

Praktische handleiding met een paar uitglidders

# Helder in herhaling over UML

Paul van der Linden

"UML for Database Design" is de poging van Eric Naiburg en Robert Maksimchuk om UML te introduceren en aan te geven hoe deze modelleertaal kan worden toegepast bij het ontwerpen van databases. Beide auteurs zijn verbonden aan de Rational Software Corporation, leverancier van het softwaretool Rational Rose.

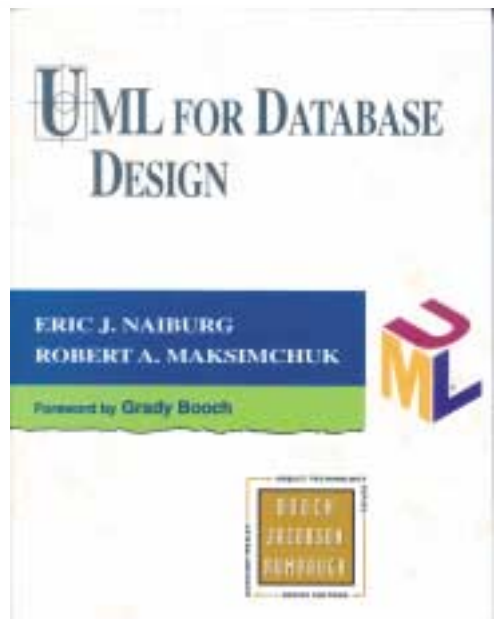
UML is een grafische notatiewijze om bedrijfsbehoeften, applicatiefuncties en database-ontwerp in kaart te brengen. Zaken die in het kader van database-ontwerp aan de orde komen zijn: business use case modelling, database requirements definition, analyse en voorlopig ontwerp, gedetailleerd ontwerp en gebruik. UML is afkomstig uit de koker van de Object Management Group (OMG) en heeft inmiddels de status bereikt van marktstandaard. De W3C-organisatie staat achter deze standaard en levert informatie over UML aan op haar website ([www.w3c.org](http://www.w3c.org)).

De auteurs hebben een pragmatische insteek gekozen. Zij geven aan mensen van de praktijk te zijn en niet theoretisch ingestelde personen.

Praktische adviezen gaan dus voor methodologisch verantwoorde opzetjes. De reden om het boek te schrijven is eenvoudig. Een ontwikkelingstraject begint bij het vaststellen van de business requirements en eindigt (hopelijk) met een werkende applicatie. In dit traject komen een aantal disciplines aan bod die weliswaar allemaal aan hetzelfde einddoel werken, maar waartussen normaliter weinig tot geen communicatie bestaat. Met name genoemd worden business-analisten, programmeurs en ontwerpers van databases. Uiteraard zou het de gehele workflow en het beoogde eindproduct ten goede komen indien de informatie-uitwisseling tussen genoemde disciplines optimaal was. Maar doordat elke discipline zijn eigen insteek kent, in een andere fase wordt ingeschakeld en de resultaten in veelal andere tools opslaat -een tekstverwerker voor de business-analist en een case-

tool voor de applicatiebouwer bijvoorbeeld- is de uitwisseling van informatie verre van optimaal.

Hier kan volgens de auteurs UML een belangrijke rol spelen. De reikwijdte van deze taal omvat immers het gehele traject vanaf de business requirements tot en met de realisatie van de applicaties. Hierdoor worden de verbanden ook expliciet gemaakt tussen de eisen die de business stelt en de software-oplossingen die daarbij horen. Naiburg en Maksimchuk geven aan dat hun boek geen leerboek UML is en ook geen boek om database-ontwerp te leren. Hiervoor bestaan andere boeken die deze onderwerpen veel vollediger behandelen. Hun oogmerk is aan te geven hoe database-ontwerpers kunnen toepassen. Dit past binnen de grotere opzet: als alle betrokken disciplines UML adopteren, ontstaan een betere communicatie en informatie-uitwisseling, waardoor projecten succesvoller kunnen zijn en de eindproducten beter worden. En natuurlijk zou een softwaretool als Rational Rose hier ook een bijdrage in kunnen leveren.



Hebben we hier een goed boek in handen? Het is in ieder geval makkelijk leesbaar geschreven. Het aantal malen dat bepaalde thema's terugkomen is bij tijd en wijle vervelend. Zo wordt nogal gehamerd op het belang van goede samenwerking tussen de drie genoemde disciplines. Dit is helder - al voordat het de eerste keer is opgeschreven. Maar misschien dat hier de praktijkervaring meespeelt: moet je iets honderd keer zeggen voordat het één keer wordt begrepen...? Door het boek heen loopt een case aan de hand waarvan de verschillende UML-diagrammen verder worden uitgelegd. De toegevoegde waarde hiervan is nogal gering, gelet op wat al in de rest van de tekst wordt uitgelegd. Af en toe komen in het boek stukken voor die op zijn minst opmerkelijk te noemen zijn. Zo worden upper-casetools gelijk gesteld met een integrale en strin-

gente (lees: rigide) benadering van het hele ontwikkeltraject. Lower-casetools worden daarentegen gezien als een partiële 'best of breed'-benadering. Dat is wel een heel eigenzinnige interpretatie van de ontwikkelingen in de case-markt. Uiteraard vormt UML hierin het sluitstuk: integraal en flexibel.

De eindconclusie moet luiden dat het boek ondanks deze uitdagingen zijn belofte inlost. Database-ontwerpers die hun eerste UML-schreden nog moeten zetten, zullen in dit boek een waardevol houvast vinden wat UML voor hun werk kan betekenen en hoe de ver-

schillende UML-diagrammen inzetbaar zijn. Gezien de zeer bescheiden omvang van dit boek moet zelfs de meest druk bezette database-ontwerper de tijd kunnen vinden de inhoud tot zich te nemen. ●

**Eric J. Naiburg en Robert A. Maksimchuk, UML for Database Design. Addison-Wesley, Object Technology Series, 2001, ISBN 0-201-72163-5. 320 pag., prijs \$ 39,99.**

Drs. P.F.H. van der Linden (paul.vanderlinden@planet.nl) is consultant.

## Compleet en lijvig

Sandra Gerrits

In "SQL - The Complete Reference" hebben auteurs Groff en Weinberg getracht SQL zo volledig mogelijk in beeld te brengen. Naast de algemene statements halen zij veel voorbeelden aan waarin de grote rdbms'en in hun implementatie van SQL verschillen van de standaard. Ook de overstap van SQL1 naar SQL2 en de uitbreidingen die op deze laatste standaard zijn gekomen, worden in dit boek behandeld. Het geheel wordt uitgebreid geïllustreerd met voorbeeldcode en tekeningen. Op de bijgevoegde cd staan een voorbeeldatabase voor het uitvoeren van de code en volledig werkende, tijdelijke installatieversies van Microsoft SQL Server 7, Oracle 8, Informix, Sybase en IBM DB2.

Het boek komt wat traag op gang. In deel 1 wordt veel aandacht besteed aan de basisstatements van SQL. Door middel van uitgebreide uitwerking van de verschillende voorbeeldcodes leggen de beide auteurs uit hoe SQL omgaat met de verwerking van de gegeven opdracht. Gelukkig trekt het tempo aan in de volgende onderdelen aan.

In deel 2 vindt de lezer aanwijzingen voor hoe hij kan omgaan met databasesystemen met behulp van SQL. Het creëren en updaten van een database passeert de revue, evenals transactie-processing en het maken van views en zaken als beveiliging en de system catalog. En passant staan Groff en Weinberg duidelijk stil bij de opbouw van een relationele database. Voldoende aandacht besteden zij aan primary keys, databaseconsistentie, data-integriteit en locking-mechanismen. De lezer raakt goed met deze begrippen bekend.

Van een iets zwaarder kaliber is het volgende onderdeel van het boek, dat ingaat op het programmeren met SQL. Embedded

SQL en dynamische SQL komen aan bod, naast onder meer ODBC, SQL\CLI en dlib API. Door middel van voorbeeldcode in C, Fortran en Cobol worden de verschillende mogelijkheden van embedded en dynamische SQL uiteengezet.

Het laatste onderdeel van deze 'complete reference' biedt een blik vooruit op uitbreidingen in SQL die nog niet of nog niet volle-

dig zijn meegenomen in de standaard en uitbreidingen daarop. Doordat veel van deze onderdelen al een weg hebben gevonden naar de verschillende dbms'en en de SQL-variant die deze systemen gebruiken, komen de uitbreidingen apart aan bod. Het inzetten van triggers en stored procedures wordt onder meer hier besproken. Ook zijn hoofdstukken gewijd aan de inzet van SQL bij gedistribueerde databases en SQL bij OO-databases. Helaas blijft de relatie tussen SQL en datawarehouses sterk onderbelicht. Het hoofdstuk daarover beslaat alleen een introductie tot datawarehousing.

Dit lijvige boekwerk bewijst niet alleen zijn nut als naslagwerk, maar geeft

bovendien een goed overzicht van de mogelijkheden van SQL in combinatie met relationele databases. ●

**James Groff, Paul Weinberg, SQL - The Complete Reference. Uitg. Osborne McGrawHill, ISBN 0072118458. 1008 pag., prijs \$ 49,99.**

Sandra Gerrits (sandra.gerrits@eniklas.com) is consultant bij eNiklas Netherlands, een in business intelligence gespecialiseerde ICT-dienstverlener.

