQL_baza_podataka.pdf
5. nepoznat, A.: Oracle nosql database key-value pairs, <http://www.oracle.com/technetwork/products/nosqlldb/overview/key-value-497224.html>
6. nepoznat, A.: Why nosql?, <http://www.couchbase.com/why-nosql/nosql-database>
7. nepoznat, A.: Key value store (Maj 2012), <http://www.aerospike.com/what-is-a-nosql-key-value-store/>
8. nepoznat, A.: Nosql (April 2014), <http://en.wikipedia.org/wiki/NoSQL#History>
9. Technologies, B.: Riak (2014), <http://basho.com/riak/>
10. Veljković, N.: Nerelacione baze podataka nosql (Jun 2013), http://poincare.matf.bg.ac.rs/~vladaf/Courses/Matf%20MNSR/Prezentacije%20Individualne/Veljkovic_NoSql_baze_podataka.pdf