

SQL misschien het best haalbare in een feilbare wereld

# Strikt relationele database is een nobel doel

Peter Verkooijen

**SQL zondigt tegen de regels van het relationele model. Moeten we ons neerleggen bij de dominantie van SQL of zijn alternatieven denkbaar? Alphora's Dataphor is het enige systeem dat de zegen van relationeel opperpriester Chris Date heeft gekregen.**

Dataphor ontstond uit puur pragmatische redenen. "We zijn oorspronkelijk met database-applicaties begonnen", zegt Bryn Rhodes, hoofd ontwikkelaar van Alphora. "We liepen steeds tegen dezelfde problemen aan waarvoor we steeds dezelfde code moesten schrijven. De beste manier om die problemen generiek op te lossen was het relationele model zo dicht mogelijk te benaderen." Alphora mengt zich met Dataphor in een oude, maar hardnekkige discussie. SQL houdt zich niet strikt aan het relationele model. "Er is een aantal basisprincipes, bouwstenen zou je kunnen zeggen, waarvoor in de SQL-wereld een andere invalshoek is gekozen dan in het relationele model", zegt analist Rick van der Lans. "Dat was al vanaf het begin. De bedenkers van SQL zijn daar al van afgeweken."

## Standaardiseren

De theoretische basis voor de relationele database is in 1970 gelegd door Ted Codd, programmeur bij IBM's San Jose Research Lab. Codd's relationele model was een manier om opslag en toegang tot data te standaardiseren op logische principes in plaats van fysieke locatie. Data zouden worden vastgelegd in tabellen met kolommen en rijen. Tussen velden in verschillende tabellen zouden logische relaties kunnen worden gelegd. Applicaties hoefden niet te weten hoe en waar de data precies waren opgeslagen om ermee te kunnen werken.

Oracle bracht in 1977 een eerste relationele database op de markt. IBM volgde in 1981 met DB2 en een Structured Query Language (SQL) als interface tussen applicatie en database. Sindsdien heeft de industrie geprobeerd SQL te standaardiseren. Codd's discipelen Chris Date, Hugh Darwen en Fabian Pascal zijn altijd luidruchtig ontevreden met die pogingen geweest. Date en Darwen hebben in 1998 in hun boek *The Third Manifesto* uiteengezet waarom SQL niet deugt en hoe het anders kan. Twee punten waarop SQL het relationele model aan zijn laars lapt zijn de dubbele rijen en NULL-waarden. "In de SQL wereld is

vanaf dag één besloten dubbele rijen toe te staan", zegt Rick van der Lans. "Dat is volgens mij de eerste grote discussie geweest. Later is daar de discussie over de NULL-waarden bijgekomen. Maar daar zijn de grondleggers van het relationele model het ook nooit over eens geweest."

De NULL-waarde introduceert een variant van drie waarden logica in het relationele model. "Dat is het fundamentele probleem", volgens Bryn Rhodes. "NULL is toegevoegd als vlag voor ontbrekende data, maar verstoort het logische systeem. Je kunt NULL niet relateren met NULL. Problemen in SQL gaan verder dan alleen die NULL-waarden. Wij proberen met Dataphor zoveel mogelijk tekortkomingen te repareren als we kunnen." Dataphor is het enige database-product waar Chris Date zich achter kan scharen. Dataphor of D4 is gebaseerd op Tutorial D uit *The Third Manifesto* en kan dingen die Oracle, IBM en Microsoft niet kunnen in hun databases.

## Regels

Een van Codd's Twaalf Regels waar alle grote SQL databases tegen zondigen is de View Updating Rule. Data uit verschillende tabellen moeten in verschillende combinaties kunnen worden gepresenteerd. Een view hoort dezelfde mogelijkheden voor data-manipulatie te bieden als een tabel. Bij de grote SQL databases is dat niet het geval. Dataphor houdt zich wel aan de regel.

**Je kunt je strikt aan het relationele model houden en nog steeds problemen hebben**

"Het hele punt van het relationele model is dat data onafhankelijk zijn van de locatie", zegt Rhodes. "Als je in de view niet met de data kunt werken ben je ernstig beperkt in je mogelijkheden. Als je met SQL in de view het updaten van de dataset mogelijk wilt maken moet je daarvoor code schrijven in de applicatie. Je moet als ontwikkelaar steeds onderscheid maken tussen een view en een tabel, terwijl dat niet zou moeten uitmaken." Een belangrijk probleem van SQL databases betreft volgens Rhodes de beperkingen in Database Level Constraints. "In SQL

heeft de ontwikkelaar weinig mogelijkheden om regels te bepalen waar de database zich aan moet houden", zegt Rhodes. "Zodra er meer dan één tabel bij is betrokken zijn regels niet af te dwingen, zonder dat je een hoop code moet schrijven. Je bent gedwongen die logica in de applicatiecode te schrijven. Er zijn limieten die de traditionele SQL-systemen wel aankunnen, maar er zijn maar weinig SQL-systemen die multitabellimieten afdwingen. Dataphor staat ook transitielimieten toe. Je kunt een limiet toepassen op niet alleen hoe data eruit zien, maar ook op hoe ze mogen veranderen. Dat kan declaratief worden geschreven in een expressie."

## Alphora heeft geen oplossing voor de NULL gevonden

Was het een fatale vergissing dat SQL is afgeweken van het relationele model? "Nee hoor", zegt Rick van der Lans. "Dat is heel bewust gedaan. Ik heb wel met die jongens gesproken. Ze zaten voor een serieuze uitdaging. Ze moesten voor het eerst op deze planeet een database server bouwen volgens die relationele principes. Als je kijkt naar wat er toen beschikbaar was aan database servers was dat een enorme stap. Omwille van prestatie hebben ze toen besloten toch water bij de wijn te doen.

Ik denk dat we dat allemaal gedaan zouden hebben." Maar we zijn nu 25 jaar verder. Met de huidige technologie moet SQL toch zijn te repareren? "Alles is te herstellen", stelt Van de Lans, "maar wat gaat dat allemaal niet kosten? Het zou een gigantische klus zijn om al die DB2 en Oracle SQL database servers te vervangen. Waar moeten we dan databases mee bouwen? Wat is het alternatief?"

Ook Dataphor maakt het nobele doel van een strikt relationele database niet waar. De uitgebreide mogelijkheden voor databaselimieten in Dataphor hebben maar deels met het relationele model te maken. "Het is ook een kwestie van ontwerp", zegt Rhodes. "Je kunt je strikt aan het relationele model houden en nog steeds problemen hebben. Het relationele model biedt een stevig fundament, maar veel problemen met SQL zijn taalkundig." Alphora heeft geen oplossing voor de NULL gevonden. "Er zijn goede argumenten om ze helemaal niet te hebben en het puur relationeel te houden", zegt Rhodes. "Maar je loopt tegen allerlei problemen aan als je ze niet hebt. We probeerden NULL-waarden te vermijden, maar hebben ze toch weer moeten introduceren. De mensen die claimen dat NULL een absolute showstopper is hebben tot dusver ook nog geen alternatief voorgesteld." Het relationeel model is in de praktijk blijkbaar niet zaligmakend. "Het is perfect als datamodel", houdt Rhodes vol. "Je kunt altijd om NULL's heen, maar niet als de applicaties die er bovenop worden ontwikkeld die ontbrekende waarden toestaan."

## Chris date: “NULL is een vreselijk ding”.

Chris Date was een van de grondleggers van het relationele model. De laatste decennia heeft hij strijd moeten voeren tegen de ketterij van SQL. De bezwaren tegen SQL lijken academisch, nauwelijks praktisch. “Die typering deugt helemaal niet”, reageert Date verontwaardigd. “Mijn mantra is al jaren dat theorie praktisch is. Het doel van de relationele theorie is dat we praktische systemen kunnen bouwen.”

*Tot welke dringende praktische problemen leidt SQL's afwijking van het relationele model?*

“Een van de problemen is dat de taal veel complexer is dan ze zou moeten zijn. Doordat SQL de theorie niet volgt is de taal moeilijker te leren, moeilijker te onderwijzen, moeilijker te documenteren, moeilijker te implementeren.”

*Waarom heeft de markt toch nu toe dan zo weinig belangstelling voor zuivere relationele alternatieven?*

“Niet-technische argumenten, zoals commerciële of politieke argumenten, geven vaak de doorslag bij beslissingen die technisch van aard zijn. Voor de eerste DB2 database verscheen hebben mensen bij IBM, waaronder ik, erop aangedrongen niet de code van het prototype te gebruiken. Het was een verschrikkelijke taal. We hadden iets veel beters nodig. Maar de mensen die de argumenten begrepen waren niet de mensen die de beslissing namen. De beslissingen werden genomen op de hoogste managementniveaus.”

*Is Alphora's Dataphor een alternatief?*

“Het is in ieder geval beter dan SQL. In de eerste release van hun product hadden die jongens heel hard geprobeerd exact te implementeren wat Hugh Darwen en ik hadden voorgeschreven in The Third Manifesto. Ze hadden geen NULL's omdat ze begrepen dat NULL's vreselijke dingen zijn. Maar ze wilden van Dataphor een federatief systeem maken dat bovenop DB2, Oracle en de rest kon draaien. Zonder NULL ging de prestaties enorm naar beneden omdat ze hun systeem moesten mappen op de SQL databases. Dus in de tweede editie hebben ze treurig genoeg de NULL toegevoegd.”

*Is een puur relationeel systeem zonder NULL dan ooit mogelijk?*

“Ja. Ik help nu iemand die een radicaal nieuwe implementatietechnologie voor het relationele model heeft waarmee we razendsnelle systemen zouden kunnen bouwen. De technologie heet het transrelationele model. De uitvinder is Steve Tarin. Maar het is erg moeilijk marktaandeel af te snoepen van Microsoft, IBM en Oracle.”

*Wanneer kunnen we de eerste transrelationele producten verwachten?*

“Daar mag ik geen uitspraken over doen. Yes, watch this space, but don't hold your breath. Het is een implementatietechnologie waarop je een werkelijk relationeel systeem met een werkelijk relationele taal kunt bouwen. Maar je kunt er donder op zeggen dat het eerste transrelationele product vanwege marktoverwegingen nog steeds een SQL interface zal hebben.”

### Het best haalbare

Rick van der Lans ziet geen directe concurrent voor een alternatief voor SQL: “We kunnen er nu al zo'n 25 jaar mee leven. We hebben er duizenden databases en tienduizenden applicaties mee gebouwd. Wat is dan het probleem? Dataphor zou interessant kunnen zijn als het een enorme prestatieverbetering zou leveren ten opzichte van bestaande systemen, maar zodra je bestaande applicaties wilt ondersteunen krijg je toch weer alle SQL-bezwaren.”

Bryn Rhodes hoopt dat SQL aan zijn eigen tegenstrijdigheden ten onder zal gaan: “SQL is een verbijsterend slechte taal. Het enige consistente aan SQL is dat het niet consistent is. Standaard SQL bestaat niet. Het aantal dialecten is ontelbaar en er komen alleen maar meer bij.” Een Noors bedrijf zou hebben geprobeerd een SQL DBMS te bouwen volgens de specificaties van de ANSI SQL standaard van 1992. “Ze ontdekten dat het onmogelijk was. De standaard zelf zit vol tegenstrijdigheden”, zegt Rhodes. Is het hele concept van een relationele database misschien aan vervanging toe? Hebben XML en Xquery de toekomst zoals sommigen beweren? “XML en XQuery zijn een enorme stap terug naar het hiërarchische model van dertig jaar geleden”, zegt Rhodes. “Waarden zijn niet altijd hetzelfde. Het relationele model

was juist ontworpen om die problemen te elimineren. Je kunt XML als een mark-up taal gebruiken. Daarvoor kan ik het nog accepteren. Maar om het uit te breiden als een algemeen datamodel is een stap terug. En de query-taal is een gruwel. XML verdwijnt, het is een rage die vanzelf zal wegebben.”

Het kan dus altijd nog erger. SQL is misschien het best haalbare in een feilbare wereld. “Het is een belangrijk onderwerp als je door de bril van Chris Date en Fabian Pascal kijkt”, zegt Rick van der Lans, “maar de rest van de wereld staat er niet echt bij stil. Soms is het wel fijn om in de IT een aantal zaken te hebben die stabiel en algemeen geaccepteerd zijn, ook al zijn ze misschien niet perfect. In die categorie zitten TCP/IP, SQL en straks waarschijnlijk ook XML. We kunnen daar dan natuurlijk over discussiëren dat we beter iets anders hadden kunnen bedenken, maar zo is het nu gewoon. We moeten leren daar nu mee om te gaan.”

**Peter Verkooijen** is freelance journalist.