

Oracle masterclass leerzame ervaring

# Experimenteren, de helicopter in en gezond verstand gebruiken

Carel-Jan Engel

**G**eregeld organiseert Oracle een zogenaamde DBA-week. Bekende goeroes en klinkende namen komen de deelnemende dba's in relatief korte indringende sessies ('specials') een aantal ins en outs vertellen die niet iedereen in zijn of haar dagelijkse bagage heeft zitten. Carel-Jan Engel schoof aan in een masterclass, onderdeel van een van de meest recente DBA-weeken.

Louter toevallig, omdat een eerder geplande datum voor ons bezoek aan een DBA-week samenviel met een bezoek van Larry Ellison aan Nederland, komen we in deze masterclass terecht, georganiseerd door Precise.

Jonathan Lewis en Anjo Kolk, twee zeer ervaren Oracle-specialisten, leiden deze bijeenkomst. Lewis -ruim zeventien jaar vertrouwd met Oracle-producten- werkt sinds twaalf jaar als onafhankelijk adviseur op het gebied van systeemontwerp en trouble-shooting. Kolk kent Oracle vrijwel even lang; hij

## *Trial-and-error is aan Lewis niet besteed*

werkte zestien jaar bij het bedrijf, van 1985 tot eind 2001. Bij Oracle in het vak gegroeid, heeft hij in zijn laatste jaren daar vooral gewerkt aan tuning en trouble-shooting. De eerste twee dagen van de masterclass worden door Jonathan Lewis verzorgd, Anjo Kolk staat op de derde dag 'voor de klas'. Beide heren



FOTO: DRE DE MAN

**ANJO KOLK (LINKS): "COMMON SENSE IS NOT COMMON PRACTICE". RECHTS CAREL-JAN ENGEL.**

weten een frisse kijk op problemen in de databasewereld op een heel plezierige manier over te brengen.

### STENEN TIJDPERK

Veel dba's die (performance)problemen ondervinden met een database waarvoor ze verantwoordelijk zijn, volgen een vaste maar niet noodzakelijk succesvolle aanpak om een oplossing te vinden. Een aantal vaste ratio's wordt gcontroleerd - staat de BufferCache Hit Ratio boven de 95 %? (of, 99%, of 99,999%, afhankelijk van de goeroe die je aanhangt).

Pas een paar bekende tips toe, en controleer of de getallen verbeteren. In feite is dit een aanpak uit het stenen tijdperk. Slechts zelden is vooraf bekend wat het effect van een gekozen maatregel is. Bij voorkeur worden meerdere maatregelen tegelijk uitgevoerd, want de database mag niet al te vaak worden gestopt en gestart. Welke maatregel verantwoordelijk is voor welk deel van het effect blijft dan ook volmaakt verborgen.

Deze methode van trial-and-error is aan Lewis niet besteed. Tijdens de verschillende sessies weet hij voortdurend te imponeren -in de goede zin van het woord- met de onvoorstelbare hoeveelheid kennis die hij van de

interne processen in de Oracle-databasekernel heeft weten te vergaren. Zijn aanpak is dan ook puur wetenschappelijk. Wanneer een nieuwe versie van de Oracle-database verschijnt, leest hij de manuals van A tot Z door. Daarbij neemt hij de inhoud niet voor waar aan, maar test hij stelselmatig de verschillende beschreven gedragingen. Dat blijkt onder meer uit de vele voorbeelden die hij geeft in zijn lessen, waarbij triviaal lijkende verschillen tussen verschillende Oracle-versies soms grote gevolgen blijken te hebben.

## LABORATORIUM

Lewis investeert verhoudingsgewijs veel tijd in wetenschappelijk laboratoriumonderzoek naar de Oracle-database. Aan de andere kant verliest hij de praktijk niet uit het oog door geregeld in het veld advieswerk uit te voeren en wereldwijd tijdens masterclasses als deze dba's te ontmoeten. Hierdoor beschikt hij over een onuitputtelijk reservoir met voorbeelden en anekdotes. Die gebruikt hij ter illustratie van de kennis die hij over zijn toehoorders uitstort. Bij het cursusmateriaal wordt ook een diskette verstrekt met daarop scripts van veel van de behandelde voorbeelden. Die kan de cursist gebruiken om zelf zijn laboratoriumomgeving te creëren waarin het geleerde in praktijk kan worden gebracht.

Lewis weet veel over interne processen en werkwijzen van de Oracle-kernel en weet deze kennis bovendien goed over te brengen. Daarmee geeft hij inzicht in de gevolgen van keuzen die zijn te maken bij het ontwerp van een systeem. In de eerste sessie worden de kosten van parser en optimizer behandeld. Aan de hand van een aantal voorbeelden wordt duidelijk gemaakt hoe met behulp van `sql_trace`, informatie over het verwerken van een simpel SQL-statement kan worden verzameld. Het voorbeeld dat Lewis gebruikt toont aan dat één SQL-statement al leidt tot zeventien interne SQL-statements die de kernel moet uitvoeren om het uit te voeren! Parsing van SQL-statements kost relatief veel tijd. Dit speelt met name bij simpele SQL-statements die worden gebruikt voor bijvoorbeeld integriteitscontroles en het maken van drop-down lists. Nu applicaties in steeds meer lagen worden gesplitst en SQL daarbij vaak dynamisch wordt gegenereerd, neemt dit probleem alleen

maar toe. Het gegenereerde SQL wordt vaak in een string aangeboden, waarbij de verschillende selectiecriteria 'hard gecodeerd' in deze string worden opgenomen. Het gevolg hiervan is dat geen twee SQL-statements meer gelijk zijn, en ieder statement altijd opnieuw moet worden vertaald. Om dit probleem het hoofd te bieden, is in Oracle 8.1 de optie `cursor_sharing` toegevoegd. De standaard waarde hiervan is 'exact'. Als de optie `cursor_sharing = force` wordt opgenomen in de initialisatieparameters zal de SQL-parser constanten in where-clauses vervangen door variabelen. Daardoor valt efficiënter met de resources om

*In de praktijk gaan aandacht en inspanning maar al te vaak uit naar de verkeerde zaken. Verlaat de platgetreden paden*

te gaan doordat hergebruik van SQL-statements vaker mogelijk zal zijn.

Een volledige samenvatting te geven van alles wat Lewis zoal behandelt, valt buiten het bestek van dit artikel. De hoeveelheid informatie die hij in twee dagen weet over te dragen, is super-geconcentreerd. Wie dat allemaal wil bevatten, moet wel uitgebreid zelf aan het experimenteren slaan. Op dezelfde diepgaande wijze gaat Lewis in op redo, latches, locks & waits, de verschillende manieren van query-optimalisatie en analyse van SQL-statements met Explain Plan. Steeds worden de stellingen daarbij bewezen met verschillende voorbeelden.

## PERFORMANCE-ERVARINGEN

Op de derde dag gaat Anjo Kolk in op performance tuning. Dat kan ook bijna niet anders, Precise, zijn huidige werkgever, bestaat van de levering van tools voor het meten van performance. Kolk deelt zijn ervaringen onder de noemer: *common sense is not common practice*. Ook hij kan zijn ervaringen illustreren met talloze praktijkvoorbeelden. De wereld van de Oracle-dba's wil hij graag bewust maken van de criteria die er echt toe doen. Een dba is vaak geneigd een reeks van ratio's te verza-

melen en vervolgens tevreden te constateren dat de performance goed moet zijn. Daar gaat het alleen niet om. Het gaat om de performance die de gebruiker ervaart. Als een systeem te langzaam is, heeft de dba het vaak gedaan. Waarom? Omdat hij nu eenmaal degene is die min of meer dagelijks in de buurt van het systeem verkeert. Dat betekent tegelijkertijd dat de andere componenten vaak over het hoofd worden gezien.

In een van Kolks praktijksituaties was sprake van een stuk PL/SQL dat enkele duizenden malen werd uitgevoerd. Nu kan de dba dat nog zo perfect tunen, als in de praktijk blijkt dat slechts eenmaal uitvoeren van de betreffende procedure voldoende is voor de functionaliteit, kan hij het nooit goed doen.

De moraal van het verhaal is dat performance tuning begint bij de vraag of uitgevoerde statements überhaupt wel nodig zijn. Kijk met een helicopterview naar het hele systeem, en meet over het totale systeem waaraan de responstijd wordt besteed. Als de database 20% van de responstijd consumeert, kan daar nu eenmaal nooit een verbetering van 50% worden gerealiseerd. Dat is precies wat de titel van Kolks presentatie zegt: in de praktijk worden de aandacht en inspanning maar al te vaak op de verkeerde zaken gericht. Geef ruimte aan gezond verstand en verlaat de platgetreden paden: begin tuning bij het ontwerp; en als er dan moet worden bijgeschaafd, neem dan het totale systeem in ogenschouw.

Het zijn drie leerzame dagen geweest, die ik iedere Oracle-dba van harte kan aanbevelen. ●

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