

Eero Karvanen

SQL-tietokantojen valvonta

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikka

Insinöörityö

10.5.2016

Tekijä(t) Otsikko	Eero Karvanen SQL-tietokantojen valvonta
Sivumäärä Aika	50 sivua 10.5.2016
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	Tietoverkot
Ohjaaja(t)	Lehtori Marko Uusitalo Team Leader Jesse Sahlman
<p>BLC Protie Oy on IT-palveluntarjoaja, jonka valvomossa valvotaan sen omaa sekä asiakasyritysten IT-infrastruktuureja ja ratkaistaan niissä esiintyviä ongelmia. Microsoft SQL Serverin tietokantojen ongelmatilanteiden ratkaiseminen kuuluu valvomon vastuualueisiin. Tietokantojen ongelmat näkyvät usein käyttäjälle sovellusten hitautena, ja ne voivat johtua lukemattomista eri tekijöistä. Ongelmien selvittäminen on usein hidasta sekä vaatii paljon asiantuntemusta tietokantojen toiminnasta.</p> <p>Työn tavoitteena oli löytää uusia tehokkaampia työkaluja tietokantapalvelinten valvontaan ja niiden ongelmatilanteiden ratkaisemiseen. Työssä testattiin kolmea eri valvontasovellusta ja niiden tietokantavalvontaominaisuuksia. Testattavat sovellukset olivat N-ablen N-central, ManageEnginen Applications Manager ja SolarWindsin Database Performance Analyzer.</p> <p>Työssä käytiin aluksi läpi tietokantojen ja erityisesti SQL Serverin teoriaa ja toimintaa. Tämän jälkeen siirryttiin testausvaiheeseen, jossa asennettiin sovellukset ja käytiin läpi niiden ominaisuuksia. Lopuksi vertailtiin sovellusten hyviä ja huonoja puolia sekä niiden eroavaisuuksia, ja niiden perusteella tehtiin päätös sovellusten ja niiden ominaisuuksien käyttöön-otosta.</p> <p>Testauksen perusteella päädyttiin hankkimaan Protien valvomon käyttöön Database Performance Analyzer ja ottamaan käyttöön Protien nykyisen valvontajärjestelmän N-centralin SQL Serverin valvontaominaisuuksia. Database Performance Analyzerin avulla pystytään valvomaan ja tutkimaan SQL-kyselyiden suoritusnopeuksia. N-centralin SQL-valvontaominaisuudet ovat helppo ottaa palvelimille käyttöön, eikä niistä jouduta erikseen maksamaan lisenssimaksuja.</p>	
Avainsanat	tietokanta, tietokantavalvonta, Microsoft, SQL Server, N-able, N-central, ManageEngine, Applications Manager, SolarWinds, Database Performance Analyzer

Author(s) Title	Eero Karvanen Monitoring SQL Server Databases
Number of Pages Date	50 pages 10 May 2016
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Data Networks
Instructor(s)	Senior Lecturer Marko Uusitalo Team Leader Jesse Sahlman
<p>BLC Protie Oy is a company providing various IT services and whose one activity is the monitoring of SQL Server databases. The primary objective here is to monitor both its own and its customers' IT infrastructure as well as to solve problems that occur in the infrastructure. The appearing problems with the databases are often noticeable to the user as slowness in application performance. These problems can be a result of various reasons, and therefore solving them is often a slow process and requires much professional expertise.</p> <p>The objective of this paper was to find new and more effective tools for database server monitoring as well as for solving the problems occurring in the servers. This study includes the testing of three different monitoring applications and a critical examination of characteristics of database monitoring. The three monitoring applications were N-able's N-central, ManageEngine's Applications Manager and SolarWind's Database Performance Analyzer.</p> <p>First, the study examines theoretical background behind databases and especially the functionality of SQL Server. This follows a testing phase in which the applications were installed and the basic characteristics defined. Finally, the found pros and cons of the applications are compared and based on this analysis, a final decision of the initialization of applications and their features is made.</p> <p>Based on the testing of applications, it was determined to deploy Database Performance Analyzer to support the monitoring department's performance. In addition, the results supported the introduction of Protie's current monitoring system N-central's SQL Server's monitoring features. Database Performance Analyzer enables the monitoring and examination of execution speed of SQL queries. Moreover, N-central's SQL monitoring is easy to implement on servers and no additional license fees need to be paid.</p>	
Keywords	database, database monitoring, Microsoft, SQL Server, N-able, N-central, ManageEngine, Applications Manager, SolarWinds, Database Performance Analyzer

Sisällys

Lyhenteet	1
1 Johdanto	3
2 Tietokannat	4
2.1 Relaatiotietokannat	4
2.2 SQL-kyselykieli	6
2.2.1 SQL-kyselyt	6
2.2.2 Muokkauk komennot	7
2.2.3 Transaktiot	9
2.2.4 Tietokantojen hallintajärjestelmät	9
2.3 Microsoft SQL Server	10
2.3.1 Versiot	10
2.3.2 T-SQL	10
2.3.3 Rakenne	11
2.3.4 Microsoft SQL Server Management Studio	12
2.3.5 Lisäominaisuudet	13
2.3.6 Tietokantojen ylläpito	14
2.3.7 SQL-kyselyn suorittaminen	16
2.3.8 Suorituskyky	18
3 Tietokantojen valvonta	20
3.1 Nykytilanne	20
3.2 Tavoitteet	21
3.3 Testauksen toteutus	22
4 Valvontajärjestelmät	22
4.1 Valvontapalvelin	22
4.2 N-able N-central	24
4.2.1 Järjestelmävaatimukset	24
4.2.2 Ominaisuudet	24
4.2.3 SQL-valvonta	26
4.2.4 Hälytykset	28

4.3	ManageEngine Applications Manager	29
4.3.1	Järjestelmävaatimukset	29
4.3.2	Asentaminen	29
4.3.3	Ominaisuudet	30
4.3.4	SQL-valvonta	31
4.3.5	Hälytykset	34
4.4	SolarWinds Database Performance Analyzer	34
4.4.1	Järjestelmävaatimukset	34
4.4.2	Asentaminen	35
4.4.3	Ominaisuudet	35
4.4.4	SQL-valvonta	37
4.4.5	Hälytykset	40
5	Vertailu	40
5.1	N-central	40
5.2	Applications Manager	42
5.3	Database Performance Analyzer	43
5.4	Johtopäätökset	45
6	Yhteenveto	46
	Lähteet	48

Lyhenteet

DBMS	<i>Database Management System</i> . Tietokannan hallintajärjestelmä on ohjelmisto, jolla hallitaan ja ylläpidetään tietokantaa.
DPA	<i>Database Performance Analyzer</i> . SolarWindsin kehittämä tietokantojen valvontasovellus.
SQL	<i>Structured Query Language</i> . Standardisoitu kyselykieli, jolla voidaan tehdä hakuja ja muutoksia relaatiotietokantaan.
DML	<i>Data Manipulation Language</i> . SQL-kielen osa-alue, jonka käskyillä hallitaan tietokantaobjekteja.
DDL	<i>Data Definition Language</i> . SQL-kielen osa-alue, jonka käskyillä hallitaan tietokannan rakennetta ja skeemaa.
DCL	<i>Data Control Language</i> . SQL-kielen osa-alue, jonka käskyillä hallitaan käyttäjien oikeuksia eri tietokantaobjekteihin.
T-SQL	<i>Transact-SQL</i> . SQL-kielen laajennus, jota Microsoft SQL Server käyttää käskykielenään.
SaaS	<i>Software as a Service</i> . Termi, jolla tarkoitetaan ohjelmiston hankkimista web-pohjaisena pilvipalveluna.
SSMS	<i>SQL Server Management Studio</i> . Microsoft SQL Serverin pääasiallinen hallintasovellus.
SSIS	<i>SQL Server Integration Services</i> . Microsoft SQL Serverin lisäominaisuus, jonka avulla voidaan tuoda data tietokantaan ulkoisista järjestelmistä.
SSAS	<i>SQL Server Analysis Services</i> . Microsoft SQL Serverin lisäominaisuus, jonka avulla voidaan analysoida ja prosessoida dataa tietokannassa.

SSRS	<i>SQL Server Reporting Services</i> . Microsoft SQL Serverin lisäominaisuus, jonka avulla voidaan luoda erilaisia raportteja tietokannan datasta.
BIDS	<i>Business Intelligence Development Studio</i> . Microsoft SQL Serverin liiketoimintatiedon (business intelligence) työkalu, jonka avulla voidaan hyödyntää SSMS:n, SSIS:n ja SSAS:n eri ominaisuuksia.
I/O	<i>Input/Output</i> . Suure, jolla kuvataan laitteen tai jonkin sen komponentin sisään- ja ulospäin menevän datan määrää.
RDP	<i>Remote Desktop Protocol</i> . Protokolla, jonka avulla voidaan etäyhteyden avulla hallita verkon yli toista tietokonetta.
SNMP	<i>Simple Network Management Protocol</i> . Protokolla, jonka avulla verkon yli voidaan hallita ja valvoa eri laitteita, kuten palvelimia, kytkimiä tai tulostimia.
WMI	<i>Windows Management Instrumentation</i> . Microsoft Windowsin ominaisuus, jonka avulla järjestelmästä voidaan kerätä informaatiota ja ilmoituksia valvontaa varten.

1 Johdanto

BLC Protie Oy on IT-palveluntarjoaja, joka ylläpitää noin 60 asiakasyrityksen IT-infrastruktuuria. Asiakasyritysten kokoluokat vaihtelevat pienistä yrityksistä monen tuhannen hengen konserneihin asti. Protie vastaa asiakasyrityksen kanssa tehdyn sopimuksen mukaisesti joko asiakkaansa koko IT-infrastruktuurista tai joistain sen osa-alueista.

Protien valvomossa valvotaan infran toimintaa ja ratkaistaan erilaisia verkon, palvelinten sekä palvelinsovellusten ongelma- ja vikatilanteita. Verkon sekä palvelinten valvonta on pitkälle automatisoitua, ja vian sattuessa siitä generoituu hälytys. Tämän jälkeen valvomossa aletaan selvittää ongelmaa ja etsiä siihen ratkaisua. Esimerkiksi palvelimia valvotaan erilaisilla mittareilla, jotka kertovat niiden suorituskyvystä sekä toiminnasta ja hälyttävät, jos jotain poikkeavaa tapahtuu.

Lähes kaikilla asiakkailla on käytössään tietokantapalvelimia, joiden ylläpitämiä kantoja eri sovellukset käyttävät. Suurin osa ylläpidossa olevista tietokantapalvelimista käyttää Microsoft SQL Server -tietokannan hallintajärjestelmää (Database Management System, DBMS). Tällä hetkellä Protien käyttämä valvontajärjestelmä valvoo tietokantoja vain prosessi- ja instanssitasolla, eli käytännössä nähdään, onko jokin tietokantapalvelu toiminnassa vai ei. Tietokantoihin voi liittyä hyvin monen tyyppisiä ongelmia, jotka näkyvät asiakkaille muun muassa eri sovellusten hitautena. Koska mahdollisia ongelmakohtia on lukemattomia, on niiden selvittäminen nykyisellä valvonnalla hidasta ja hankalaa. Tämän takia tarvitaan nykyistä kattavampaa tietokantavalvontaa.

Tämän insinööriyön tarkoituksena on löytää entistä parempia tietokantavalvontatyökaluja Protien käyttöön. Testausta varten asennetaan kaksi eri tietokantavalvontasovellusta, jotka ovat ManageEngine Applications Manager, sekä SolarWinds Database Performance Analyzer (DPA). Näiden lisäksi otetaan käyttöön ja testataan nykyisen valvontajärjestelmän N-centralin tietokantavalvontaominaisuuksia. Sovelluksille perustetaan oma palvelin jolla ne pyörivät, ja testauksessa käytetään Protien oman tuotantoympäristön SQL-tietokantapalvelimia. Testausten perusteella vertaillaan valvontasovellusten käyttöä ja niiden ominaisuuksia, sekä arvioidaan niiden soveltuvuutta Protien kaltaiselle IT-palveluntarjoajalle.

2 Tietokannat

Tietokannalla tarkoitetaan säännönmukaisesti järjestettyä tietokokoelmaa. Tietokannan tiedoilla on myös aina yhteys toisiinsa. Erilaiset taulukot, arkistot tai kortistot ovat tietokantoja. Informaatioteknologiassa tietokannalla tarkoitetaan sähköistä tietokantaa, jota hallitaan jollakin tietokannan hallintajärjestelmällä. [1.] [2.]

Tietotekniikassa käytetään taulukkorakennetta ryhmittämään tietokannan sisältämää dataa. Yrityksen keräämät tiedot omista asiakkaistaan ja työntekijöistään ovat yksi esimerkki tietokannasta. Tiedot tallennetaan tietokannan tauluihin, joista niitä pystytään myöhemmin tarpeen vaatiessa hakemaan tai ryhmittelemään haluttujen ominaisuuksien mukaan. Tietokannan yksi taulu voi sisältää esimerkiksi perustiedot yrityksen asiakkaista ja toinen taulu tietoja asiakkaiden tekemistä hankinnoista. Tietokanta perustuu johonkin tietomalliin, joka määrittelee sen rakenteen, sekä sille suoritettavat toimenpiteet, eli niin kutsutun käsitteistön.

Erilaisia tietomalleja:

- Relaatiomalli
- Oliomalli
- Hierarkkinen malli
- Verkkomalli [3.] [4.]

2.1 Relaatiotietokannat

Relaatiotietokannat perustuvat nimensä mukaisesti relaatiomalliin, joka perustuu taas predikaattilogiikkaan. Relaatiotietokanta rakentuu tauluista, joilla on yhteys toisiinsa. Jokaisella relaatiotaululla on aina oma nimensä. Taulu sisältää rivejä, eli tietueita, joista aina yksi rivi kuvaa tiettyä kohdetta reaali maailmassa. Rivin sarake sisältää aina kohteen jonkin ominaisuuden tai arvon. Sarakkeet vastaavasti nimetään havainnollistamaan arvon merkitys, ja niille määritellään myös rajoitteet, mitä arvoja kullekin sarakeelle voidaan tallentaa. Rajoitteilla voidaan esimerkiksi määritellä, että tietylle sarakeelle voidaan tallentaa pelkästään alfanumeerinen arvo, päivämäärä, vuosiluku tai henkilön ikä. Rajoitteet ottavat myös kantaa siihen, voiko kenttä olla tyhjä. [4.] [5.]

Perusavain
↓
Yritys

Yrtunnus	Nimi	Postinro	Postitmp	Puhelin
111	Mäkitieto Oy	90100	Oulu	08-112233
222	Mikropaja Oy	90250	Oulu	08-123123
121	Systeemityö Oy	02150	Espoo	
122	Kokotieto Tmi	90120	Oulu	050-3233311
223	Teematupa Oy	90160	Oulu	08-998877
225	Mikrotieto Oy	00100	Helsinki	040-7788665

Kuvio 1. Relaatiotietokannan taulu, joka esittää eri yritysten tietoja. Perusavain tässä taulussa on yritystunnus ensimmäisessä sarakkeessa. [6.]

Jokainen taulu sisältää perusavaimen, joka koostuu yhdestä tai useammasta sarakkeesta. Jokaisella tietueella on oltava yksilöllinen perusavain, joka erottaa sen taulun muista tietueista. Perusavain ei saa myöskään puuttua yhdeltäkään taulun riviltä, eikä sen arvo voi olla NULL, jolla usein tietokannoissa kuvataan puuttuvaa arvoa. Tietokantojen taulujen yhteydet rakentuvat taulujen perusavaimista. Taulun viiteavain viittaa relaatiotietokannan toisen taulun perusavaimeen. [7.]

Perusavain
↓
Yritys

Yrtunnus	Nimi	Postinro	Postitmp	Puhelin
111	Mäkitieto Oy	90100	Oulu	08-112233
222	Mikropaja Oy	90250	Oulu	08-123123
121	Systeemityö Oy	02150	Espoo	
122	Kokotieto Tmi	90120	Oulu	050-3233311
223	Teematupa Oy	90160	Oulu	08-998877
225	Mikrotieto Oy	00100	Helsinki	040-7788665

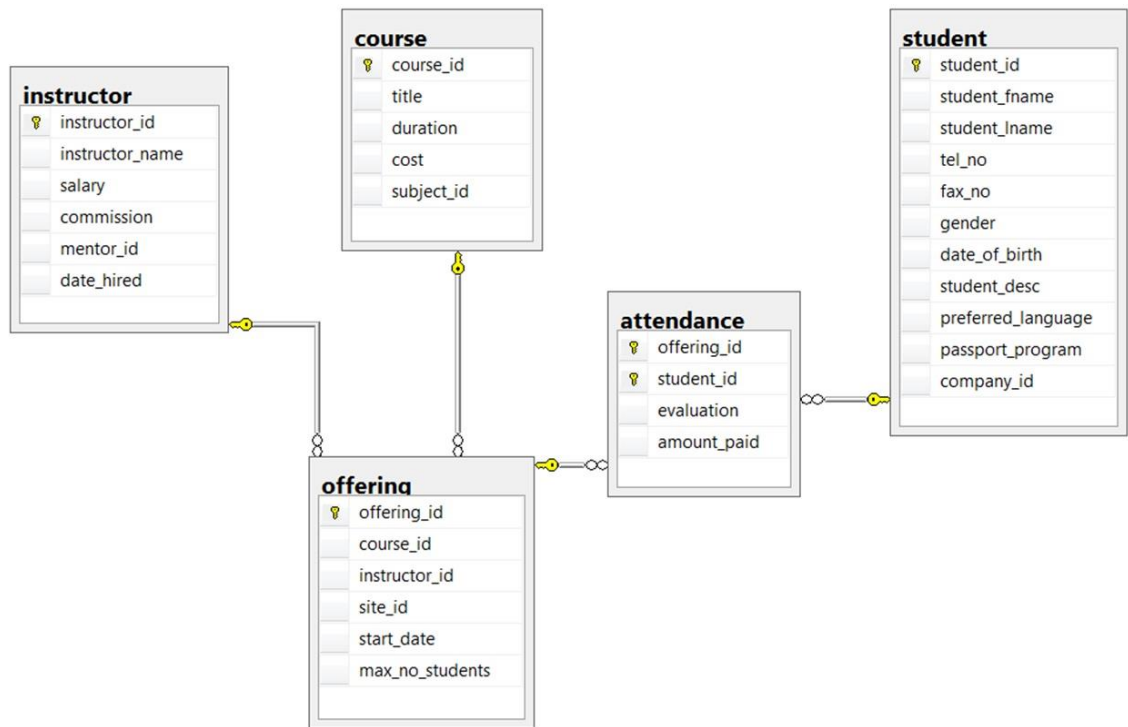
Viiteavain
↓

Henkilö

Hnro	Etunimi	Sukunimi	Syntymävuosi	Aloituspvm	Yrtunnus
1	Mikko	Mikkonen	1965	1.2.1999	111
2	Tiina	Taitava	1970	12.3.2000	222
3	Riina	Rovasti	1980	30.5.2002	121
4	Liisa	Leppänen	1963	2.3.1990	121
5	Pekka	Ponteva	1985	1.6.2004	122
6	Raili	Tenhunen	1961	2.6.1985	225

Kuvio 2. Relaatiotietokanta, jossa on kaksi taulua. Alemman taulun viiteavain viittaa ylemmän taulun perusavaimeen. [6.]

Tietokannan skeemalla tarkoitetaan loogista kuvausta tietokannan rakenteesta. Se voidaan esittää graafisesti taulujen ja sen sisältämien sarakkeiden välisinä suhteina.



Kuvio 3. Kuvassa on esitetty graafisesti kurssijärjestelmän tietokannan skeema. [8.]

2.2 SQL-kyselykieli

SQL (Structured Query Language) on standardisoitu kyselykieli, jota käytetään eri tietokantojen hallintajärjestelmissä. Se on relaatiomallin tunnetuin sovellus, vaikka se ei olekaan täysin relaatiomallin mukainen. Käytännössä kuitenkin kaikki relaatiomalliin perustuvat tietokannat ymmärtävät SQL-kieltä. Kieli koostuu kyselyistä sekä kolmesta osa-alueesta, joilla tehdään muokkauksia tietokantaan. Osa-alueet ovat Data Manipulation Language (DML), Data Definition Language (DDL) sekä Data Control Language (DCL). [9.]

2.2.1 SQL-kyselyt

SELECT-komennolla haetaan SQL-tietokannasta dataa. Se on SQL-kielen yleisin operaatio, eikä itsessään tee mitään muutoksia kantaan, vaan sen voidaan ajatella toimivan ns. read-only-periaatteella lukien tietokantaa. SELECT-komennon kanssa käytetään aina FROM-sanaa, joka määrittelee taulut, joista haluttua tietoa etsitään. Lisäksi voidaan käyttää myös WHERE-sanaa, jolla määritellään ehdot haettaville riveille. Alla on muutama esimerkki yksinkertaisista SQL-kyselyistä.

Valitse kaiken datan taulusta Asiakkaat:

```
SELECT * FROM Asiakkaat;
```

Valitse kaikki ne miesasiakkaat, jotka asuvat Oulussa:

```
SELECT * FROM Asiakkaat  
WHERE Kaupunki='Oulu'  
AND Sukupuoli='Mies';
```

2.2.2 Muokkauskomennot

DML-komennoilla muokataan dataa tietokannassa. DML-komentojen avulla voidaan muun muassa lisätä uutta dataa tauluihin, päivittää, poistaa tai yhdistää arvoja.

Data Manipulation Language (DML) esimerkkejä:

- INSERT (lisää)
- UPDATE (päivitä)
- DELETE (poista)
- MERGE (yhdistä)

DDL-komennoilla muokataan tietokannan rakennetta ja sen määrittelyjä. Niiden avulla voidaan luoda, muokata sekä tyhjentää kokonaisia tietokantoja ja tauluja tai esimerkiksi nimetä uudelleen tai poistaa tietty sarake.

Data Definition Language (DDL) esimerkkejä:

- CREATE (luo)
- ALTER (muuta)
- TRUNCATE (tyhjennä)
- DROP (pudota)

DCL-komennoilla hallitaan käyttäjien oikeuksia suorittaa eri komentoja tietokantaan. Tiettylle käyttäjälle voidaan määrittää esimerkiksi oikeus suorittaa vain SELECT-komentoja johonkin tauluun, kun taas joku toinen käyttäjä saa myös muokata ja poistaa taulusta tietoja.

Data Control Language (DCL) esimerkkejä:

- GRANT (myönnä)
- REVOKE (kumoa)
- DENY (kiellä)

Seuraava esimerkki muuttaa asiakas Minna Kuuselan osoitteen tietokantaan:

```
UPDATE Asiakkaat
SET Osoite='Kauppiaantie 12',Postinumero=20100,Kaupunki='Turku',Maa='Suomi'
WHERE Etunimi='Minna'
AND Sukunimi='Kuusela';
```

Alla oleva lause luo TestiDB-tietokantaan taulun nimeltä FyysisetOminaisuudet, jonka perusavaimena toimii ensimmäisen sarakkeen juokseva numero. Muut sarakkeet tietokannassa ovat Etunimi, Sukunimi, Ika, Pituus ja Paino. Etunimi ja Sukunimi ovat tekstimuodossa ja muut ovat kokonaislukuja. Tämän jälkeen tietokantaan syötetään Juuso Peltolan ja Kalle Järvisen tiedot, ja lopuksi ne tulostetaan SELECT-lauseella näytölle.

```
CREATE TABLE TestiDB.FyysisetOminaisuudet
(id INTEGER PRIMARY KEY AUTOINCREMENT,
Etunimi VARCHAR(MAX),
Sukunimi VARCHAR(MAX),
Ika INTEGER,
Pituus INTEGER,
Paino INTEGER);

INSERT INTO FyysisetOminaisuudet(Etunimi,Sukunimi,Ika,Pituus,Paino)
VALUES ('Juuso','Peltola',28,181,79);
INSERT INTO FyysisetOminaisuudet(Etunimi,Sukunimi,Ika,Pituus,Paino)
VALUES ('Kalle','Järvinen',24,185,88);

SELECT * FROM FyysisetOminaisuudet;
```

id	Etunimi	Sukunimi	Ika	Pituus	Paino
1	Juuso	Peltola	28	181	79
2	Kalle	Järvinen	24	185	88

Kuvio 4. Edellisessä esimerkissä luotu taulu tulostettuna näyttää tältä. [8.]

2.2.3 Transaktiot

SQL-tietokannan käsittely tapahtuu käytännössä niin kutsuttuina transaktioina. Yhtä transaktiota voidaan ajatella käsittelykokonaisuutena, joka sisältää yhden tai useamman peräkkäisen tietokantaoperaation eli SQL-komennon. Transaktio voidaan jakaa neljään eri vaiheeseen, joista kaksi viimeistä ovat toisensa pois sulkevia. [10.]

SQL-transaktion vaiheet:

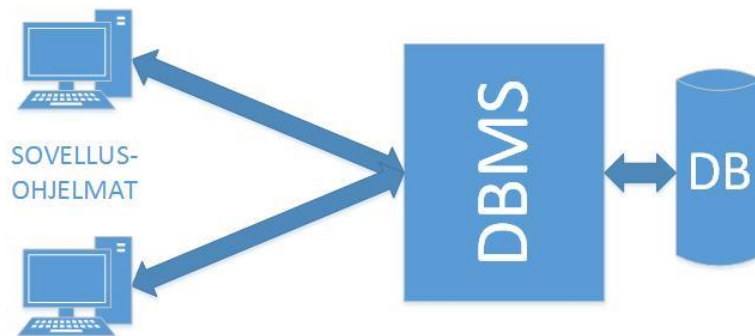
1. Transaktion aloittaminen
2. Operaatioiden suorittaminen
3. Jos ei virheitä, niin COMMIT
4. Jos virheitä, niin ROLLBACK

Esimerkkinä transaktiosta voidaan ajatella rahansiirtoa tililtä toiselle. Tuolloin operaatiot voidaan jakaa kahteen osaan. Raha nostetaan ensin toiselta tililtä pois, jonka jälkeen sama summa rahaa talletetaan toiselle tilille. Jos molemmat operaatiot onnistuvat, niin suoritetaan COMMIT-komento, joka tekee muutokset molemmille tileille, eli tietokantaan. Jos molemmat tai edes toinen operaatioista epäonnistuu, suoritetaan ROLLBACK-operaatio, joka peruuttaa kaikki transaktion tekemät muutokset tileille, eli tietokantaan. [11.]

2.2.4 Tietokantojen hallintajärjestelmät

Tietokannan hallintajärjestelmä, eli database management system (DBMS), voidaan määritellä sovelluksena, joka hoitaa kaikki tietokantaoperaatiot. Sen avulla voidaan perustaa tietokanta ja ylläpitää sitä. Muita vaatimuksia DBMS:lle on esimerkiksi sen tietoriippumattomuus, yhteiskäyttöisyys ja yhteensopivuus. DBMS on linkki loppukäyttäjän käyttämän sovellusohjelman ja itse tietokannan välissä. Sen avulla tallennetaan dataa tietokantaan, ylläpidetään kantaa sekä muokataan sitä. [9.]

Tietokannan hallintajärjestelmä:



Kuvio 5. DBMS on linkki tietokannan ja sitä hyödyntävien sovellusohjelmien välillä.

Suosituimpia tietokantojen hallintajärjestelmiä nykyään ovat muun muassa Microsoft SQL Server, Oracle Database, IBM DB2, MySQL ja PostgreSQL. Yhteistä näille kaikille hallintajärjestelmille on se, että niitä käytetään SQL-komennoilla. Jokainen näistä voidaan asentaa palvelimelle, jolloin ne pystyvät palvelemaan useaa käyttäjää samanaikaisesti.

2.3 Microsoft SQL Server

2.3.1 Versiot

Microsoft SQL Server on Microsoftin kehittämä ja myymä versio SQL-tietokannan hallintajärjestelmästä. Microsoft markkinoi tuotetta monilla eri lisensseillä, joiden nimitykset vaihtelevat eri versioiden välillä. Uusin versio on SQL Server 2016, joka on julkaistu helmikuussa 2016. Tätä edeltävät versiot, jotka ovat edelleen yhä tuen piirissä, ovat SQL Server 2005, 2008, 2008 R2, 2012 ja 2014. Lisäksi on olemassa erikoisversioita, kuten Microsoftin Software as a Service (SaaS) pilviversio Azure SQL Database. Microsoft on myös ilmoittanut julkaisevansa SQL Serverin Linux-käyttöjärjestelmälle vuonna 2017. [12.]

2.3.2 T-SQL

Microsoft SQL Server käyttää pääasiallisena käskykielenään Transact-SQL:ää (T-SQL), jonka Microsoft on kehittänyt yhteistyössä Sybasen kanssa. T-SQL on standardisoidun

SQL-kielen laajennus, ja tuo siihen lisäominaisuuksia, jotka liittyvät muun muassa transaktioiden hallintaan, rivien ja virheiden käsittelyyn sekä lokaaleiden muuttujien määrittelyyn. T-SQL:llä luodaan, hallitaan ja muokataan tietokantoja sekä lisätään ja muokataan dataa niiden sisällä. Sen avulla myös hallitaan ja valvotaan koko SQL-palvelininstanssin toimintaa. [13.] [14.]

2.3.3 Rakenne

Tietokanta koostuu yhdestä tai useammasta tietokantatiedostosta. Jokaisella tietokannalla on mdf-päätteinen ensisijainen tietokantatiedosto sekä vaihtoehtoisesti myös ndf-päätteisiä toissijaisia kantatiedostoja. Kaikki tietokannan sisältämä data sijaitsee näissä mdf- ja ndf-päätteisissä tiedostoissa. Transaktiot kirjataan ldf-päätteiseen lokitiedostoon. Tietokannan palautustyypistä riippuen transaktioloki sisältää joko tietokannassa sillä hetkellä käynnissä olevat transaktiot (recovery model simple) tai transaktiot pidemmältä aikaväliltä – yleensä viimeisimpään varmistukseen asti (recovery model full).

SQL Server sisältää varsinaisten käyttäjien tietokantojen (user databases) lisäksi järjestelmätietokantoja (system databases), joita se käyttää järjestelmän hallintaan ja ylläpitoon. Alla olevassa listauksessa on esiteltyä niiden nimet ja sisältö. Näiden lisäksi myös SQL Serverin lisäominaisuudet voivat luoda omaan käyttöönsä tietokantoja.

master:

- Järjestelmän kokoonpano ja asetukset [15.]

msdb:

- Aikataulutetut tehtävät
- Pääasiassa SQL Server Agentin käytössä
- Myös mm. Service Brokerin käytössä [16.]

model:

- Malli (template) uusille tietokannoille [17.]

tempdb:

- Väliaikainen käsiteltävissä oleva data [18.]

Resource:

- Helpottaa järjestelmän päivittämistä
- Piilotietokanta järjestelmäobjekteille
- Read-only [19.]

Tietokantatiedostoihin data ryhmitellään 8 kilotavun suuruisiin sivuihin, joilla jokaisella on oma järjestysnumerosa. Sivun on SQL-palvelimen perusyksikkö tietokannan datalle. Tietokannan hallintajärjestelmä kuitenkin allokoii datalle tilaa levyiltä aina kahdeksan sivun

kokoisina lukualueina. Yksi lukualue voi sisältää dataa yhdeltä (uniform extent) tai useammalta (mixed extent) objektilta, kuten taululta. Yleensä SQL-palvelin allokoii uudelle objektille tilaa jaetulta lukualueelta, mutta jos objekti kasvaa yli kahdeksan sivun kokoiseksi, niin se alkaa allokoimaan sille jatkossa tilaa aina omalta lukualueelta. [20.]

SQL-kyselyiden nopeuttamiseksi tietokantaan voidaan luoda indeksejä, joista yleisin on niin kutsuttu B-tree-rakenne. B-tree-rakenteessa tietyn sarakkeen kaikki arvot kopioidaan indeksiin, ja niihin liitetään osoitin, eli pointteri, osoittamaan arvon alkuperäistä muistipaikkaa tietokannassa. Tämän jälkeen arvot indeksissä lajitellaan pienimmästä suurimpaan, mikä usein nopeuttaa sarakkeeseen kohdistuvia SQL-kyselyitä huomattavasti. Tällöin SQL-palvelimen ei tarvitse käydä koko taulun saraketta alusta loppuun läpi löytääkseen hakemansa datan. Toisaalta indeksit myös vievät tietokannasta tilaa, ja lisäksi hidastavat datan muokkausta, koska muokkaukset täytyy tehdä taulun lisäksi tällöin myös indeksiin. Klusteroitu indeksi (clustered index) on indeksi, jonka mukaan taulun rivit on järjestetty tietokantaan. Jokainen taulu voi sisältää maksimissaan yhden klusteroidun indeksin, ja ensisijainen avain on myös automaattisesti taulun klusteroitu indeksi, ellei indeksiä erikseen määritetä joksikin muuksi. [21.] [22.]

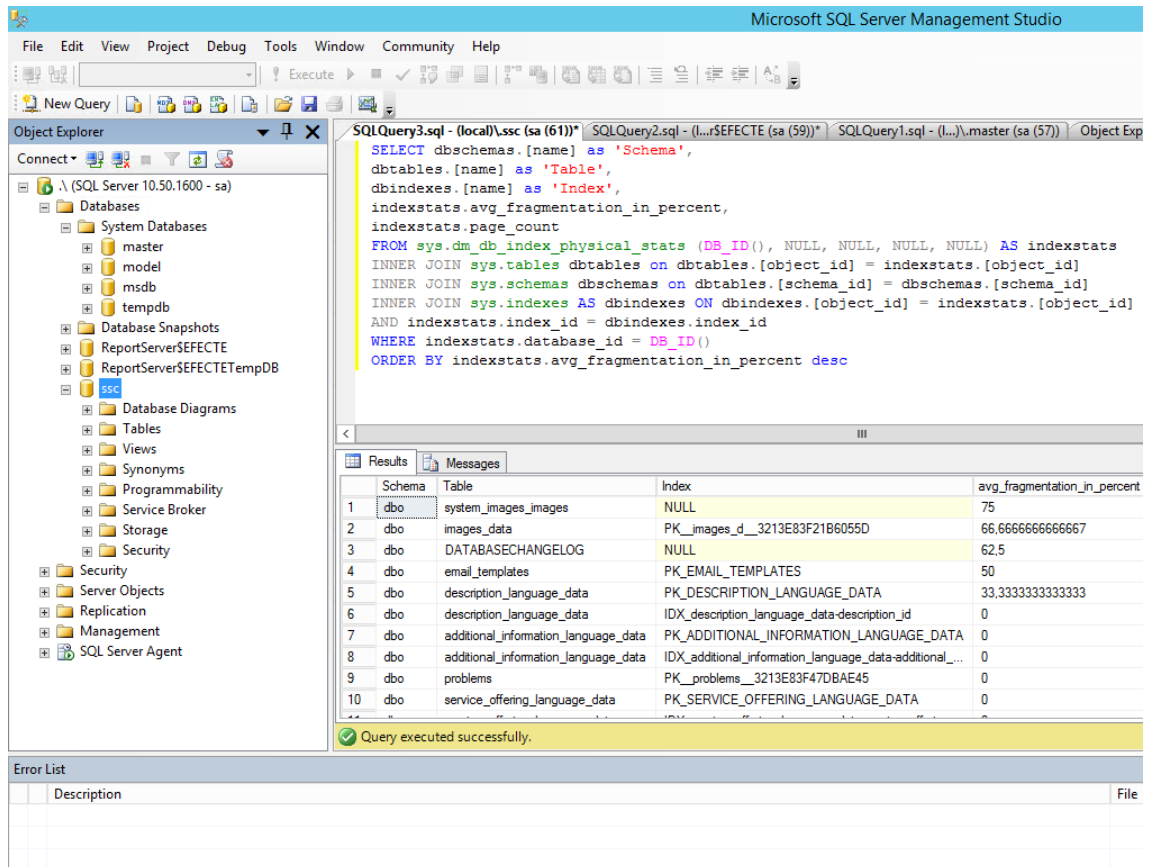
SQL Server optimoi hakuja myös ylläpitämällä tilastoja (statistics) taulujen sarakkeista. Saraketilasto luodaan aina indeksin luonnin yhteydessä, mutta tietokannan ylläpitäjä voi luoda niitä myös käsin, tai asetuksista riippuen DBMS voi luoda niitä automaattisesti. Tilastot sisältävät tietoja muun muassa saatavilla olevista indekseistä, sarakkeiden tietueiden määrästä ja datan tiheydestä. Nämä auttavat SQL-palvelinta valitsemaan optimaalisen toteutustavan SQL-kyselyille. [23.]

2.3.4 Microsoft SQL Server Management Studio

SQL Serverin pääasiallinen hallintasovellus on SQL Management Studio (SSMS), jolla hallitaan, konfiguroidaan ja ylläpidetään SQL-palvelinta. Sen graafisen käyttöliittymän keskeisenä navigointityökaluna on Object Explorer, jolla päästään hallinnoimaan SQL-palvelimen kaikkia eri objekteja. [14.]

Microsoft SQL Server Management Studio tärkeimpiä ominaisuuksia:

- Tietokantojen luonti
- Tietokantojen rakenteiden muokkaus
- Taulujen, indeksien, rivien ym. objektien muokkaus
- Tietokantojen suorituskyvyn analysointi ja optimointi
- Tietokantojen varmistus
- SQL-kyselyiden ja muiden komentojen ajaminen suoraan kantaan



Kuvio 6. Kuvakaappaus Microsoft SQL Server Management Studiosta, missä näkyy Object Explorer vasemmassa palkissa.

2.3.5 Lisäominaisuudet

Microsoft SQL Server sisältää paljon lisäominaisuuksia, jotka eivät ole välttämättömiä tietokantojen toiminnalle, mutta jotka voidaan halutessa asentaa ja ottaa käyttöön. Osa näistä palveluista pyörii suoraan osana tietokantaprosessia ja osa taas omana prosessinaan sen rinnalla. Seuraavassa listauksessa on lueteltuna tärkeimmät lisäpalvelut ja niiden ominaisuudet.

Integration Services (SSIS)

- Datan hakeminen tietokantaan ulkoisista järjestelmistä
- Datan muokkaaminen SQL-muotoon
- Datan tallettaminen tietokantaan [24.] [25.]

Analysis Services (SSAS)

- Datan analysointi
- Datan prosessointi
- Datan louhinta (data mining) [24.] [26.]

Reporting Services (SSRS)

- Datan raportointiympäristö
- Web-käyttöliittymä [24.] [27.]

Business Intelligence Development Studio (BIDS)

- Käyttää SSIS-, SSAS- ja SSRS-palveluita
- Helpottaa liiketoimintatiedon (business intelligence) prosessien hallintaa
- Pohjautuu Visual Studioon [28.]

Data Tools

- Korvasi BIDS:n SQL Server 2012:ssa
- Voidaan asentaa osana SQL Serveriä ja/tai Visual Studiota [29.]

Visual Studio

- Microsoftin ohjelmankehitysympäristö
- Tietokannan skeeman graafinen esittäminen
- Graafinen SQL-kyselyiden luonti
- Data Tools -integraatio [14.] [30.]

SQL Server Agent

- Suorittaa aikataulutettuja tehtäviä (jobs)
- Tehtävillä mm. varmistetaan ja huolletaan tietokantoja [31.]

Replication Services

- Datan kopioiminen ja synkronisointi
- Transaktioiden kopioiminen ja synkronisointi
- Toimii palvelimelta palvelimelle tai palvelimelta sovellusohjelmaan [14.]

Service Broker

- Välittää viestejä SQL-palvelimien ja -tietokantojen välillä
- Toimii sovellusten välisenä rajapintana [32.]

Full-Text Search Service

- Pystyy indeksoimaan kaikkea tietokannan sisältämää dataa
- Nopeuttaa ja laajentaa SQL-kyselyitä [33.]

SQLCMD

- Komentokehote-sovellus SQL Serverille
- Voidaan käyttää T-SQL-komentojen skriptauksessa [34.]

2.3.6 Tietokantojen ylläpito

SQL Serverin tietokantoja huolletaan ja varmistetaan erilaisilla huoltosuunnitelmillä (maintenance plans), jotka sisältävät tehtäviä (tasks). SQL Server Agent voi suorittaa

huoltosuunnitelmia aikataulutetusti, tai ne voidaan luoda manuaalisesti ajettaviksi. Huoltotehtävät ylläpitävät tietokantoja esimerkiksi järjestelemällä taulujen indeksejä uudelleen tai päivittämällä niiden tilastoja. Töiden suunnittelulla ja toteutuksella on iso merkitys tietokantojen toimintaan ja niiden nopeuteen. Huonosti toteutettu ylläpito voi myös hidastaa ja heikentää tietokannan toimintaa entisestään. Tietokannan ominaisuuksista ja käytävästä sekä datan luoteesta riippuu, millaiset huoltosuunnitelmat sille sopivat parhaiten. Joitain tietokantoja halutaan varmistaa viidentoista minuutin välein, kuin toisille taas riittää viikoittainen varmistus. Myös se, miten usein tietokannan data päivittyy, vaikuttaa huoltotoimenpidetarpeisiin ja niiden aikataulutukseen. Alla olevassa listauksessa on esiteltynä SQL Server Management Studion Maintenance Plans -hakemiston alla valittavissa olevat huoltotehtävät (maintenance plan tasks).

Back Up Database

- Tietokannan varmistus
- Transaktiologioiden varmistus

Check Database Integrity

- Tietokannan eheyden tarkistus

Execute SQL Server Agent Job

- Suorittaa valmiin SQL Server Agent -työn

Execute T-SQL Statement

- Suorittaa T-SQL-komentoja

History Cleanup

- Tyhjentää msdb-tietokannasta historiatietoja
- Estää msdb-tietokannan paisumista

Maintenance Cleanup

- Poistaa vanhoja tiedostoja
- Käytetään erityisesti vanhojen varmistusten poistamiseen

Notify Operator

- Lähettää ilmoituksen ylläpitäjälle

Rebuild Index

- Rakentaa indeksit uudelleen

Reorganize Index

- Järjestää indeksit uudelleen

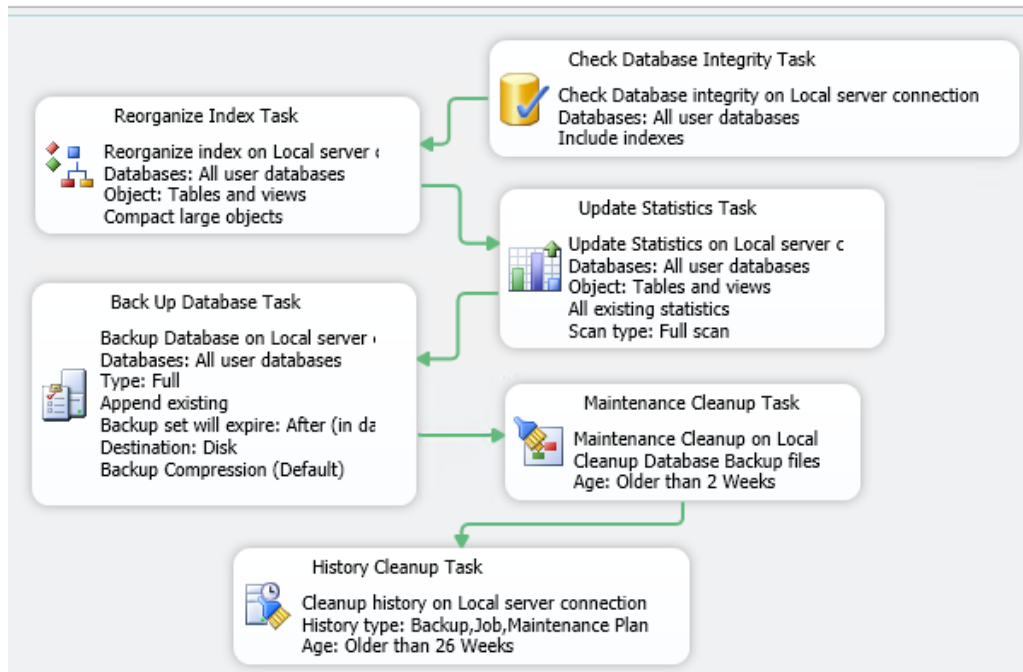
Shrink Database

- Tiivistää tietokannan dataa
- Pientää tietokannan kokoa levyllä

Update Statistics

- Päivittää tilastot

Name	Backup_and_maintenance	
Description		
Schedule	Occurs every week on Wednesday at 3:00:00 AM. ...	Run as
		SQL Server Agent service account



Kuvio 7. Kuvassa on esimerkki viikoittaisesta SQL-palvelimen huoltosuunnitelmasta. Ensimmäinen suoritettava työ on 'Check Database Integrity', jonka valmistettua edetään nuolten osoittamassa järjestyksessä aina seuraavaan tehtävään.

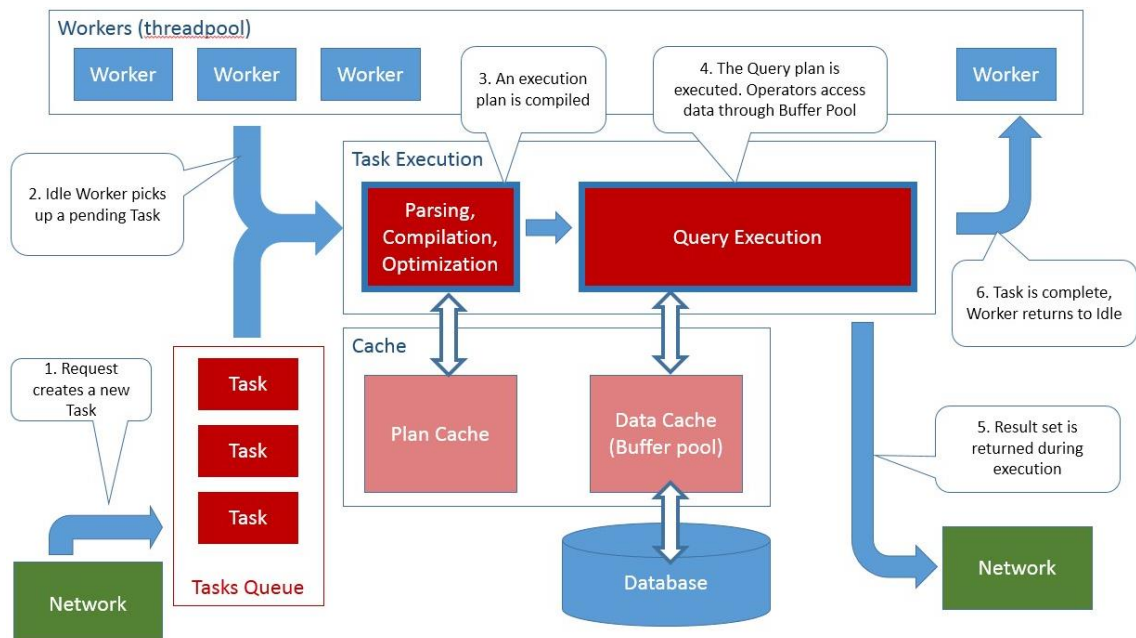
2.3.7 SQL-kyselyn suorittaminen

Tietokantaa käyttävien sovellusten suorittamat transaktiot sisältävät SQL-kyselyitä, joita kutsutaan myös pyynnöiksi (request). Jokaista pyyntöä SQL-palvelin käsittelee omana tehtävänä (task), ja pyynnön saapuessa palvelimelle se asetetaan tehtäväjonoon (task queue) odottamaan suoritusta. Tehtävän suorittamisen hoitaa prosessorin säie (worker thread), joka on käytännössä prosessorin pienin suoritusyksikkö. Kun vapaana oleva säie alkaa suorittamaan tehtävää, se suorittaa sen valmiiksi alusta loppuun asti ja tekee kyselyn määrittämät muutokset tai lähettää haetun datan takaisin sovellusohjelmalle. [35.]

Tehtävän varsinaisen suorittamisen alussa kysely analysoidaan ja sen suorittamista varten luodaan suunnitelma (query plan). Suunnitelmaa varten käydään läpi tilastot sekä

indeksit, ja palvelin yrittää näiden avulla löytää optimaalisen tavan päästä haluttuun dataan käsiksi. Eri vaihtoehdoille lasketaan niin sanotusti hinta (price), eli se, paljonko dataa täytyy kyselyn aikana lukea, ja kuinka paljon muistia sekä prosessoriaikaa suunnitelma vaatii. Jos sama kysely toistuu usein, niin palvelimella on hyvin todennäköisesti valmis suunnitelma keskusmuistissa sijaitsevassa suunnitelmavälimuistissa, ja se voi ryhtyä heti suorittamaan tehtävää. [36.]

SQL-kysely voidaan jakaa eri vaiheisiin, jotka tehtävän aikana suoritetaan aina yksi kerrallaan. Jokainen näistä vaiheista hakee tai prosessoi dataa kyselyn määrittämällä tavalla. Tietokannasta haettava data luetaan keskusmuistissa sijaitsevasta välimuistista (buffer cache), ja sitä käsitellään 8 kilotavun kokoisina sivuina. Jos haettavaa sivua ei löydy välimuistista, niin se pitää kopioida ensin levyltä tietokannasta välimuistiin. Myös kyselyn välitulokset tallennetaan keskusmuistiin. Jos tehtävä tekee muutoksia tietokantaan, se käytännössä ensin kirjoittaa niistä lokikirjauksen välimuistiin, jonka jälkeen se muokkaa kyseistä sivua välimuistissa. Ennen kuin koko muutoksen sisältävä transaktio valmistuu, suoritetaan COMMIT-komento, joka kirjoittaa kaikki transaktion aikana tehdyt lokikirjaukset levyille transaktiolokiin. SQL-palvelin kirjaa tietyin väliajoin muutokset myös tietokantaan, ja tätä hetkeä kutsutaan tallennuspisteeksi (checkpoint). Tehtävän lopuksi palvelin lähettää sovellusohjelmalle vastauksen, ja säie on tämän jälkeen valmis suorittamaan uutta tehtävää. [36.]



Kuvio 8. Kuvassa on SQL-kysely loogisesti havainnollistettuna sen eri suoritusvaiheisiin. [36.]

2.3.8 Suorituskyky

Sovellusohjelman lähettämä pyyntö voidaan ajallisesti jakaa kahteen eri kategoriaan, jotka ovat odotusaika (wait time) ja varsinainen suoritus-aika (processing time). Kun tehtävä joutuu missä tahansa suoritusvaiheessa jostain syystä odottamaan, niin SQL-palvelin kirjaa ylös odotuksen keston ja sen syyn. Mahdollisia erilaisia kirjattavia odotusai- katyyppejä on olemassa kymmeniä erilaisia. Tietokannan suorituskykyä arvioitaessa tulee odotusajan lisäksi ottaa huomioon myös tehtävän varsinaiseen suorittamiseen ku- luva aika. [37.]

SQL-kyselystä muodostettu pyyntö odottaa aluksi tehtäväjonossa vapautuvaa proses- sorin säiettä. Säikeiden kokonaismäärä riippuu käytettävissä olevista suorittimista, ja tehtävän jonossa viettämä odotusaika riippuu säikeiden määrän lisäksi sitä edellä ole- vien tehtävien määrästä sekä niiden suoritusajasta. Tehtävän analysointiin käytetty aika riippuu kyselyn monimutkaisuudesta, ja optimaalisen toteutus suunnitelman laskemiseen kuluu enemmän aikaa, mikäli sitä ei löydy suoraan suunnitelmavälimuistista. Ajantasais- ten tilastojen pohjalta saadaan paremmin optimoitua tehtävän toteutus. Myös indeksit tai niiden puute vaikuttaa siihen, minkälaiseen suunnitelmaan päädytään, ja samoin sen arvioituun hintaan. [36.] [37.]

Tehtävän varsinaisessa suoritusvaiheessa data luetaan aina välimuistista, mutta jos sitä ei sieltä löydy, joudutaan se ensin kopioimaan levyiltä tietokannasta keskusmuistiin. Si- vua lukiessa tai muokatessa sille asetetaan niin kutsuttu salpa (latch), joka määrittää miten muut säikeet voivat sivua käyttää. Salpoja on olemassa viisi eritasoista, jotka on lueteltu alla. [38.]

Salvat (latches):

- KP – Keep (pidä)
- SH – Shared (jaettu)
- UP – Update (päivitä)
- EX – Exclusive (pois sulkeva)
- DT – Destroy (tuhoa)

Jos säie esimerkiksi muokkaa tietokannan sivua, niin se asettaa siville exclusive-salvan. Tällöin muut tehtävät joutuvat odottamaan salvan vapautumista, ennen kuin pystyvät lu- kemaan kyseistä sivua. Jos taas säie vain lukee sivulta dataa, niin se asettaa shared- salvan, jolloin sivua pystyvät myös muut säikeet lukemaan, mutta eivät muokkaamaan.

	KP	SH	UP	EX	DT
KP	Y	Y	Y	Y	N
SH	Y	Y	Y	N	N
UP	Y	Y	N	N	N
EX	Y	N	N	N	N
DT	N	N	N	N	N

Kuvio 9. Taulukkokuvasta nähdään eri salpojen vaikutukset sivujen toimintaan. Risteävät Y-merkkiset operaatiot onnistuvat samaan aikaan, kun taas N-merkkisissä operaatioissa jälkimmäinen tehtävä joutuu odottamaan salvan vapautumista. [39.]

Säikeiden lisäksi myös transaktiot asettavat rajoitteita samanaikaiselle pääsulle dataan, ja näitä rajoitteita kutsutaan lukoiiksi (locks). Lukoilla on kolme päätyyppiä, jotka ovat shared (S), update (U) sekä exclusive (X), ja näiden sisällä on lukuisia eri alatyyppejä, joilla on eri merkityksiä. Salvat asetetaan fyysiseen dataan, eli SQL-tietokannan kohdalla sivuihin, kun taas lukot asetetaan loogiseen dataan, eli esimerkiksi tauluihin tai riveihin. Niiden toiminta on joka tapauksessa hyvin salpojen tapaista, eli kun transaktio lukitsee tietokannasta taulun exclusive-lukolla muokatakseen sitä, niin muut transaktiot eivät voi tällöin lukea sieltä mitään dataa, vaan joutuvat odottamaan. [40.]

Taulujen datan järjestyksellä on vaikutus kyselyn suoritusnopeuteen. Kun indeksittömästä sarakkeesta haetaan tietoa, joutuu säie usein käymään läpi koko taulun alusta loppuun (table scan) löytääkseen haluamansa datan. Sarakeindeksissä tietueet ovat sen sijaan järjestelty pienimmästä suurimpaan, ja sieltä etsiminen on usein huomattavasti nopeampaa. Vielä sitäkin nopeampi haku voidaan suorittaa klusteroidusta indeksistä, joka käytännössä tarkoittaa sarakkeen mukaan järjestettyä taulua. Tällöin data on myös fyysisesti muistissa tallennettu sarakkeen mukaiseen järjestykseen.

SQL-palvelimen suorituskyky koostuu lopulta hyvin monen asian summasta. Eri prosessorimallit tarjoavat palvelimen käyttöön eri määrän säikeitä suorittamaan tehtäviä, ja keskusmuistin määrä vaikuttaa siihen, kuinka paljon dataa voidaan lukea suoraan nopeasta välimuistista levyn sijaan. Hyvin suunnitellut kyselyt, sekä ajantasaiset tilastot auttavat optimaalisen toteutussuunnitelman laatimisessa, ja järkevästi rakennetut ja päivitettyt indeksit nopeuttavat datan hakua tauluista. Myös levynopeudella on merkitystä levyllä sijaitseviin lokeihin tai kantoihin kohdistuviin luku- ja kirjoitusoperaatioihin. Palvelimen resurssienkäyttöä tarkkailemalla ja odotusaikatilastoja tutkimalla saadaan tärkeää tietoa, jolla pystytään tarvittaessa parantamaan SQL-palvelimen ja sen tietokantojen toimintaa.

3 Tietokantojen valvonta

3.1 Nykytilanne

Palvelinten valvontaan käytetään Protieellä N-ablen N-central-ohjelmistoa. N-central on web-pohjainen etävalvonta- ja -hallintasoftware verkon eri laitteille, kuten kytkimille, reitittimille, palvelimille ja työasemille. Kun palvelin tuodaan N-central-valvonnan piiriin, sille asennetaan valvonta-agentti-sovellus, joka kerää tietoja palvelimen resursseista ja toiminnasta. Ylläpitäjä voi tämän jälkeen määrittää, mitä kaikkea palvelimelta halutaan valvoa. N-Central pystyy myös agentin valvontadatan tunnistamaan palvelimelta erilaisia rooleja, joiden mukaan se automaattisesti tuo tietyt palvelut valvonnan piiriin. Samalla se myös asettaa eri mittareihin ennalta määritellyt raja-arvot palvelimen tyyppin mukaan. Seuraavissa listauksissa on esitelty Protien valvontaa Windows- ja SQL-palvelimille. Nämä kohteet lisätään valvontaan automaattisesti kaikilta uusilta palvelimilta, ja näihin tehdään jälkepäin asiakas- tai palvelinkohtaisesti muutoksia tai lisäyksiä aina tarpeen mukaan.

N-Central-valvonta Windows-palvelimille:

- Valvonta-agentti (agent.exe)
- Probe-yhteys (yhteys valvovaan palvelimeen)
- Prosessori
- Keskusmuisti
- Levyasemat

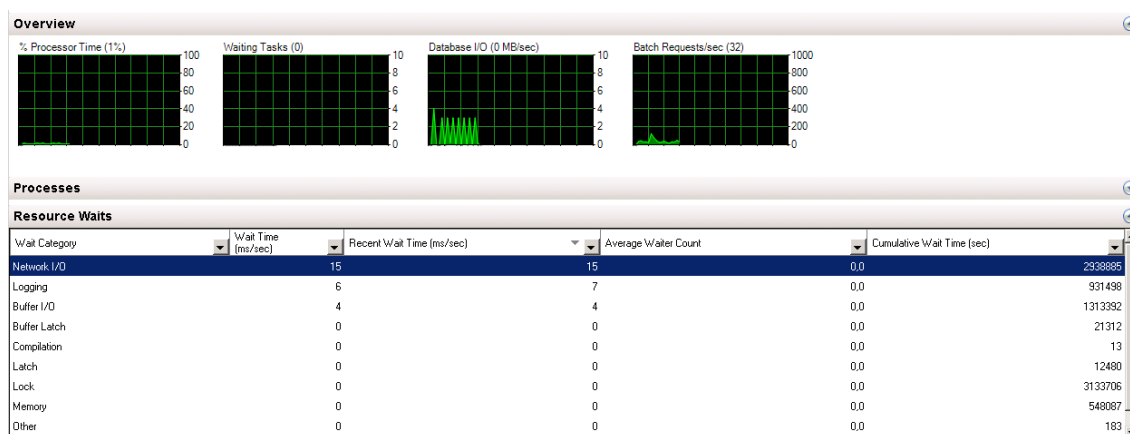
N-Central valvonta SQL-palvelimille:

- Kaikki SQL Server -Windows-palvelut palvelimelta
- Levyjen kokonais-I/O (Input-/Output-levyoperaatiot)
- Hälyrajojen muutoksia levyasemille ja keskusmuistille

Agentin ja muiden palveluiden kohdalla valvotaan sitä, ovatko ne käynnissä. Probe-yhteydelle, muistille, prosessorille, levyasemille ja levyjen I/O-arvolle on asetettu raja-arvot, joiden ylittyessä aiheutuu hälytys, mistä lähtee automaattisesti sähköposti Autotask-toiminnanohjausjärjestelmään. Autotask generoi sähköpostin pohjalta työpyynnön, jonka jälkeen valvomon henkilökunta alkaa tutkimaan ja korjaamaan tilannetta. Esimerkiksi levyasemien kohdalla valvotaan niiden täyttymistä, ja SQL-palvelimilla tuo raja-arvo on asetettu oletuksena 85 prosenttiin. Valvottavia kohteita skannataan asetuksista riippuen 5-15 minuutin välein. N-central säilyttää valvontadatan viimeisimmän kolmen kuukauden ajalta. Valvontadatan arkistoidaan myös Report Manager -järjestelmään, joka on N-centralin lisäosa. Report Managerista voi ajaa erilaisia raportteja ulos valvontadatan pohjalta,

ja se on asetettu säilyttämään kaikkein tärkein data tietokannassaan periaatteessa ikuisesti.

SQL-palvelimen ongelmatilanteiden selvityksessä apuna on siis N-centralin valvontadata. Hälytysten ja automaattisten tikettien avulla saadaan nopeasti tieto siitä, jos jokin palvelu tai instanssi lakkaa toimimasta tai jos palvelimen resurssit ovat normaalia kovemmassa käytössä. N-centralista saa RDP-etäyhteyden (Remote Desktop Protocol) palvelimelle, josta tilannetta päästään tutkimaan tarkemmin. Muun muassa Windowsin tapahtumienvälvonta, resurssienvälvonta, suorituskyvyn valvonta ja tehtävienhallinta auttavat ongelmatilanteen selvittämisessä. Tärkein työkalu niin SQL-palvelimen hallinnassa, kuin vianselvityksessäkin on SQL Server Management Studio, joka sisältää muun muassa SQL-lokin ja Activity Monitorin. Activity Monitorilla saadaan reaaliajassa tarkempaa tietoa muistinkäytöstä, tietokantojen I/O-lukemista ja käyttäjien avoinna olevista sessioista. Sillä voidaan tarkastella mitkä SQL-kyselyt vievät palvelimelta eniten resursseja, ja mitä resursseja kyselyt joutuvat palvelimella eniten odottamaan (wait time).



Kuvio 10. Kuvakaappaus SSMS:n Activity Monitorista, jossa on laajennettuna Resource Waits -näkyvä.

3.2 Tavoitteet

Nykyinen N-central-valvonta kertoo tällä hetkellä instanssi- ja palvelutasolla SQL-tietokantojen toiminnasta. Tietokantojen suorituskykyongelmien tutkimisessa käytetään lukuisia eri työkaluja. Ongelmien selvittäminen on usein työlästä ja vaatii paljon aikaa ja asiantuntemusta. Jos palvelimen fyysisten resurssien käytössä ei suoraan ole mitään poikkeavaa, joka aiheuttaisi valvonnassa hälytyksen, niin tilannetta päästään tutkimaan tarkemmin vasta, kun asiakas raportoi sovellusohjelman hitaudesta.

Protie on jo vuosia jatkuvasti kehittänyt ja automatisoinut valvontaa, jotta ongelmien korjaamisen sijaan voitaisiin keskittyä yhä enemmän ennaltaehkäisevään työhön. Tietokantavalvonnan tehostaminen on seuraava looginen askel tässä kehityksessä.

3.3 Testauksen toteutus

Ennen insinööriyön varsinaista aloittamista vertailtiin eri tietokantavalvontasovelluksia, joista testattaviksi valittiin ManageEnginen Applications Manager ja SolarWindsin Database Performance Analyzer. Lisäksi testataan N-centralin SQL-valvonnan eri ominaisuuksia, joita ei aikaisemmin ole vielä Protiellä hyödynnetty tietokantojen valvonnassa. Näihin sovelluksiin päädyttiin, koska ne ovat keskenään erilaisia, ja näin pystytään monipuolisesti tutkimaan erityyppisiä ratkaisuja tietokantavalvonnan tehostamiseksi.

Testauksen aluksi perustettiin uusi virtuaalinen Windows 2012 R2 -palvelin, johon asennettiin DPA ja Applications Manager. N-central on jo asennettuna omalle palvelimelleen ja tuotannossa, joten sen eri valvontaominaisuuksia voitiin ryhtyä suoraan testaamaan.

4 Valvontajärjestelmät

4.1 Valvontapalvelin

Testausosuus aloitettiin perustamalla virtuaalinen Windows 2012 R2 -palvelin Applications Managerille, sekä Database Performance Analyzerille. Ennen palvelimen perustamista selvitettiin asennettavien ohjelmistojen laitteistovaatimukset, ja näiden perusteella päätettiin palvelimen ominaisuudet.

Valvontapalvelimen ominaisuudet:

- Käyttöjärjestelmä: Windows Server 2012 R2 Standard 64-bit
- Kovalevy: 80Gt (NetApp-levyjärjestelmästä)
- Keskusmuisti: 8Gt
- Prosessorit: 2x2 Intel Xeon E5-2660 v2 2.20GHz

Asennus suoritettiin VMWare vSphere -virtuaalipalvelinalustalle Protien omaan palvelinverkkoon valmiista Windows 2012 R2 Standard -palvelinpohjasta (VM template). Koska kyseessä on virtuaalinen palvelin, resurssien lisääminen tai kapasiteetin kasvattaminen

onnistuu tarvittaessa jatkossa helposti. Keskusmuistin sekä prosessorien lisäämistä varten palvelin täytyy sammuttaa, mutta levytilaa voidaan allokoida lisää levyjärjestelmästä koska tahansa palvelimen ollessa päällä. Palvelimen käyttöönottoon liittyi eri vaiheita, jotka on kerätty alla olevaan listaukseen.

Valvontapalvelimen käyttöönotto:

- Ominaisuuksien määrittely
- Perustaminen
- Nimeäminen ja toimialueeseen liittäminen
- Verkkoasetusten määrittäminen (mm. staattinen IP-osoite)
- Windowsin aktivointi
- F-Securen asennus
- VMWare Toolsin päivitys
- SQL Server 2014 SE SP1:n asennus
- Päivittäminen (Windows Update)
- N-centralin valvontaan lisääminen
- Dokumentointi

Palvelinta perustettaessa sinne asennettiin myös SQL Server 2014 Standard Edition SP1 Applications Managerin ja Database Performance Analyzerin valvontadatojen tietokantoja varten. Asentaminen tapahtui Microsoftin Partner-portaalista hankitusta DVD-imagesta. SQL Server asennettiin alla olevassa kuvassa näkyvillä ominaisuuksilla. Asennuksen jälkeen palvelin vielä päivitettiin Windows Updaten avulla.

Information about the Setup operation or possible next steps:

Feature	Status
Management Tools - Complete	Succeeded
Client Tools Connectivity	Succeeded
Client Tools SDK	Succeeded
Client Tools Backwards Compatibility	Succeeded
Management Tools - Basic	Succeeded
Database Engine Services	Succeeded
Full-Text and Semantic Extractions for Search	Succeeded
Integration Services	Succeeded
Reporting Services - Native	Succeeded
Analysis Services	Succeeded
SQL Browser	Succeeded
SQL Writer	Succeeded
SQL Client Connectivity	Succeeded
SQL Client Connectivity SDK	Succeeded

Kuvio 11. DPA:n valvontadatan tietokantaa varten perustettu SQL Server ja sen ominaisuudet.

4.2 N-able N-central

4.2.1 Järjestelmävaatimukset

N-ablen ilmoittamat N-centralin järjestelmävaatimukset riippuvat valvottavien laitteiden määrästä. Alla olevissa listauksissa on esitelty järjestelmävaatimukset tuhannelle ja kuudelle tuhannelle valvottavalle laitteelle.

Valmistajan ilmoittamat järjestelmävaatimukset, kun 1000 valvottavaa laitetta:

- 75 Gt kovalevytilaa
- 4 Gt keskusmuistia
- 2-ytiminen prosessori, Intel Xeon 55xx tai uudempi
- Tietokanta valvontadatalle: PostgreSQL
- Käyttöjärjestelmä: Centos 6 (x64) Linux
- Verkon nopeus: 100 Mbps [41.]

Valmistajan ilmoittamat järjestelmävaatimukset, kun 6000 valvottavaa laitetta:

- 300 Gt kovalevytilaa
- 16 Gt keskusmuistia
- 8-ytiminen prosessori, Intel Xeon 55xx tai uudempi [41.]

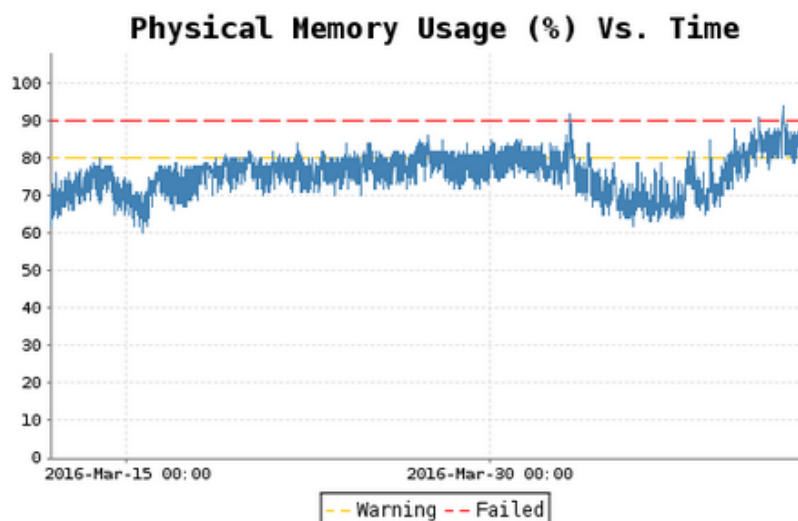
4.2.2 Ominaisuudet

N-central on IT-palveluntarjoajille suunnattu web-pohjainen sovellus, jolla valvotaan verkkolaitteita, palvelimia ja työasemia. Palvelinten osalta valvotaan pääasiassa niiden resurssienkäyttöä sekä niiden pyörittämiä palveluita ja prosesseja, joista valvotaan käytännössä sitä, ovatko ne käynnissä. Palvelinvalvonta on toteutettu valvonta-agentti-sovelluksella, joka asennetaan valvottaville palvelimille. Agentti kerää palvelimesta tietoja, valvoo koneen resursseja sekä palveluita ja lähettää datan valvontapalvelimelle.

Service	Status	Transition	Last Scan Time
Agent Status	✓	2016-Mar-31 23:34	2016-Apr-11 20:45
Connectivity	✓	2016-Mar-31 23:34	2016-Apr-11 20:43
CPU	✓	2016-Mar-31 23:34	2016-Apr-11 20:44
Disk - C:	✓	2016-Mar-31 23:34	2016-Apr-11 20:44
Disk I/O - _Total	✓	2016-Apr-01 16:59	2016-Apr-11 20:44
LocalIP	✓	2016-Apr-08 09:30	2016-Apr-11 20:44
Memory	⚠	2016-Apr-05 16:16	2016-Apr-11 20:44
Patch Status	✓	2016-Mar-31 23:34	2016-Apr-11 16:59
Windows Server Clock Drift	✓	2016-Apr-11 19:50	2016-Apr-11 20:45
Windows Service - Ignite PI Server	✓	2016-Apr-05 19:24	2016-Apr-11 20:44
Windows Service - ManageEngine Applications Manager	✓	2016-Apr-05 19:24	2016-Apr-11 20:44
Windows Service - SQL Server (MSSQLSERVER)	✓	2016-Apr-01 16:59	2016-Apr-11 20:44
Windows Service - SQL Server Agent (MSSQLSERVER)	✓	2016-Apr-05 16:59	2016-Apr-11 20:44

Kuvio 12. N-centralin näkymä valvottavista kohteista SQL-palvelimella.

N-centralin palvelimien resurssienvälvonta sisältää useita eri mittareita valvottavista resursseista. Esimerkiksi muistinkäytön osalta valvotaan sekä fyysistä että virtuaalista muistia, ja suoritinkäytön lisäksi valvontadataan tallentuu myös prosessoria palvelimella eniten kuormittavat sovellukset.



Kuvio 13. Graafinen esitys erään SQL-palvelimen fyysisen muistin käytöstä viimeisen kuukauden ajalta N-centralista.

4.2.3 SQL-valvonta

SQL-palvelimien resurssien osalta N-central valvoo tällä hetkellä prosessorin, keskusmuistin sekä palvelimen levyasemien käyttöä. Lisäksi valvotaan SQL Server -Windows-palveluita. N-central sisältää myös SQL-tietokantojen valvontatyökaluja, jotka otetaan käyttöön ja testataan. Niiden sisältämät mittarit on kerätty alla oleviin luetteloihin.

SQL Server (per instanssi):

- Active transactions – aktiiviset transaktiot
- Log files size (KB) – lokitiedostojen koko (Kt)
- Data files size (KB) – tietokantojen koko (Kt)
- Transactions/second – transaktiot sekunnissa
- Number of user connections – käyttäjien istuntojen määrä
- Average waiting time – keskimääräinen odotusaika
- Deadlocks/second – lukkoristiriitoja sekunnissa

SQL Database Information (per instanssi, per tietokanta):

- Database Name – tietokannan nimi
- Log files size (KB) – lokitiedostojen koko (Kt)
- Transactions per second – transaktiot sekunnissa
- Data files size (KB) – tietokantatiedostojen koko (Kt)

SQL Server Locks (per instanssi):

- Lock Waits per second – lukkojen odotusten määrä sekunnissa
- Average wait time in milliseconds – keskimääräinen odotusaika (ms)
- Lock timeouts per second – lukkojen aikakatkaisuja sekunnissa
- Number of deadlocks per second – lukkoristiriitoja sekunnissa

SQL Memory Manager (per instanssi):

- Connections memory (KB) – yhteyksien ylläpitoon käytetty muisti (Kt)
- Lock memory (KB) – lukkojen ylläpitoon käytetty muisti (Kt)
- Optimizer memory (KB) – optimoijan käyttämä muisti (Kt)
- SQL cache memory (KB) – suunnitelmavälimuistin koko (Kt)
- Total server memory (KB) – palvelimen muistin koko (Kt)
- Memory Grants Pending – odottavat muistivaraukset

SQL Server – Buffer Manager (per instanssi):

- Buffer cache hit ratio – välimuistin käyttö (%)
- Page reads per second – sivujen luvut sekunnissa
- Page writes per second – sivujen kirjoitukset sekunnissa
- Page life expectancy (hr) – sivun odotettavissa oleva elinikä (t)
- Lazy writes per second – lazy writer kirjoittaa vanhoja sivuja välimuistista levyille (vapauttaa tilaa välimuistista)

SQL Transaction Information (per instanssi):

- Latch waits per second – salpojen odotusten määrä sekunnissa
- Full Scans per second – ilman indeksiä suoritettut haut tauluista sekunnissa
- User connections – käyttäjien yhteyksien määrä
- Logins per second – kirjautumisten määrä sekunnissa

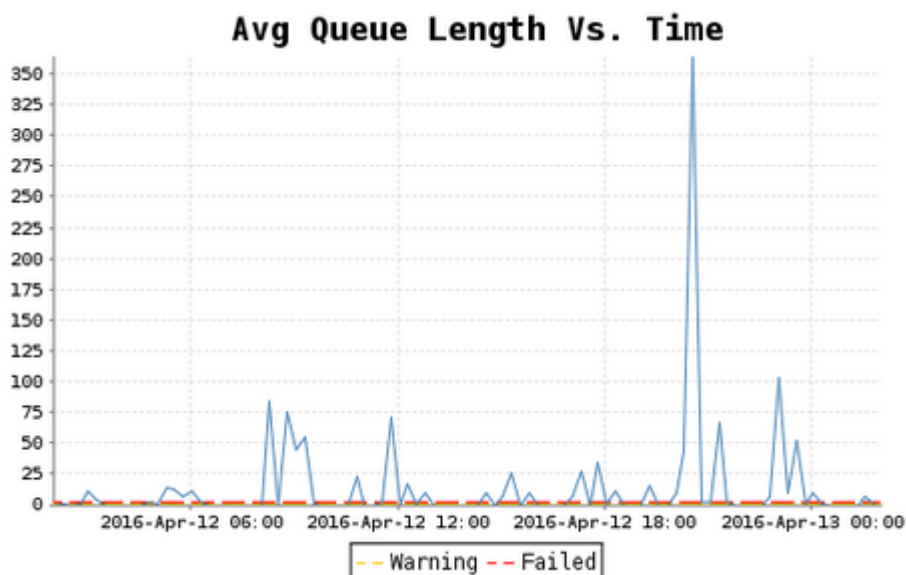
Description	Value	State	Thresholds
		✓ Normal	
Connections memory (KB)	504	Warning	
		Failed	
		✓ Normal	
Lock memory (MB)	215.78	Warning	
		Failed	
		✓ Normal	
Optimizer memory (MB)	1.45	Warning	
		Failed	
		✓ Normal	
SQL cache memory (MB)	100.34	Warning	
		Failed	

Kuvio 14. Kuva SQL Server Memory Manager -valvonnasta, jossa näkyy palvelimen sen hetkistä muistinkäyttöä.

Levyasemien I/O-valvonta on säädetty valvomaan SQL-palvelimen kaikkien levyjen I/O-operaatioita, ja testausta varten muutetaan valvonta mittaamaan myös levyasemia yksitellen, jolloin saadaan tarkempaa dataa tietokantalevyille kohdistuvista levyoperaatioista. I/O-valvonta sisältää alla luetellut mittarit.

Disk I/O:

- Current Queue Length – levyjonon pituus
- Avg Queue Length – keskimääräinen jonon pituus
- Write Operations/Second – kirjoitusoperaatioita sekunnissa
- Read Operations/Second – lukuoperaatioita sekunnissa
- Total Operations/Second – levyoperaatioita sekunnissa
- Data Written (KB) – dataa kirjoitettu (Kt)
- Data Read (KB) – dataa luettu (Kt)
- Total Data Sent/Received (KB) – dataa lähetetty/vastaanotettu (Kt)
- Idle Time (%) – levyn odotusaika
- Disk Time (%) – levyoperaatiot jonossa
- Disk Read Time (%) – lukuoperaatiot jonossa
- Disk Write Time (%) – kirjoitusoperaatiot jonossa



Kuvio 15. Kuvassa näkyy SQL-palvelimen tietokantalevyn levyjonon vaihtelua 24 tunnin aikana.

4.2.4 Hälytykset

N-centraliin on Protiellä tehty erilaisia sääntöjä, jotka määrittävät valvottavat kohteet ja niiden hälyrajat, kun palvelin lisätään valvontaan. Kun palvelin lisätään valvontaan, N-central tunnistaa, että kyseessä on SQL-palvelin ja asettaa sille valvonnan ja hälytykset SQL-palvelimille tehtyjen sääntöjen mukaan. Tämän jälkeen valvottavia kohteita voidaan poistaa tai lisätä palvelimelle ja mittareiden hälyrajoja voidaan myös muokata. Mittareissa on kolme eri tilaa, jotka ovat normal (normaali), warning (varoitusta) ja failed (kriittinen).

Sähköpostihälytykset säädetään ilmoitusasetuksilla, ja myös ne asetetaan automaattisesti palvelimille ennalta luodun säännön mukaan, palvelimen tyypistä riippuen. Sähköpostista generoituu automaattisesti työpyyntö Protien toiminnanohjausjärjestelmään, jonka jälkeen tilannetta aletaan tutkia. Hälytykset on pääsääntöisesti määritelty niin, että kun warning- tai failed-raja ylitetään, niin siitä generoituu hälytys sähköpostitse.

4.3 ManageEngine Applications Manager

4.3.1 Järjestelmävaatimukset

Valmistajan ilmoittamat järjestelmävaatimukset, kun 0-250 valvottavaa kohdetta:

- 20 Gt kovalevytilaa
- 2 Gt keskusmuistia
- 1.4 GHz prosessori
- Tietokanta valvontadatalle: SQL Server tai MySQL
- Käyttöjärjestelmä: Windows Server, Windows tai Linux
- Verkkoselain: IE, Firefox, Chrome
- Tuetut DBMS:t: Oracle, SQL Server, DB2, Informix, MySQL, Sybase, PostgreSQL, MongoDB, Cassandra, Redis, Couchbase [42.]

Valmistajan ilmoittamat järjestelmävaatimukset, kun 250-1000 valvottavaa kohdetta:

- Yksi admin-valvontapalvelin ja 2-3 hallintapalvelinta
- 40 Gt kovalevytilaa
- 2 Gt keskusmuistia
- 1.8 GHz prosessori [42.]

Valmistajan ilmoittamat järjestelmävaatimukset, kun yli 1000 valvottavaa kohdetta:

- Yksi admin-valvontapalvelin ja vähintään 4 hallintapalvelinta
- Vähintään 100 Gt kovalevytilaa
- 2-4 Gt keskusmuistia
- 2 * 1.8 GHz prosessori [42.]

4.3.2 Asentaminen

ManageEngine Applications Managerista asennettavaksi versioksi valittiin Professional Edition. Enterprise Edition on tätä laajempi asennus, joka tarvitsee yhden niin kutsutun admin-palvelimen lisäksi myös muita hallintapalvelimia. Versioiden välillä ei ole muuten mitään toiminnallista eroa, mutta Enterprise Edition tuo järjestelmään parempaa vikasietoisuutta ja skaalautuvuutta. Käytössä on kokeiluversio, jossa on 30 päivän ajan testattavana kaikki maksullisen version ominaisuudet. Asennuksen jälkeen Applications Managerin hallintaan kirjaudutaan verkkoselaimella, jossa asetusvelhon avulla perustetaan valvontadatalle oma SQL-tietokanta, ja lisätään testausta varten SQL-instanssit valvontaan.

ManageEngine Applications Manager 12 Setup

Microsoft SQL Server Database Setup

ME

Host Name: localhost

Port Number: 1433

Database: AMDB

User Name: sa

Password: [masked]

Instance: --Named Instance--

InstallShield

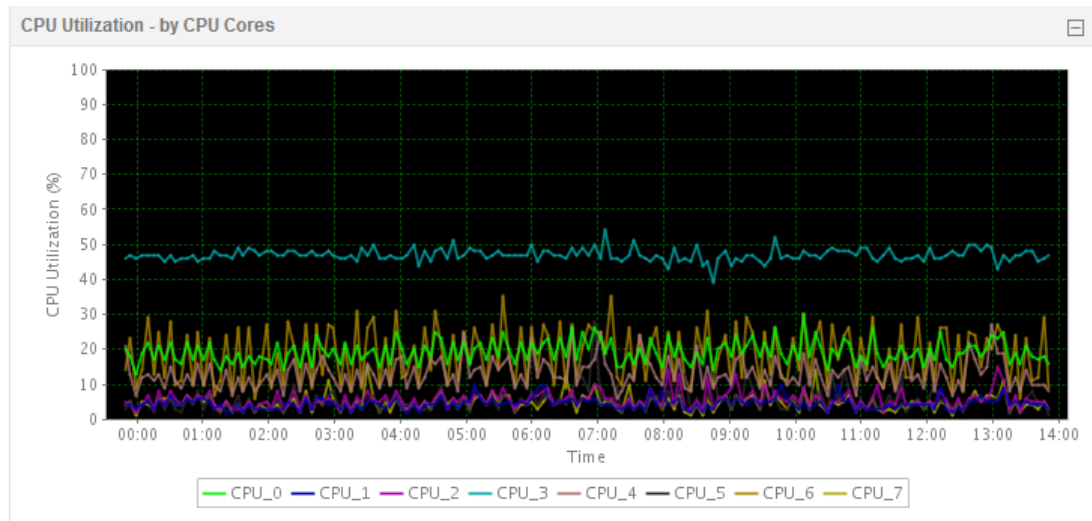
< Back Next > Cancel

Kuvio 16. SQL-tietokannan perustaminen Applications Managerin valvontadataa varten.

4.3.3 Ominaisuudet

Applications Manager on ManageEnginen kehittämä web-pohjainen sovellus palvelimien sekä niiden ohjelmien ja palvelujen valvontaa varten. N-centraliin verrattuna painopiste on enemmän palvelinten sovellusten toiminnan ja suorituskyvyn valvonnassa, ja siksi se myös sisältää N-centralia huomattavasti enemmän sovelluskohtaisia mittareita. Toisaalta Applications Managerin avulla ei voida valvoa verkkolaitteita, kuten kytkimiä, reitittimiä tai palomuureja, mutta se voidaan integroida toimimaan ManageEnginen verkonvalvontasovelluksen OpMangerin kanssa samassa web-hallintaympäristössä.

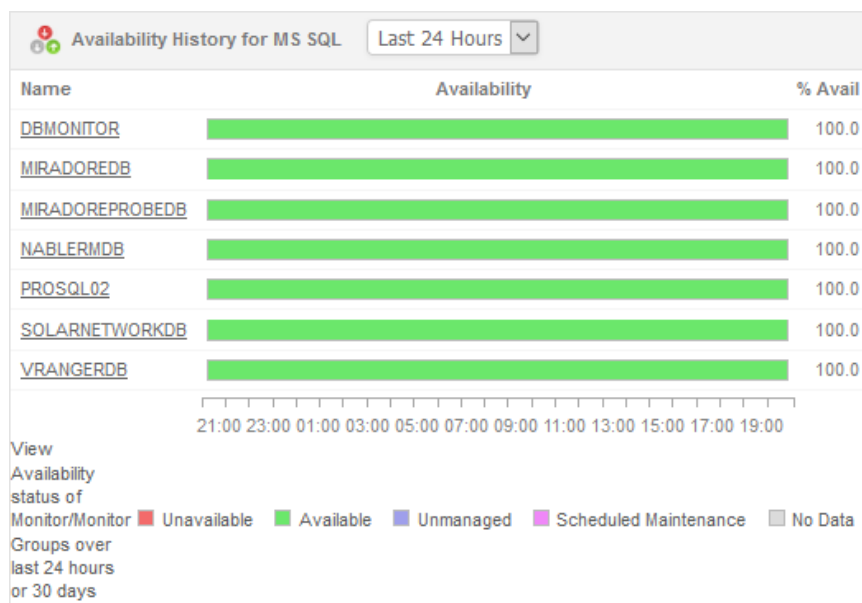
Kohteita lisätään valvontaan palvelin- tai sovelluskohtaisesti. Windows-palvelimista voidaan valvoa niiden resurssienkäytön lisäksi esimerkiksi prosesseja ja palveluita, verkko-liikenteen määrää, tapahtumalokia sekä ajoitettuja tehtäviä. Palvelimille ei asenneta erikseen agentti-sovellusta, vaan valvonta toimii WMI-infrastruktuurin (Windows Management Instrumentation) ja SNMP-protokollan avulla (Simple Network Management Protocol).



Kuvio 17. Applications Mangerin piirtämä kuvaaja prosessorien kuormituksesta eräällä Windows-palvelimella.

4.3.4 SQL-valvonta

ManageEngine Applications Managerin SQL-valvonnan etusivu sisältää listauksen valvottavista palvelimista sekä niiden tilatiedon viimeisen 24 tunnin tai 30 päivän ajalta. Lisäksi palvelimen välimuistin käyttö, yhteyksien määrä, muistin määrä ja verkkoyhteyden latenssi on esitetty graafisesti viimeisen tunnin ajalta. Tarkempaa dataa voidaan tarkastella palvelinkohtaisesti nimeä klikkaamalla.

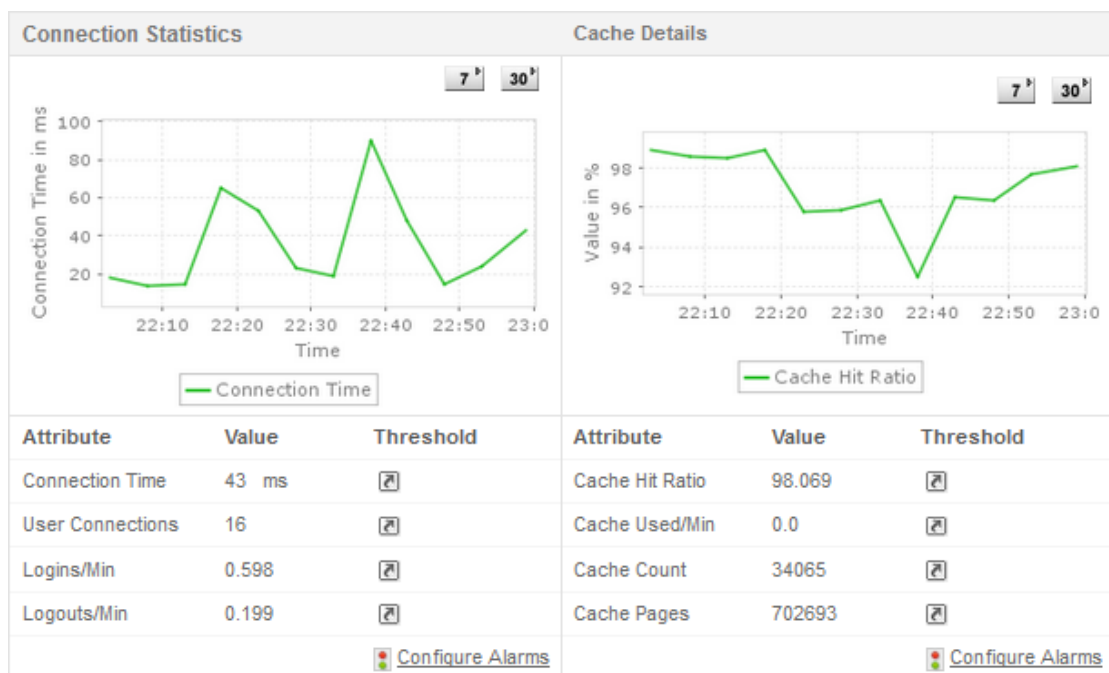


Kuvio 18. Applications Managerin SQL-valvonnan etusivulla nähdään valvottavien palvelimien tilatiedot viimeisen 24 tunnin tai 30 päivän ajalta.

SQL-palvelinta tarkastellessa sen valvonnan yleisnäkymässä nähdään palvelimen resurssimittareista niiden tämänhetkiset tiedot. Resurssimittarit on jaettu ryhmiin niiden tyyppin mukaan, ja jokaisella näistä ryhmistä on näkymässä oma laatikko. Jokaisessa laatikossa on myös graafinen esitys yhdestä tai useammasta sen mittarista viimeisen tunnin ajalta.

Applications Managerin resurssimittarit palvelimen yleisnäkymässä:

- Monitor Information – perustiedot valvottavasta palvelimesta
- Today's Availability – palvelimen tila tältä vuorokaudelta
- Memory usage – muistinkäytön mittarit
- Connection Status – tiedot yhteyksistä
- Cache Details – suunnitelmavälimuistinkäytön mittarit
- Lock Details – lukkojen mittarit
- SQL Statistics – kyselyiden mittarit
- Latch Details – salpojen mittarit
- Access Method Details – datahakujen mittarit
- Job Details – tiedot SQL-palvelimen aikataulutetuista tehtävistä (jobs)
- Database Details – tiedot tietokannoista



Kuvio 19. SQL-palvelimen mittarit on Applications Managerissa jaettu eri kategorioihin, joilla on omat laatikkonsa palvelimen päänäkymässä. Kuvassa on palvelimen yhteyksien sekä suunnitelmavälimuistin tiedot.

Palvelimen tietokannoilla on myös omat näkymänsä, joissa on tarkempia tietoja niiden resurssienkäytöstä ja toiminnasta. Tietokannan tiedoista nähdään muun muassa sen asetustietoja, kuten onko se säädetty päivittämään tilastot automaattisesti, sekä kyseisen tietokannan palautustyyppi.

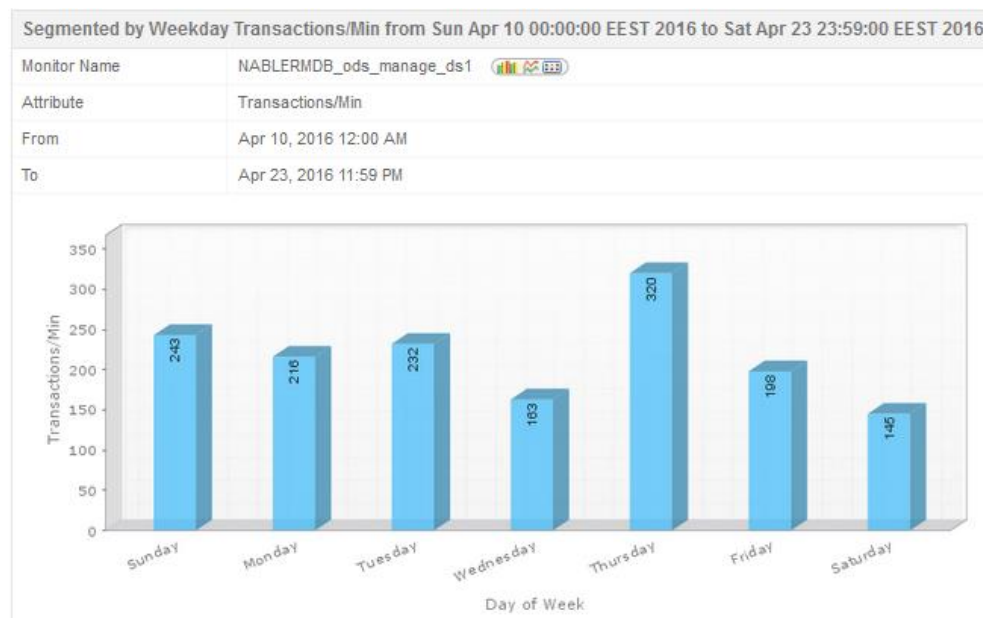
Tietokannan resurssimittarien yleisnäkymä:

- Database Details – perustiedot tietokannasta
- Today's Availability – tietokannan tila kuluvalta vuorokaudelta
- Datafile Details – tietokannan koko
- Log File Size – lokitiedoston koko ja tietoja sen käytöstä
- Transaction Details – transaktioiden mittarit
- Log Flush Details – mittarit lokitietojen siirroista välimuistista levyille
- Database Properties – tietokannan asetukset

Database Properties					
Attribute	Value	Threshold	Attribute	Value	Threshold
Creation Date	2012-09-11		Auto Shrink	False	
Database Mode	SIMPLE		Auto Create Statistics	True	
Page Verify	CHECKSUM		Auto Update Statistics	True	
Compatibility Level	100		Configure Alarms		

Kuvio 20. Applications Manager näyttää myös tietoja tietokannan asetuksista.

Palvelimen mittareiden kehitystä voidaan tarkastella halutulla aikavälillä siirtymällä historiatietoihin, joissa mittareista voidaan luoda erilaisia kuvaajia ja diagrammeja. Valmiit pohjat sisältävät muun muassa näkymän, jossa mittarin kehitystä voidaan tarkastella halutulta ajanjaksolta viikonpäivien mukaan. Samankaltainen diagrammi voidaan myös luoda vuorokauden tuntien suhteen, jolloin pystytään tarkastelemaan mittarin kehitystä eri vuorokaudenaikoina pidemmältä aikaväliltä kuin vain yhden päivän ajalta.



Kuvio 21. Pylväsdiagrammi, jossa nähdään kahden viikon ajalta transaktioiden keskimääräinen määrä minuutissa suhteessa viikonpäiviin.

Palvelimen tiedoissa on lisäksi suorituskykynäkymä, jossa voidaan luoda erilaisia listoja esimerkiksi raskaimmista kyselyistä, palvelimen odotustyypeistä tai indekseistä. Näitä listoja voidaan kohdentaa myös yksittäiseen tietokantaan. Valittavina ovat esimerkiksi listat eniten käytetyistä kyselyistä sekä hitaimmista kyselyistä. Missing Indexes -listaus näyttää SQL-palvelimen ylläpitämän tilaston puuttuvista indekseistä, joiden luonti voi mahdollisesti nopeuttaa kyselyiden toteutusta.

4.3.5 Hälytykset

Application Managerissa mittareilla on kolme mahdollista tilaa, jotka ovat critical (kriittinen), warning (varoitusta) ja clear (normaali). Hälyrajat konfiguroidaan ensin hälyrajaprofiileihin (threshold profiles), joihin voidaan määrittellä prosentuaaliset tai numeeriset arvot. Tämän jälkeen luodun profiilin voi liittää yhdelle tai useammalle valvottavalle kohteelle, yhteen tai useampaan eri mittariin. Valvottavista kohteista voi luoda ryhmiä, joihin hälyrajaprofiilit kohdistetaan, tai hälyrajat voidaan myös määrittää tietyn tyyppisille kohteille samalla kertaa, kuten esimerkiksi kaikille SQL-palvelimille. Kun hälyrajat määritellään mittareihin, ne voidaan liittää samalla ennalta luotuun toimintaprofiiliin (action profile), jossa määritellään esimerkiksi sähköpostin lähetys tukeen silloin kun hälytysraja ylittyy.

4.4 SolarWinds Database Performance Analyzer

4.4.1 Järjestelmävaatimukset

Valmistajan ilmoittamat järjestelmävaatimukset:

- 3 Gt kovalevytilaa per SQL-instanssi
- 4 Gt keskusmuistia
- Tietokanta valvontatallalle: Oracle tai SQL Server
- Käyttöjärjestelmä: Linux/Unix tai Windows Server
- Verkkoselain: IE, Firefox, Chrome
- Java Virtual Machine 1.6 tai 1.7
- Tuetut DBMS:t: Oracle, SQL Server, SAP ASE, DB2, MySQL [43.]

4.4.2 Asentaminen

SolarWinds Database Performance Analyzeristä ladattiin ja asennettiin palvelimelle uusin versio, joka oli testin alkaessa 10.1.313. Käytössä on kokeiluversio, joka sisältää kaikki täyden version ominaisuudet. Ainoana erona maksulliseen versioon on rajattu kokeilu-aika, joka saatiin ennen testin alkua pidennettyä normaalista 14 päivästä 28 päivään. Asennus sujui nopeasti, minkä jälkeen DPA:n hallintaan päästiin kirjautumaan verkkoselaimella. Asetusvelhon avulla perustettiin ensin valvontaa varten oma SQL-tietokanta ja luotiin DPA:lle omat tunnukset tuohon tietokantaan. Tämän jälkeen rekisteröitiin SQL-instanssit testausta varten valvontaan.

Kuvio 22. SQL-instanssin rekisteröiminen DPA:n valvontaan.

4.4.3 Ominaisuudet

Database Performance Analyzer on ainoastaan tietokantojen valvontaan kehitetty sovellus, ja sen pääominaisuutena on tietokannan suorituskyvyn mittaaminen SQL-kyselyiden keston avulla. Se pitää kirjaa kaikista valvottaviin tietokantoihin kohdistuneista SQL-kyselyistä ja jakaa niiden keston ajallisesti eri odotusaikoihin sekä kyselyn varsinaiseen

suoritus aikaan. Odotusaikoja eri näkökulmista tutkimalla voidaan löytää palvelimen resursseista ja kyselyistä niin kutsuttuja pullonkauloja, joihin puuttumalla voidaan parantaa palvelimen toimintaa.

Wait	Queries	CPU	Mem	Disk	Sess	Type
						MS SQL 2008 R2
						MS SQL 2008 R2 SP1
						MS SQL 2014
						MS SQL 2008 R2 SP1
						MS SQL 2012 SP1
						MS SQL 2005 SP2

Kuvio 23. DPA:n etusivulla näky valvottavien SQL-palvelinten tilatiedot.

DPA tarjoaa valvontadatan lisäksi tietoa datan tulkintaan ja antaa myös vinkkejä, joiden avulla palvelimen suorituskykyä voidaan mahdollisesti parantaa. Esimerkiksi kaikki eri odotustyyppit on selitetty, sekä lisäksi mahdolliset ratkaisut niistä johtuvien ongelmien selvittämiseksi. DPA analysoi kyselyitä myös käskytasolla ja ehdottaa sen pohjalta raskaimpien ja eniten aikaa vievien kyselyiden tehostamiseksi ratkaisuja, kuten indeksin luomista hauissa paljon käytettyihin sarakkeisiin.

PAGEIOLATCH_EX

Exclusive mode page IO latch request. The PAGEIO latches are a subset of BUF latches used when the buffer and associated data page or the index page is in the middle of an IO operation. PAGEIOLATCH waittypes are used for disk-to-memory transfers.

Resolved By

DBAs, Developers, System Administrators and Storage Administrators

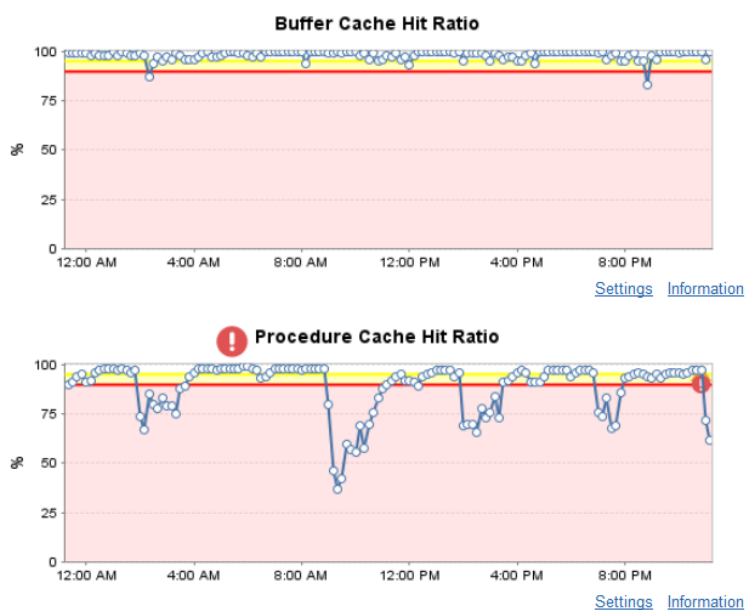
Solutions

1. Tune the SQL statement waiting on this waittype. Inefficient SQL statements read more data than necessary and will try to acquire more latches than necessary as a result.
2. Significant wait time for this waittype may suggest disk I/O subsystem issues.

Kuvio 24. DPA:n infosivu PAGEIOLATCH_EX-odotustyyppistä, jossa selitetään sen syy, sekä mahdolliset ratkaisut siitä johtuvien ongelmien selvittämiseksi.

Vaikka DPA:n valvonta keskittyykin nimenomaan SQL-palvelimen suorituskyvyn mittaamiseen kyselyiden odotusaikojen avulla, valvoo se myös SQL-palvelimien resurssien-

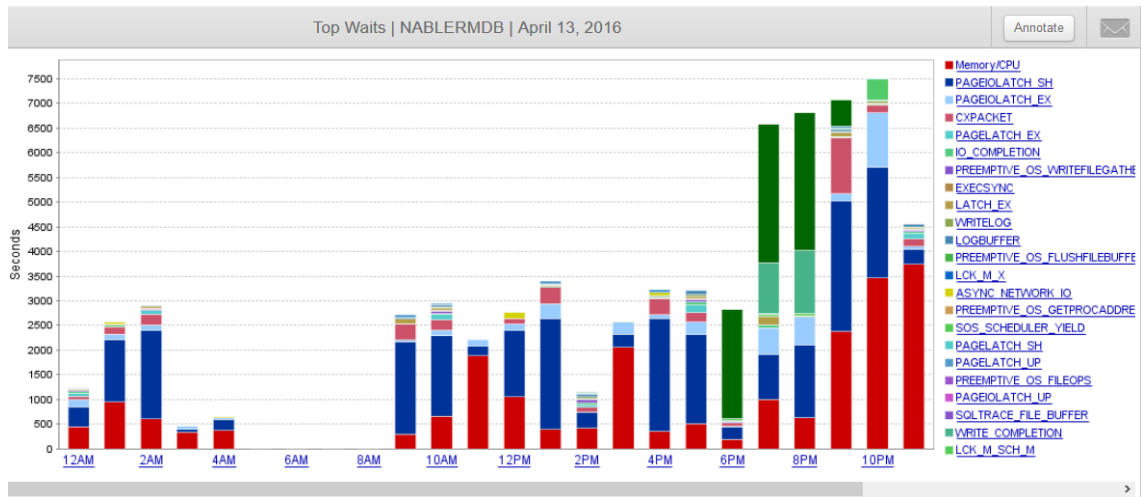
käyttöä. Kyselyitä ja niiden odotusaikojen suhteen voi verrata palvelimen resurssien käyttöön, ja etsiä niistä syy-seuraussuhteita. Palvelinten resurssienvälvön osat sisältävät prosessorin, muistin, levyn ja verkkoliikenteen valvonnan, joissa jokaisessa on lukuisia eri mittareita. Esimerkiksi muistinkäyttöä valvotaan kahdeksalla eri mittarilla, joilla valvotaan sekä SQL-instanssin välimuistia, että koko palvelimen muistinkäyttöä. Osa valvottavista mittareista vaatii maksullisen VMWare-lisäosan, jonka avulla voidaan valvoa virtuaalipalvelimien kohdalla myös niiden isäntäpalvelimen (host) resurssien käyttöä.



Kuvio 25. DPA:n piirtämä 24 tunnin kuvaaja välimuistin ja suunnitelmavälimuistin käytöstä.

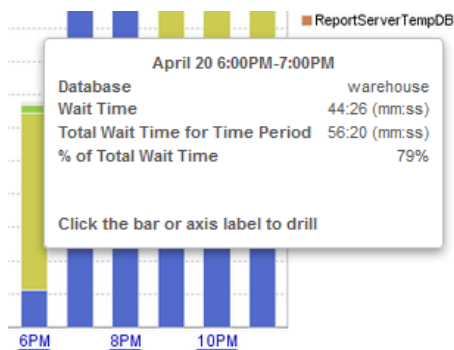
4.4.4 SQL-valvonta

Database Performance Analyzer tarjoaa SQL-palvelimen suorituskyvyn tutkimiseen pääasiassa erilaisia pylväsiagrammeja, joita voidaan skaalata tarpeen mukaan. Suurin osa diagrammeista näyttää SQL-palvelimeen kohdistuneiden kyselyiden kokonaiseston halutulla ajanjaksolla. Pylväät jaotellaan erivärisiin osiin, joiden merkitys riippuu siitä, mistä näkökulmasta odotusaikojen halutaan tutkia. Odotusaikojen voidaan tarkastella esimerkiksi raskaimpien kyselyiden mukaan, jolloin valitulta ajanjaksolta lasketaan kyselyn kaikkien suorituskertojen odotusaikojen summa omaksi värikseen pylväisiin. Kyselyiden lisäksi pylväät voidaan jaotella muun muassa odotustyyppien, ohjelmien, tietokantojen ja käyttäjien mukaan.








Kuvio 26. Pylväsdiagrammi, jossa kyselyiden odotusaika esitetään yhden vuorokauden tuntien suhteen, ja kokonaisodotusaika on jaoteltu eri väreihin odotustyyppien mukaan.

Diagrammia skaalaamalla voidaan datasta havaita trendejä eri ajanjaksoilla. Pylväitä klikkaamalla päästään tarkastelemaan valittua ajanjaksoa tarkemmin. Kun hiiren osoittimen vie minkä tahansa pylvään väriin päälle, tulee näkyviin tarkempaa dataa valitusta kohteesta. Lisäksi pylväsdiagrammin sivussa on aina selitykset eri väreille, joita klikkaamalla saadaan niistä lisätietoa. SQL-kyselyiden odotusaikoja tutkiessa voidaan esimerkiksi tarkastella valittua kyselyä T-SQL-kielenä.



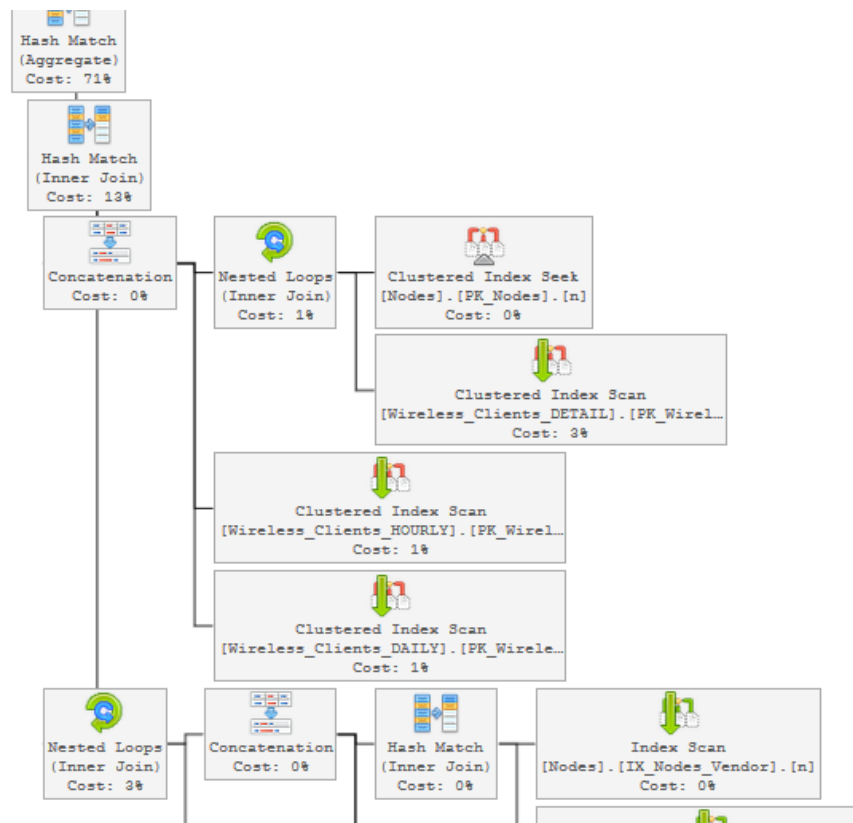
Kuvio 27. Diagrammissa on esitetty eri tietokantoihin kohdistuneiden kyselyiden kokema odotusaika eri väreillä. Osoittamalla hiirellä haluttua väriä saadaan tarkempaa tietoa valvontadatasta.

Diagrammien alapuolella on DPA:n ehdotuksia siitä, mitä kyselyitä mahdollisesti kannattaisi tutkia tarkemmin. Lisäksi diagrammin alle saadaan valittua palvelimen resurssien käytön eri mittareita, jolloin odotusaikojen muutoksia voidaan suoraan verrata palvelimen resurssienvontadataan tarkasteltavalla ajanjaksolla.

April 14	
	Query 5142641082 accounted for 20% of instance execution time, plan has changed more...
	Query 3111430980 accounted for 9% of instance execution time, 100% in Memory/CPU more...
	Query 4757252257 accounted for 5% of instance execution time, high executions (5,167) during 9AM hour more...
	Query 4183323189 accounted for 4% of instance execution time, high executions (5,919) during 1PM hour more...
	Query 3311840830 accounted for 4% of instance execution time, 100% in Memory/CPU more...

Kuvio 28. DPA ehdottaa käyttäjälle mahdollisia ongelmallisia kyselyitä tarkastettavaksi.

DPA kertoo myös kyselyiden kokonaisodotusajan lisäksi niiden suorituskertojen määrän sekä yhden suorituskerran keskimääräisen keston. Myös kyselyiden loogisia toteutus suunnitelmia voidaan tarkastella graafisesti.



Kuvio 29. SQL-kyselyn toteutussuunnitelman graafista esitystä.

4.4.5 Hälytykset

Database Performance Analyzerin resurssimittareilla voi olla neljä eri tilaa, jotka ovat critical (kriittinen), warning (varoitus), broken (rikki) ja normal (normaali). Hälyrajat muokataan jokaiselle palvelimelle erikseen. Email-hälytykset konfiguroidaan alerts-välilehdeltä, ja ne voidaan määrittää resurssimittareiden hälyrajojen mukaan joko yhdestä tai kaikista resursseista samalla kertaa. Kun hälytys on luotu, voidaan se määrittää yhdelle tai useammalle palvelimelle. Lisäksi voidaan luoda hälytysryhmiä, joihin liitetään halutut palvelimet ja aikaisemmin luodut hälytykset. Hälytyksiä voidaan asettaa myös kohteista, joita DPA:n valvonnassa ei muuten käsitellä. Esimerkiksi SQL Serverin aikataulutetuista tehtävistä voidaan luoda automaattisia hälytyksiä. Hälytysten lisäksi voidaan aikatauluttaa sähköpostitse lähetettäviä raportteja valvontadatasta.

5 Vertailu

Testaukset toteutettiin Protien omassa tuotantoympäristössä, jonka SQL-palvelimia valvottiin valvontasovelluksilla koko testausjakson ajan. Samalla kerättyä valvontadataa analysoitiin, ja sen perusteella tehtiin myös joitain muutoksia tietokantapalvelimille ja niiden tietokantoihin. Tässä luvussa arvioidaan ja vertaillaan testattujen valvontasovellusten hyödyllisyyttä Protien tietokantavalvontaan. Testaustulosten perusteella tehtiin lopulta päätös siitä, mitkä ratkaisut otetaan Protien valvomoon käyttöön.

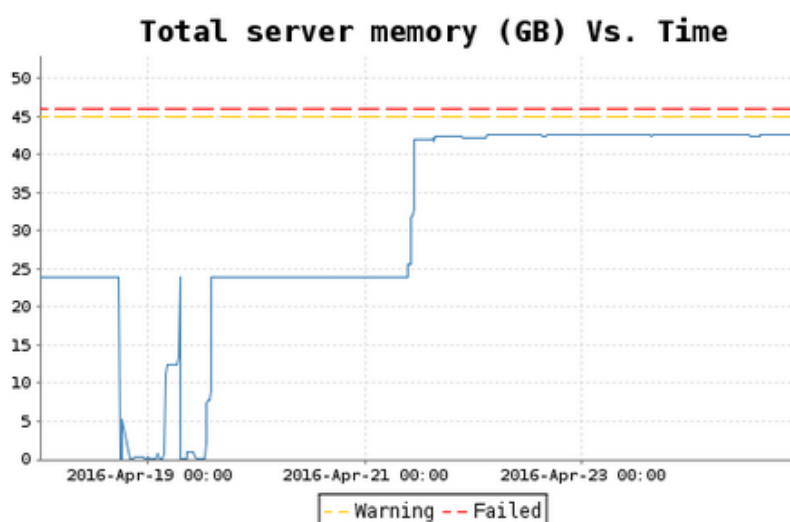
5.1 N-central

N-centralin tietokantavalvonta sisältää monia hyödyllisiä mittareita, joiden perusteella pystytään seuraamaan tietokantapalvelimen toimintaa, ja vikatilanteessa ne auttavat ongelman syyn rajaamisessa. Kaikki N-centralin valvomat mittarit ovat SQL-palvelimen ylläpitämiä tilastoja, jotka kertovat palvelimen toiminnasta. SQL Server ei tallenna mittareiden tietoa mihinkään, ja monet näistä mittareista ovat monimutkaisten SQL-kyselyiden takana. N-central tallentaa SQL Serverin oleellisimpien mittareiden arvot helposti ymmärrettävään ja nopeasti saatavilla olevaan muotoon. Kerätty valvontadata voidaan esittää graafisesti, jolloin mittareiden kehitystä on helppo tarkastella pidemmällä aikavälillä. Tämä helpottaa vikatilanteen ymmärtämistä, koska yksittäiset mittareiden lukemat eivät useinkaan kerro palvelimen toiminnasta mitään ilman vertailukohtaa aikaisempaan, kyseiselle palvelimelle tai tietokannalle normaaliin tilanteeseen.

SQL-palvelimen kohdalla pelkkä palvelimen muistinkäytön valvominen ei riitä, koska SQL Server varaa yleensä palvelimelta kaiken vapaana olevan keskusmuistin oman välimuistinsa käyttöön ja vapauttaa sitä tarpeen mukaan muille prosesseille dynaamisesti. 'Buffer cache hit ratio (%)' on esimerkki hyvin oleellisesta mittarista, jota N-central valvoo. Se kertoo, kuinka suuri osa kyselyistä kohdistuu välimuistissa valmiina oleviin sivuihin, eli käänteisesti se kertoo miten suuri osa kyselyiden hakemasta datasta joudutaan hakemaan levyiltä. Transaktioiden määrän sekä sivujen lukujen ja kirjoitusten määrän kehityksestä nähdään sen sijaan nopeasti, ovatko SQL-kyselyt lisääntyneet tai niiden kuormitus palvelimelle kasvanut.

Vaikka N-central nopeuttaa ongelmatilanteiden selvitystä ja antaa lähtötietoja vianrajaukseen, laajempi selvitystyö on kuitenkin tehtävä palvelimella. Tämä työ vaatii tekijältä usein laajaa tietämystä tietokantojen toiminnasta.

SQL-mittareita N-centralin valvontaan lisättäessä huomattiin yhdellä SQL-palvelimella tietokantainstanssin muistin olevan alle 30 gigatavua – huolimatta siitä, että kyseiselle virtuaalipalvelimelle oli allokoitu 48 gigatavua keskusmuistia. Windowsin tiedoista nähtiin, että palvelimella pyöri Windows 2008 R2 Standard -käyttöjärjestelmä, joka tukee enintään 32 gigatavua keskusmuistia. Käyttöjärjestelmä päivitettiin Enterprise-versioon, jonka jälkeen SQL-instanssin muistinkäyttö saatiin nostettua 44 gigatavuun. 4 gigatavua jätettiin tarkoituksella käyttöjärjestelmän ja muiden prosessien käyttöön rajoittamalla SQL-instanssin muistinkäyttöä SQL Server Management Studiosta.



Kuvio 30. Kuvaajassa nähdään SQL-instanssin muistinkäyttö, joka oli alle 25 Gt ennen palvelimen päivittämistä uudelle lisenssille. SQL Server oli pysäytettyä palvelimen päivitysoperation ajan, minkä vuoksi muistinkäyttö on ensin käynyt nollassa.

5.2 Applications Manager

Applications Manager on N-centralin tapaan eräänlainen palvelinvalvonnan kokonaisratkaisu. SQL-valvonnan osalta sen ominaisuudet ovat N-centralia monipuolisemmat, ja se sisältää N-centralia enemmän eri mittareita SQL-palvelimen valvontaan. Erityisesti tietokantakohtaisia mittareita on huomattavasti enemmän kuin N-centralissa. Applications Manager kertoo kannoista myös joitain oleellisia asetustietoja. Tietokannoista nähdään nopeasti esimerkiksi taulujen koot sekä indeksien käyttötilastoja. Myös mittareiden historiatietojen tarkastelu on hieman N-centralia monipuolisempaa. Kuvaajien skaalaus on vapaampaa, ja valvontadatan esittämiseen on tarjolla useita erilaisia vaihtoehtoja.

Applications Manager sisältää mittareiden valvonnan lisäksi myös muita hyödyllisiä ominaisuuksia tietokantojen valvontaan. Aikataulutettujen tehtävien suoritusten valvominen on näistä yksi hyödyllinen ominaisuus. Käytännössä jokaisella SQL-palvelimella on – tai ainakin pitäisi olla – aikataulutettuja tehtäviä, joilla huolletaan tietokantaa. Erityisesti indeksien järjestely ja tilastojen päivittäminen ovat tärkeä osa tietokannan ylläpitoa. Lisäksi SQL Serverin omat varmistukset tietokannasta on toteutettu huoltosuunnitelmissa. Applications Manager valvoo niiden suorituksia, ja sen voi asettaa hälyttämään jos tehtävä jostain syystä epäonnistuu.

Sovelluksen Performance-välilehti sisältää lisäksi käteviä listauksia, joilla voidaan analysoida tietokantojen toimintaa SQL-kyselyiden tasolla. Esimerkiksi SQL Server Management Studion näkymä Activity Monitorista, jossa nähdään tiedot eri odotustyypeistä, on saatavilla. Yksi hyödyllinen listaus on SQL-palvelimen 'Cost of Missing Indexes' -listaus, jonka tietojen perusteella työn aikana erääseen paljon hauissa käytettyyn tauluun luotiin uusi indeksi nopeuttamaan hakuja taulusta. Kyseessä on kyseisen tietokannan muihin tauluihin suhteutettuna keskikokoinen taulu, johon kohdistui runsaasti hakuja, ja jotka SQL-palvelimen optimoija toteutti lähes jokaisella suorituskerralla skannaamalla taulun datan kokonaan. Indeksien luonnin jälkeen palvelin siirtyi pääosin etsimään taulusta dataa luodun indeksin avulla.

OBJECT NAME	INDEX NAME	USER_SEEKS	USER_SCANS
parameters	NULL	0	14
parameters	idx_export_parameters	0	0
parameters	idx_export_parameters	28	0
parameters	idx_protie_taskid	20306	0

Kuvio 31. Kuvassa on näkyvässä Applications Managerin listauksen perusteella parameters-tiluun luotu indeksi idx_protie_taskid ja tauluun kohdistuneiden kyselyiden määrä.

Applications Manager kerää SQL-palvelimelta eri kyselyillä löytyvät palvelimen toiminnasta kertovat oleelliset tiedot keskitetyksi yhteen, siistiin ja helposti ymmärrettävään muotoon. Hälytykset on N-centralin tapaan helppo kohdistaa yhdelle tai useammalle palvelimelle samalla kertaa. Kuten N-centralissa, palvelimet voidaan jaotella asiakasryhmiin, ja tarvittaessa asiakkaan tunnukselle voidaan rajata pääsy vain omien palvelintensa valvontadataan, mikä on Protien kaltaiselle IT-palveluntarjoajalle tärkeä ominaisuus.

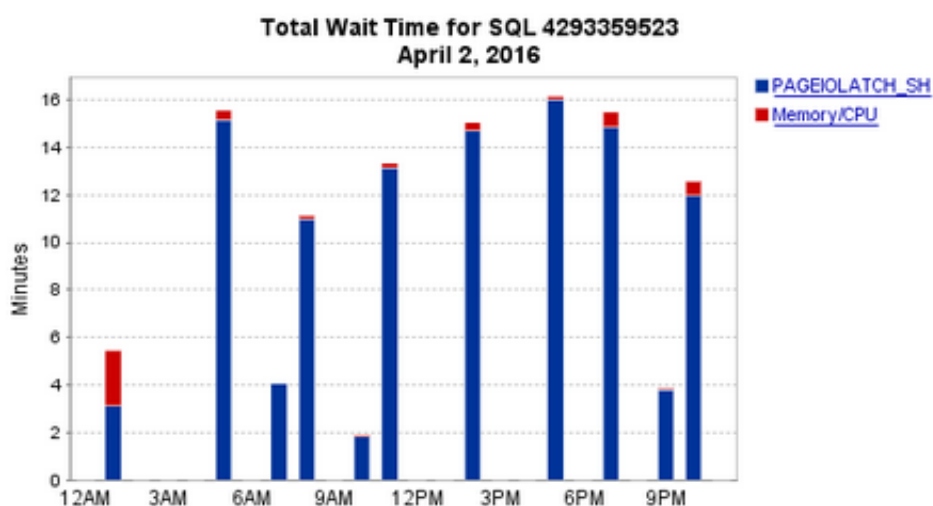
5.3 Database Performance Analyzer

Database Performance Analyzerin valvontaominaisuudet eroavat paljon Applications Managersta ja N-centralista, ja se on pelkästään tietokantojen valvontaa varten kehitetty sovellus. DPA sisältää paljon samoja resurssimittareita kuin kaksi muuta testattua sovellusta, mutta sen tärkein ominaisuus on tietokannan suorituskyvyn mittaaminen sen odotusaikoja tutkimalla.

DPA:n idea esittää kaikki tärkeimmät graafit pylväsdiagrammeina, jotka esittävät odotusaikojen muutoksia eri näkökulmista, on yksinkertainen ja tehokas. Datan tutkimalla voidaan loogisella päättelyllä löytää syitä tietokantojen suorituskykyongelmiin. Ongelmien selvittäminen DPA:n avulla vaatii käyttäjältä ymmärtämystä tietokantojen toiminnasta, mutta monessa tilanteessa se nopeuttaa selvitystyötä huomattavasti SQL Serverin omiin työkaluihin verrattuna. DPA tutkii palvelimen toimintaa suoraan kyselytasolla, ja sen valvontadatan perusteella voidaan usein rajata ongelman syy melko tarkasti esimerkiksi jonkin resurssin tarpeeseen, tiettyyn kyselyyn, käyttäjään tai puuttuvaan indeksiin.

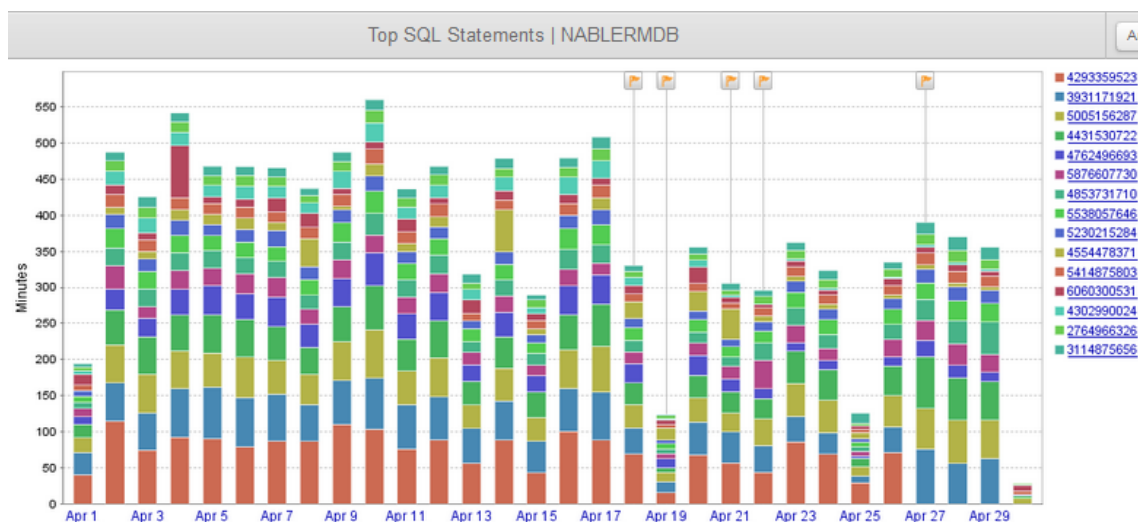
DPA:n valvontadatan avulla tehtiin testauksen aikana muutamia muutoksia tietokantapalvelimelle. Tutkimalla odotusaikoja eri ohjelmien kohdalla havaittiin, että SQL Server Management Studio aiheutti useina päivinä viikossa 20–30 % palvelimen kokonaisodotusajasta. Tämän jälkeen SSMS:n suorittamia kyselyitä tutkimalla nähtiin, että raskaimmat kyselyt liittyivät indeksien rakentamiseen ja tilastojen päivittämiseen. SSMS:n kolme kertaa viikossa suoritettavassa huoltosuunnitelmassa oli virheellisesti ajastettu indeksien uudelleenrakentamisen (rebuild index) jälkeen tilastojen päivitys (update statistics) koko kantaan. Tämä siis siitä huolimatta, että rebuild index -työssä päivitetään jo indeksien tilastot. Ratkaisuna asetettiin palvelin päivittämään update statistics työssä tilastot vain taulujen sarakkeisiin.

DPA:lla kyselyitä tutkimalla saatiin myös lyhennettyä koko palvelimen raskaimman päivittäisen kyselyn odotusaikaa murto-osaan aikaisemmasta. Kyseessä oleva kysely hakee useamman kerran vuorokaudessa yli 190 miljoonan rivin taulusta sen yhden sarakkeen pienimmän arvon. Kyseinen taulu sisältää tietoja tapahtumista, ja haetussa sarakkeessa on noiden tapahtumien tapahtuma-aikoja. Kysely siis hakee taulun vanhimman tapahtuman tapahtuma-ajan, käyttääkseen tuota tietoa tämän jälkeen eräässä tietokantaoperaatioissa kannan toisessa taulussa. Löytääkseen tuon sarakkeen pienimmän arvon joutuu palvelin tutkimaan taulun jokaista riviä. SQL-palvelin käsittelee dataa 8 kilotavun kokoisina sivuina, ja data on tietokannassa järjestetty riveittäin. Koska kyseinen taulu on yli 23 gigatavun kokoinen, sitä ei löydy välimuistista, vaan se joudutaan hakemaan aina levytä tietokannasta. Luetut sivut sisältävät paljon dataa myös taulun muista sarakkeista. Käytännössä siis suuri osa taulun datasta kopioidaan jokaisella suorituskeralla levytä palvelimen välimuistiin, jotta yhden sarakkeen pienin arvo saadaan selville.



Kuvio 32. Kuvassa nähdään palvelimen raskaimman kyselyn yhden päivän suorituskerrojen odotusaika.

DPA:n tietojen mukaan raskaimman kyselyn lähes koko odotusaika oli tyyppiä PAGEIOLATCH_SH, joka tarkoittaa sitä, että tehtävä odottaa sivujen lukua levytä välimuistiin. Jokainen kyselykerta kesti keskimäärin lähes 15 minuuttia, ja kyselyn odotusaika oli päivittäin 6-11 % koko tietokantapalvelimen odotusajasta. Selvittyjen tietojen pohjalta tauluun luotiin indeksi, joka sisälsi tapahtuma-aika-sarakkeen arvot järjestyksessä pienimmästä suurimpaan. Kyselyn suoritusajaksi lyheni tämän jälkeen olemattomiin.



Kuvio 33. Kuvasta nähtävissä ongelmallisen kyselyn päivittäinen odotusaika ennen ja jälkeen indeksinluonnin. Kyseessä oleva indeksi näkyy kuvassa oranssilla värillä (4293359523).

DPA tarjoaa käyttäjälleen yksityiskohtaista valvontadataa, jota olisi erittäin vaikea tai jopa mahdotonta hakea SQL-palvelimelta ilman DPA:n kaltaista työkalua. Se ei sisällä kuitenkaan odotusaikojen valvonnan lisäksi juurikaan mitään muita ominaisuuksia. DPA on joka tapauksessa erittäin hyödyllinen ja tehokas ohjelma suorituskykyongelmien selvittämiseen ja tietokannan nopeuden optimoimiseen.

5.4 Johtopäätökset

Testausten perusteella päätettiin hankkia Protien valvomon käyttöön Database Performance Analyzer tietokantojen suorituskykyongelmien tutkimista varten. Lisäksi N-centraliin tehdään SQL-palvelimille sääntö, joka tuo kaikki sen sisältämät SQL-mittarit palvelimilla valvontaan.

N-centralin SQL-valvonta voidaan ottaa käyttöön kaikilla palvelimilla nopeasti hyvin pienellä vaivalla ja ilman lisäkustannuksia. Tämän jälkeen kaikista SQL-palvelimista saadaan kerättyä niiden tärkeimmistä mittareista valvontadataa myöhempää tarkastelua varten. Tästä on varmasti apua tulevaisuudessa tietokantaongelmien selvityksessä, kun palvelimen suorituskykyä voidaan verrata aikaisempaan valvontadataan, jolloin palvelin toimi vielä normaalisti.

Database Performance Analyzer oli testausten perusteella ylivoimainen sovellus tietokantojen suorituskykyongelmien juurisyyn selvityksessä. Sen yksinkertainen toteutus sopii hyvin työkaluksi Protien tietokantojen valvontaan, koska DPA:n valvontadataa on helppo lukea ja ymmärtää. Sovelluksen avulla haastavampiakin tietokantaongelmia ratkaistakseen ei välttämättä tarvitse olla kokenut tietokanta-asiantuntija. DPA myös säästää aikaa nopeuttamalla vianrajausta, ja sitä voidaan käyttää lisäksi tietokantojen suorituskyvyn optimoinnissa.

Database Performance Analyzeristä tullaan ainakin aluksi hankkimaan pieni lisenssi, jolla valvotaan ja tutkitaan suurimpia sekä hitaimpia tietokantapalvelimia. Asiakkaan toiveesta voidaan lisätä palvelimia valvontaan tarpeen mukaan. Database Performance Analyzeriä tullaan käyttämään myös tietokantojen suorituskyvyn optimointiin niin, että palvelinta ei välttämättä tarvitse pysyvästi jättää sen valvontaan. VMWare-lisäosan lisenssin hankinnasta ei vielä ole tehty päätöstä, mutta on mahdollista että myös se tullaan hankkimaan.

ManageEnginen Applications Manager on SQL-tietokantojen valvontaan myös hyödyllinen sovellus ja sisältää useita käteviä ominaisuuksia, joita N-centralissa tai DPA:ssa ei ole. Kyseessä on kuitenkin sovellus, joka on suunniteltu palvelimien kokonaisvaltaiseen valvontaan, ja myös sen lisenssimaksut ovat tästä syystä suhteellisen korkealla. Applications Manager antoi kuitenkin hyvää perspektiiviä tutkimustyöhön. Se auttoi havaitsemaan kahdessa muussa sovelluksessa niiden puutteita ja vahvuuksia, vaikka sitä ei lopulta valittukaan Protien SQL-valvonnan työkaluksi.

6 Yhteenveto

Työn aikana tutustuttiin ensin SQL Serverin teoriaan, jonka jälkeen asennettiin testattavat valvontasovellukset ja aloitettiin varsinainen testaustyö. Valvonnan kohteeksi rajattiin tarkoituksella Microsoft SQL Server -palvelimet, jotta aiheeseen voitaisiin syventyä tarpeeksi perusteellisesti työn aikana. Protien ylläpitämistä tietokantapalvelimista valtaosa käyttää SQL Serveriä tietokannan hallintajärjestelmänään.

Työn jaksotus oli hyvin selkeä, jota korosti erityisesti testattavien uusien sovellusten rajalliset kokeiluajat. Kaikki testausvaiheen työ piti saada valmiiksi niiden aikana. Database

Performance Analyzerin normaali 14 päivän kokeiluaika saatiin sovittua sovellusvalmistajan kanssa 28 päivään ennen testauksen alkua. Näin työn testausvaihetta voitiin pidentää lähes kuukauteen.

Tavoitteena oli löytää uusia tehokkaampia tietokantavalvontatyökaluja Protien valvomon käyttöön, ja se saavutettiin. Uudeksi valvontasovellukseksi hankitaan Database Performance Analyzer. Sillä pystytään valvomaan tietokantojen toimintaa – erityisesti sitä miten nopeasti SQL-kyselyt palvelimella suoritetaan. DPA:n valvontadatan perusteella voidaan entistä nopeammin ja helpommin löytää tietokantojen toimintaa hidastavia tekijöitä tai ongelmallisia kyselyitä. Lisäksi Protien pääasiallisen palvelinten valvontajärjestelmän N-centralin SQL-valvontaominaisuudet otetaan jatkossa Protien valvomon käyttöön.

Database Performance Analyzerin käyttö Protien valvomossa tulee ajan myötä muokautumaan tarpeen mukaan. Aluksi tarkoituksena on valvoa DPA:n avulla suurimpia, tärkeimpiä sekä ongelmallisimpia tietokantainstansseja. DPA tukee SQL Serverin lisäksi myös muita järjestelmiä, ja näistä esimerkiksi Oracle- ja DB2-tietokantoja saatetaan tulevaisuudessa lisätä valvontaan. N-centralin valvontaominaisuudet otetaan sen sijaan käyttöön kaikille SQL-palvelimille.

Lähteet

- 1 Beal, Vengie. What is Database (DB)? Webodia Definition. Verkkodokumentti. <<http://www.webopedia.com/TERM/D/database.html>> Luettu 1.3.2016.
- 2 A-ajokorttikoulu – Tietokanta. Verkkodokumentti. <<https://a-ajokorttikoulu.wikispaces.com/Tietokanta>> Luettu 25.4.2016.
- 3 Database Models: Hierarcical, Network, Relational, Object-Oriented, Semistructured, Associative and Context. Verkkodokumentti. <<http://unixspace.com/context/databases.html>> Luettu 1.3.2016.
- 4 Laine, Harri. Tietomalli.pdf. Verkkodokumentti. <<https://www.cs.helsinki.fi/u/laine/tikape/k03/tietomalli.pdf>> 15.1.2003. Luettu 1.3.2016.
- 5 Tiedonhallinta, relaatiotietokanta. Verkkodokumentti. <http://edu.phkk.fi/opiskelu/thmate/th_osa3.htm> Luettu 1.3.2016.
- 6 Hiltunen, Maarit. Relaatiotietokannan perusteet. Verkkodokumentti. <http://www.okol.org/verkkokurssit/datanomi/tietojarjestelmien_kaytto_ja_kehittaminen/tietokantasuunnittelun_perusteet/relaatiotietokannan_perusteet/perusteet.htm> Luettu 2.3.2016.
- 7 Ekonoja, Antti; Lahtonen, Tommi; Mäntylä, Jukka. Relaatiotietokantojen peruskäsitteet. Verkkodokumentti. <<http://appro.mit.jyu.fi/doc/tiedonhallinta/tietokannat/index2.html>> 5.11.2004. Luettu 2.3.2016.
- 8 Watt, Adrienne. Database Design, Chapter 8, The Entity Relationship Data Model. Verkkodokumentti. <<https://opentextbc.ca/dbdesign01/chapter/chapter-8-entity-relationship-model/>> Luettu 2.3.2016.
- 9 Huotari, Jouni. SQL-kielen perusteet. Verkkodokumentti. <<http://student.labranet.jamk.fi/~huojo/opetus/IIZO3030/SQLopas.pdf>> Luettu 2.3.2016.
- 10 Laiho, Martti; Wendelius, Mika. SQL Transactions, Teoriaa ja käytännönharjoituksia. Verkkodokumentti. <http://myy.haaga-helia.fi/~dbms/dbtechnet/download/SQL-Transactions_handbook_FI.pdf> Luettu 3.3.2016.
- 11 Laiho, Martti. SQL-transaktiot. Verkkodokumentti. <http://myy.haaga-helia.fi/~ict03d/rdbms/mats/Transaktiot_Toipuminen.pdf> 22.11.2005. Luettu 3.3.2016.
- 12 Microsoft SQL Server. Verkkodokumentti. <https://en.wikipedia.org/wiki/Microsoft_SQL_Server> Luettu 3.3.2016.
- 13 What is SQL Server? Verkkodokumentti. <<http://searchsqlserver.techtarget.com/definition/SQL-Server>> Luettu 3.3.2016.
- 14 SQL Server Management Studio. Verkkodokumentti. <https://en.wikipedia.org/wiki/SQL_Server_Management_Studio> Luettu 14.3.2016.
- 15 Master Database. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/ms187837.aspx>> 4.3.2016. Luettu 14.3.2016.

- 16 Msdb Database. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/ms187112.aspx>> 4.3.2016. Luettu 14.3.2016.
- 17 Model Database. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/ms186388.aspx>> 4.3.2016. Luettu 14.3.2016.
- 18 Tempdb Database. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/ms190768.aspx>> 4.3.2016. Luettu 14.3.2016.
- 19 Resource Database. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/ms190940.aspx>> Luettu 14.3.2016.
- 20 Understanding Pages and Extents. Verkkodokumentti. <<https://technet.microsoft.com/en-us/library/ms190969%28v=sql.105%29.aspx>> Luettu 15.3.2016.
- 21 Sahgal, Varoon. How do database indexes work? Verkkodokumentti. <<http://www.programmerinterview.com/index.php/database-sql/what-is-an-index/>> Luettu 16.3.2016.
- 22 Saghal, Varoon. Clustered vs. Non Clustered Index. Verkkodokumentti. <<http://www.programmerinterview.com/index.php/database-sql/clustered-vs-non-clustered-index/>> Luettu 16.3.2016.
- 23 Santos, Donabel. Understanding SQL Server Statistics. Verkkodokumentti. <<http://blog.idera.com/sql-server/understanding-sql-server-statistics/>> 25.4.2011. Luettu 16.3.2016.
- 24 Uwujaren, Janice. What Is the Difference between Microsoft SSRS, SSIS and SSAS? Verkkodokumentti. <<http://smallbusiness.chron.com/difference-between-microsoft-ssrs-ssis-ssas-34689.html>> Luettu 16.3.2016.
- 25 SQL Server Integration Services. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/ms141026.aspx>> Luettu 16.3.2016.
- 26 Analysis Services. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/bb522607.aspx>> Luettu 16.3.2016.
- 27 Reporting Services. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/ms159106.aspx>> Luettu 16.3.2016.
- 28 Introducing the Business Intelligence Development Studio. Verkkodokumentti. <<http://searchsqlserver.techtarget.com/feature/Introducing-the-Business-Intelligence-Development-Studio>> Luettu 16.3.2016.
- 29 Wood, Dan. SQL Server Data Tools – Clearing up the Confusion. Verkkodokumentti. <<http://blog.nwcadence.com/sql-server-data-tools-clearing-up-the-confusion/>> 9.8.2013. Luettu 16.3.2016.
- 30 Microsoft Visual Studio. Verkkodokumentti. <https://en.wikipedia.org/wiki/Microsoft_Visual_Studio> Luettu 16.3.2016.
- 31 SQL Server Agent. Verkkodokumentti. <<https://technet.microsoft.com/en-us/library/ms189089%28v=sql.105%29.aspx>> Luettu 16.3.2016.

- 32 SQL Server Service Broker. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/bb522893.aspx>> Luettu 16.3.2016.
- 33 What is the full-text search engine?. Verkkodokumentti. <http://help.arcgis.com/EN/ARCGISSERVER/10.0/HELP/ARCGIS_SERVER_DOTNET_HELP/index.html#/0093000000wv000000> 18.11.2013. Luettu 16.3.2016.
- 34 Sqlcmd Utility. Verkkodokumentti. <<https://msdn.microsoft.com/en-us/library/ms162773.aspx>> Luettu 16.3.2016.
- 35 Kehayias, Jonathan; Stellato, Erin. SQL Server Performance Tuning Using Wait Statistics: A Beginner's Guide. Verkkodokumentti. <http://downloads.red-gate.com/simpletalk/whitepaper_wait_statistics.pdf> Luettu 8.4.2016.
- 36 Understanding how SQL Server executes a query. Verkkodokumentti. <<http://rusanu.com/2013/08/01/understanding-how-sql-server-executes-a-query/>> 1.8.2013. Luettu 8.4.2016.
- 37 How to analyse SQL Server performance. Verkkodokumentti. <<http://rusanu.com/2014/02/24/how-to-analyse-sql-server-performance/>> 24.2.2014. Luettu 8.4.2016.
- 38 Colley, Derek. Explanation of SQL Server IO and Latches. Verkkodokumentti. <<https://www.mssqltips.com/sqlservertip/3088/explanation-of-sql-server-io-and-latches/>> 30.10.2013. Luettu 8.4.2016.
- 39 Ozawa, Masayuki. Seas. Verkkodokumentti. <<http://www.slideshare.net/masayukiozawa/seas-11657724>> 19.2.2012. Luettu 8.4.2016.
- 40 Understanding Locking in SQL Server. Verkkodokumentti. <<https://technet.microsoft.com/en-us/library/aa213039%28v=sql.80%29.aspx>> Luettu 9.4.2016.
- 41 N-central, Technical requirements. Verkkodokumentti. <<http://www.n-able.com/products/n-central#scroll-reqs>> Luettu 25.3.2016.
- 42 Applications Manager, System Requirements. Verkkodokumentti. <https://www.manageengine.com/products/applications_manager/system-requirements.html> Luettu 25.3.2016.
- 43 Database Performance Analyzer for SQL Server, System Requirements. Verkkodokumentti. <<http://www.solarwinds.com/database-performance-analyzer-sql-server.aspx#scroll-systemRequirements>> Luettu 25.3.2016.