

Никола Миленковић,¹ Милош Радосављевић²

ПОБОЉШАЊЕ РАДА ИНФОРМАЦИОНОГ СИСТЕМА КРОЗ ОПТИМИЗАЦИЈУ БАЗЕ ПОДАТАКА

UDK: 005.591.1:004.65

Оригинални научни рад

Rezime

Оптималне перформансе базе података доприносе бољем коришћењу ресурса, осигуравајући да су улагања у хардвер и инфраструктуру максимизирана. То значи уштеду трошкова и већи поврат улагања. Штавише, поједностављене операције базе података ослобађају вредне рачунарске ресурсе, омогућавајући предузећима да се фокусирају на иновације, развој производа и стицање конкурентске предности.

У овом раду ћемо проћи начине за побољшање перформанси базе података. Применом ових техника, предузећа могу да убрзају своје апликације, смање време застоја и пруже својим корисницима искуство без премца које подстиче лојалност и покреће успех на данашњем тржишту са великом конкуренцијом.

Кључне рећи: оптимизација, SQL, анализа упита.

¹ Naucni saradnik, Institut primenjenih nauka u Beogradu, Lomina 2, R. Srbija, nikola.milenkovic@gmail.com

² Istrazivac saradnik, Institut primenjenih nauka u Beogradu, Lomina 2, R. Srbija, mradosavljevic93@gmail.com

Увод

Базе података су срж сваке пословне апликације и одговорне су за прикупљање, складиштење и преузимање великих количина података. Ово централизовано прикупљање података је од виталног значаја за пословање организације (Duan, 2009), јер преноси битне информације и олакшава критичне пословне процесе.

Због тога ће лоцирање проблема са базом података и њихово благовремено решавање ради побољшања перформанси базе података обезбедити да ваше базе података функционишу на врхунској ефикасности и да су поремећаји у пословању сведени на минимум.

У овом чланку ћемо детаљније погледати како да побољшате перформансе ваше базе података да бисте обезбедили искуство без проблема уз одржавање продуктивности у целој организацији. Када је укупна перформанса базе података мањкава, било да се ради о лошој структури података или проблему индексирања, продуктивност и профитабилност могу бити погођени у целој организацији.

Анализа системског окружења

Пошто су сервери базе података домаћин свих ваших процеса и покретачка снага перформанси апликација, императив је да у сваком тренутку имају довољно хардвера и ресурса.

Ако имате време одговора дуже од нормалног или друге проблеме са перформансама, можда је време да проверите ЦПУ, меморију и простор на диску сервера.

Важно је да пазите на време спремности вашег ЦПУ-а, што вам говори колико је пута ваш систем покушао да користи ЦПУ, али није могао јер није било довољно доступних ресурса (Maesaroh, 2022).

Ово ће вам помоћи да боље разумете искоришћеност вашег ЦПУ-а и утврдите да ли треба да надоградите на већи ЦПУ. На крају крајева, што је ваш ЦПУ моћнији, то ће боље моћи да обрађује више апликација и захтева, чиме се побољшава перформансе базе података.

Ако вашим серверима недостаје меморија, квар ваше базе података је неминован. Један од начина да утврдите да ли вам је потребно више меморије је да проверите број грешака страница које ваш систем има.

Ако је број грешака велики, то значи да вашим серверима понестаје (или потпуно нема) доступне меморије. Додељивање више меморије вашим серверима ће сигурно помоћи у оптимизацији перформанси базе података.

Када постављате нови сервер базе података, најбоље је да ускладиштите своје податке, евиденцију и датотеке резервних копија на одвојеним дисковима. Не само да ће ово минимизирати фрагментацију диска уз побољшање перформанси, већ ће такође бити изузетно

згодно у случају опоравка од катастрофе (Györfödi, 2015).

Такође је важно узети у обзир типове дискова на вашем серверу. Милиони И/О операција се могу користити за приступ или враћање потребних података за један упит.

Одабир ССД (енг. Solid State Drive) хард дискова може да обезбеди снагу вашем SQL серверу, Oracle бази података или другом систему за управљање релационим базама података за оптималне перформансе.

Оптимизација рада базе података

Део подешавања перформанси базе података може укључивати оптимизацију упита, јер проблеми са перформансама често могу бити повезани са лошим упитима.

На најосновнијем нивоу, упит је захтев у реалном времену за податке из базе података. Да бисте побољшали перформансе базе података, добра је идеја да оптимизујете најчешће упите које ваш сервер базе података прима (Janjua, 2022).

Да бисте започели процес оптимизације упита, требало би да циљате одређене упите који имају значајан утицај на време извршења упита, укључујући упите који су повремено или доследно спори или имају црвене заставице.

Ручно оптимизовање упита може бити тешко и дуготрајно, тако да коришћење оптимизатора упита или ангажовање спољних сарадника за оптимизацију може помоћи у побољшању перформанси базе података.

Поред упита, индекси су још један суштински елемент базе података. Индексирање ствара „структуру“ која помаже да ваши подаци буду организовани, што заузврат олакшава њихово лоцирање (Marathe, 2022).

Правилно индексирање може побољшати перформансе ваше базе података јер повећава ефикасност процеса преузимања података, штедећи ваше и системско време и труд.

Праћење ко – и шта – има приступ вашој бази података у сваком тренутку је кључно за стално побољшање перформанси базе података.

Разумевање свих апликација и услуга које приступају вашој бази података може вам помоћи да одредите уска грла у перформансама за одређени извор.

Ако једна услуга има проблема са перформансама, али сте у могућности да утврдите да је она садржана у тој једној услузи, то ће вероватно бити брзо решење са минималним утицајем на ваше друге апликације.

Међутим, ако имате прекиде у целој бази података, то може бити већи проблем са серверима или хардвером.

Дефрагментација података је један од најефикаснијих приступа који можете предузети да повећате перформансе базе података.

Са подацима који се стално уписују и уклањају из ваше базе података, они ће неизбежно постати фрагментирани, што може успорити процес преузимања података или ометати план извршења упита.

Када дефрагментирате своје податке, релевантни подаци се могу груписати заједно, омогућавајући И/О операцијама да раде брже и ефикасније.

Примарни кључеви јединствено идентификују запис у табели базе података. Можда већ дуже време креирате примарне кључеве, али да ли то радите исправно?

Много пута администратори база података и програмери додељују вредности поља које се могу дуплирати и које се морају поновити или су склони грешкама у куцању и другим грешкама. Као опште правило, требало би да покушате да користите поље идентитета као примарни кључ где год је то могуће. Поља идентитета су цели бројеви и стога су мање склона грешкама у куцању и грешкама. Такође заузимају мање меморије, остављајући више простора за рад са базом података и побољшавајући укупне перформансе ваше базе података. Пошто су поља идентитета попут корисничких ИД-ова или УУИД-ова, већа је вероватноћа да ће бити јединствена.

Оптимизација упита ради смањења времена извршења

У већини сценарија, перформансе ваше базе података могу бити директно повезане са структуром ваших упита. У SQL бази података, начин на који структуришете или пишете своје упите утиче на време одговора базе података, односно колико дуго је потребно за преузимање података. У наставку можете прочитати о неким методама и техникама које се обично користе за оптимизацију упита базе података.

Много пута, администратори база података и програмери лежерно користе подупите да би преузели потребне податке из више табела за одређени услов. Ако нисте свесни компромиса коришћења подупита, то може изазвати много проблема са перформансама ваше базе података (Patil, 2017).

Подупити се покрећу према надређеним упитима. Ово имплицира да ће сваки подупит чекати да се изврши док се извршавање надређеног упита не заврши (Patil, 2017). Штавише, сваки подупит се покреће исти број пута као родитељски упит на једном реду табеле у вашој бази података. Ово се не дешава у случају повезивања више табела (JOIN), пошто се два упита у повезивањем покрећу као један упит. Због тога би требало да покушате да користите овај начин увезивања више табела уместо подупита кад год и где

год је то могуће за брже време одговора на упит (Rautmare, 2016).

MySQL има уграђену команду PROCEDURE ANALYSE() коју можете укључити у своје упите. Ова команда помаже у анализи ваших упита и аутоматски препоручује боље типове и величине података за колоне у вашој бази података. Ово вам, заузврат, помаже да разумете које промене можете да унесете у своје колоне да бисте смањили укупну величину табеле у смислу потрошене меморије (Stjepanović, 2015). Можете користити ову команду са било којим упитом и користити информације за реструктурирање табела, колоне и поља у вашој бази података.

Појединачни упит читања, писања, брисања или ажурирања ради најбрже када имате једну операцију читања, писања, брисања или ажурирања. Међутим, у стварној апликацији, групне операције ће бити уобичајени захтев (Tahaghoghi, 2006).

Високо ефикасна и ефикасна база података може дозволити истовремено извршавање више упита без блокирања извршења другог упита. Ово даје већу контролу конкурентности вашем систему базе података. Међутим, покретање појединачних упита одузима доста времена. Покретање сваког од ових упита један по један за обављање исте операције ажурирања или брисања смањује укупну конкурентност система базе података. Уместо тога,

можете покренути ажурирање и брисати упите у групама и покренути их заједно да бисте повећали истовременост ваше базе података. Ово доводи до бржег извршавања, остављајући простора за извршавање других упита, што вашој бази података даје већу истовременост.

Коришћење апликација за оптимизацију рада базе

SolarWinds Database Performance Analyzer надгледа време одговора ваших СКЛ изјава. Може да шаље упозорења и даје аналитику и предлоге за оптимизацију ваше базе података (Wahyudi, 2022). Он идентификује уска грла у перформансама у вашој бази података и може да надгледа базу података у било ком окружењу од тестирања, преко постављања, до производње. Штавише, можете да визуелизујете перформансе кроз графиконе и графиконе на контролној табли.

SQL Sentry врши аутоматизовано праћење ваше базе података. Брзо дијагностикује проблеме у вези са недостајућим подацима како би ваша апликација остала нетакнута. Ефикасно оптимизује сва ваша окружења базе података за управљање више података користећи постојећу инфраструктуру.

Paessler вам помаже да надгледате упите у бази података, везе, одговоре и још много тога преко прилагодљиве контролне табле. Такође можете да га користите за надгледање одређених скупова података

из ваше базе података и слање упозорења која ће вам саветовати да предузмете мере како бисте спречили застој базе података (Wang, 2023).

Redgate вам одмах саопштава статус свих ваших сервера и ваше базе података. Можете га користити за дијагностицирање базе података и прилагођавање упозорења о потенцијалним проблемима који се односе на стање вашег сервера базе података.

Закључак

Морамо да водимо рачуна о перформансама наших база података. Споре и неефикасне операције базе података могу довести до фрустрирајућих кашњења, деградиране функционалности апликације и на крају отерати кориснике. Супротно томе, база података високих перформанси обезбеђује несметан приступ критичним информацијама у реалном времену, пружајући беспрекорно корисничко искуство које омогућава да клијенти буду ангажовани и лојални.

Усавршавањем ефикасних техника оптимизације базе података, предузећа могу откључати мноштво предности. Побољшане перформансе базе података омогућавају апликацијама да брзо обрађују трансакције, смањујући ризик од уских грла током периода највеће употребе. Повећана ефикасност значи да базе података могу да поднесу већа оптерећења и скалу како би задовољиле растуће захтеве проширења базе корисника без угрожавања перформанси.

Циљ подешавања перформанси базе података је да се минимизира време одговора на ваше упите тако што ћете на најбољи начин искористити ресурсе вашег система. Најбоље коришћење ових ресурса укључује минимизирање мрежног саобраћаја, диск И/О и ЦПУ времена. Овај циљ се може постићи само разумевањем логичке и физичке структуре ваших података, апликација које се користе на вашем систему и начина на који конфликтна употреба ваше базе података може утицати на перформансе.

Најбољи начин да избегнете проблеме са перформансама је да обезбедите да проблеми са перформансама буду део ваших текућих развојних активности. Многа од најзначајнијих побољшања перформанси се остварују кроз пажљив дизајн на почетку циклуса развоја базе података. Да бисте најефикасније оптимизовали перформансе, морате да идентификујете области које доносе највећа повећања перформанси у најразличитијим ситуацијама. Фокусирајте своју анализу на ове области.

Literatura

1. Duan, S., Thummala, V., Babu, S. (2009) Tuning database configuration parameters with iTuned. Proceedings of the VLDB Endowment, 2(1): 1246-1257
2. Győrödi, C., Győrödi, R., Pecherle, G., Olah, A. (2015) A comparative study: MongoDB vs. MySQL. u: 2015 13th

- International Conference on Engineering of Modern Electric Systems (EMES), IEEE, 1-6
3. Janjua, J.I., Khan, T.A., Zulfiqar, S., Usman, M.Q. (2022) An Architecture of MySQL Storage Engines to Increase the Resource Utilization. u: 2022 International Balkan Conference on Communications and Networking (Balkan Com), IEEE, 68-72
 4. Maesaroh, S., Gunawan, H., Lestari, A., Tsaourie, M.S.A., Fauji, M. (2022) Query Optimization In MySQL Database Using Index. International Journal of Cyber and IT Service Management, 2(2): 104-110
 5. Marathe, A.P., Lin, S., Yu, W., El, G.K., Larson, P.Å., Sun, C. (2022) Integrating the Orca Optimizer into MySQL. u: EDBT, 2-511
 6. Patil, M.M., Hanni, A., Tejeshwar, C.H., Patil, P.M. (2017) A qualitative analysis of the performance of MongoDB vs MySQL database based on insertion and retrieval operations using a Web/android application to explore load balancing: Sharding in MongoDB and its advantages. u: 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), IEEE, 325-330
 7. Rautmare, S., Bhalerao, D.M. (2016) MySQL and NoSQL database comparison for IoT application. u: 2016 IEEE international conference on advances in computer applications (ICACA), IEEE, 235-238
 8. Stjepanović, D., Savić, M., Jokić, J., Marić, S. (2015) Performance measurements of some aspects of multi-threaded access to key-value stores. u: 2015 23rd Telecommunications Forum Telfor (TELFOR), IEEE, 831-834

9. Tahaghoghi, S.M., Williams, H.E. (2006) Learning MySQL: Get a Handle on Your Data. O'Reilly Media, Inc
10. Wahyudi, J., Asbari, M., Sasono, I., Pramono, T., Novitasari, D. (2022) Database Management Education in MYSQL. Edumaspul: Jurnal Pendidikan, 6(2): 2413-2417
11. Wang, B., Dai, L., Liao, B. (2023) System Architecture Design of a Multimedia Platform to Increase Awareness of Cultural Heritage: A Case Study of Sustainable Cultural Heritage. Sustainability, 15(3): 2504

Nikola Milenković, Miloš Radosavljević

IMPROVING THE OPERATION OF THE INFORMATION SYSTEM THROUGH DATABASE OPTIMIZATION

Abstract

Optimal database performance contributes to better resource utilization, ensuring that hardware and infrastructure investments are maximized. This means cost savings and a higher return on investment. Furthermore, simplified database operations free up valuable computing resources, allowing businesses to focus on innovation, product development and gaining a competitive advantage. In this paper, we will go over ways to improve database performance. By applying these techniques, businesses can speed up their applications, reduce downtime, and provide

their users with an unparalleled experience that fosters loyalty and drives success in today's highly competitive marketplace.

Key words: optimization, SQL, query analysis.

Datum prijema (Date received): 11.08.2023.

Datum prihvatanja (Date accepted): 23.10.2023.