

Carel-Jan Engel over ervaringen rond het Oracle-rdbms

Divide et Impera

Lang voordat er ook maar enig benul van bits en bytes bestond, wist Julius Caesar zijn vijanden te onderwerpen volgens het principe 'Verdeel en Heers'. Daarbij werd de vijand moegestreden door interne, meestal etnische, conflicten aan te wakkeren. De verzwakte tegenstander kon daarna makkelijk worden overwonnen. One-liners als die van Caesar spreken mensen nog steeds aan, ook in ons vak. Er is zelfs een project voor verwerking van recursieve programma's naar hem vernoemd: JuliusC, een interpreter die C-programma's analyseert om tot efficiëntere code te komen. Dat het mechanisme 'Verdeel en Heers' nog steeds werkt zie ik bijna dagelijks in mijn advieswerk. Meestal vervullen de leveranciers de rol van Julius Caesar en nemen de gebruikers de rol van te overwinnen vijand op zich. Nog wranger wordt het daarbij te constateren dat de gebruikersorganisatie op meesterlijke wijze zelfstandig in staat blijkt de interne verdeling tot stand te brengen, waarna de leverancier alleen nog maar orders hoeft te scoren. Let wel, orders met een beperkt of geen nut, maar wel voor aanzienlijke bedragen.

Enkele weken geleden was ik in Londen bij een grote organisatie om Oracle Data Guard te komen uitleggen, installeren en configureren. Op het oog lijkt het een droomproject. De organisatie heeft voorzien in drie computer-ruimtes; de centrale ruimte op het hoofdkantoor, een tweede ruimte in een andere vleugel van hetzelfde gebouw en tenslotte een uitwijkcentrum op zo'n 40 kilometer afstand. Er lijkt voldoende bewustzijn en budget aanwezig voor een goede oplossing. De verbinding tussen beide vleugels heeft een capaciteit van 2 Gigabit, naar de uitwijklocatie ligt een 40 Megabit-verbinding. De eerste twee systemen zullen via Data Guard volledig synchronoos worden bijgehouden. Om te kijken of dat haalbaar is hebben we een korte benchmark opgezet, die 200 Megabyte aan redo-informatie genereert. In de praktijk levert het drukste moment van de dag ongeveer 750 MB op in vijf uur, de benchmark representeert dus iets meer dan 25 procent van de dagelijkse load. De servers blijken de benchmark in één minuut en 13 seconden te kunnen verwerken, terwijl er door de synchrone Data Guard verwerking gegarandeerd wordt dat geen gegevensverlies kan optreden. Wordt deze garantie losgelaten door de verbinding asynchrone te maken, dan kan de benchmark in 1 minuut en 8 seconden worden verwerkt. Dit is al met al voldoende vertrouwenwekkend om geen performance-verlies te verwachten tijdens het alledaagse gebruik.

Verder was ook aandacht besteed aan de belangrijkste foutoorzaak in IT-systemen: de menselijke factor. Iemand die per ongeluk een tabel weggooit, wordt niet geholpen met replicatie. De tabel zal in de tweede database immers net zo snel worden weggegooid. Door het wegschrijven van de wijzigingen op de standby database synchronoos te laten verlopen, maar het verwerken van de wijzigingen met enkele uren vertraging uit te voeren, wordt een buffer ingebouwd. Als een tabel wordt weggegooid, kan de standby database worden bijgewerkt tot vlak voor het fatale moment, en dan worden geopend. De weggegooidde tabel kan zo worden teruggehaald. Tenslotte was er nog het uitwijkcentrum. De asynchrone Data Guard-replicatie levert het risico op van enkele seconden dataverlies, maar dat was acceptabel. De risico's waren dus perfect verdeeld over de verschillende computer-ruimten en konden dan ook prima worden beheerst. Dat is alleen wat anders dan wat ik bedoelde met de opening van deze column. Die sloeg op de organisatie zelf. Aan het begin van mijn bezoek vroeg ik wat nu precies de eisen van het management waren met betrekking tot beschikbaarheid en hersteltijd. Daaruit bleek dat door het management nog geen eisen waren vastgelegd. Ik begrijp dan ook niet dat er zoveel budget kan worden vrijgemaakt om zoveel dure installaties te kopen, terwijl er niet kan worden gemeten of er aan de eisen wordt voldaan. Niet omdat er onvoldoende wordt gemeten, maar omdat er geen eisen zijn! Erger werd het nog toen bleek dat de afdeling die verantwoordelijk was voor ontwikkeling en beheer van de intra/internet-applicatie, waarvan het hele bedrijf afhankelijk was, zijn eigen weg ging. De databases waren perfect beschermd, alleen kon de applicatie de andere computer-ruimtes niet bereiken! Uitwijk was dus nutteloos. Dat zie ik helaas veel te vaak: gescheiden opererende afdelingen voor systeemontwikkeling, database-beheer, netwerk, gegevensopslag, server-beheer enzovoort. Voor de chefs van die afdelingen heeft het één voordeel: als er iets mis gaat, kan je altijd naar iemand anders wijzen. Kandidaat-schuldigen zijn er genoeg. Voor de leveranciers is het een walhalla, er kunnen talloze elkaar overbodig makende oplossingen worden verkocht. Divide et Impera!

Carel-Jan Engel (cjpengel.dbalert@xs4all.nl) is onafhankelijk Oracle-specialist op het gebied van database-ontwerp, trouble-shooting en tuning.