

Napredne baze podataka

Mehanizmi pohrane podataka
u bazama podataka



• Ideja tehnologije baze podataka

- Osnovna ideja tehnologije baza podataka jest da pojedina aplikacija ne stvara svoje vlastite datoteke na disku već umjesto toga **manipulira podacima na posredan način**, služeći se uslugama specijaliziranog softvera – **DBMS**

Temeljne komponente DBMS-a razvijene su u **svrhu**:

- stvaranja
- čitanja
- ažuriranja
- brisanja podataka,

i nazivaju se **mehanizmima pohrane podataka**.

- Većina DBMS-a podržava **višestruke mehanizme** za iste baze podataka
- Organizacija podataka u bazi podataka opisuje strukturu pohrane podataka te metode pristupa odnosno određuje sve **identifikacijske i operacijske karakteristike podataka** koji će biti pohranjeni u sustavu

• Organizacija pohrane podataka

- Baza podataka = pohranjena zbirka podataka
- Podatci = skup zapisa (uključuje tekstualne datoteke veličine 1 bit)
- Zapis se može promatrati kao zbir srodnih područja koje sadržavaju elementarne podatke
- Podatkovne datoteke
 - mogu postojati na primarnoj i na sekundarnoj memoriji
 - zbog svoje velike količine podatkovne datoteke vrlo brzo mogu premašiti kapacitet glavne memorije – iz tog razloga u pravilu se pohranjuju u sekundarnu
- Dobra organizacija podataka
 - *brzo vrijeme odaziva (dohvata) podataka*
 - *podržavanje velike količine transakcija, odnosno prometa*
 - *postizanje velike gustoće pohranjenih podataka uz mali gubitak prostora*

• **Mehanizmi pohrane podataka u MySQL-u**

- MySQL baza je optimizirana kako bi bila brza nauštrb funkcionalnosti
- Nasuprot tome, vrlo je stabilna i ima dobro dokumentirane module i ekstenzije te podršku od brojnih programskih jezika

• Razlike u mehanizmima pohrane (1)

• Transakcijska podrška

- Podrazumijeva više dodatne memorije, memorijskog prostora, i iskoristivosti CPU-a
- Mnoge aplikacije ne potražuju transakcije stoga je brže i efikasnije korištenje netransakcijskih mehanizama

• Karakteristike na razini tablice

- MySQL omogućuje da upravljač tablica ima CHECKSUM atribut koji je vidljiv u TABLES u INFORMATION_SCHEMA
- Da li tablica ima vrijednost CHECKSUM ovisi o mehanizmu pohrane
- Trenutno samo MyISAM podržava

• Zaključavanje

- MySQL podržava mogućnost zaključavanja cijele tablice
- Mehanizmi pohrane mogu implementirati svoje vlastite metode zaključavanja ne bi li omogućili zaključavanje na više razina (npr. cijela kolona za veću konkurentnost)

• Razlike u mehanizmima pohrane (2)

- **Indeks implementacija**

- Različite aplikacije mogu imati koristi od različitih strategija indeksiranja
- Postoji nekoliko uobičajenih metoda korištenih za svaku pohranu

- **Strani ključ**

- Koristi se za provedbu referencijskog integriteta među tablicama
- Mnogo aplikacija ne zahtjeva strane ključeve pa ih mnogi mehanizmi pohrane niti ne podržavaju

- **Buffering**

- Podatak, indeks i log buffer obrađuju se pomoću mehanizma pohrane
- Dok neki mehanizmi podržavaju mnogostruke buffere drugi ih u određenim područjima ne podržavaju uopće

• Razlike u mehanizmima pohrane (3)

• **Datoteka za pohranu**

- neki mehanizmi pohrane pohranjuju podatke i indekse u samopodržane datoteke (tablica može biti kopirana kopiranjem datoteke)
- drugi pak koriste centralizirane metapodatke čime ni tablice ne mogu biti jednostavno kopirane kopiranjem datoteke

• **Backup**

- Postoje mehanizmi s alatima pohrane koji dozvoljavaju konzistentnim ne blokiranim kopijama da se izvedu
- S druge strane postoje i takvi koji će prouzročiti grešku ako je kopiranje pokrenuto dok je tablica u uporabi

Karakteristike Database Enginea

Storage Engine Features

Feature	MyISAM	Memory	InnoDB	Archive	NDB
Storage limits	256TB	RAM	64TB	None	384EB
Transactions	No	No	Yes	No	Yes
Locking granularity	Table	Table	Row	Row	Row
MVCC	No	No	Yes	No	No
Geospatial data type support	Yes	No	Yes	Yes	Yes
Geospatial indexing support	Yes	No	Yes ^[a]	No	No
B-tree indexes	Yes	Yes	Yes	No	No
T-tree indexes	No	No	No	No	Yes
Hash indexes	No	Yes	No ^[b]	No	Yes
Full-text search indexes	Yes	No	Yes ^[c]	No	No
Clustered indexes	No	No	Yes	No	No
Data caches	No	N/A	Yes	No	Yes
Index caches	Yes	N/A	Yes	No	Yes
Compressed data	Yes ^[d]	No	Yes ^[e]	Yes	No
Encrypted data ^[f]	Yes	Yes	Yes	Yes	Yes
Cluster database support	No	No	No	No	Yes
Replication support ^[g]	Yes	Yes	Yes	Yes	Yes
Foreign key support	No	No	Yes	No	No
Backup / point-in-time recovery ^[h]	Yes	Yes	Yes	Yes	Yes
Query cache support	Yes	Yes	Yes	Yes	Yes
Update statistics for data dictionary	Yes	Yes	Yes	Yes	Yes

Table Storage Engines Feature Summary ver 8.

Feature	MyISAM	Memory	InnoDB	Archive	NDB
<u>B-tree indexes</u>	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>	No	No
<u>Backup/point-in-time recovery (note 1)</u>	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>
<u>Cluster database support</u>	No	No	No	No	<u>Yes</u>
<u>Clustered indexes</u>	No	No	<u>Yes</u>	No	No
<u>Compressed data</u>	<u>Yes (note 2)</u>	No	<u>Yes</u>	<u>Yes</u>	No
<u>Data caches</u>	No	N/A	<u>Yes</u>	No	<u>Yes</u>
<u>Encrypted data</u>	<u>Yes (note 3)</u>	<u>Yes (note 3)</u>	<u>Yes (note 4)</u>	<u>Yes (note 3)</u>	<u>Yes (note 3)</u>
<u>Foreign key support</u>	No	No	<u>Yes</u>	No	<u>Yes (note 5)</u>
<u>Full-text search indexes</u>	<u>Yes</u>	No	<u>Yes (note 6)</u>	No	No
<u>Geospatial data type support</u>	<u>Yes</u>	No	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>
<u>Geospatial indexing support</u>	<u>Yes</u>	No	<u>Yes (note 7)</u>	No	No
<u>Hash indexes</u>	No	<u>Yes</u>	No (note 8)	No	<u>Yes</u>
<u>Index caches</u>	<u>Yes</u>	N/A	<u>Yes</u>	No	<u>Yes</u>
<u>Locking granularity</u>	Table	Table	<u>Row</u>	<u>Row</u>	<u>Row</u>
MVCC	No	No	<u>Yes</u>	No	No
<u>Replication support (note 1)</u>	<u>Yes</u>	<u>Limited (note 9)</u>	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>
<u>Storage limits</u>	256TB	RAM	64TB	None	384EB
<u>T-tree indexes</u>	No	No	No	No	<u>Yes</u>
<u>Transactions</u>	No	No	<u>Yes</u>	No	<u>Yes</u>
<u>Update statistics for data dictionary</u>	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>	<u>Yes</u>

Notes:

1. Implemented in the server, rather than in the storage engine.
2. Compressed MyISAM tables are supported only when using the compressed row format. Tables using the compressed row format with MyISAM are read only.
3. Implemented in the server via encryption functions.
4. Implemented in the server via encryption functions; In MySQL 5.7 and later, data-at-rest tablespace encryption is supported.
5. Support for foreign keys is available in MySQL Cluster NDB 7.3 and later.
6. InnoDB support for FULLTEXT indexes is available in MySQL 5.6 and later.
7. InnoDB support for geospatial indexing is available in MySQL 5.7 and later.
8. InnoDB utilizes hash indexes internally for its Adaptive Hash Index feature.
9. See the discussion later in this section.

•Kriteriji za odabir mehanizma pohrane

- Frekvencija čitanja podataka u odnosu na pisanje
- Zahtjeva li se transakcijska podrška
- Je li potrebna uporaba stranih ključeva
- Upotreba indexa
- Veličina tablica
- Prenosivost na druge OS / arhitekture
- Zahtjevi daljnjih nadogradnji i prilagodba novim zahtjevima

Usporedba transakcijskih i netransakcijskih mehanizama pohranjivanja

- Transakcijski mehanizmi su sigurniji
 - Čak i u slučaju zastoja MySQL-a ili problema sa sklopovljem, moguće je spasiti podatke
 - Moguće je korištenje rollback operacije radi poništavanja izmjena
 - U slučaju neuspjeha ažuriranja, sve promjene se poništavaju
- Transakcijski mehanizmi pohranjivanja pružaju bolju istodobnost za tablice na kojima se vrši mnogo ažuriranja i čitanja istovremeno
- Netransakcijski mehanizmi pohranjivanja su brži
- Netransakcijski mehanizmi zahtijevaju manje prostora na disku, te je potrebno manje memorije za izvršavanje ažuriranja

• Postavljanje mehanizma pohrane

```
CREATE TABLE t1 (i INT) ENGINE = INNODB;  
CREATE TABLE t2 (i INT) ENGINE = CSV;  
CREATE TABLE t3 (i INT) ENGINE = MEMORY;
```

```
ALTER TABLE t ENGINE = InnoDB;
```

```
--postavljanje pretpostavljenog mehanizma pohrane za sesiju  
SET default_storage_engine=NDBCLUSTER;
```


•MyISAM

- Pretpostavljeni tip tablice u MySQL-u do verzije 5.7
- Optimiziran za kompresiju i brzinu
- Podržava full-text index
- Jednostavna prenosivost između različitih OS-ova
- Podržava velike tablice (do 256TB)
- Fiksne ili dinamične veličine tablica
- Zaključavanje na razini tablica
- Podatci unutar jedne tablice mogu biti pohranjeni na različiti lokacijama, pa čak i na različitim sustavima
- Velike tablice mogu biti komprimirane u manje read-only tablice koje će zauzimati manje diskovnog prostora (*myisampack*)
- **Ne podržava transakcije**
- Uporaba kod sustava orijentiranih ka dohvatu podataka (skladišta podataka, određene web aplikacije)

• InnoDB

- Pretpostavljeni mehanizam pohrane u MySQL-u od verzije 5.7
- Potpuna podrška transakcijama i osiguravanje ACID svojstava *bez kompromisa* prema kompresiji i brzini
- Koristi međuspremnik radi veće brzine pohrane
- Odgovara učinkovitosti MyISAM tablice
- Podržava tablice veličine do 64TB
- Također se pri pokretanju provjerava ispravnost tablica
- Podrška stranim ključevima, commit, rollback
- Jedan od mehanizama sa 'najviše opreme' u MySQL-u
- Uporaba u većini primjena

•Archive

- Pohrana velikih setova podataka kojima se pristupa manje često, u manjem komprimiranom obliku
- Pogodno za pohranjivanje 'povijesnih' podataka
- Omogućava ubacivanje i selektiranje podataka, ali ne i brisanje, zamjenu i ažuriranje
- Ne podržava indeksiranje (prilikom čitanja pretražuju se cijele tablice)
- Smanjena brzina komunikacije sa podacima zbog kompresije koja se odvija on-the-fly prilikom umetanja odnosno dohvaćanja podataka
- Ne podržava transakcije

• Federated

- Omogućuje pristup podacima na lokalnom serveru sa udaljenih MySQL baza podataka bez korištenja replikacije ili klastera
- Povezivanje nekoliko fizički različitih MySQL poslužitelja
- na lokalnom serveru nalazi samo jedna datoteka sa ekstenzijom .frm koja sadrži format tablice te konekcijski string koji pokazuje na udaljenu tablicu, dok se na udaljenom serveru nalazi datoteka koja sadrži format tablice kao i same podatke
- operacije na lokalnom serveru se automatski vrše na udaljenim tablicama (u lokalnim tablicama se ne spremaju nikakvi podaci)
- Uporaba za **data martove** – poboljšanje performansi za analitičke izvještaje

•Memory (HEAP)

- spremanje sadržaja tablice direktno u memoriju
- Tablice posebne namjene sa sadržajem podataka
- prednost je brzina pohrane (ali zahtjeva **dostatnu količinu RAM-a**)
- Iako je moguće kreirati privremenu tablicu, standardna memory tablica je trajna te će se prilikom gašenja servera isprazniti, a ne obrisati u cijelosti
- Prilikom stvaranja memory tablice stvara se samo jedna datoteka na disku koja sadrži format tablice
- Session management, caching
- Većinom za dohvat / čitanje podataka

•CSV

- Sprema podatke u tekstualne datoteke koristeći format pisanja gdje su vrijednosti odvojeni zarezom
- Prilikom kreiranja CSV tablice stvaraju se dvije datoteke
 - Format tablice je spremljen u datoteci sa ekstenzijom .frm
 - u .CSV datoteci spremaju podaci iz tablice
- Uporaba u slučajevima kada je podatke potrebno dijeliti s drugim aplikacijama koje koriste CSV format zapisa podataka
- **Ne podržava indekse**

•MERGE

- MyISAM merge tablica sama ne sadržava podatke već se odnosi na skupinu identičnih MyISAM podtablica koje se koriste kao jedna
- sve podtablice sadrže jednake informacije za stupce i indekse
- najčešće koriste kod analiziranja periodičnih podataka koji su spremljeni u više tablica

•ISAM

- metoda indeksiranja podataka radi brzog dohvaćanja
- prvobitni i jedini mehanizam pohranjivanja sve do izlaska MySQL 3.23 verzije, kada je uvedena novija MyISAM verzija
- dva načina pristupa redcima, sekvencijalno ili pomoću indeksa
- dopušteno korištenje maksimalno 16 indeksa, te 16 dijelova ključa po ključu
- tablice veličine do 4GB

• NDB

- Network Database
- NDBCLUSTER
- visoko redundantna verzija MySql-a prilagođena okolini distribuiranog računarstva
- Pohranjivanje tablica u podatkovnim čvorovima kojima se lako može pristupiti sa drugih MySQL servera

•Kombiniranje mehanizama - primjer

- Sustav u potpunosti ACID podrživ, no optimiziran za brzi dohvat podataka (web aplikacija)
 - Kreiranje tablica na InnoDB mehanizmu za sve funkcionalnosti sustava
 - Uklanjanje redundancije i normalizacija
 - Postavljanje indeksa
 - Izrada MyISAM tablica za podatke koji trebaju biti pretraživi
 - Izrada okidača za punjenje MyISAM tablica
 - Prilikom unosa u InnoDB tablicu i svih N n-torki koje sadrži, u MyISAM sestrinsku tablicu pune se samo one n-torke po kojima podatci trebaju biti pretraživi
 - Postavljanje full-text indeksa

•Usporedba brzine rada / dohvata podataka

- Potrebno je kreirati jednake tablice koristeći **MyISAM**, **InnoDB** i **Memory** mehanizme pohranjivanja
- Jednak broj redaka i stupaca, random brojevi 1 – 10000
- Svaka tablica sadrži 1 milijun redaka
- Nakon izvođenja operacija izračunavaju se srednje vrijednosti izvođenja pojedinih operacija nad svakom tablicom, te se rezultati uspoređuju

Br	Naredba
1.	SELECT * FROM ime_tablice;
2.	SELECT count(*) FROM ime_tablice;
3.	SELECT COUNT(DISTINCT random_values1) FROM ime_tablice;
4.	SELECT * FROM ime_tablice WHERE random_values1=random_values2 OR random_values1=random_values3 OR random_values1=random_values4;
5.	SELECT * FROM ime_tablice ORDER BY random_values1;
6.	SELECT SUM(random_values1) FROM ime_tablice;
7.	SELECT AVG(random_values1) FROM ime_tablice;
8.	SELECT MIN(random_values1), MAX(random_values1) FROM ime_tablice;
9.	SELECT MIN(random_values1),MAX(random_values1), MIN(random_values2),MAX(random_values2), MIN(random_values3), MAX(random_values3), MIN(random_values4), MAX(random_values4) FROM ime_tablice;
10.	SELECT COUNT(*), COUNT(DISTINCT random_values2), AVG(random_values3), AVG(random_values4) FROM ime_tablice;

Redni broj operacije	myisam	innodb	memory
1.	0,0016	0,0335	0,0014
2.	0,0004	1,0805	0,0004
3.	0,5710	1,5985	0,4322
4.	0,2941	1,4099	0,2119
5.	0,9077	2,2902	0,6283
6.	0,2946	1,4246	0,2117
7.	0,2967	1,4035	0,2183
8.	0,2089	1,2110	0,1450
9.	0,4374	1,5907	0,3309
10.	0,9106	2,1400	0,7308

Redni broj operacije	myisam	innodb
1.	114,29%	2392,86%
2.	100%	270125%
3.	132,12%	369,85%
4.	138,79%	665,36%
5	144,47%	364,51%
6.	139,16%	672,93%
7.	135,91%	642,92%
8.	144,07%	835,17%
9.	132,19%	480,72%
10.	124,60%	292,83%

•Rezultat mjerenja

- Budući da se podaci spremaju unutar same memorije, **memory mehanizam pohranjivanja se pokazao najbrži**
- Još uvijek iznimno brz, MyISAM mehanizam pohranjivanja se pokazao za nijansu sporiji
- Izuzetak je izvođenje prebrojavanja redaka, gdje su rezultati identični - ovo je rezultat činjenice da MyISAM, kao niti memory, stvarno ne prebrojava retke poput InnoDB-a jer je broj redaka kod ovih mehanizama pohranjivanja u svakom trenutku poznat

• Nakon mjerenja

- Rezultati analize pokazali su kako je MyISAM značajno brži mehanizam pohranjivanja od InnoDB
- Posebno je naglašena veća brzina kod MyISAM mehanizma pohranjivanja u bazama podataka gdje se često koristi selektiranje podataka, budući da se u tom području InnoDB pokazao znatno inferiorniji
- InnoDB mehanizam pohranjivanja sadrži visoki integritet podataka sa cijenom smanjenih performansi
- MyISAM, bez takvih opterećenja, se može koristiti u slučajevima gdje je naglasak na brzini, međutim sa cijenom gubitka podataka u slučaju pada sustava
- „Koji je mehanizam pohranjivanja najbolji“ = „Koji je mehanizam pohranjivanja najprikladniji za...?“

• Za kraj

- **Različiti mehanizmi** pohrane omogućuju MySQLu mnogo **različitih mogućnosti**, također postoje mnogi načini pohrane razvijeni od drugih tvrtki
- **InnoDB**, danas najčešće korišten mehanizam pohrane, razvijen je od više različitih tvrtki
- S obzirom da se **jedan mehanizam pohrane ne može savršeno uklapati u svaku situaciju** korištenje više mehanizama omogućuje pohranu koja bi bila primjenjiva u različita područja primjene
- Najbolji **način skladištenja i pretraživanja** velikih količina podataka s obzirom na tehnologiju u suštini predstavljaju osnovu svakog informacijskog sustava, tj. temelj svakog poslovnog subjekta koji svoje poslovanje bazira na dostupnosti kvalitetnih i brzih informacija