



Selvitysraportti

**Tietokannan hallinta IIO30200
MySQL nykytila ja tulevaisuus
MySQL Query Analyzer**

**Jarkko Aalto (D4410)
Ossi Rantapuska (E1141)**

**Raportti
Huhtikuu 2010**



CREATIVE COMMONS LISENTOITU

<http://creativecommons.org/licenses/by-nc-sa/1.0/fi/>

Ohjelmistotekniikan koulutusohjelma



**JYVÄSKYLÄN
AMMATTIKORKEAKOULU**

SISÄLTÖ

KUVIOT	1
1 JOHDANTO	3
2 MYSQL TIETOKANTA JÄRJESTELMÄ.....	3
2.1 MySQL nykytilanne.....	3
2.1.1 Sun ostaa MySQL:n	3
2.1.2 Oracle ostaa Sunin	4
2.1.3 Nykyiset MySQL:n heikkoudet.....	4
2.1.4 MySQL:n nykypäivä.....	4
2.2 MySQL tulevaisuuden näkymät.....	4
2.2.1 MySQL kehityksen alalla.....	5
2.2.2 Oracle keskustelee MySQL-tietokannan tulevaisuudesta	6
2.2.3 Falcon tietokantamoottori on hyllytetty.....	6
3 MYSQL QUERY ANALYZER	7
3.1 Johdanto.....	7
3.2 Yleiskuva	8
3.3 Asennus.....	9
3.4 Käyttö.....	10
3.5 Query Analyzerin käyttötapauksia	12
3.6 Ongelmatilanteita	17
LÄHTEET.....	18

KUVIOT

Kuvio 1. MySQL Query Analyzerin arkkitehtuuri	8
Kuvio 2. MySQL Query Analyzerin Konfiguraatio.....	9

Kuvio 3. MySQL Enterprise Dashboard	10
Kuvio 4. Query Analyzerin perusnäkyvä.	11
Kuvio 5. Kyselystä tehty Explain.....	12
Kuvio 6. Query Analyzer käytössä ohjelmakehityksessä ja laadunvarmistuksessa...	13
Kuvio 7. Query Analyzer ja näytteenotto tuotantokyselyistä ei-ruuhkaisina aikoina .	14
Kuvio 8. Query Analyzer ja tuotannon osajoukko monitorointi	15

1 JOHDANTO

Selvitysraportti on osa tietokannan hallinta IIO30200 -opintojakson toteutusta, jossa opiskelijat perehtyvät ryhmäkohtaisesti johonkin DBMS työkaluun tai esittelevät jonkun tietokannan hallintatyökalun. Kaikki selvitysraportit ovat Creative Commons -lisensoituja. Oletusarvoisesti käytetään Attribution-Noncommercial-Share Alike 3.0 -lisenssiä, jolloin raportteja saa käyttää ei-kaupallisiin tarkoituksiin, ks.

<http://creativecommons.org/licenses/by-nc-sa/3.0/>.

Ryhmämme tehtävänä on selvittää MySQL-tietokannan nykytila ja tulevaisuudennäkymät ja lisäksi MySQL Query Analyzer -työkalun esittely.

2 MYSQL-TIETOKANNANHALLINTAJÄRJESTELMÄ

2.1 MySQL:n nykytilanne

Oracle on yrittänyt ostaa vuonna 2006 silloin vielä itsenäisen MySQL:n. Yrityskauppojen epäonnistuttua Oracle hankki vuonna 2006 Innobasen, jonka InnoDB-tietokantamoottori oli tuolloin käytössä MySQL:ssä. Tällöin kaupan arveltiin aiheuttavan MySQL:lle pahoja ongelmia. Tällöin avoimen MySQL joutuminen enimmäkseen suljettuja tuotteita tarjoavan Oraclen hoteisiin olisi voinut merkitä, että toinen merkittävä avoimen koodin tietokanta Postgres olisi saanut selvästi merkittävämmän aseman tietokantamarkkinoilla.

2.1.1 Sun ostaa MySQL:n

Vuonna 2008 Sun osti MySQL-tietokantayhtiön miljardilla dollarilla. Sun on halukas kehittämään MySQL:ää avoimen lähdekoodin tuotteena, koska Sunilla ei ollut omaa kilpailevaa tuotetta tietokantasegmentillä. Mutta parin kuukauden jälkeen Sunin johdoto halusi lisätä tuotteeseen suljettuja palasia, kuten varmuuskopioinnin.(Lehto, T. 2010.).

2.1.2 Oracle ostaa Sunin

Vuonna 2009 Oracle tekee ostotarjouksen ja ostaa Sunin ja samalla MySQL tietokannan 7,4 miljardilla dollarilla. Ostaessaan Sunin se on antanut kymmenen sitoutumista siitä että kehitys jatkuu, mutta se ei lupaa sitoutua niihin.

2.1.3 Nykyiset MySQL:n heikkoudet

Tuotantotiimi näkee julkaistun MySQL 5 - tuoteuudistuksessa virheenä sen, että he lisäsivät sellaisia ominaisuuksia, joita asiakkaat eivät edes ole ottaneet käyttöön. (Lehto, T. 2010.)

2.1.4 MySQL:n nykypäivä

MySQL on nykypäivän suosituin Open Source tietokanta kahdesta pääsyystä: pienempi kokonaisomistushinta verrattuna kilpaileviin vaihtoehtoihin, sekä suorituskykynsä, luotettavuutensa ja helppokäyttöisyytensä vuoksi. Nämä asiat auttavat minkä tahansa kokoisia yrityksiä rakentamaan halpoja, saatavuudeltaan hyviä online-sovelluksia jotka skaalatutuvat kun liiketoiminta kasvaa. Kun nämä sovellukset ja MySQL-palvelimet, jotka pyörittävät niitä, kasvavat määrissä ja tulevat liiketoiminnan kannalta kriittisiksi. Tällöin katkokset ja hitaat vasteajat tulevat kalliimmiksi.

Tavallisista päivityksistä Enterprise-palvelimiin ja mikrotason SQL-analytiikkaan, MySQL Enterprise ja sen sarja tietokanta- ja monitorointiohjelmistoja ja tuotantoa tukevia palveluita on kehitetty silmälläpitäen sitä, että jokainen sovellus ja ylläpidetty MySQL-palvelin toimii korkeimmilla mahdollisilla tietoturvan, suorituskyvyn ja saatavuuden tasoilla.

2.2 MySQL tulevaisuuden näkymät

Erottuaan Sunin palvelusta Michael "Mony" Widenius on ryhtynyt kehittämään 20 hengen tiimillä gpl-lisenssin alaisuudessa tietokantaa nimeltään Mariadb, koska he eivät voi käyttää nimeä MySQL, koska kaupallisten asiakkaiden oikeudet ovat Sunilla ja se ei salli nimen käyttöä. (Lehto, T. 2010).

2.2.1 MySQL kehityksen alalla

Edwards Screver, Oraclen tärkein yritysohjelmistoarkkitehti ja sen avoimen lähdekoodin strategian päällikkö kaavailee pääalustuksena MySQL:in käyttäjille, että MySQL:n jatkuva kehittäminen on Oraclen edun mukaista. (Babcock, C, 2010.)

Haastattelussa maanantaina 12.4.2010 Screven sanoi, että hän aikoi kertoa käyttäjäperustelun "miksi MySQL on tärkeä Oraclelle". Avoimen lähdekoodin järjestelmä on niin erilainen kuin 11g-tietokanta-järjestelmä. Tämän takia Oracle on vähän huolissaan siitä, että MySQL vaikuttaa tulevaan Oraclen myyntiin. (Babcock, C, 2010.)

"MySQL on pieni, kevyt, helppoa asentaa", Screven pani merkille. "Nämä ovat mielenkiintoisia ominaisuuksia. Niillä tavoitetaan asiakassegmentti johon Oracle ei muuten pääse. On järkevä investoida koko ajan MySQL:n jatkuvaan kehitykseen", hän sanoi. Hän vakuuttaa uudelleen MySQL:n jatkuvan kehityksen, mitä hän kutsuu nimellä "State of the Dolphin". MySQL käyttää vielä Delfiiniä yrityssymbolina. (Babcock, C, 2010.)

Screven ilmoitti beeta MySQL 5.5 olevan saatavilla. Sen InnoDB tietokantamoottori, jota käytetään usein lisäosana MySQL:n käyttäjien toimesta, tulee oletusarvoiseksi varastointimoduuliksi. MySQL sai alkunsa nopeana, lue vain tietokanta ohjelmana, joka oli yksinkertainen toteuttaa web-sivujen palveluita varten. Sekä kaupallisesti tuettu versio että vapaasti ladattava, avoimen lähdekoodin MySQL:ssä InnoDB yhdistetään toisiinsa ja käyttäjille. (Babcock, C, 2010.)

Lisäksi MySQL Workbench release 5.2 on paranneltu ja mahdollistaa MySQL-tietokannan ylläpitäjän rakentaa visuaalisesti tietomallin, joka toteutetaan MySQL:llä. MySQL Monitoria on parannettu analysoimaan paremmin kyselyssä paljastamaan mitä tietokannan resurssit tarvitsevat ja missä pullonkaulat sijaitsevat. (Babcock, C, 2010.)

Oracle on myynyt erillisen tuotteen, joka toimii InnoDB:n tietokantamoottorilla, kuumatuki joka sallii datan pysyvät varmuuskopiot, kun tietokanta toimii. Oraclessa MySQL:n ja InnoDB:n Hot Backup-toiminto sisällytetään pakettiin, Screven sanoi. (Babcock, C, 2010.)

Olemme päättäneet, että teemme MySQL:stä paremman ja aiomme jatkaa investointeja. Monet haastatelluista kysyvät mitä motivaatiota Oraclella oli hankkia MySQL (osana Sun Microsystems)," sanoi Screven yrittäen poistaa pelot. (Babcock, C, 2010.)

Useita kokonaisjärjestelmän suorituskyvynparannuksia on myös tehty, hän lisäsi. Ei ennen yleistä jakoa, jota ei saada ennen MySQL 5.5:n julkaisupäivää, hän lisäsi. (Babcock, C, 2010.)

2.2.2 Oracle keskustelee MySQL-tietokannan tulevaisuudesta

Mikä teki MySQL:n onnistuneeksi, hän selitti, että se on kevyt ja helppo asentaa ja hoitaa. "On hyvin helppoa ohjelmoijille aloittaa MySQL:n kanssa, se on hyvin tärkeä ominaisuus, tästä me haluamme pitää kiinni.", hän sanoi. "Me haluamme käyttää – ja meidän tarvitsee käyttää – MySQL:ää. Tämän osoittaa osa tietokantamarkkinoista, jossa me olemme nykyisin Oraclen tietokannan kanssa mukana." (Prince, B. 2010.)

Oracle julkaisi 13.4.2010 MySQL 5.5 beta-julkaisun, jonka parempia piirteitä ovat metatietojen lukitseminen, parannukset kahdentumisessa ja muissa parannuksissa.

Tietokannassa esitellään InnoDB ilman mitään lisäkustannuksia. MySQL 5.5 lisäksi Oracle myös ilmoitti ohjelmistovapautuksen MySQL Clusterista 7.1, joka esittelee MySQL Cluster Managerin auttaakseen käyttäjiä automatisoimaan MySQL Clusterin tietokannan hallinnon. (Prince, B. 2010.)

2.2.3 Falcon tietokantamoottori on hyllytetty

Falcon tietokantamoottori on hyllytetty, sanoo Edward Screven. "Luulen, että Falcon tietokanta moottori on vanhentunut. Falcon oli Sunin, tai itse asiassa MySQL AB:n vastaus InnoDB:n Oraclen ostamiseen", Screven sanoi. "Ihmiset olivat huolestuneita – mikä on väärin – että me jotenkin yrittäisimme vahingoittaa MySQL:ää tekemällä jotakin InnoDB:lle. Tietenkään me emme tee mitään InnoDB:lle lukuun ottamatta sitä, että teemme siitä paremman. Joten nyt kun me olemme saattaneet ryhmät yhteen, Falconilla ei ole paikkaa maailmassa." (Prince, B. 2010.)

Screven sanoi tulevaisuudesta, että Oracle pyrkii tekemään helpommaksi ihmisille muuttaa dataan edestakaisin MySQL:n ja Oraclen tietokantojen välillä ja tuomme Oraclen piirteitä kuten Secure Backupin, Enterprise Manager ja Audit vault

MySQL:ään. Kuitenkaan käyttäjien ei pitäisi odottaa lisää yhdentymistä Oraclen koko sovellussalkussa. (Prince, B. 2010.)

"Nyt meidän perinteiset tuotteet, kuten Back Office ja CRM-sovellukset luottavat paljon hyvin erikoistuneeseen Oraclen tietokannan toiminnallisuuteen ja minä arvelen, että ei olisi mahdollista rakentaa MySQL:n tällaista toiminnallisuutta täysin muuttamatta MySQLin luonnetta", hän sanoi. (Prince, B. 2010.)

Kaiken kaikkiaan on hyvin tärkeää Oraclelle, että MySQL pysyy "kaikkialla läsnä olevana". "Olisi hullua suututtaa kaikki ne ihmiset, jotka käyttävät MySQL:ä ilmaiseksi ja rakentavat sen päälle sovelluksia. Oracle toivoo, että monet niistä lopulta muuttuvat maksaviksi asiakkaiksi. Niinpä se on tärkeä meille yritysnäkökulmasta pitääkseen yhteisön elinkelpoisena ja pitääkseen jäsenet innostuneena MySQL:stä." (Prince, B. 2010.)

3 MYSQL QUERY ANALYZER

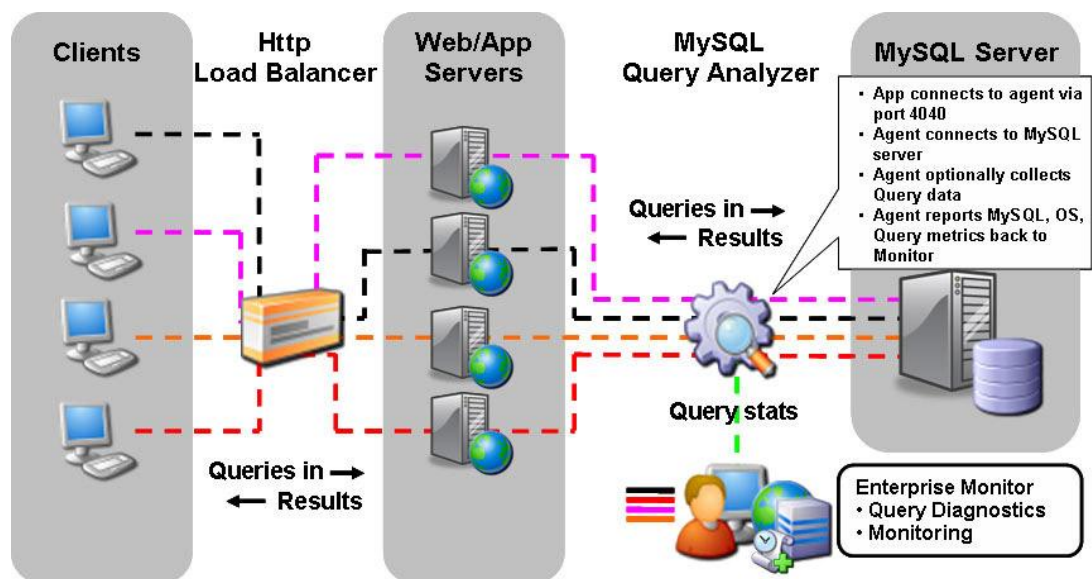
3.1 Johdanto

Yksi suurimmista haasteista MySQL:n päälle rakennettujen sovellusten kehityksessä ja ylläpidossa on se, että ajan myötä MySQL-palvelimien määrä tuppaa monistumaan nopeaan tahtiin. Erään MySQL-käyttäjille tehdyn kyselyn mukaan keskimääräinen MySQL:n DBA ylläpitää viidestä kymmeneen fyysistä MySQL-palvelinta. Sama kysely paljasti myös että he, jotka käyttävät MySQL:n replikaatiota skaalaamaan suurten tietoliikennemäärien web-sovelluksia, lisäävät uuden lukuorjan topologiaansa joka toinen kuukausi - ja jokaisella lisätyllä orjalla tietysti on omat ylläpito- ja monitorointitarpeet. MySQL Enterprise Monitor auttaa hallitsemaan useampia MySQL-palvelimia vähemmällä ajalla ja vaivalla.

3.2 Yleiskuva

MySQL Query Analyzer on MySQL Enterprisen mukana tuleva työkalu, jonka avulla voidaan kehittää sovelluksen suorituskykyä. Sillä kyetään monitoroimaan kyselyjen suorituskykyä ja määrittämään tarkasti mikä osa SQL-koodista toimii mahdollisesti pullonkaulana. Ongelmakyselyt voidaan suodattaa ja analysoida tietyltä halutulta aikaväliltä, mikä tekee ongelmakohtien löytämisestä vielä helpompaa. (MySQL Query Analyzer Overview)

Kyselyt näytetään koostettuna näkymänä kaikista MySQL-palvelimista, ja Query Analyzerin avulla pystytään myös tarkastelemaan suorituskertojen määriä, suoritusaikojen yhteismääriä ja tietojen kokonaismäärää. MySQL Query Analyzerin avulla on mahdollista kehittää SQL-koodia aktiivisen kehityksen aikana ja jatkuvasti monitoroida ja optimoida työn alla olevia kyselyitä. (MySQL Query Analyzer Overview)



Kuvio 1. MySQL Query Analyzerin arkkitehtuuri

MySQL Query Analyzer antaa käyttäjälle syvällisen näkymän SQL-koodista tiivistäen kaikki kyselyt ja koostaen suoritustilastot yksittäiseen paneeliin. Kaikki tämä mahdol-

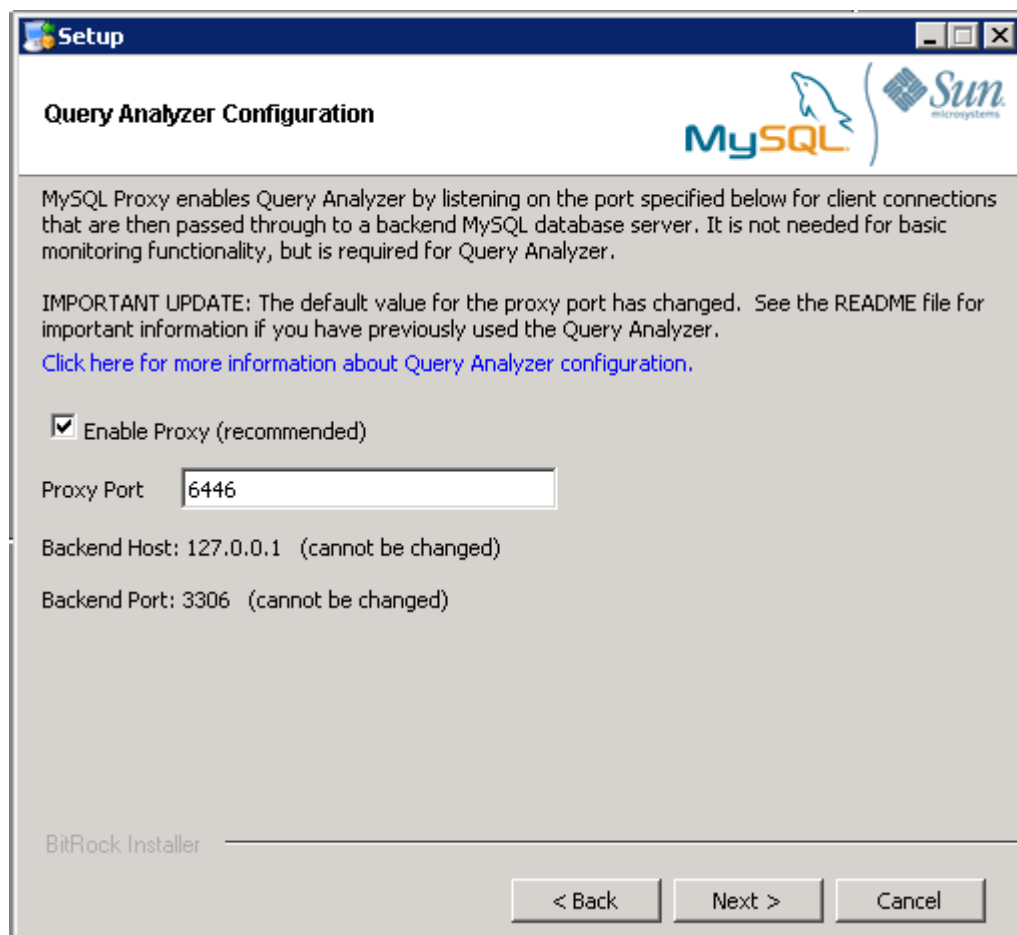
listaa parhaan mahdollisen suorituskyvyn MySQL:llä. (MySQL Query Analyzer Overview)

3.3 Asennus

MySQL Query Analyzer asentuu Service Agents-asennuspaketin (Palveluagentit) kanssa. Se kytketään päälle asennuksen aikana. Sen tilaa pystytään vaihtamaan Collection- ja Pass thru-tilojen välillä.

Ennen Agentin asentamista tarvitsee olla asennettuna Service Manager (Palvelumanageri), sillä palveluagentit kommunikoivat sen kanssa. Palvelumanagerin asennuksen ja konfiguroinnin jälkeen voidaan asentaa palveluagentit.

Palveluagenttiasennuksen edetessä tiettyyn pisteeseen tulee Query Analyzeriin liittyvä asetusikkuna (Kuvio 2).



Kuvio 2. MySQL Query Analyzerin Konfiguraatio

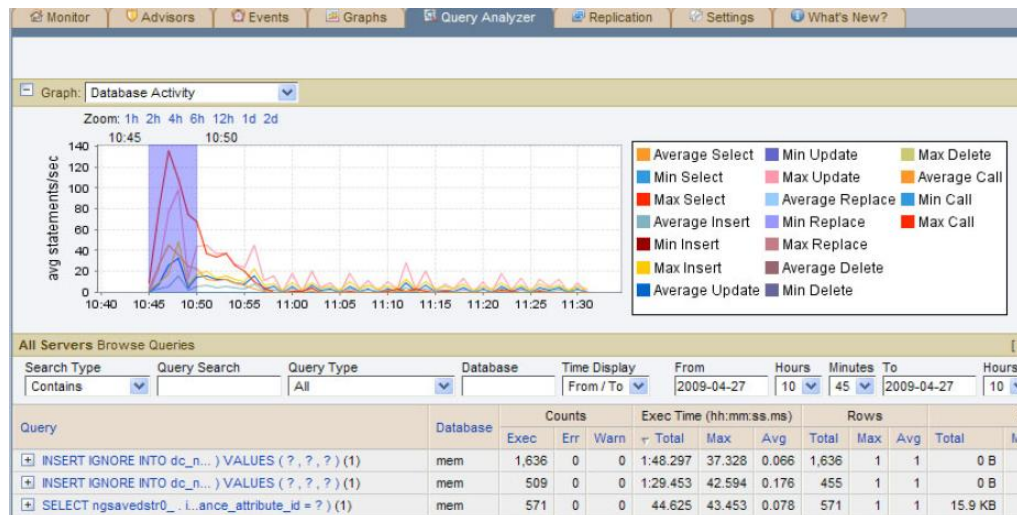
Jotta Query Analyzer saadaan käyttöön, tulee valita kohta "Enable Proxy (recommended)", ja kertoa portti jota haluaa analyzerin kuuntelevan. Query Analyzer on integroitu osaksi MySQL Enterprise Dashboardia, joten onnistuneen asennuksen jälkeen siihen pitäisi päästä Dashboardin kautta käsiin. Kuviossa 3 näkyy MySQL Enterprise Dashboardin perusnäky.



Kuvio 3. MySQL Enterprise Dashboard

3.4 Käyttö

Valitsemalla Dashboardin ylävalikosta "Query Analyzer" pääsee kuvan 3 mukaiseen näkymään. Näkymässä näytetään tiivistelmä kaikista suoritetuista kyselyistä tilastoina.



Kuvio 4. Query Analyzerin perusnäky.

Huonoimman suorituskyvyn kyselyistä näytetään yksityiskohtaisia tilastoja jotta voidaan analysoida minkä asioiden kombinaatiot johtavat suorituskyvyn heikkenemiseen:

- Näkymät huonoimman suorituskyvyn lausekkeista
- Suoritus aika tietyltä tapahtumalta
- Suorituksen aika ja päivämäärä
- Mistä sessiosta, käyttäjästä, säikeestä kysely on lähtöisin
- Lähde / päämääräpalvelin ja MySQL-palvelin
- Integroitu EXPLAIN huonoimman suorituskyvyn kyselyistä

EXPLAIN

Sen jälkeen kun ongelmakysely on tunnistettu, voidaan käyttää EXPLAIN-lausetta analysoimaan

- Onko indeksejä käytetty tehokkaasti
- Onko tiedostojen järjestäminen tarpeen
- Mitä tauluja ja sarakkeita kyselyissä käytetään eniten

Esimerkki syötetystä Explain-lausekkeesta:

```
EXPLAIN SELECT part_num FROM `inventory`.`parts` WHERE
(`vendor` = "foo") ORDER BY `delivery_datetime` DESC LIMIT 100;
```

Esimerkki palautetusta lauseesta:

```
ID select_type table type possible_keys key key_len ref rows Extra
```

Canonical Query Example Query Explain Query									
Explain of a query that occurred during the Time Span									
Explain									
id	select_type	table	type	possible_keys	key	key_len	ref	rows	extra
1	PRIMARY	p	ALL	null	null	null	null	16334	Using where
2	DEPENDENT SUBQUERY	payment	ref	idx_fk_customer_id,ix_customer_paydate	ix_customer_paydate	2	sakila.p.customer_id	14	Using index

hide

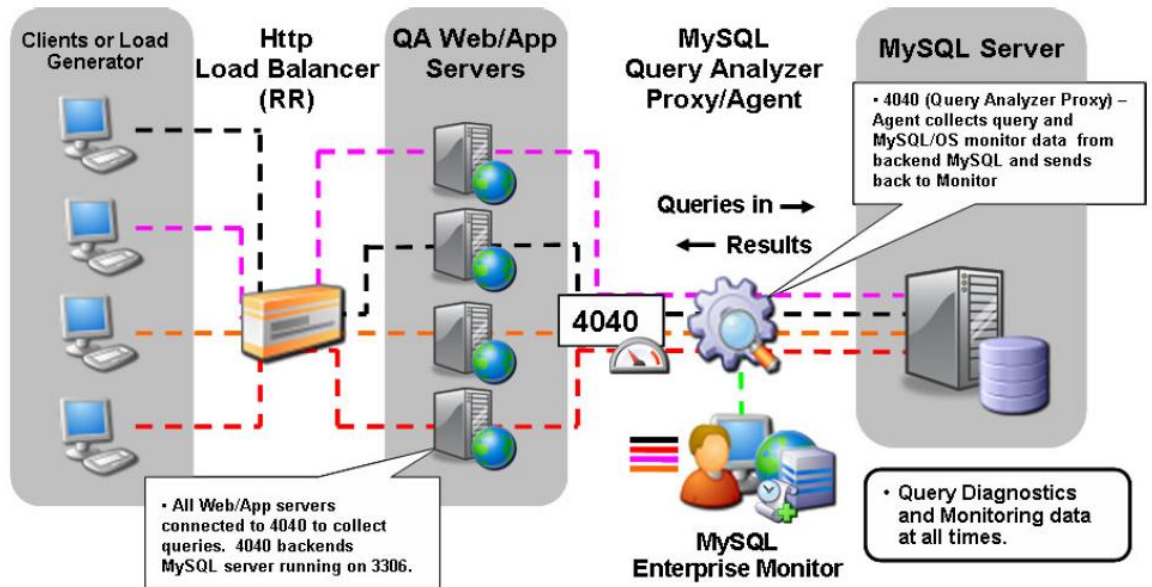
Kuvio 5. Kyselystä tehty Explain.

3.5 Query Analyzerin käyttötapauksia

Query Analyseriä voidaan käyttää kehityksessä, laadunvarmistuksessa ja kehitysympäristöissä SQL-koodin diagnosointiin ja optimointiin. Alla on listattuna yleisimpiä tapauksia:

Kyselyiden monitorointi kehityksessä ja laadunvarmistuksessa

Ohjelmoijat voivat käyttää Query Analyseria ennakoivaan monitorointiin. He kykenevät sen avulla myös virittämään ohjelmakoodia ennen käyttöönottoa. Tässä tapauksessa sovellukset liitetään suoraan porttiin jota palveluagentti kuuntelee (kustomoita-vissa, oletusarvona portti 4040). Yleinen tapaus on seuraavanlainen:



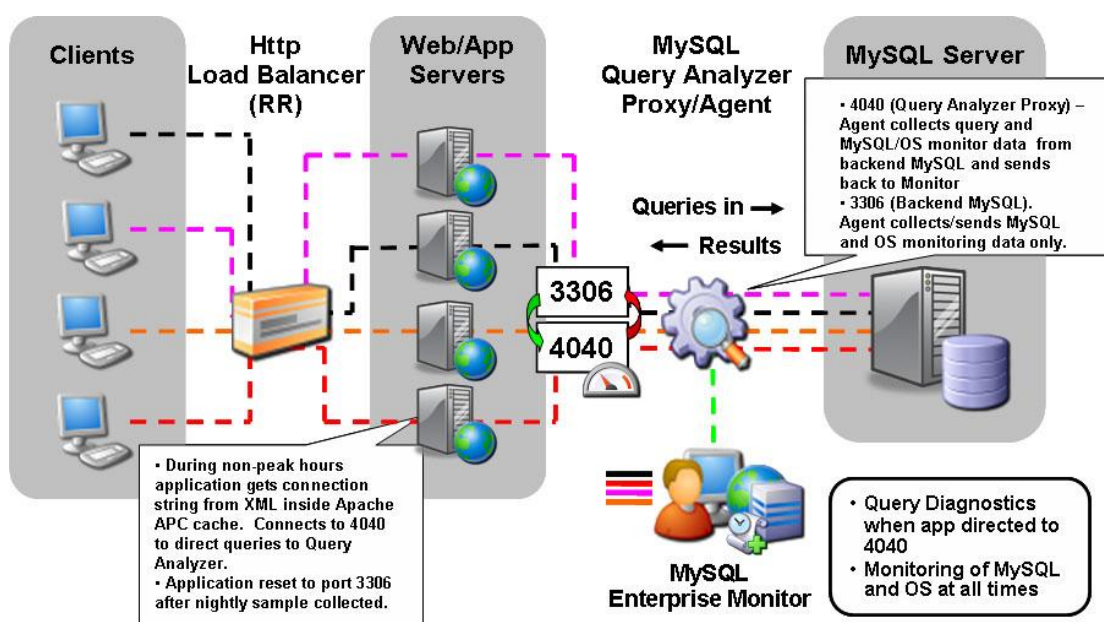
Kuvio 6. Query Analyzer käytössä ohjelmakehityksessä ja laadunvarmistuksessa

On tärkeää huomata, että palveluagentti jatkaa MySQL- ja käyttöjärjestelmäspesifisten tietojen keräämistä Enterprise Monitorille niin pitkään kuin sovellus on yhteydessä palveluagentin porttiin.

Kyselyjen monitorointi tuotantoympäristöissä

Query Analyzer on arvokas tietokannan hallinnoijille ja järjestelmänvalvojille, koska se auttaa löytämään sellaiset SQL-koodit, joita ajetaan liian usein, joiden resurssivaatimukset palvelimien suhteen ovat korkeat, tai jotka ovat vasta lähiaikoina laitettu käytäntöön ja aiheuttavat suorituskyvyn pudotusta. On tosin tärkeää huomata, että Query Analyzer itse myös pudottaa hieman suorituskykyä, kun sen tarvitsee osoittaa kyselyitä palveluagentille. Vaikutuksen suuruun riippuu monitoroidusta sovelluksesta, keski- ja huippukyselytaakoista ja kyselyn/palautettavan datan sisällöstä. Query Analyzerin negatiivinen vaikutus suorituskykyyn on kuitenkin hyötyihin nähden sen verran pieni, että monien tietokannan hallitsijoiden mielestä on hyväksyttävää pitää Query Analyseria toiminnassa tuotannon joka vaiheessa, ainakin joidenkin sovellusten tapauksessa.

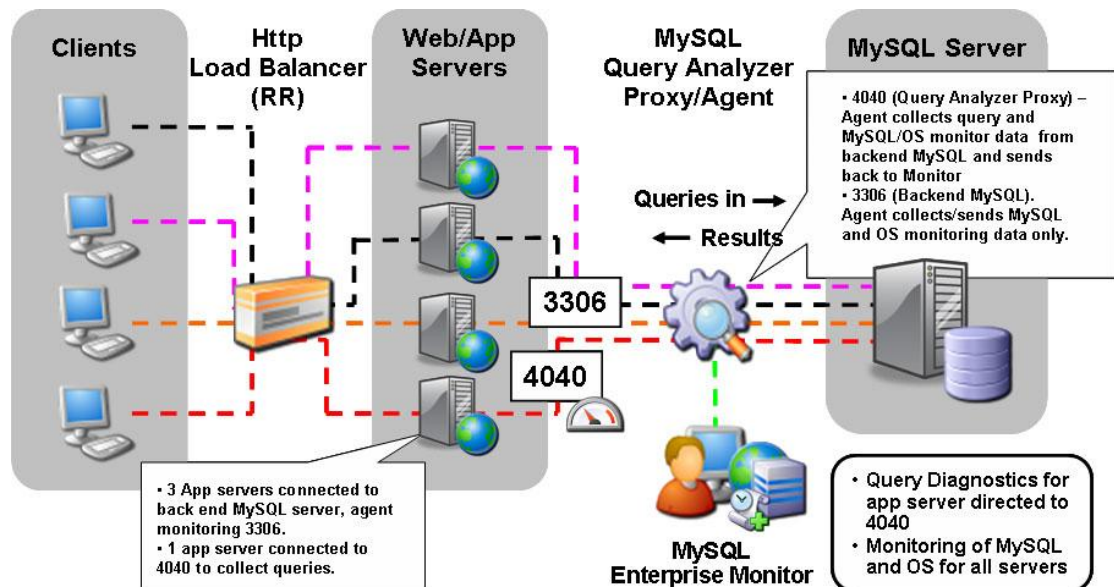
Query Analyzerille on myös vaihtoehtoisia hyödyntämistapoja. Yksi sellainen on suunniteltu näytteidenotto kyselyistä sellaisina ajankohtina jolloin kyselyjen määrä ei ole huipussaan. Työkalulla voidaan myös monitoroida tuotantojärjestelmien osajoukkoja jotka käsittelevät tyypillisiä monitoroidun sovelluksen taakkoja. Alla on esimerkiksi siitä, kuinka Query Analyzerilla voidaan tehdä otos tuotantokyselyvirrasta ei-ruuhkaisina aikoina.



Kuvio 7. Query Analyzer ja näytteenotto tuotantokyselyistä ei-ruuhkaisina aikoina

Tässä käytössäsovellus ja web-palvelimet saavat yhteysmerkkijojensa tiedostoista jotka on tallennettu sovelluspalvelimen välimuistiin tai tiedostojärjestelmään. Ei-ruuhkaisina aikoina sovellus käynnistetään uudestaan ottamaan yhteys palvelinagentin porttiin. Sovellus osoittaa kyselyt määrättyyn porttiin, jossa ne kerätään ja koostetaan sekä lähetetään Enterprise Monitorille analysoitavaksi. Kerätyt kyselyt ovat tyypillisiä näytteitä sellaisista kyselyistä, joita tehdään läpi vuorokauden, joten DBA pystyy näkemään mahdolliset pullonkaulat ja virittämään sovellusta nopeammin toimivaksi. Kun näyte on otettu, sovellus käynnistetään jälleen uudestaan ottaakseen yhteyden taustalla olevaan MySQL-palvelimeen.

Alla on esimerkki siitä, kuinka Query Analyzeria voidaan käyttää monitoroimaan tuotantojärjestelmien osajoukkoja jotka joutuvat käsittelemään tyypillisiä monitoroidun sovelluksen kyselyitä.



Kuvio 8. Query Analyzer ja tuotannon osajoukko monitorointi

Tässä käytössäsovelluspalvelimet ovat yhteydessä MySQL-taustapalvelimeen tai kuormitukseltaan tasapainotettuun luku-orja farmiin. Näistä yksi on konfiguroitu osoittamaan kyselyitä dedikoidulle luku-orjalle joka taas on kon-

figuroitu kuuntelemaan kyselyitä palvelinagentin portista. Kyselyt, jotka ohjataan tälle lukuorjalle, ovat tyypillisiä muillekin lukuorjille osoitettuja kyselyitä, joten DBA pystyy erottamaan pullonkaulat ja optimoimaan suorituskkyä. Käyttötapaa voi myös yhdistää tuotannon näytteistämisen kanssa niin, että tietty sovelluspalvelin osoittaa kyselyitä vain analysoitavaksi tietyllä aikavälillä. Jälleen on tärkeää muistaa, että palveluagentti jatkaa tiedon keräystä ja lähetystä Enterprise Monitorille niin pitkään kun sovellus on yhdistetty määritettyyn porttiin.

3.6 Ongelmatilanteita

Jos Query Analyzerin kanssa tulee ongelmia: joko informaatiota ei näy tai kyselyiden koko joukko jota odotetaan ei näy Query Analyzerin sivulla, on muutamia asioita joita voi tarkistaa.

Varmistaakseen että järjestelmä on konfiguroitu oikein kyselyjen analysointia varten, tarkista seuraavat asiat:

- Tarkista, että agentti on käynnissä tarkistamalla agentin loki ja palvelimen tila MySQL Service Managerista
- Tarkista agentin asetukset. Esimerkiksi Plugins-parametrin pääkonfiguraatio-tiedostossa (mysql-monitor-agent.ini) täytyy olla seuraava proxy-plugin: "plugins=proxy,agent"
- Tarkista että pystyt saamaan yhteyden agentin kautta tausta-MySQL-serveriin. Voit tehdä tämän tarkistamalla MySQL-clientilla. Tällöin täytyy syöttää samat asetukset kuin jos olisit ottamassa yhteyttä alkuperäiseen palvelimeen, mukaanlukien käyttäjänimi ja salasana
- Tarkista että MySQL client-sovellus on konfiguroitu käyttämään valittua proxy-porttia, ei todellista MySQL-porttia
- Tarkista että Query Analyzer on päällä palvelimella.

LÄHTEET

Lehto, T. 2010. Kannattiko Mysql myydä sunille? Tietokone 2/2010. Sanoma Magazines: Forssa.

MySQL Query Analyzer Overview. 2010. Viitattu 17.3.2010.

<http://www.mysql.com/products/enterprise/query.html>.

Sun Microsystems. MySQL Query Analyzer White Paper. 2009. Viitattu 1.4.2010. Ladattavissa sivulta <http://www.mysql.com/>.

Babcock, C, 2010. Oracle MySQK Development Matters. Viitattu 13.4.2010. <http://www.informationweek.com/news/software/database/showArticle.jhtml?articleID=224400002&subSection=Open+Source>

Prince, B. 2010. Oracle Discusses MySQL Database Plans. Viitattu 13.4.2010. <http://www.eweek.com/c/a/Database/Oracle-Discusses-MySQL-Database-Plans-854185/>