

Klaas Brant over IBM's DB2 Universal Database

De Ultieme Waarheid

Ik neem u deze keer mee op reis naar het Griekenland van zo'n 4 eeuwen BC. Om precies te zijn naar Delphi, waar het legendarische Orakel sprak. De oude Grieken waren ervan overtuigd dat dat de ontegenzeggelijke waarheid was. Nu, zo'n 25 eeuwen later, hebben we last van een ander orakel, namelijk Oracle. Hun opperhoofd heeft ons al diverse malen de toekomst als waarheid voorspeld. Dat zit de Oracle-mannen blijkbaar in de genen, want zelfs onze eigen Carel-Jan kan u De Waarheid vertellen. Over The Grid, deze keer. Knap voor iemand die koffiedik in de tuin gooit om de katten eruit te houden. Het probleem dat ik met Oracle-voorspellingen heb, is dat ze nogal Oracle-minded zijn. Zoiets van: "if the only tool you have is a hammer, every problem looks like a nail". Oracle 10g kan alleen op kleine, lokale schaal werken, dus dan zal grid-computing wel lokaal zijn. We zullen zien, wellicht gaat grid-computing wel helemaal nergens heen en moet Larry zijn gummetje tevoorschijn halen om de "g" van zijn Oracle 10g af te halen. Het zou niet de eerste keer zijn dat de Oracle de oplossing heeft voor een probleem dat niet bestaat...

Wat is dan wél de toekomst? Ik weet het niet, maar ik gooi dan ook mijn koffiefilter altijd netjes in de groenbak. We moeten volgens mij de toekomst in elk geval niet zoeken in moderne fratsen, maar in oplossingen voor hedendaagse problemen. Weet u wat de best bezochte trainingen zijn? Die over performance. Soms heeft een database mooie features maar moet je er iets voor doen om ze te activeren. Neem nu bijvoorbeeld DB2's Materialized Query Tables (MQT). Een prachtige manier om veel resources te besparen. Het idee is simpel: zorg dat je het resultaat van complexe (b.v. aggregatie) bewerkingen opslaat. Komt later dezelfde bewerking in SQL nog eens voorbij dan kun je slim gebruik maken van het reeds bekende resultaat. Natuurlijk zullen nieuwe INSERT, UPDATES en DELETES het resultaat ongeldig maken en moeten we op een slimme manier het opgeslagen resultaat aanpassen. Hiervoor zijn twee mogelijkheden: onmiddellijk of uitgesteld. En als het onmiddellijk moet dan kan het wellicht ook op een slimme manier. Denk maar eens aan het opslaan van AVG(salary) GROUP BY Department. Een update van een enkel salaris zal niet het gemiddelde van alle afdelingen in de war gooien. Dus is een slimme aanpak gewenst. Een ander probleem is het herkennen van de opgeslagen bewerking in de SQL. Ik heb u reeds in het verleden ooit verteld over query re-write. Ook voor MQT is het



herschrijven van SQL belangrijk. Soms kan het voorkomen dat een complexe bewerking met behulp van boolean algebra is om te zetten zodat de MQT weer gebruikt kan worden. DB2's optimizer zal proberen dit te doen.

Verder is natuurlijk de exploitatie van nieuwe technologie en hardware iets wat een database sneller kan maken, maar waarvoor de gebruikers niet te veel willen betalen. De truc die veel software leveranciers uithalen is het hercompileren onder de nieuwe infrastructuur. Als dat werkt, dan ondersteunt men de nieuwe infrastructuur. Onzin natuurlijk! DB2 Versie 8 for OS/390 wordt op dit moment grotendeels herschreven om de 64 bits adressering van de nieuwe z-serie machines te ondersteunen. Een hels karwei, want aan de buitenkant lijkt het of er niks veranderd terwijl er aan de binnenkant erg veel aangepast moet worden. Natuurlijk kun je een zoekalgoritme in

de databasebuffers hetzelfde houden, maar ze worden soms erg inefficiënt als de omvang van de buffers in de gigabytes begint te lopen. Soms is er voor een dergelijke aanpassing research nodig. Neem nu bijvoorbeeld de huidige limiet van 225 tabellen in een join. Wat gebeurt er met de SQL optimalisatie als je dat naar 1000 zou oprekken? Komt er niet ergens een punt dat de algoritmes dol draaien of gewoon walgelijk veel cpu-tijd nodig hebben? En mocht u nu denken "Wie heeft er nu meer dan 225 tabellen in een

join nodig?" dan is het antwoord: u! Pakketten als bijvoorbeeld SAP kunnen met hun SQL-generator werkelijk immense SQL's genereren. Zo was er in DB2 een limiet van 32 MB voor de text van het SQL-statement. Meer dan genoeg? Niet echt... hij staat nu op 64 MB en gaat binnenkort zelfs fors omhoog omdat ook voor sommige applicaties 64 MB niet voldoende is. Heeft u wel eens 64 MB ingetypt? SQL generators zijn werkelijk meester in het generen van onzin.

Wellicht komen bij Carel-Jan de applicaties binnenkort uit het stopcontact, maar ik blijf toch nog even met twee benen op de grond. En mocht Carel-Jan op zoek zijn naar de oplossing voor de katten in zijn tuin, dan kan ik hem aanraden zelf een kat te nemen. Bij mij komen namelijk nooit vreemde katten in de tuin...

Klaas Brant (kbrant@kbce.nl) is DB2-specialist en directeur van KBCE b.v. Meer informatie over DB2 is te vinden op www.kbce.nl en www.db2-times.com