

De twee vermeende grote blunders

'The Third Manifesto' kritisch bekeken (1)

Maurice Gittens

In een drietal artikelen zal Maurice Gittens een kritische blik werpen op het boek 'Foundation for future database systems; The Third Manifesto' van Hugh Darwen en Chris Date. In dit artikel identificeert hij een aantal logische inconsistenties.

Volgens Hugh Darwen en Chris Date, de auteurs van het boek 'Foundation for future database systems; The Third Manifesto' (verder TM genoemd), staat de grondregel *Alle logische verschillen zijn grote verschillen* en zijn uitvloeisel *Alle logische fouten zijn grote fouten* centraal in hun werk aan dit boek.

Dit artikel identificeert een aantal logische inconsistenties in de dissertatie van Darwen en Date door uit te gaan van stellingen als *Logisch geldige gevolgtrekkingen mogen slechts uitgaan van vooronderstellingen die en logisch geldig zijn en ook relevant zijn*. Waar het pertinent is wordt E.F. Codd aangehaald om de argumentatie kracht bij te zetten.

De twee grote blunders

De TM wijst twee zogenoemde 'grote blunders' aan in de huidige stand van de integratie van object-georiënteerde en relationele systemen. Deze zo vernoemde *grote blunders* zijn:

- de equivalentie tussen object-klassen en relatie-variabelen (relvars);
- het toekennen van een zogenoemde Object Identifier aan tuples in relatie-variabelen.

Volgens de TM bestaat er geen algemeen aanvaarde definitie van een object-georiënteerd systeem

De primaire stelling die ik in dit artikel poneer is dat de door Date en Darwen gepresenteerde substantiatie voor de bewering dat bovenstaande keuzes grote blunders zijn, formeel gezien, niet valide is. In deze is mijn claim niet dat Date en Darwen onjuist

zijn in hun conclusies met betrekking tot de vermeende 'grote blunders'. Nee, mijn bewering in deze beperkt zich tot de claim dat hun stellingname met betrekking tot de 'grote blunders' niet op logisch geldige wijze is afgeleid.

Mijn secundaire stelling is dat de vermeende grote blunders niet met recht als zodanig zijn vernoemd. De secundaire stelling wordt mede onderbouwd door middel van aanhalingen uit 'Extending the Database Relational Model To Capture More Meaning' van E.F. Codd.

De eerste blunder

Blunder 1 betreft de equivalentie tussen Relatie-variabelen en Object-klassen. Volgens Date en Darwen is het gelijkstellen van relatie-variabelen en object-klassen een blunder. Deze sectie gaat in op de onderbouwing of misschien eerder het gebrek aan onderbouwing van deze bewering in de TM maken. Gegeven de context van logische geldigheid waarin Date en Darwen de uiteenzetting in de TM plaatsen wordt in dit verband de volgende vraag gesteld: Op basis van welke in logische zin geldige argumenten concluderen Date en Darwen dat het gelijkstellen van relatie-variabelen en object-klassen een blunder is?

De bewering dat het gelijkstellen van relatie-variabelen en object-klassen een blunder is wordt in de TM het eerst gemaakt op pagina 15 van het boek. De uiteenzetting in dat hoofdstuk wordt door Date en Darwen *informeel* genoemd.¹ Date en Darwen stellen niet alleen dat het onjuist is om relatie-variabelen gelijk te stellen aan object-klassen, maar tegelijkertijd stellen ze dat het, in plaats daarvan, juist is om domeinen gelijk te stellen aan object-klassen. De reden die de auteurs voor deze stellingname geven is dat het en voor domeinen² en voor object-klassen feitelijk is dat hun respectievelijke waarden door een verzameling bestemde operatoren gemanipuleerd worden.

Op mijn beurt stel ik dan de vraag: *Geldt niet voor relatie-variabelen dat hun waarden door bestemde operatoren gemanipuleerd worden?* Wat is dan bijzonder aan de operatoren van domeinen dat er voor zorgt dat deze wezenlijk anders zijn dan de operatoren van andere relationele abstracties?

Stel ik definieer, conform de vrijheden geboden door de TM definitie van domeinen, een domein dat precies de verzameling operatoren kent die de relationele algebra voor relatie-waarden definieert. Wat is dan het wezenlijke en dus ook het logische

verschil tussen zulk een domein-waarde en een relatie-waarde? Beide worden per slot van rekening door *precies* dezelfde verzameling operatoren gemanipuleerd.

Door de, volgens Date en Darwen, informele aard van de uiteenzetting in hoofdstuk 2 van de TM ben ik op zoek gegaan naar de formele onderbouwing voor de stelling van Date en Darwen, dat de equivalentie tussen relatie-variabelen en object-klassen een blunder is. Afgezien van herhalingen van in essentie het bovenstaande argument, blijkt de TM niet een formele onderbouwing voor de bovenstaande claim te hebben gegeven. Ook een uitvoerige e-mail-discussie over dit onderwerp met Hugh Darwen heeft deze onderbouwing niet aan het licht gebracht. Het blijft tevens toch ook opmerkelijk te noemen dat er volgens de TM maar een mogelijke manier bestaat om object-georiënteerde en relationele systemen te integreren. Waarom zouden er niet meerdere manieren, ieder met zijn eigen voor- en nadelen, kunnen bestaan? Als laatste vermeld ik dat er volgens de TM geen algemeen aanvaarde definitie van een object-georiënteerd systeem bestaat. Het gebrek aan zulk een definitie weerhoudt Date en Darwen desondanks niet van het maken van vrij absolute en ongenueanceerde uitspraken over object-georiënteerde systemen en hun relatie tot relationele systemen. Kan op basis van de genoemde door de TM gepresenteerde argumenten met recht worden gesteld dat het gelijkstellen van relatie-variabelen en object-klassen een grote blunder is?

Orthogonaal

Zijn de eigenschappen van domeinen orthogonaal ten opzichte van de ondersteuning het relationele model?

Volgens de TM zijn domeinen wat betreft interne structuur en ondersteunde operatoren niet beperkt³. Dit volgt uit de principiële TM benadering: "The question as to what datatypes are supported is orthogonal to the question of support of the relational model". Deze stellingname staat haaks op de invalshoek van E.F. Codd. Volgens Codd zijn simpele domeinen, zonder interne structuur (vanuit het perspectief van de DBMS), voorwaardelijk voor het relationele model. "A relational database is a time-varying collection of data, all of which can be accessed and updated as if they were organized as a collection of time-varying tabular (nonhierarchical) relations of assorted degrees defined on a given set of *simple* domains".

De ondersteuning van complexe domeinen noodzaakt de datamodelleur te kiezen tussen de inzet van complexe domeinen en relaties. De vraag wanneer de datamodelleur dient te kiezen voor de inzet van een relatie dan wel een complex domein, is volgens pagina 381 van de TM "still somewhat open". Anders gezegd: de TM stelt er nog niet uit te zijn onder welke omstandigheden de datamodelleur een object-klasse (dus een TM-domein) dient in te zetten en wanneer de datamodelleur een relatie dient in te zetten in zijn datamodellen omdat beide opties mogelijk zijn. In het licht van de vermeende eerste 'grote blunder' en de stellingname van Codd is dit feit opmerkelijk te noemen.

Blunder 2: Het gebruik van Object Identifiers in relationele tuples

Het gebruik van Object Identifiers bij de integratie van relationele en object-georiënteerde systemen benoemt de TM als de tweede grote blunder. De redenen die in de TM voor deze stellingname worden gepresenteerd zijn:

- Het zogenoemde *information principle* van Codd; dat vrij vertaald stelt dat alle informatie in een relationele database in termen van waarden (values) geformuleerd dient te kunnen worden. Interactie tussen verschillende delen van relationele databases zouden slechts op basis van de vergelijking van waarden plaats moeten vinden;
- Een andere reden dat de TM geeft voor het verwijderen van pointers, is het feit dat volgens Codd het gebruik van pointers foutgevoeliger en minder intuïtief is dan het vergelijken van waarden;⁴
- Op pagina 417 in de paragraaf met als titel "OBJECT IDS UNDERMINE INHERITANCE" wordt gesteld dat Object Identifiers niet goed aansluiten bij het overervingsconcept dat doorgaans door object-georiënteerde systemen wordt ondersteund.

Ondersteuning van complexe domeinen noodzaakt de datamodelleur te kiezen

Gezien de context van logisch correct redeneren waarin Date en Darwen de TM hebben geplaatst, stel ik met betrekking tot de eerste twee argumenten de volgende vraag: wat is logisch gezien de relevantie van deze argumenten? Het feit dat Codd pointers verwerpt omdat ze *moeilijk* en/of *foutgevoelig* zouden zijn, is niet relevant als substantiatie voor de verwerping van object identifiers in relationele systemen. Hiermee wil ik natuurlijk niet zeggen dat Codd met deze stelling geen gelijk zou kunnen hebben.

Op verschillende plaatsen in de TM blijkt dat de TM pointers en Object Identifiers als synoniem gebruikt. Dit is een abstractie-fout, niet in het minst omdat identiteit een onderscheidend kenmerk is van ieder element van een wiskundige verzameling. Op basis van de identiteit van elementen van wiskundige verzamelingen is een primaire operatie als *het tellen* (het enumereren als u dat liever hebt) mogelijk. Zonder identiteit zou men niet in staat zijn om twee elementen van een wiskundige verzameling te onderscheiden. De vraag of twee elementen van een wiskundige verzameling wel of niet aan elkaar gelijk zijn, staat namelijk geheel los van de vraag of twee elementen van wiskundige verzamelingen wel of niet onderscheidbaar zijn.

Het argument dat Object Identifiers inheritance zouden ondermijnen is eveneens een misslag. De gepresenteerde argumentatie toont hoogstens aan dat Object Identifiers ondermijnend zijn voor het inheritance-model van de TM. Het is niet aangetoond dat de vermeende problemen ook voor andere inheritance-modellen bestaan. Wanneer we tevens niet nalaten op te merken dat Codd geen enkele moeite had met de ondersteuning van Object Identifiers (die hij 'surrogates' noemt, zie literatuurverwijzing 2); kunnen we dan anders dan twijfels koesteren of Object Identifiers terecht door TM zijn verworpen? Is het niet bijvoorbeeld aannemelijk dat de TM Object Identifiers per abuis met pointers heeft verward?

Candidate keys

Het is verhelderend om de verwerping van (object) identiteit door de TM vanuit een andere invalshoek te beschouwen. In de relationele algebra zoals deze in hoofdstuk 4 van de TM wordt gepresenteerd, is geen plaats ingeruimd voor de identiteit van tuples of relaties. Volgens bladzijde 152 van de TM is het vereist dat iedere relatie-variabele minimaal een candidate key heeft. Op bladzijde 153 wordt vervolgens, vrij vertaald, gesteld dat candidate keys op unieke wijze tuples in relatie-variabelen dienen te identificeren. Is vanuit het perspectief van de relationele algebra van de TM, de vereiste dat er minimaal een candidate key dient te bestaan niet arbitrair? Duidelijk is dat deze vereiste niet volgt uit de relationele algebra van de TM! Waar volgt het dan wel uit? Is het niet mogelijk dat de TM identiteit uit de relationele algebra verwerpt, terwijl het via een achterdeur identiteit vermomd als candidate keys in relationele systemen introduceert?

Conclusies

De auteurs Date en Darwen hebben hun werk aan het boek 'Foundation for future database systems; The Third Manifesto' in een context van logische correctheid geplaatst door de stelling *Alle logische fouten zijn grote fouten* als een grondslag aan te wijzen. In dit artikel is beargumenteerd dat de norm van logische correctheid in dit boek met betrekking tot de twee vermeende grote blunders niet is gehaald.

Ten eerste geldt dat de vermeende grote blunders niet op basis van pertinente argumenten als zodanig zijn aangemerkt. Ten tweede geldt dat, uitgaande van het begrippendefinitiekader van de TM, er niet is vastgesteld dat de vermeende grote blunders inderdaad als met recht blunders te noemen zijn. Als laatste kan worden aangemerkt dat de gepubliceerde mening van E.F. Codd (Extending the Database Relational Model to Capture more Meaning) op beide punten van dispuut tegenovergesteld is aan de stellingname van de TM.

Noten

1. Alle paginaverwijzingen betreffen de tweede editie van de TM.
2. Zoals domeinen door de TM zijn gedefinieerd.
3. Appendix C op pag. 378 behandelt een hieraan verwant onderwerp.
4. Pointers en Object Identifiers worden in de TM als een en hetzelfde gezien, zie bijvoorbeeld pagina 27 en pagina 417.

Literatuur

- C.J. Date, Hugh Darwen (2000) *Foundation for Future Database Systems*, Addison-Wesley Publishing Company.
- E.F. Codd, (1979) "Extending the Database Relational Model to Capture more Meaning".

Maurice Gittens (maurice@gittens.nl) is zelfstandig IT-consultant.