

Poređenje brzine odziva MS SQL servera i MySQL-a

Stanišević Ilja, Vićentić Mladen

Akademija strukovnih studija Zapadna Srbija,
Odsek u Valjevu,
Valjevo, Srbija

ilja.stanisevic@vipos.edu.rs, mladjen.vicentic@vipos.edu.rs

Obradović Slobodan

ITS Visoka škola strukovnih studija za IT
Beograd, Srbija

slobo.obradovic@gmail.com

Sažetak— U nameri da izaberemo optimalan sistem za upravljanje bazom podataka za informacioni sistem Visoke poslovne škole strukovnih studija VIPOS u Valjevu, izvršena su testiranja najpredlaganijih opcija (MS SQL Server i MySQL). Umesto oslanjanja na postojeće alate za testiranje (*eng. benchmark*), razvojni tim je razvio svoj alat koji je raden u istoj tehnologiji kao i školski sistem i koji je namenjen testiranju onih aspekata koji su relevantni za konkretnu situaciju u kojoj se sistem realizovao. U ovom radu je opisana primenjena metodologija i način realizacije testnog alata, kao i dati prikaz ostvarenih rezultata.

Ključne riječi- sistem za upravljanje bazom podataka; Microsoft SQL Server; MySQL; benchmark alati i testovi;

I. UVOD

Prilikom izrade informacionog sistema za potrebe Visoke poslovne škole strukovnih studija VIPOS Valjevo, nametnulo se pitanje izbora sistema za upravljanje bazom podataka. Različiti članovi razvojnog tima su imali različite utiske i predloge, u zavisnosti od njihovog iskustva. Bilo je, stoga, potrebno razviti objektivni metod i alat za evaluaciju, prilagođeni potrebama Škole, koji će dati adekvatan i argumentovan odgovor na ovo pitanje. U ovom radu opisujemo razvoj i primenu ovog metoda i alata, te dajemo prikaz ostvarenih rezultata.

II. TEHNIČKI PREDUSLOVI

Informacioni sistem Visoke poslovne škole VIPOS u Valjevu je predviđen da radi u lokalnoj mreži (LAN) škole po modelu klijent-server. Mreža poseduje četiri servera (kontrolor domena, rezervni kontroler domena, server baze podataka i server kontrole izlaska na internet) i oko 120 radnih stanica. Kontrolori domena i server baze podataka rade pod Windows 8 server operativnim sistemom dok su na radnim stanicama prisutne različite verzije Windows operativnog sistema (Windows 10, Windows 7 i Windows XP), ali svi imaju instalirano .NET framework 3.5 okruženje. Sam sistem je realizovan u jeziku C#.NET, u razvojnom okruženju Visual Studio. Budući da je nabavka nove opreme sporadična i zavisi od finansijskih mogućnosti Škole i izdvajanjima od strane Ministarstva prosvete, nauke i tehnološkog razvoja Republike

Srbije, neophodno je obezbediti kompatibilnost sa starijim sistemima i radnim stanicama.

Rukovodstvo Škole se opredelilo, iz bezbedonosnih razloga, da sistem bude dostupan isključivo u školskom okruženju, tj. lokalnoj mreži škole, te da ne bude podržan rad preko interneta. Sistem ima modul za prijavljivanje i proveru prava pristupa, a postoji i kontroler domena, sa definisanim ulogama i pravima pristupa, tako da se sistem za upravljanje bazom podataka može osloniti na bezbedonosne primitive operativnog sistema.

Kad su u pitanju funkcionalni zahtevi sistema, on treba da podrži rad:

- studentske službe (registar studenata, upise, overavanje semestra, kolokvijume, ispite, izdavanje raznih uverenja, zapisnika i sl.);
- organizaciju nastave (poveravanje nastave, planiranje nastave, evidenciju održanih časova, evidenciju realizacije plana nastave na semestralnom i godišnjem nivou);
- aktivnosti mentora (praćenje uspešnosti i efikasnosti nastave, i to na nivou predmeta (evidenciju uspešnosti studija na svakom predmetu), smeru (zbirni podaci za pojedine studijske programe i godine studija), pojedinačnog studenta (evidencija uspešnosti svakog studenta pojedinačno) i ispitnog (kolokvijumskog) roka (brojeve studenata koji su prijavili, izašli i položili dati kolokvijum/predmet kako za pojedine predmete, tako i zbirno za studijski program i godinu studija);
- kadrovske službe (evidencije podataka o zaposlenima i njihovo prisustvo u Školi).

Iz opisa funkcionalnosti sistema može se zaključiti da sistem neće biti previše opterećen aktivnostima unosa podataka, ali će biti značajno opterećen generisanjem različitih izveštaja. Zapravo, najveći deo unosa, kao i generisanje dokumenata potrebnih nastavnicima i studentima Škole se odvija na samo jednom mestu, u studentskoj službi Škole. Ove aktivnosti se odvijaju u realnom vremenu (na šalterima), te se kao primarni kriterijum evaluacije odgovarajućeg skladišta podataka nameće brzina, pri čemu je najvažnija brzina pretraživanja baze.

III. IZBOR METODOLOGIJE

Da bi mogli objektivno odrediti performanse sistema za upravljanje bazom podataka, potrebno je izvršiti odgovarajuća merenja prikladnim softverskim alatom za testiranje - benchmarkom. Kvalitetan benchmark za sisteme za upravljanje bazom podataka, po Greju [1], bi morao da ima sledeće osobine:

- **relevantnost**, tj. mora pokriti proveru onih osobina koje su od interesa što većem broju potencijalnih korisnika;
- **portabilnost**, tj. mora biti primenljiv na više različitih sistema za upravljanje bazom podataka;
- **jednostavnost**, tj. mora biti lako primenljiv pri čemu ne zahteva utrošak previše resursa;
- **prilagodljivost**, tj. mora biti primenljiv kako na velike tako i na male računarske sisteme, kao što mora biti primenljiv i na nove računarske arhitekture.

Ove osobine su, po svojoj prirodi, često kontradiktorne, pa je potrebno napraviti adekvatan kompromis između njih [2]. Tako, npr. osobina jednostavnosti može biti u suprotnosti sa zahtevima portabilnosti.

Veliki je broj parametara koji mogu uticati na ostvarene rezultate testiranja. Među njima se nalaze hardver (brzina centralne procesne jedinice (CPU), broj CPU-a i broj jezgara, pristupna brzina memoriji, količina centralne memorije, brzina sabirnice, brzina tvrdog diska), operativni sistem (način rada sa nitima - threading), način upravljanja memorijom i realizacija zaključavanja, broj uspostavljenih konekcija ka klijentima, šema podataka, količina pohranjenih podataka, tip aplikacije kojom se pristupa bazi podataka, konfiguracija baze podataka (dodeljena veličina keša za upite, maksimalan broj konekcija, upotrebe indeksa, definisanje mrežnih protokola korišćenih za pristup bazi) [3].

Razvoj tehnologije baza i skladišta podataka doveo je do pojave alata za testiranje koji ispituju i evaluiraju različite aspekte sistema za upravljanje bazama podataka. Tako imamo benchmarke procesiranja transakcija, relacione benchmarke, objektno orjentisane i objektno-relacione benchmarke, XML benchmarke, benchmarke baza koje koriste sistemi za podršku odlučivanju (DSS), benchmarke za non-SQL baze, benchmarke za baze u oblaku [2], [4]. Posebna organizacija, Savet za performanse procesiranja transakcija (Transaction Processing Performance Council – TPC) izdaje standarde za benchmarke namenjene relacionim bazama i verifikuje korektnost ovih alata [4].

Na tržištu danas imamo veliki broj paketa za testiranje namenjenih bazama podataka, kao što su Quest [5], STS Soft [6], HammerDB [7] i mnogi drugi. Značajan broj njih je softver otvorenog koda, tako da njihova primena ne zahteva dodatna finansijska sredstva. Ipak, ovde se javlja jedan značajan problem. Većina benchmark alata su kreirani da budu što univerzalnije primenljivi, pa samim tim, u krajnju ocenu ulaze i kriterijumi nebitni za konkretnu namenu. U našem slučaju, gotovo sigurno bi bili obuhvaćene ocene osobina kao što su rad sa velikim brojem korisnika, rad velikom količinom podataka, rad u internet okruženju. Ove osobine za izbor baze za informacioni sistem Škole nemaju nikakav značaj. Postoje i benchmark alati namenjeni evaluaciji usko specijalizovanih

baza [8], kao i alati koji prioritet u ispitivanju daju manjem broju osobina [9], ali ni ti alati nisu pogodni za određivanje najbolje baze podataka za naš sistem, usled ograničenosti svog domena primene.

Imajući sve ovo u vidu, razvojni tim informacionog sistema se opredelio za drugačiju strategiju. Umesto korišćenja gotovog alata za testiranje performansi opšte namene, krenulo se u realizaciju testnog programa koji treba da testira i obezbedi objektivnu evaluaciju za osobine i okruženje koje će u potpunosti biti prilagođeno zahtevima školskog informacionog sistema. Ovakav sistem ne bi zadovoljio uslove portabilnosti i prilagodljivosti (što, u konkretnom slučaju, i nije od značaja), ali bi maksimalno zadovoljio uslove relevantnosti (bio bi dizajniran za konkretnu situaciju i, kao takav, bio bi maksimalno relevantan za tu situaciju) kao i jednostavnosti (testirale bi se samo osobine bitne za konkretn slučaj).

IV. ODREĐIVANJE DOMENA TESTIRANJA

Kako je projektovana baza podataka za školski sistem relaciona, to je i alat za testiranje predviđen da radi sa relacionim modelom. Kao osnovni kriterijum evaluacije je usvojena brzina odziva. Pri tome, posebno su testirani odzivi na različite upite, kako za pretraživanja (SELECT), tako i akcione (INSERT, UPDATE i DELETE). Posebno su merene vrednosti sa logičkim uslovom (postojanje WHERE klauzule u upitu), a posebno bez ovakvog uslova. Posebno su uzimani u obzir upiti nad jednom relacijom, a posebno nad više relacija, tj. kad postoji JOINT klauzula. Merenja su vršena pri različitim količinama podataka koji postoje u bazi, pri čemu je tražene logičke uslove zadovoljavalo 20% postojećih slogova.

Najveća količina podataka pri transakcionim obradama se dešava početkom školske godine, kada se upisuju novi studenti, kao i kada postojeći studenti upisuju narednu ili obnavljaju prethodnu godinu. Broj slogova obuhvaćenih ovim transakcijama se kreće između 150-250, u zavisnosti od broja studenata upisanih na pojedinu godinu studija., te nisu od interesa transakcije koje bi uključivale veći broj slogova.

Da bi procenili količinu redova u relacijama, morali smo utvrditi koji objekat u sistemu je najbrojniji. Pregledom u prethodne podatke, utvrđeno je da su to prijave za kolokvijume. Broj ovih objekata, na godišnjem nivou možemo proceniti na osnovu sledećeg obrasca:

$$PK = \sum_{i=0,m,d} (\Gamma_i * \alpha_i * BS_i * BK_i) \quad (1)$$

pri čemu su nam:

- i – vrsta studija (osnovne strukovne, master strukovne i osnovne studije na daljinu);
- PK – broj prijava kolokvijuma;
- Γ_i – broj godina trajanja studija na i-toj vrsti studija;
- α_i – koeficijent aktivnih studenata na i-toj vrsti studija;
- BS_i – broj studenata na i-toj vrsti studija;
- BK_i – prosečan broj kolokvijuma u toku godine na i-toj vrsti studija;

Škola je akreditovana za upis ukupno 165 studenata na osnovne strukovne studije, 50 studenata na master strukovne

studije i 32 studenta na osnovne strukovne studije – studiranje na daljinu. Na osnovnim strukovnim studijama i studijama na daljinu prosečan broj kolokvijuma na godišnjem nivou je 14, dok je na master strukovnim studijama 6. Osnovne strukovne studije traju 3 godina, a master strukovne studije traju 2 godine. Koeficijenti aktivnih studenata na osnovnim studijama je 0,38, na studijama na daljinu 0,15, a na master studijama 0,57. Napomenimo da ovde nije uzet u obzir broj uspešno upisanih studenata u sledeću godinu studija, budući da se on približno kompenzuje brojem ponovnih izlazaka na kolokvijume. Kada ove vrednosti uvrstimo u jednačinu (1) dobijamo:

$$PK = PK_o + PK_m + PK_d \quad (2)$$

pri čemu su:

$$PK_o = 3 * 0,38 * 165 * 14 = 2.633,4$$

$$PK_m = 2 * 0,57 * 50 * 6 = 342$$

$$PK_d = 3 * 0,15 * 32 * 14 = 201,6$$

te uvršćivanjem u jednačinu (2) dobijamo:

$$PK = 2633,4 + 342 + 201,6 = 3177 \quad (3)$$

Dakle, na godišnjem nivou je procenjeno da će najopterećenija relacija imati prirast od 3,177 redova, što bi na trogodišnjem nivou iznosilo 9,531 red. Imajući u vidu trend opadanja broja studenata koji je prisutan već duži period, a koji će se, prema podacima o broju srednjoškolaca i osnovaca u Kolubarskom regionu, kao i celoj Zapadnoj Srbiji, nastaviti i narednih godina, testiranja će biti vršena do maksimalnog opterećenja relacije od 10.000 slogova. Merenja pri većem opterećenju nisu od veće važnosti u konkretnom slučaju.

Slične procene su vršene i za ostale realacije sa većim brojem objekata (npr. prijave kolokvijuma, prijave ispita, evidencija realizacije održanih časova) i procena opterećenja nigde nije prešla navedene vrednosti. Većina tabela, ipak, ima znatno manje slogova, najčešće 10-1000 slogova. Ovde spadaju registri nastavnika, zaposlenih, kabineta, studijskih programa, nastavnih predmeta, poveravanja nastave, aktivnih studenata, diplomiranih studenata...

Konačno, budući da je relaciona baza podataka predviđena projektom striktno normalizovana i nalazi se u Boyce-Coddovoj normalnoj formi [10], od značaja je i efikasnost sistema za upravljanje bazom podataka pri realizaciji spajanja relacija korišćenjem JOINT iskaza. Testiranja su zato morala uključiti i ove slučajeve.

Prilikom obrade malog broja slogova (npr. pojedinačnog unosa), kašnjenja prouzrokovana realizacijom mrežnih protokola značajno prevazilaze vreme odziva samog sistema za upravljanje bazom podataka, te mogu dovesti do pogrešnih rezultata. Stoga su merenja vršena za obradu minimalno 100 slogova, a maksimalno 10,000.

V. TEHNIČKO OKRUŽENJE

Da bi merenja bila što adekvatnija, izvođena su u realnom radnom okruženju u kojem će se izvršavati i informacioni sistem Škole. U okviru lokalne mreže, računar namenjen za server baze podataka ima procesor Intel Pentium CPU G2130

na 3.2 GHz i poseduje 8GB RAM memorije. Server radi pod operativnim sistemom Windows Server 2008 Enterprise without Hyper-V pod SP2, 64-bitna verzija. Tvrdi disk je podeljen na dve particije od po 232 GB, pri čemu se sistemi za upravljanje bazom podataka nalaze na primarnoj particiji..

Što se tiče softvera korišćenog za izradu sistema, testni alat je bio realizovan u programskom jeziku C#.NET, isto kao i informacioni sistem Škole. Softverska platforma je bila .NET Framework verzija 3.5, što je posledica postojanja starijih računara u školskoj lokalnoj mreži, pa je, radi kompatibilnosti, za informacioni sistem prihvaćena verzija 3.5 kao standard, što je onda primenjeno i na testni alat. Iz istog razloga, izvršna verzija je podešena za 32-bitni procesor. Kao programersko okruženje, korišćen je MS Visual Studio Community 2017, verzija 15.9.14.

Izbor sistema koji će se testirati je bio određen i tehničkim mogućnostima, kao i iskustvom članova razvojnog tima. Škola nije posedovala neke druge komercijalne sisteme (npr. Oracle DB, IBM DB2, Informix, ...), niti su članovi razvojnog tima imali iskustva u radu sa nekim drugim sistemima otvorenog koda (npr. PostgreSQL, Ingres, SQLite...). Procenjeno je da bi uvođenje ovih sistema pri realizaciji sistema Škole uzrokovalo veći utrošak raspoloživih resura (bilo novca, bilo vremena), tako da oni nisu uzeti u razmatranje.

Za testiranje su izabrana dva najčešća sistema za upravljanje bazom podataka: MS SQL Server (verzija 2016, Standard, 64-bitna) i MySQL (verzija 8.0.16 – MySQL Community Server – GPL). U preliminarno razmatranje je bio uključen i MS Access (verzija Professional Plus 2016), ali je pokazao izuzetno loše karakteristike prilikom testiranja na lokalnom računaru, pa nije uzet u obzir za testiranje u mrežnom okruženju [11].

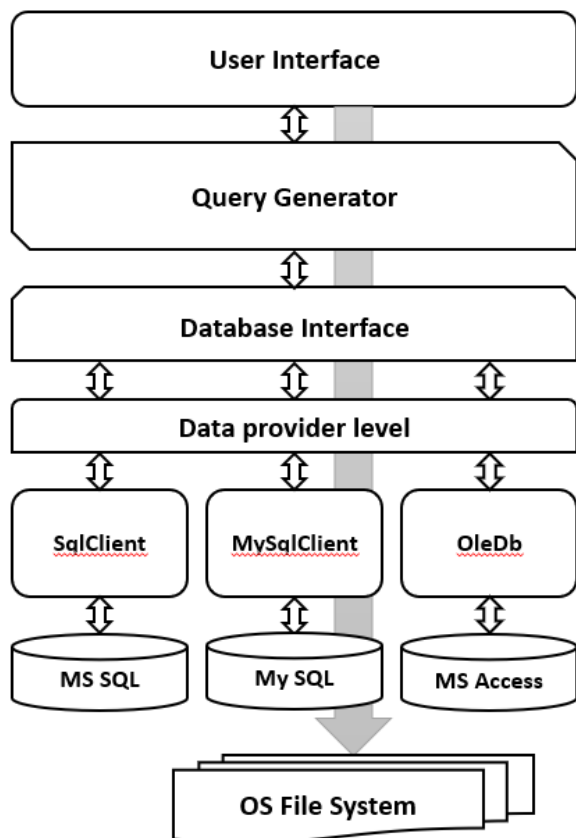
MS SQL Server predstavlja najčešće korišćeni sistem baza podataka u poslovnim organizacijama, dok je MySQL na trećem mestu [12]. U ukupnoj upotrebi, najčešće korišćen sistem je Oracle DB, dok su MySQL i MS SQL Server na drugom i trećem mestu [13]. U praksi se može reći da MS SQL Server dominira za poslovne aplikacije u okviru organizacionih subjekata, dok MySQL dominira u izradi aplikacija za internet, u kombinaciji sa Linux operativnim sistemom, Apache serverom i PHP programskim jezikom. Popularnosti MySQL-a doprinosi i njegovo korišćenje od strane mnogih alata za izradu web stranica, kao što je npr. WordPress. MS Access, kao sastavni deo paketa paketa Microsoft Office se, uprkos brojnim funkcionalnim ograničenjima, nalazi na petom mestu liste najkorišćenijih sistema baza podataka [13].

VI. IZRADA ALATA ZA EVALUACIJU

Konceptualni dizajn sistema za testiranje je dat na SI.1. Izvorni alat je u sebi uključivao i modul za testiranje MS Accessa. Ovim dizajnom je omogućeno i relativno jednostavno dodavanje opcija za testiranje i drugih sistema. Sve što je potrebno je dodavanje adekvatnog obezbeđivača podataka (eng. *data provider*) i dodavanja jednostavne metode koja s njim ostvaruje vezu u okviru interfejsa baze podataka.[14].

Težilo se da sam testni softver bude što jednostavniji za upotrebu, ali da, s druge strane, sadrži sve potrebne

funkcionalnosti. Korisniku je omogućeno da jednostavnim izborom opcije iz kontrole combo-boxa izabere sistem za upravljanje bazom podataka koji želi testirati (MS SQL, MySQL, Access). Potom su mu na raspolaganju četiri osnovna komandna dugmeta za osnovne SQL iskaze (SELECT, UPDATE, INSERT, DELETE) koje želi testirati. Opciono može pokrenuti metodu za automatsko punjenje tabele unesenim brojem redova. Takođe mu je omogućeno da opciono zadaje logičke uslove (WHERE klauzule) za SELECT, UPDATE, i DELETE iskaze. Ukoliko ih ne navede, upit se izvršava bez WHERE klauzule. Dodatno mu je omogućeno da izabere koja polja i kako će menjati prilikom ažuriranja sloga naredbom UPDATE (tj. definisanje SET klauzule).



Slika 1. Dizajn sistema za testiranje

Jednim boksom za čekiranje korisnik može aktivirati i uključivanje JOINT klauzule koja vrši spajanje više tabela. Korišćene su dve tabele između kojih postoji višestruka funkcionalna zavisnost, tj. M:N odnos, te je uključena i treća tabela, tabela veze. Ova struktura je izabrana za testiranje jer predstavlja najopštiji slučaj spajanja relacija [10]. Ukoliko je boks za čekiranje "include JOIN clause" čekiran, sistem za testiranje tada automatski puni sekundarnu bazu i uspostavlja relaciju preko tabele veze sa primarnom relacijom. U generisanu SQL naredbu se dodaje JOINT klauzula i, tako se prilikom izvršavanja efikasno može meriti odziv kad su referisane sve tri tabele.

Korisnik ne mora unositi celokupne SQL komande, već samo unosi logičke uslove za WHERE klauzulu, kao i nove vrednosti za SET klauzulu pri pokretanju UPDATE komande. Na osnovu ovih unosa generator upita generiše korektne SQL upite i šalje ih interfejsu baze podataka na obradu i dalje izvršavanje.

Sistem vrši merenje vremenskog intervala od prosleđivanja SQL komande obezbeđivaču podataka do prijema odziva, tj. rezultata izvršene operacije (npr. podataka u tabelarnom obliku kao rezultata SELECT komande, ili do prijema indikatora uspešno izvršenog upita za akcione upite). Interval se meri u sistemskim otkucajima (eng. *ticks*), pri čemu je jedan otkucaj ekvivalent 100 ns, odnosno predstavlja desetmilioni deo sekunda.

Rezultati izvršenih merenja, kao i SQL komande koje su inicirale akciju, se sukcesivno upisuju u list boks kontrolu. Na kraju sesije merenja korisnik ove rezultate može upisati u datoteku koja se nalazi na tvrdom disku i da je kasnije koristi u analizi rezultata i daljoj obradi podataka. Korisniku su na raspolaganju još dve funkcionalnosti, trenutni prikaz sadržaja tabele, te opcija pražnjenja sadržaja svih tabela testirane baze podataka.

VII. PREZENTACIJA REZULTATA

Vršena su merenja vremena odziva za sledeće slučajeve: bezuslovni INSERT, SELECT i DELETE; uslovni (sa WHERE klauzulom) SELECT, UPDATE i DELETE, te složeni SELECT pri postojanju veze između tabela (JOIN klauzula). Svaka od ovih situacija je merena pri različitom broju redova u tabelama, od 100 do 10,000 redova. Za svaku testiranu kombinaciju je vršeno ukupno 7 merenja. Najveća i najmanja izmerena vrednost su odbacivane, a rezultat je predstavljala aritmetička sredina preostalih pet merenja.

Posebno je uzeta u obzir osobina programske platforme .NET Framework, kao i JIT interpretera da se prilikom pokretanja određenih modula vrši učitavanje većih blokova koda u centralnu memoriju, što izaziva određenja kašnjenja, što može dovoditi do zbunjujućih rezultata. Dešava se da kraći i jednostavniji upiti koji su prvi zadati traju nekoliko redova veličina duže nego znatno kompleksniji i zahtevniji upiti koji se zadaju nakon te inicijalizacije. Zbog toga je pre svakog merenja izvedeno nekoliko upita koji nisu bili evidentirani kao rezultati merenja.

U Tabeli I je dato vreme odziva prilikom izvršenja INSERT komande za Microsoft SQL Server i MySQL, u zavisnosti od broja slogova koji se unosi u tabelu. Vreme je dato u sekundama.

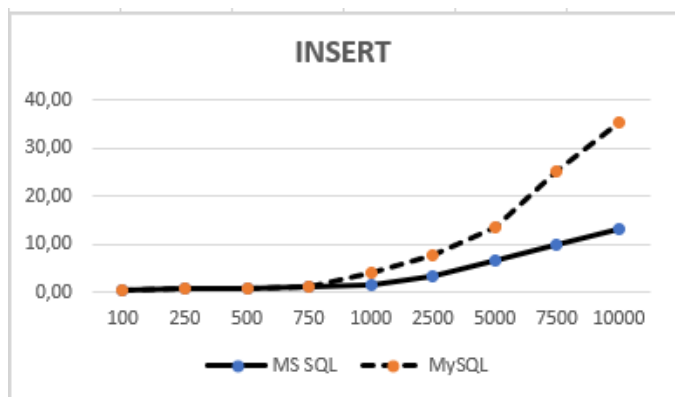
TABELA I. VREME ODZIVA INSERT KOMANDE

MS SQL	0,21	0,59	0,77	1,02	1,31	3,34	6,49	9,76	13,09
MySQL	0,34	0,63	0,87	1,29	3,97	7,59	13,48	25,36	35,44
slogova	100	250	500	750	1000	2500	5000	7500	10000

Ovde treba napomenuti da je unos podataka vršen separatno, tj. da je prilikom upisa svakog sloga uspostavljena konekcija ka bazi podataka, koja je, nakon uspešno

realizovanog upisa i ukidana. Nije vršena korekcija ove pojave, budući da je na isti način realizovan i unos podataka u realnom sistemu, te se ovako dobijeni rezultati mogu smatrati validnim.

Grafički prikaz vremena odziva je dat na Sl.2.



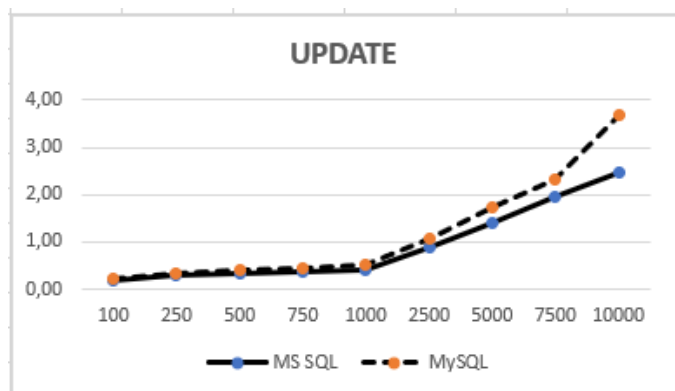
Slika 2. Grafički prikaz vremena odziva INSERT komande

U Tabeli II je dato vreme odziva prilikom izvršenja UPDATE komande za MS SQL Server i MySQL, u zavisnosti od broja slogova. Značajno kraće vreme izvođenja u odnosu na vreme realizacije INSERT komande, premda je UPDATE najzahtevnija akciona komanda SQL-a je rezultat uspostavljanja samo jedne konekcije sa samom bazom, a ažuriranje se potom realizuje serijski nad većim brojem slogova.

TABELA II. VREME ODZIVA UPDATE KOMANDE

MS SQL	0,18	0,30	0,32	0,36	0,42	0,89	1,40	1,95	2,48
MySQL	0,21	0,35	0,41	0,44	0,51	1,08	1,72	2,34	3,67
slogova	100	250	500	750	1000	2500	5000	7500	10000

Grafički prikaz vremena odziva je dat na Sl.3. Primetno je da je razlika vremena odziva između MS SQL-a i MySQL-a značajno manja nego pri merenju vremena odziva realizacije INSERT komande. Razlog ovome leži u činjenici da je MS SQL Server realizovan u istoj, Microsoftovoj tehnologiji, pa i već pomenuto višestruko uspostavljanje konekcije značajno kraće traje u ovom slučaju, nego kad je reč o MySQL-u (tj. obezbeđivač podataka (*eng. data provider*) je efikasniji za MS SQL.



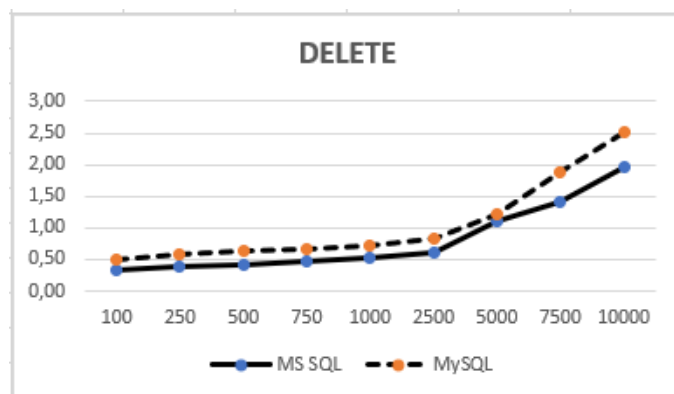
Slika 3. Grafički prikaz vremena odziva UPDATE komande

U Tabeli III je dato vreme odziva prilikom izvršenja DELETE komande za MS SQL Server i MySQL, u zavisnosti od broja slogova

TABELA III. VREME ODZIVA DELETE KOMANDE

MS SQL	0,32	0,38	0,41	0,48	0,54	0,62	1,11	1,40	1,96
MySQL	0,51	0,58	0,63	0,67	0,73	0,83	1,21	1,87	2,51
slogova	100	250	500	750	1000	2500	5000	7500	10000

Grafički prikaz vremena odziva pri realizaciji DELETE akcione komande je dat na Sl.4.



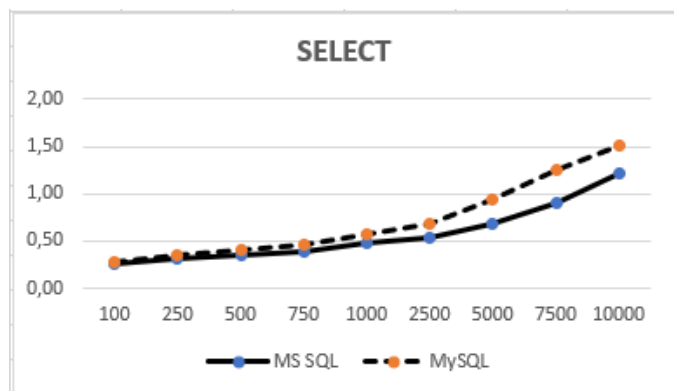
Slika 4. Grafički prikaz vremena odziva DELETE komande

Vreme odziva u sekundama prilikom izvršenja SELECT komande za MS SQL Server i MySQL, nad jednom tabelom, u zavisnosti od broja n-torki u relaciji, dato je u Tabeli IV.

TABELA IV. VREME ODZIVA SELECT KOMANDE

MS SQL	0,26	0,32	0,35	0,39	0,48	0,54	0,69	0,90	1,21
MySQL	0,28	0,36	0,41	0,46	0,57	0,68	0,94	1,26	1,52
slogova	100	250	500	750	1000	2500	5000	7500	10000

Grafički prikaz vremena odziva pri realizaciji SELECT upita nad jednom relacijom je dat na Sl.5.



Slika 5. Grafički prikaz vremena odziva SELECT komande

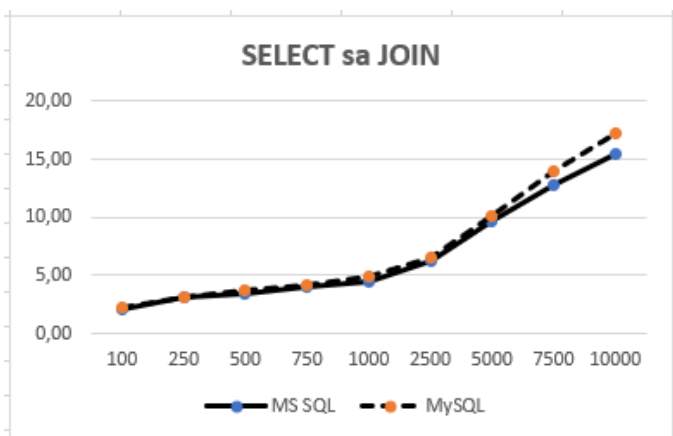
Najvažniji parametar procene predstavlja rezultat merenja vremena odziva prilikom izvršavanja SELECT upita kod kog postoji veza između više relacija. Vreme odziva u sekundama

prilikom izvršenja SELECT komande koja u sebi sadrži i JOIN klauzulu, za MS SQL Server i MySQL dato je u Tabeli V.

TABELA V. VREME ODZIVA SELECT KOMANDE KOJA U SEBI SADRŽI JOIN KLAUZULU

MS SQL	2,13	3,10	3,49	3,97	4,40	6,21	9,60	12,80	15,40
MySQL	2,18	3,19	3,74	4,23	4,92	6,53	10,12	13,98	17,24
slogova	100	250	500	750	1000	2500	5000	7500	10000

Grafički prikaz vremena odziva pri realizaciji SELECT upita koji sadrži i JOIN klauzulu, dat je na Sl.6.



Slika 6. Grafički prikaz vremena odziva SELECT komande koja u sebi sadrži i JOIN klauzulu

Budući da je slično istraživanje sprovedeno i u lokalnoj laboratorijskoj radnoj stanici, gde se čitav sistem nalazi na jednom računaru [11], interesantno je uporediti rezultate kako bi se uočio uticaj mrežnih protokola na vremena odziva. Tabela VI nam daje prikaz srednje vrednosti vremena odziva za svaku komandu u lokalnu i u mrežnom okruženju, te kašnjenje prouzrokovano mrežnim protokolima.

TABELA VI. POREĐENJE IZMERENOG VREMENA ODZIVA U MREŽNOM I LOKALNOM OKRUŽENJU

	INSERT	UPDATE	DELETE	SELECT	SELECT /JOIN
MS SQL u mreži	4,06	0,92	0,80	0,57	6,79
MS SQL u lokalno	1,72	0,01	0,37	0,04	0,11
mrežno kašnjenje	2,34	0,91	0,43	0,53	6,68
MySQL u mreži	9,89	1,19	1,06	0,72	7,347
MySQL u lokalno	7,98	0,01	0,65	0,10	0,14
mrežno kašnjenje	1,91	1,18	0,41	0,62	7,21

ZAKLJUČAK

Premda razlike ostvarenih vremena nisu prevelike (sa izuzetkom izvršenja INSERT komande), ipak se mora primetiti da je MS SQL server ostvario bolje rezultate od MySQL-a po svim merenim parametrima. Stoga je upravo ovaj DBMS izabran za realizaciju školskog sistema. U dosadašnjoj eksploataciji nije dolazilo do zastoja u radu usled neadekvatnog

odziva baze podataka i može se reći da se izbor pokazao kao ispravan.

Na kraju napomenjemo da cilj ovog istraživanja nije bio da pruži opštu i apsolutnu ocenu testiranih sistema, već da ih uporedi u specifičnom, realnom okruženju prilagođenom korisničkim potrebama u konkretnom slučaju.

Takođe je pokazano da je moguće, umesto oslanjanja na razne već postojeće, ali često neadekvatne benchmark testove, napraviti sistem koji će testirati upravo one elemente koji su nam od interesa i na osnovu toga argumentovano doneti odluku primerenu konkretnom problemu.

LITERATURA

- [1] Gray, J. (ed.). "The Benchmark Handbook for Database and Transaction Processing Systems 2nd edition". Morgan Kaufmann, 1993;
- [2] Darmont, Jérôme, "Object Database Benchmark", Encyclopedia of Information Science and Technology, I-III, Idea Group Publishing, pp.2146-2149, 2005;
- [3] MySQL, "MySQL Performance Benchmarks – Measuring MySQL Scalability and Throughput", A MySQL Technical White Paper, March 2005, raspoloživ na: <http://www.jonahharris.com/osdb/mysql/mysql-performance-whitepaper.pdf>;
- [4] Darmont, Jérôme, "Data Processing Benchmarks", Encyclopedia of Information Science and Technology, Third Edition, pp.146-152, 2014;
- [5] Quest, Benchmark Factory For Databases, raspoloživ na: <https://www.quest.com/products/benchmark-factory/>;
- [6] STS Soft, DatabaseBenchmark, raspoloživ na <http://stssoft.com/products/database-benchmark/>;
- [7] HammerDB, raspoloživ na: <https://www.hammerdb.com/>;
- [8] Cudre-Mauroux Philippe, Kimura Hideaki, Lim Kian-Tat, Rogers Jennie, Madden Samuel, Stonebraker Michael, Zdonik Stanley B., Brown Paul G. "SS-DB: A Standard Science DBMS Benchmark" XLDDB 2010, Stanford University, CA, Oct. 6-7, 2010.
- [9] McKnight William, Dolezal Jake and Barker Roger, "Cloud Database Performance Benchmark, Product Profile and Evaluation: Actian Vector and Microsoft SQL Server", MCG Global Services, February 2018.
- [10] Alagić Suad, „Object-oriented Database Programming“, Springer-Verlag, 1989;
- [11] Stanišević I., Obradović S., Vičentić M., „Database Management System Selection“, International Scientific Conference UNITECH 2019, pp. II-33 – II-38, Gabrovo, Bulgaria, EU, 15-16 November 2019;
- [12] Emison J.M., "2014 State of Database Technology", Information Week, San Francisco, CA, Rep. R7770314, 2014;
- [13] DB-Engines.com, DB-Engines Ranking, on-line, August 2019., raspoloživo na: http://db-engines.com/en/ranking_trend;
- [14] Stanišević I., Obradović S., Vičentić M., „DBMS Response Speed Testing System“, International Scientific Conference UNITECH 2019, pp. II-39 – II-43, Gabrovo, Bulgaria, EU, 15-16 November 2019;

ABSTRACT

In order to select optimal DBMS for IS developed for Valjevo Business School, testing two available options (i.e. MS SQL and MySQL) was performed. Instead of using benchmark software, simple software to test features relevant for the system has been developed. This paper describes applied methodology and displays realised results.

MS SQL AND MySQL RESPONSE RATE COMPARISON

Stanišević Ilja, Obradović Slobodan, Vičentić Mladen