Performance: extra schijf toevoegen aan het systeem doet wonderen

Schaalbaarheid storage met SQL Server

Peter ter Braake

DAS, NAS en SAN. RAID 0, 1, 5 of een combinatie daarvan. Files en filegroups. Het is niet eenvoudig een strategie te ontwerpen om de data tier van een systeem goed in te richten. In dit artikel wordt een en ander op een rijtje gezet.

Ten eerste zijn er de verschillende opslagmogelijkheden. Heb ik genoeg aan Direct Attached Storage (DAS) of moet ik investeren in Network Attached Storage (NAS) of een Storage Area Network (SAN)? Daarnaast is fault tolerance, en dus RAID, belangrijk om over na te denken. Heb ik een '24 by 7' systeem? Welke downtime is acceptabel, zowel voor gepland onderhoud als in geval van calamiteiten? En om het nog complexer te maken spelen zowel de uiteindelijke performance als het gemak waarmee het systeem is te administreren een rol.

Hoewel NAS wel is genoemd als één van de storage-mogelijkheden, wordt verder niet te veel op deze optie ingegaan. Microsoft raadt af om NAS devices te gebruiken in combinatie met SQL Server. De mogelijkheid om SQL Server datafiles op een NAS device te plaatsen zijn per default zelfs uitgeschakeld. De voornaamste reden is dat er problemen gaan optreden met het recovery-proces. Dit proces draait tijdens het opstarten van SQL Server en zorgt ervoor dat we, na een onvoorziene shutdown, niet met inconsistente databases te maken krijgen. Met de tussenkomst van het netwerk om bij de data te komen is dat mechanisme niet meer gegarandeerd. Bovendien kan de tussenkomst van het 'normale' netwerk grote performance-problemen opleveren aangezien het DBMS een groot beslag legt op het IO-subsysteem. Als aan bepaalde randvoorwaarden is voldaan, is het gebruik van een NAS device echter wel mogelijk. Hierover is meer te vinden op http://support.microsoft.com/kb/q304261.

RAID

Uiteraard is voor een database de keuze van het juiste RAID-level van belang, ongeacht of u kiest voor een eigen disk-systeem voor SQL Server (DAS) of voor het inrichten van een SAN. Het gekozen RAID-niveau bepaalt grotendeels de fault-tolerance van het systeem en is dus van invloed op de beschikbaarheid van de database. Daarnaast wordt uiteraard de performance beïnvloed door de keuze van het RAID-level. Laten we op een rijtje zetten welk onderdeel van SQL Server waar en met welk RAID-level neergezet moet worden.

Om te beginnen kan voor elke database onderscheid gemaakt worden tussen de datafile(s) en de logfile(s). Het is heel sterk aan te raden om de data- en logfiles op fysiek aparte schijven te zetten, zowel om niet de data en de log kwijt te raken als een disk het begeeft als om disk contention te voorkomen. De logfile zal gezien de aard van het gebruik grotendeels sequentieel benaderd worden, waarbij de nadruk ligt op schrijfopdrachten. De combinatie van veel (en snel) schrijven en de belangrijkheid van de logfile voor de meeste recovery-strategieën, maakt dat voor de logfile op RAID-1 (mirroring) is aan te raden. RAID-10 zou een nog betere performance kunnen geven met dezelfde fault tolerance als RAID-1. Meestal is de performance van RAID-1 echter meer dan genoeg en wegen de extra kosten van een RAID-10 dus niet op tegen het performance voordeel.

> RAID-0 is goedkoop en levert een prima performance op vanwege de striping

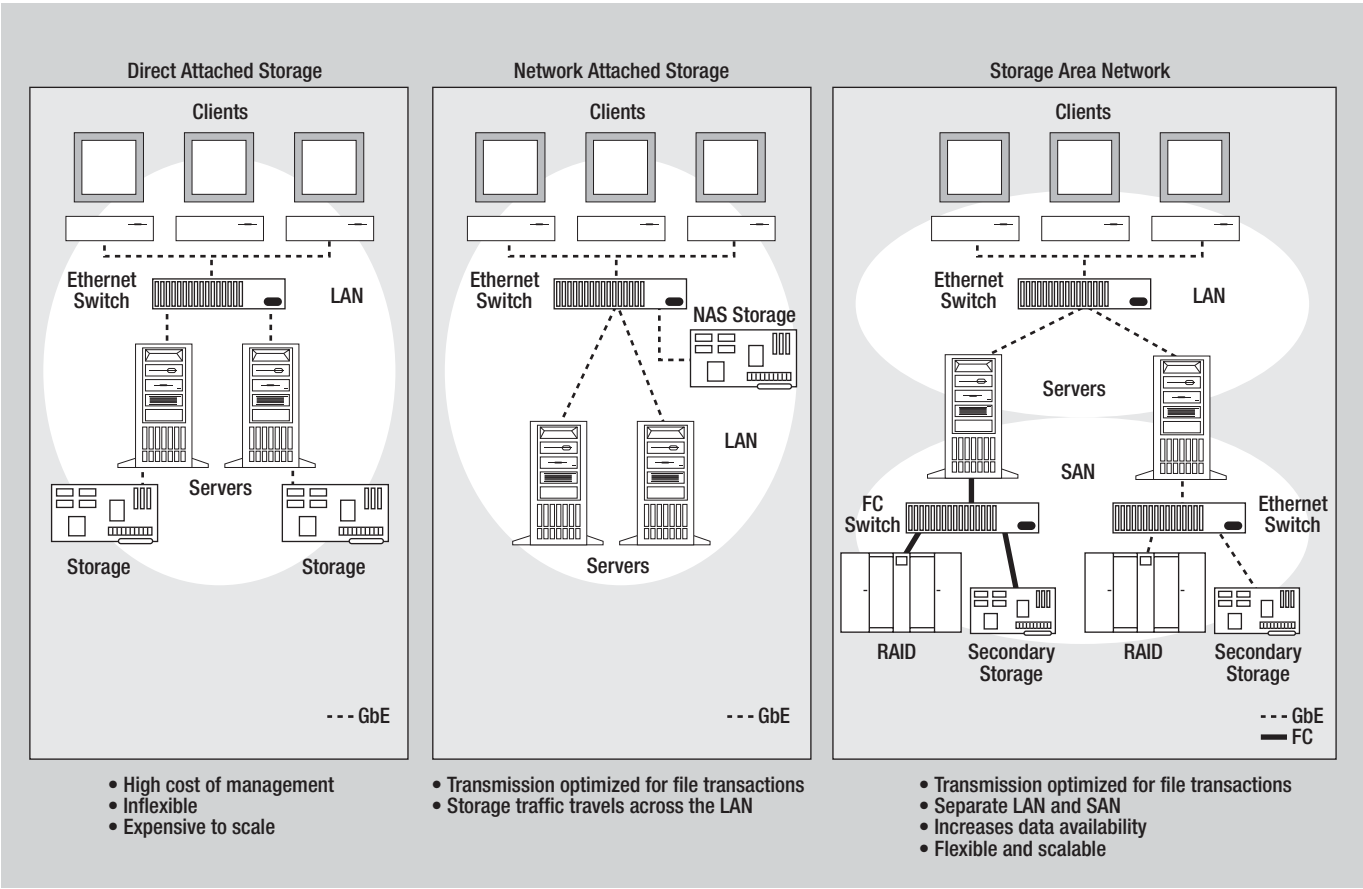
Het kiezen van de juiste RAID-configuratie voor de datafiles is lastiger. In een database waar zowel availability als recoverability niet van groot belang zijn, is RAID-0 een goede keuze: het is goedkoop en levert een prima performance op vanwege de *striping*. Gecombineerd met RAID-1 voor de logfile (en een bulk-logged of full recovery model) kunnen de data altijd vanuit

de log teruggehaald worden in het geval dat er een disk uitvalt. De hele database zal in dit geval echter wel offline zijn totdat de restore-operatie van alle benodigde backups compleet is.
De volgende optie is om de data op een RAID-5 (striping with parity) configuratie te zetten. De beschikbaarheid van de database wordt nu beter. Als een schijf uitvalt werkt de database gewoon door. De performance zal echter meteen dramatisch afnemen aangezien voor elke IO-operatie de parity-informatie gebruikt moet worden om de data te reconstrueren. Gezien de relatief slechte schrijf-performance van RAID-5 is deze configuratie alleen een optie als er weinig datamodificaties (transacties) zijn. Meestal ligt schrijven echter niet op het kritieke pad gezien de lazy-write capaciteiten van SQL Server.

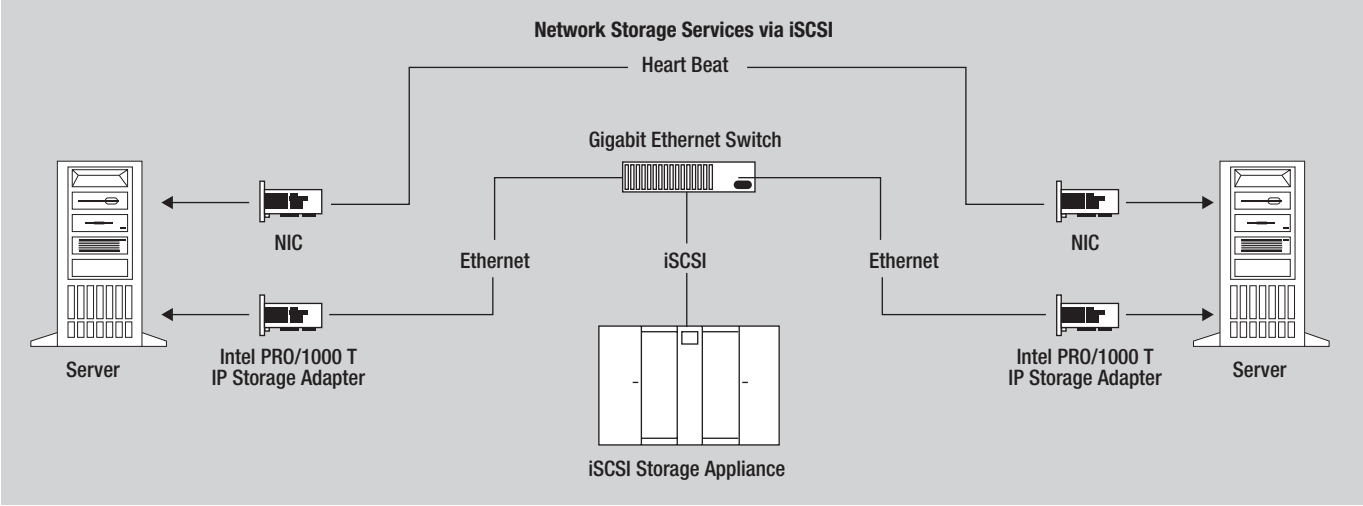
De beste en ook veruit de duurste optie, zowel wat betreft performance als wat betreft fault tolerance, is om de datafiles op een RAID-10 (mirroring and striping) systeem te zetten. De striping zorgt voor goede IO performance omdat de IO over meer fysieke disks verdeeld wordt. De mirroring zorgt voor goede fault tolerance. Heeft u een grote database die 24 uur per dag beschikbaar moet zijn en beschikt u over het benodigde budget, dan is dit de aangewezen weg. Meer informatie over RAID-configuraties is te vinden in 'Inside Microsoft SQL Server 2000' van Kalen Delaney.
Uiteraard speelt het soort systeem in de bovenstaande discussie ook een rol. Een OLTP-systeem met veel random IO en veel schrijfopdrachten stelt andere eisen aan het disk-subsysteem dan een DSS (decision support system), waar de nadruk op leesopdrachten zal liggen. Daarnaast zal een DSS-systeem vaker query's te verwerken krijgen die grote resultaatsets of grote tussenresultaten opleveren. Dit kan resulteren in het veel gebruiken van TempDB, zodat het voor de performance interessant wordt deze op een aparte fysieke schijf te zetten.

Hardware

Het spreekt voor zich dat een hardware-matige RAID-oplossing verre te verkiezen valt boven een software-matige oplossing. Evenzo is het duidelijk dat elk RAID-level een minimum aantal schijven vereist. Ook voor de capaciteitsplanning (hoe groot is de database en wat is de te verwachten groei?) is het aantal schijven dat men kiest van belang. Eén grote disk zal een slechtere performance opleveren dan tien kleine schijven die in totaal dezelfde opslagcapaciteit bieden.
Met RAID-oplossingen wordt al gebruikgemaakt van meer schijven. Door in het database-ontwerp slim gebruik te maken van files en filegroups kan eventueel nog wat extra winst behaald worden. SQL Server heeft de mogelijkheid meer datafiles per database te maken. Deze files worden specifiek op een (logische) drive gezet en kunnen gegroepeerd worden in filegroups. Van elk database-object, zoals met name tabellen en indexen, kan bij het



Afbeelding 1: Opslagconfiguraties.



Afbeelding 2: Configuratie van twee geclusterde servers met een gedeelde iSCSI SAN.

maken aangegeven worden in welke filegroup SQL Server dat object moet opslaan. Op deze manier zijn tabellen die vaak tegelijk worden geraadpleegd op verschillende disks te zetten, door ze in andere filegroups te plaatsen. Hiermee bereikt men een soort van load balancing tussen de verschillende schijven met als doel hotspots te voorkomen.

Naast de harde schijven zelf speelt de diskcontroller-kaart bij externe schijven natuurlijk een grote rol. De kaart is een potentiële bottleneck van elk systeem. Het is dus van belang een snelle kaart te kopen en het liefst één die meer channels ondersteunt. In een OLTP-systeem, waar de schijf relatief veel tijd besteedt aan zoeken, zal een channel niet snel overbelast raken, zodat een kaart gemakkelijk meer disks kan aansturen. In een DSS-omgeving met veel sequentiële leesopdrachten (table/index scans) is het aan te raden minder disks per channel te configureren. Overigens is in zo'n systeem extra intern geheugen misschien wel een even goede (of zelfs betere) manier om de IO performance te verbeteren.

SAN

Over het acronym SAN kan enig misverstand bestaan omdat het voor twee verschillende dingen wordt gebruikt. De acronym SAN wordt soms gebruikt als afkorting van System Area Network. In dat geval gaat het over een speciaal snel en extra betrouwbaar netwerk tussen servers of clusters van servers. Dit wordt gebruikt in plaats van een gewone LAN- of WAN-verbinding tussen de servers. Een SAN-verbinding levert grote voordelen op in multi-tier, gedistribueerde systemen die veel netwerkverkeer veroorzaken. Alleen SQL Server 2000 Enterprise Edition ondersteunt dit. Naast de benodigde hardware dienen slechts de juiste netwerkprotocollen ingesteld te worden (zie SQL Server books online voor meer informatie).
Naast het bovenstaande wordt de acronym SAN gebruikt voor Storage Area Network. Dit is een speciaal netwerk, dus los van het normale bedrijfsnetwerk, waaraan opslagmedia hangen. Dit kunnen RAID sets zijn met verschillende levels. De SAN biedt deze sets als logische volumes aan de servers van uw netwerk aan. Hierdoor is het erg eenvoudig om voor elke toepassing de optimale opslagwijze te bieden. De database server wordt dus simpelweg aan de SAN gekoppeld en ziet de geconfigureerde logische volumes. Afbeelding 1 toont naast andere opslagconfiguraties, ook een typische SAN-configuratie. Bij het gemak om verschillende RAID levels naast elkaar aan te bieden, is het ook mogelijk om een SAN dubbel uit te voeren (gemirrored) om op die manier het ultieme in fault tolerance te krijgen (kijk voor meer informatie bijvoorbeeld op http://www.findarticles.com/p/articles/mi_m0BRZ/is_10_22/ai_98977109).

Het grote voordeel van het gebruik van een SAN ten opzichte van storage die rechtstreeks aan de server hangt, is de schaalbaarheid. Een SAN kan over het algemeen een veel grotere opslagcapaciteit aan dan Direct Attached Storage. Is de capaciteit niet meer toereikend, dan is toevoegen van extra schijfruimte geen enkel probleem. Los van deze schaalbaarheid gaat een SAN ook efficiënter om met de beschikbare hoeveelheid opslagcapaciteit dan wanneer we meer servers hebben met elk hun eigen storage device. Immers, één server kan gebrek hebben aan schijfruimte, terwijl een andere server nog veel ruimte over heeft. Toch is er een capaciteitsprobleem dat opgelost moet worden.
Naast de beschikbare capaciteit is ook het centrale beheer van die grote hoeveelheid opslagruimte een voordeel van SAN's boven direct gekoppelde schijfcapaciteit. Wat af en toe over het hoofd gezien wordt, is dat een database de SAN wel moet delen met andere applicaties die gebruikmaken van de gedeelde opslag. Uiteindelijk moet de database de beschikbare bandbreedte dus delen.

Backup-strategie

Een ander vaak genoemd voordeel van een SAN is dat het de mogelijkheid biedt een backup device zoals een tapestreamer in de SAN-configuratie op te nemen. In de SAN kan men dan een

backup-strategie configureren die vervolgens op de achtergrond draait, zonder dat het gewone netwerk daardoor extra belast wordt. Het is natuurlijk van belang om backups vanuit SQL Server te blijven maken. Een SQL Server backup garandeert immers dat op correcte wijze wordt omgegaan met lopende transacties, waarmee database consistency wordt gegarandeerd. Deze SQL Server backups kunnen echter gewoon naar een schijf in de SAN geschreven worden. De SAN backup schrijft ze vervolgens weg naar het uiteindelijke backup-medium.
Gezien alle bovenstaande voordelen van een SAN ten opzichte van DAS ben je bijna geneigd te zeggen dat iedereen met een SAN-systeem werkt. De waarheid is echter dat SAN's lange tijd alleen interessant zijn geweest voor grote systemen waarbij geld geen rol speelt. SAN's zijn traditioneel uitgerust met een Fibre Channel-netwerk. De bijbehorende infrastructuur, hardware en software is erg duur. Ook hebben de meeste bedrijven niet de technische kennis in huis die nodig is om een optisch netwerk te onderhouden en dus is scholing nodig.

> SQL Server heeft de mogelijkheid om meer datafiles per database te maken

Met de opkomst van iSCSI (Internet Small Computer System Interface) komt SAN echter ook binnen het bereik van kleinere bedrijven en systemen. iSCSI is een protocol dat SCSI commando's verpakt in TCP/IP pakketjes waarmee block IO over IP-netwerken mogelijk wordt. Er is dus geen dure optische hardware of specialistische kennis nog om een SAN gebaseerd op iSCSI (IP SAN) operationeel te krijgen. Op performance vlak kan een IP-netwerk natuurlijk niet op tegen een optisch netwerk. Maar met 1Gigabit IP-netwerken is de performance voor de meeste systemen die in het midden- en kleinbedrijf draaien, inclusief afdelings-databases binnen grotere organisaties, meer dan genoeg. Er zijn zelfs tests van hardware-fabrikanten bekend die aantonen dat de performance van hun IP SAN-oplossing gelijkwaardig is aan optische SAN-oplossingen. De optische oplossingen zijn wel een factor twee tot vier duurder dan de fibre optic-alternatieven.
Microsoft ondersteunt iSCSI binnen de Windows 2000 en Windows Server 2003 operating systemen. Dat wil zeggen dat een 'gewone' ethernet-kaart afdoende is en dat software-matige iSCSI initiators de vertaling van SCSI commando's naar TCP/IP pakketjes voor zijn rekening neemt. Uiteraard kost dit enig intern geheugen en belangrijker CPU cycles. Dit kan oplopen tot 10 procent aan extra gebruikte CPU cycles, die dus niet ten goede komen aan SQL Server.
Het alternatief is de aanschaf van een speciale iSCSI Host Bus Adapter (HBA), dit is een iets duurdere oplossing. Een HBA neemt echter het verpakken van SCSI commando's over van het OS, waarmee dus performance-winst gehaald wordt. Op het moment dat de server veel IO moet doen, wat in het geval van SQL Server natuurlijk zo is, is het gebruik van een iSCSI HBA dus aan te raden.

Failover clustering

Alle moeite en investeringen in een SAN zijn over het algemeen niet de moeite waard als de server zelf vervolgens een single point of failure wordt. We zien SAN's dan ook bijna altijd in combinatie met clustering. Afbeelding 2 schetst een typische configuratie van twee geclusterde servers met een gedeelde iSCSI SAN. SQL Server maakt gebruik van de clustering-mogelijkheden die het operating systeem biedt. Zowel Windows 2000 als Windows Server 2003 ondersteunen failover clustering via Microsoft Clustering Services (MSCS). Beide servers uit afbeelding 2 houden via een heartbeat van elkaar bij of ze nog operationeel zijn. Eén van de servers handelt alle verzoeken van clients af en fungeert dus als de actieve SQL Server. Op het moment dat deze server uitvalt, neemt de andere server het automatisch over. Omdat beide servers het storage-systeem delen, is er op het gebied van dataconsistentie geen enkel probleem.
Het moge duidelijk zijn dat het hart van elk database-syteem de opslag van data is. Dus wordt de waarde van een database-systeem in grote mate bepaald door de capaciteit, de betrouwbaarheid en de snelheid van het opslagsubsysteem. De te volgen strategie en de aan te schaffen hardware zijn in hoge mate afhankelijk van de eisen die aan het systeem gesteld worden en het budget dat beschikbaar is.
Het managen van alles dat met dataopslag te maken heeft, is dan ook een belangrijke taak van de DBA. Vooraf moet goed in kaart gebracht worden wat de eisen zijn die aan een systeem gesteld worden: hoeveel opslagcapaciteit is nodig; hoeveel geplande en ongeplande downtime is acceptabel; en welke performance moet gehaald worden. Voor het laatste moet ideaal gezien een baseline neergezet worden waartegen men het systeem met regelmaat kan controleren.

Conclusies

Veel performance-problemen kan men door het kiezen van de juiste hardware oplossen. Als de problemen echter veroorzaakt worden door een slechte architectuur van het gehele systeem (van de datalaag tot en met presentatielaag) zal investeren in hardware niet meer dan een lapmiddel zijn. Tunen tegen slechte query's is onbegonnen werk. Voordat men echter aan de slag gaat met files en filegroups om performance-problemen aan te passen, is het echter wel aan te raden de hardware eens onder de loep te nemen. Wellicht dat een extra schijf toevoegen aan het systeem al wonderen doet.

Peter ter Braake (pbraake@computrain.nl) is productspecialist SQL Server bij Compu'Train.