



Oracle 10g: High Availability

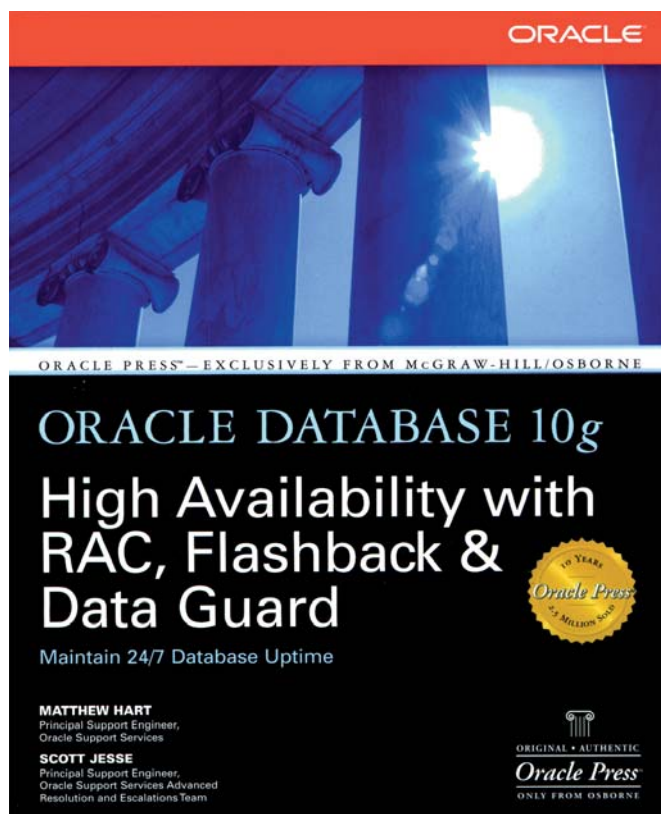
Matthew Hart en Scott Jesse

Deze keer bespreek ik een boek uit de reeks van Oracle Press. Het boek behandelt mijn favoriete onderwerp: High Availability.

Over de auteurs

Beide auteurs werken als Principal Support Engineer bij Oracle Support. Hoe je het wendt of keert, die werkgeversrelatie heeft

invloed op de benadering in het boek. Dat begint al in de inleiding, waar al vroeg wordt uitgelegd dat benadering van *High Availability* van databases het beste kan worden gerealiseerd met de databasesoftware zelf. Dat kost natuurlijk (Oracle-) licenties, maar per saldo is dat volgens de auteurs toch het meest efficiënt. Met een boek als dit nemen ze een forse uitdaging aan: het zijn nogal wat onderwerpen om samen in één boek te behandelen.



Boek: Oracle 10g: High Availability with RAC, Flashback and Data Guard, 421 pagina's
 Schrijvers: Matthew Hart en Scott Jesse
 Uitgever: Osborne McGraw Hill / Oracle Press
 ISBN: 0-07-225428-9

Het boek

Om de uitdaging van de veelheid aan onderwerpen aan te pakken kozen de auteurs een origineel begin. Hoofdstuk 1 beschrijft een fictieve handel in houtschroeven, die vervolgens een reeks aan onverwachte situaties meemaakt, met systeemuitval als gevolg natuurlijk. Ieder van die situaties leidt vervolgens tot een mogelijke oplossing, die de beschikbaarheid verbetert. Dat begint met een reeks nieuwe features van 10g, op het gebied van resource-beheer, tuning en RAC (Grid). De benadering is wat luchtig, alle problemen zijn met Oracle natuurlijk zonder meer op te lossen, tot en met een logisch probleem waarvoor een paar tabellen een paar uur in tijd moeten worden teruggedraaid met gebruikmaking van de flashback optie. Het komt voor dat dit kan, maar dat is maar hoogst zelden. In de praktijk is er vrijwel altijd sprake van afhankelijkheid van de tabellen in een schema, en één of een paar tabellen kunnen zelden worden teruggezet zonder de gegevensintegriteit aan te tasten.

Hoofdstuk 2 behandelt een reeks 'standaard' database-features, variërend van Enterprise Manager via partitioning tot transportable tablespaces. Deze features maken het uitvoeren van taken voor de DBA vaak een stuk gemakkelijker, en ze zorgen er soms inderdaad voor dat deze taken kunnen worden uitgevoerd zonder dat de database *down* hoeft te worden gebracht, maar voor een *High Availability* boek vind ik het wat ver gezocht. Het boek lijkt niet geheel vreemd van invloed van Oracle Marketing. Dat gevoel wordt mede gevoed door het terugkomende voorbeeld waarbij wordt gemigreerd van Solaris

naar Linux: was dat niet de grote hype van Oracle ten tijde dat het boek werd geschreven?

Hoofdstuk 3 loopt door een reeks tuning-hulpmiddelen van Oracle 10g heen. In het verlengde van hoofdstuk 2: een langzame query geeft de gebruiker misschien het idee dat het systeem niet beschikbaar is, maar het is er een beetje met de haren bijgesleept.

In hoofdstuk 4 begint het echte werk. Er wordt een stappenplan doorgenomen voor het inrichten van RAC op een paar tweedehands Linux machines met een Firewire schijf er tussen. Ik heb het stappenplan niet zelf doorlopen. Dat heeft meer met een gebrek aan tijd dan gebrek aan interesse te maken. Ik focus zelf op Data Guard, en heb daarin nog zoveel zaken niet ontdekt dat ik me nog geen tijd gun om RAC te verkennen. Als ik dat nu zou gaan doen zou ik overigens kiezen voor het gebruik van de 'kant-en-klare' VMWare uitprobeerversies die Oracle op OTN ter beschikking stelt. Hoofdstuk 5 gaat vervolgens over het beheer van de cluster, zoals het toevoegen en verwijderen van nodes. Aan het eind nog twee pagina's over grid-control. Hoofdstuk 6 borduurt op dit beheer voort met de behandeling van het service-concept.

Hoofdstuk 7 begint met Data Guard. Als eerste worden de voordelen van *Physical* en *Logical standby* databases aangegeven. En ja hoor, ook hier wordt aangegeven dat *Logical standby* databases het mogelijk maken een zogenaamde *rolling upgrade* uit te voeren. Een *rolling upgrade* is het upgraden van de database terwijl de applicatie door kan blijven draaien. Dat is met de oplossing van Oracle niet mogelijk. De claim is dan ook onterecht, zoals Ken Jacobs ook toegaf in het interview dat ik eerder met hem voor Optimize had. Ook hier dringt het verhaal van marketing door in het boek.

De stappenplannen voor het maken van de standby databases (*Physical* en *Logical*) zien er redelijk uit. Ze zijn niet echt volledig, de lezer zal wat door het boek moeten bladeren om alle informatie compleet te krijgen, maar dat is hooguit onhandig. De veelheid aan mogelijke opties voor het transport van de redo log informatie wordt erg summier behandeld.

Waar RAC een aantal hoofdstukken kreeg toebedeeld: setup, management en toepassing, wordt Data Guard in één hoofdstuk behandeld. Het beheren van de Data Guard komt dan ook nog uitgebreid aan de orde, inclusief het omschakelen naar de standby database. Het geheel wordt in eerste instantie command-line georiënteerd beschreven, pas aan het einde wordt ook de Data Guard GUI nog beschreven. Mijns inziens een

terechte keuze: *High Availability* vereist dat je weet wat je doet, wat er achter de schermen gebeurt. Een GUI verbergt in dat geval gewoon te veel voor de beheerder.

In de resterende hoofdstukken worden RMAN, Flashback, Streams en Oracle Net nog behandeld. Flashback heeft naar mijn mening een zeer beperkte betekenis op het gebied van *High Availability*, net als Streams. RMAN heeft zijn plaats, dat spreekt voor zich, maar het streven naar compleetheid veroorzaakt daardoor mede oppervlakkigheid. Oracle Net verdient meer aandacht dan het in het boek krijgt.

Doelgroep

Het boek is bestemd voor de DBA met ervaring, die een omgeving met verhoogde beschikbaarheid wil gaan opzetten. Het feit dat beide auteurs bij Oracle werkzaam zijn verklaart dat RAC in dit boek voorkomt. Oracle zet RAC nadrukkelijk in de markt als een oplossing voor *High Availability*. Ik ben zelf een uitgesproken RAC-scepticus, als het gaat om hoge beschikbaarheid. De praktijkverhalen over problemen met RAC-installaties, zowel tijdens het inrichten als tijdens het gebruik, zijn mijns inziens te talrijk om RAC voor hoge beschikbaarheid te zetten. In de begintijd, van RAC met 9i, was dat meer het geval dan nu, maar ook nu komen de *horror stories* nog te vaak voorbij. Ik sluit niet uit dat RAC op termijn ook voor *High Availability* van Oracle databases gemeengoed wordt, maar het is daarvoor nu nog te vroeg.

Conclusie

Hart en Jesse zetten met dit boek een aardig verhaal over *High Availability* neer, waarin een aantal belangrijke aspecten wordt doorgenomen. Het is naar mijn mening vooral een introductie in het onderwerp, zonder dat echt diep op de materie wordt ingegaan. De flaptekst claimt een volledige handleiding met praktijkoefeningen, maar dat wordt niet waargemaakt. De hoofdstukken over 'standaard features' van de Oracle software hadden achterwege kunnen blijven, ten gunste van de zaken die er in *High Availability* planning meer toe doen. Ik vind dat het boek te veel uitgaat van de technologie, de oplossingen. Om *High Availability* aan te pakken is het noodzakelijk te vertrekken bij de eisen van de gebruikersorganisatie. Deze raakt in een technologie-gedreven aanpak nogal eens gemengd tussen goede bedoelingen, die veel kosten en een leuke uitdaging vormen, maar net niet doen wat er van mag worden verwacht.

Carel-Jan Engel werkt als onafhankelijk Oracle-consultant. Hij is lid van het Oak Table Network. E-mail: cjpengel.dbalert@xs4all.nl