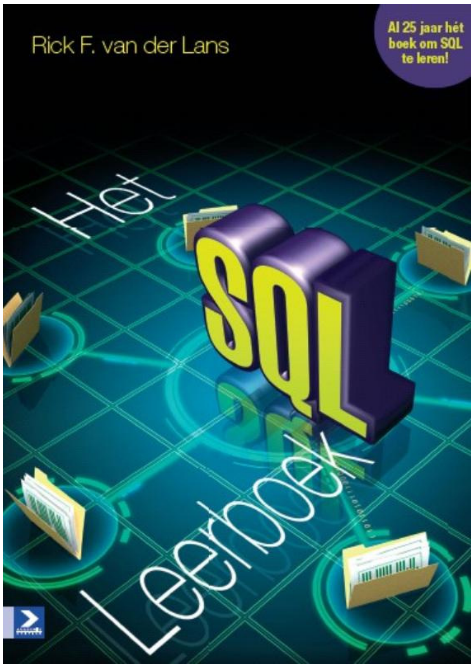
# Het SQL Leerboek – zevende editie
# Introductie tot de verzamelingenleer en de logica




Auteur: Rick F. van der Lans
Versie: 1.0
Datum: Februari 2012

Hoofdstuk 1

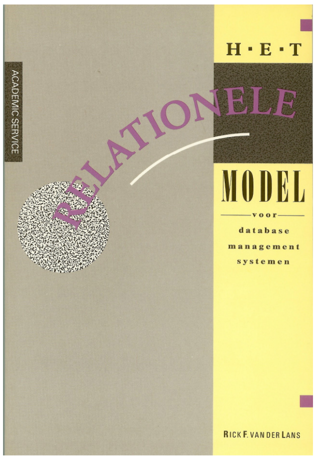
# Inleiding

## 1.1 Inleiding

Zoals duidelijk aangegeven in *Het SQL Leerboek*, is SQL gebaseerd op het relationele model, en de grondbeginselen en theorieën van het relationele model zijn gebaseerd op twee takken van de wiskunde, de verzamelingenleer en de predikatenlogica. Dit document bevat uiteraard geen complete behandeling van deze twee onderwerpen, want voor beide is een geheel boek nodig. We hebben ons beperkt tot beschrijvingen van die begrippen die relevant zijn voor het relationele model. Dit hoofdstuk moet dan ook voornamelijk gezien worden als een korte, niet geheel formele introductie. Voor meer uitvoerige beschrijvingen van de verzamelingenleer en de predikatenlogica verwijzen we naar onder andere de boeken van Stoll[1] en Vader[2].

Indien u vragen of opmerkingen op dit document hebt, laat ons dat weten via email adres sql@r20.nl

Opmerking: Deze tekst in dit document is een bewerking van een hoofdstuk uit mijn boek *Het Relationele Model voor Database Management Systemen* dat in 1988 is uitgegeven, maar dat niet meer beschikbaar is; zie figuur 1.1. Omdat het voor sommigen toch interessant is om iets van de achtergrond van SQL te weten, is besloten deze tekst hier opnieuw beschikbaar te maken.




**Figuur 1.1** *De kaft van het boek Het Relationele Model voor Database Management Systemen*

[1] Stoll R.R., *Set Theory and Logic*, Dover Publications Inc., 1979.

[2] Vader R., *Wiskunde voor Software Constructeurs*, Academic Service, 1987.

Hoofdstuk 2

# De verzamelingenleer

## 2.1 Geschiedenis van de verzamelingenleer

De verzamelingenleer is de tak van de wiskunde die zich bezighoudt met de eigenschappen van verzamelingen. De leer der verzamelingen is opgezet door de Duitse wiskundige Georg Ferdinand Ludwig Phillip Cantor (1845-1918); zie figuur 2.1.

De fundamentele relatie van de verzamelingenleer is die van het *element-zijn*, aangeduid door het symbool $\in$. Een verzameling is in de wiskunde de naam voor elk geheel van op een bepaalde wijze onderscheiden 'dingen'. Deze dingen worden *elementen* van een verzameling genoemd. De essentie van Cantor's concept van een verzameling is dat een groep objecten als een geheel beschouwd wordt. Buiten een formele definitie van een verzameling definieerde Cantor eveneens bewerkingen op verzamelingen: de unie, de doorsnede, het verschil en het product. Vanaf 1879 publiceerde hij een reeks artikelen over de verzamelingenleer. De verzamelingenleer vormt thans de grondslag van de moderne wiskunde.

**Figuur 2.1** *Georg Ferdinand Ludwig Phillip Cantor - grondlegger van de verzamelingenleer*

## 2.2 Verzamelingen en elementen

We definiëren een verzameling als volgt:

*Definitie: Een verzameling is een collectie objecten die een geheel vormen.*

Volgens deze definitie vormen bijvoorbeeld de namen van alle kleuren een verzameling, alle gehele getallen vormen een verzameling, alle paspoortnummers vormen een verzameling en alle records in een bestand vormen een verzameling. De objecten van een verzameling worden *elementen* genoemd. Voorbeelden van de elementen van de verzameling kleuren zijn 'rood', 'wit' en 'geel'. Dat een element $e$ deel uitmaakt van een verzameling $V$ wordt als volgt weergegeven:

$$e \in V$$

Dat een element $e$ *geen* deel uitmaakt van een verzameling $V$ wordt als volgt weergegeven:

$$e \notin V$$

Een verzameling kan elk willekeurig aantal elementen bevatten, twee, vier, vijf miljard maar ook nul. De verzameling met nul elementen wordt de *lege verzameling* genoemd.

De verzamelingenleer kent een speciale, algemeen geaccepteerde notatie. De elementen van een verzameling kunnen met een formele wiskundige notatie of met een grafische presentatie weergegeven worden. Met de formele notatie kan de inhoud van een verzameling door middel van het opsommen van alle elementen of met een formule weergegeven worden. Veronderstel dat de verzameling $V$ bestaat uit de elementen 1 tot en met 5 uit de verzameling natuurlijke getallen (deze verzameling wordt met het symbool $N$ weergegeven). De verzameling $V$ kan ten eerste door middel van opsomming van alle elementen geformuleerd worden:

$$\{ 1, 2, 3, 4, 5 \}$$

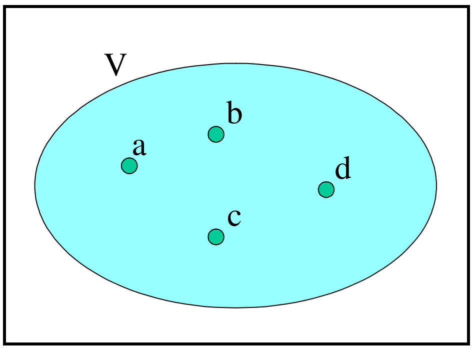
en ten tweede met een formule:

$$\{ e \mid e \in N \text{ en } e > 0 \text{ en } e < 6 \}$$

Deze laatste formulering moet als volgt gelezen worden: De verzameling $V$ bestaat uit alle elementen $e$, waarvoor geldt dat $e$ een element is van de verzameling $N$, groter dan 0 en kleiner dan 6.

Een lege verzameling wordt met twee accolades of met het symbool $\varnothing$ weergegeven.

Grafisch zijn de elementen van een verzameling met een zogenaamd *venn-diagram* weer te geven. Deze diagrammen zijn genoemd naar de Engelse wiskundige John Venn (1834-1923). In een venn-diagram worden de elementen van een verzameling voorgesteld door punten binnen een gesloten contour. In sommige gevallen worden bij de punten de 'namen' van de elementen geplaatst. In figuur 2.2 zijn de elementen van een verzameling genaamd $V$ weergegeven. Een venn-diagram is feitelijk een grafische weergave van een verzameling die bestaat uit een opsomming van elementen. Om tekentechnische redenen zijn venn-diagrammen niet geschikt voor het weergeven van verzamelingen met een groot aantal elementen. Ze zijn daar ook niet voor bedoeld.




**Figuur 2.2** *Voorbeeld van een venn-diagram*

**Figuur 2.3** *John Venn - uitvinder van de venn-diagram*

Het aantal elementen van een verzameling wordt het *kardinaalgetal* genoemd. Het kardinaalgetal van de verzameling { a, b, c, d } is dus vier, want er zijn vier elementen. De lege verzameling heeft per definitie een kardinaalgetal van nul.

Twee verzamelingen $X$ en $Y$ zijn *gelijk* als elk element uit $X$ ook element is van $Y$ en omgekeerd. Dit wordt als volgt weergegeven:

$$X = Y$$

De volgorde van de elementen in een verzameling heeft geen betekenis. De verzameling { a, b, c } is bijvoorbeeld gelijk aan { c, a, b }. Ofwel:

$$\{ a, b, c \} = \{ c, a, b \}$$

Als de verzamelingen $X$ en $Y$ niet gelijk zijn, wordt dit als volgt weergegeven:

$$X \neq Y$$

Als verzameling $X$ geen enkel element bevat dat niet een element is van verzameling $Y$, dan is $X$ een *deelverzameling* van $Y$. Dit wordt als volgt weergegeven:

$$X \subset Y$$

Als geldt dat $X$ een deelverzameling is van $Y$, dan noemen we $Y$ een *omvattende verzameling* van $X$. Dit wordt als volgt weergegeven:

$$Y \supset X$$

De lege verzameling is dus per definitie een deelverzameling van elke andere verzameling. Zij bevat namelijk geen enkel element, dus ook geen enkel element dat niet een element is van de andere verzameling. Om diezelfde reden is elke verzameling ook een deelverzameling van zichzelf. Dus, ongeacht het aantal elementen van de verzameling $X$ geldt dat $\varnothing \subset X$ en $X \subset X$.

Twee verzamelingen zijn *conjunct* als ze een aantal elementen gemeenschappelijk hebben. De verzamelingen { a, b, c } en { b, c, d } zijn conjunct, omdat ze de elementen b en c gemeenschappelijk hebben. Twee verzamelingen zijn *disjunct* als ze geen enkel element gemeenschappelijk hebben. Bijvoorbeeld, de verzamelingen { a, b, c } en { d, e, f } zijn disjunct.

Tot nu toe zijn we er steeds van uitgegaan dat een verzameling alleen enkelvoudige elementen bevat, zoals het getal 4 of de kleur 'rood'. In principe kan een element ook zelf weer een verzameling zijn. Een paar voorbeelden:

$\{ \{1\}, \{2\}, \{3\}, \{4\} \}$
$\{ \{1,2,3\}, \{1,2\}, \varnothing \}$
$\{ \{ \{1\}, \{2\} \}, \{ 2, \{3,4,9\}, 5, 4 \}, \{4\} \}$

## 2.3 Bewerkingen met verzamelingen

Vanuit verzamelingen zijn andere verzamelingen te vormen. De verzamelingenleer kent hiervoor verschillende bewerkingen:

- unie of vereniging
- intersectie of doorsnede
- verschil
- product

De *unie* van twee verzamelingen $X$ en $Y$ is de verzameling van alle elementen die tot $X$, tot $Y$ of tot beide behoren.

$$Z = X \cup Y$$

of

$$Z = \{ e \mid e \in X \text{ of } e \in Y \}$$

Bijvoorbeeld:

$$\{ a, b, c \} \cup \{ d, a \} = \{ a, b, c, d \}$$

Rechts van het gelijkteken staat de verzameling elementen die het resultaat vormt van de unie van de twee verzamelingen die links van het gelijkteken staan. De verzamelingen die bewerkt worden, noemen we *operands*. De verzamelingen $X$ en $Y$ zijn dus operands.

De *intersectie* van twee verzamelingen $X$ en $Y$ is de verzameling van alle elementen die zowel tot $X$ als tot $Y$ behoren.

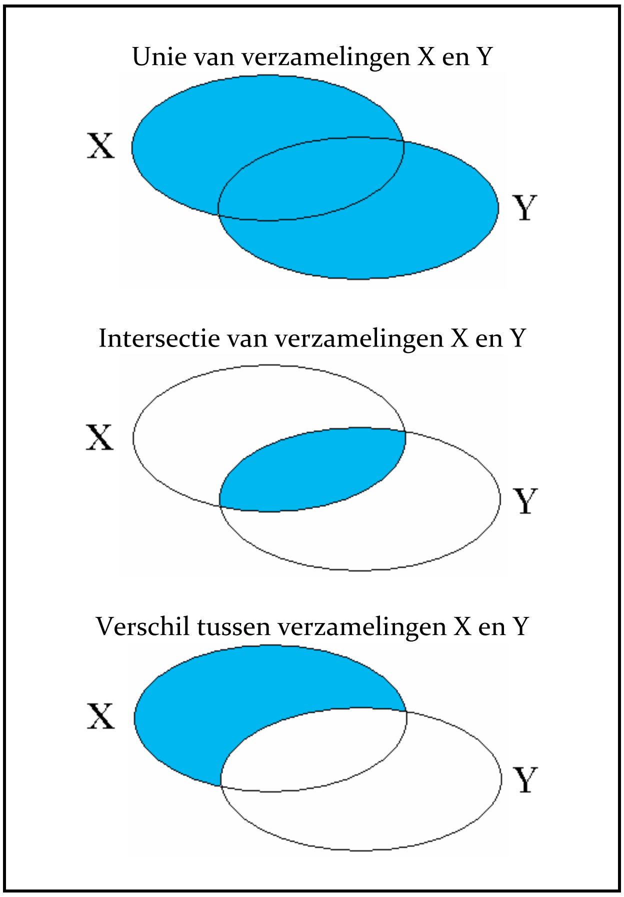
$$Z = X \cap Y$$

of

$$Z = \{ e \mid e \in X \text{ en } e \in Y \}$$

Bijvoorbeeld:

$$\{ a, b, c \} \cap \{ d, a, b, e \} = \{ a, b \}$$




**Figuur 2.4** *Bewerkingen op verzamelingen*

Rechts van het gelijkteken staat de verzameling elementen die het resultaat vormt van de intersectie van de twee verzamelingen links van het gelijkteken. Het resultaat bevat alle elementen die in *beide* verzamelingen voorkomen.

Het verschil van twee verzamelingen $X$ en $Y$ is de verzameling van alle elementen die wel tot $X$ behoren, maar niet tot $Y$.

$$Z = X - Y$$

of

$$Z = \{ e \mid e \in X \text{ en } e \notin Y \}$$

Bijvoorbeeld:

{ a, b, c, d } - { d, e } = { a, b, c }

Rechts van het gelijkteken staat de verzameling elementen die het resultaat vormt van het verschil van de twee verzamelingen links van het gelijkteken. Het resultaat bevat alle elementen uit de meest linkse verzameling die *niet* in de tweede verzameling voorkomen.

In de volgende paragraaf behandelen we de productverzameling.

Er gelden enkele algemene regels, wetten feitelijk, die op bewerkingen van toepassing zijn. Elk van deze wetten zullen we definiëren en toelichten.

1. $A \cup (B \cup C) = (A \cup B) \cup C$ en $A \cap (B \cap C) = (A \cap B) \cap C$

   Deze twee wetten worden de associatieve wetten voor respectievelijk de bewerkingen unie en intersectie genoemd. Een bewerking $\theta$ is associatief als geldt dat $(A\ \theta\ B)\ \theta\ C$ gelijk is aan $A\ \theta\ (B\ \theta\ C)$ voor elke combinatie van $(A, B, C)$. Haakjes kunnen bij associatieve bewerkingen weggelaten worden, dus $A \cup (B \cup C)$ is gelijk aan $A \cup B \cup C$.

2. $A \cup B = B \cup A$ en $A \cap B = B \cap A$

   Deze twee wetten worden de *commutatieve wetten* voor respectievelijk de bewerkingen unie en intersectie genoemd. Een bewerking $\theta$ is commutatief indien de volgorde van de operands onderling verwisseld kan worden zonder dat dit invloed heeft op het resultaat van de bewerking. Met andere woorden, een bewerking $\theta$ is commutatief als $A\ \theta\ B$ gelijk is aan $B\ \theta\ A$ voor elk paar $(A, B)$. Merk op dat $A - B \neq B - A$; de bewerking 'verschil' is dus een voorbeeld van een niet-commutatieve bewerking.

3. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ en $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

   Deze wetten worden de *distributieve wetten* genoemd. De analogie van eigenschappen van unie en intersectie met eigenschappen van optellen en vermenigvuldigen van getallen is zeer opvallend.

4. $A \cup \varnothing = A$ en $A \cap \varnothing = \varnothing$ en $A - \varnothing = A$

   Toevoeging van de lege verzameling aan een niet-lege verzameling heeft geen invloed. Hieruit is af te leiden dat, als $A \cup B = A$, dan geldt dat $B = \varnothing$ of $B \subset A$.

5. $A \cup A = A$ en $A \cap A = A$

   Deze wetten worden de idempotentie wetten genoemd.

6. $A \cup (A \cap B) = A$ en $A \cap (A \cup B) = A$

   Deze twee wetten worden de wetten van DeMorgan genoemd.

## 2.4 Tupel, productverzameling en relatie

Voor het relationele model is de relatie uit de verzamelingenleer belangrijk. Om dit begrip te kunnen definiëren maken we gebruik van twee andere begrippen: n-tupel en productverzameling.

> *Definitie: Een geordende n-tupel bestaat uit n elementen en wordt aangegeven als $<a_1, a_2, \ldots, a_n>$, waarbij element $a_i$ afkomstig is van verzameling $V_i$ $(1 \leq i \leq n)$.*

Twee geordende n-tupels, of kortweg *tupels*, $<a_i, a_2, \ldots, a_n>$ en $<b_1, b_2, \ldots, b_n>$ zijn gelijk, indien voor elke i geldt dat $a_i = b_i$ $(1 \leq i \leq n)$. Indien een tupel uit twee elementen bestaat, spreken we van een *paar* en bij drie elementen van een *tripel*

> *Definitie: De productverzameling van de verzamelingen $V_1, V_2, \ldots, V_n$ $(n>0)$ is de verzameling tupels $\{ <a_1, a_2, \ldots, a_n> \mid a_i \in V_i \}$.*

Een dergelijke productverzameling wordt als volgt weergegeven:

$$V_1 \times V_2 \times \ldots \times V_n$$

De productverzameling wordt ook wel het *cartesisch product* genoemd (genoemd naar René Descartes).

Veronderstel dat $U = \{ a, b \}$, $V = \{ 1, 2, 3 \}$ en $W = \{ 4, 5 \}$. De productverzameling van $V$ en $W$ is dan:

$$V \times W = \{ < 1,4 >, < 1,5 >, < 2,4 >, < 2,5 >, < 3,4 >, < 3,5 > \}$$

en de productverzameling van $U \times V \times W$ is:

$$
\begin{aligned}
U \times V \times W = \{ & < a, 1, 4 >, < a, 1, 5 >, < a, 2, 4 >, \\
& < a, 2, 5 >, < a, 3, 4 >, < a, 3, 5 >, \\
& < b, 1, 4 >, < b, 1, 5 >, < b, 2, 4 >, \\
& < b, 2, 5 >, < b, 3, 4 >, < b, 3, 5 > \}
\end{aligned}
$$

De productbewerking is associatief, dus $(U \times V) \times W$ is gelijk aan $U \times (V \times W)$, maar niet commutatief. $U \times V$ levert namelijk andere elementen op dan $V \times U$, want het geordende paar <a,4> is *niet* gelijk aan <4,a>.

> *Definitie: De relatie R op de verzamelingen $V_1$, $V_2$, …, $V_n$ is een deelverzameling van het cartesisch product $V_1 \times V_2 \times \ldots \times V_n$.*

Voorbeelden (*R1*, *R2* en *R3* zijn relaties):

$$
\begin{aligned}
R1 &\subset A \times B \\
R2 &\subset A \times B \times C \times D \\
R3 &\subset A \times A
\end{aligned}
$$

Als een relatie gebaseerd is op twee verzamelingen (zoals *R1* en *R3*), dan spreken we van een *binaire relatie*, en bij *n* verzamelingen spreken we van een *n-aire relatie*. Conclusie (het $\Leftrightarrow$ symbool betekent 'is gelijkwaardig met'):

$$R \subset V_1 \times V_2 \times \ldots \times V_n \Leftrightarrow R \subset \{ < a_1, a_2, \ldots, a_n > \mid a_1 \in V_1 \text{ en } a_2 \in V_2 \text{ en } \ldots \text{ en } a_n \in V_n \}$$

Een speciaal soort binaire relatie is de *functie*.

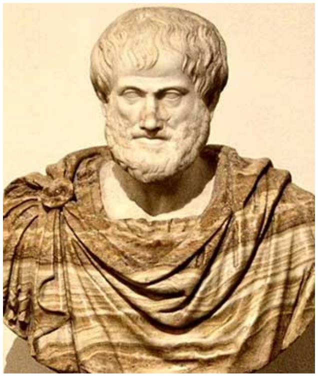
> *Definitie: Een functie is een deelverzameling van het cartesisch product van twee verzamelingen A en B, als elk element uit de verzameling A maximaal één keer voorkomt in de verzameling tupels.*

Hoofdstuk 3

# De logica

## 3.1 Geschiedenis van de logica

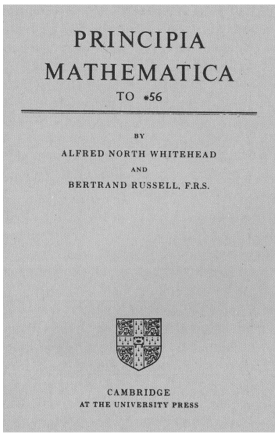
Onder *logica* wordt de wetenschap of kunst van het redeneren verstaan, of met andere woorden, de wetenschap die zich bezighoudt met de wetten van het correcte denken en redeneren. De principes van de logica dateren van een hele tijd terug. De bekende Griekse filosoof Aristoteles (384-322 v.Chr.) was de grondlegger van de klassieke logica; zie figuur 3.1. Aristoteles bestudeerde de klassieke logica als zelfstandig onderdeel van de filosofie. Hij zocht naar een methode om uit beweringen andere beweringen, die reeds impliciet in de beweringen vastliggen, af te leiden. Dit proces om beweringen af te leiden wordt *deductieve redenering* genoemd. We geven een voorbeeld. Uit de beweringen 'alle vogels hebben twee poten' en 'elke mus is een vogel' kan de bewering 'elke mus heeft twee poten' afgeleid worden. Aristoteles zelf noemde dit zelf *Analytica*. Later, ongeveer 200 jaar na Christus, gaf Alexander van Aphrodisias (ook een Grieks wijsgeer) het werk van Aristoteles de naam 'logica'.



**Figuur 3.1** *Aristoteles - filosoof*

Aristoteles zag alleen filosofie als toepassingsgebied van de logica. Pas rond 1850 ontstond het idee om logica in de wiskunde toe te passen. Er werd gezocht naar een strengere axiomatische opbouw en een consequent gebruik van symbolen. Aristoteles maakte in zijn theorieën alleen gebruik van natuurlijke taal. Het uiteindelijke resultaat van dit onderzoek werd en wordt nog steeds de *symbolische logica* genoemd (andere namen zijn de mathematische of formele logica). De symbolische logica werd gecreëerd door de Italiaanse wiskundige Giuseppe Peano (1858-1932) en de Duitser Gottlob Frege (1848-1925). Peano ontwierp onder andere een symbolenschrift (dat we nog steeds gebruiken) voor de symbolische logica.

In 1910 schreven twee zeer vooraanstaande Engelse wiskundigen, Alfred North Whitehead (1861-1947) en Bertrand Arthur William Russell (1872-1970) het werk *Principia Mathematica*[3]; zie figuur 3.2. Zij toonden aan dat alle wiskunde tot logica is terug te brengen.


PRINCIPIA
MATHEMATICA
TO *56

BY
ALFRED NORTH WHITEHEAD
AND
BERTRAND RUSSELL, F.R.S.

CAMBRIDGE
AT THE UNIVERSITY PRESS

**Figuur 3.2** *De titelpagina van de verkorte versie van het Principia Mathematica*

De symbolische logica heeft twee hoofdtakken: *propositielogica* en *predikatenlogica*. De predikatenlogica is gebaseerd op de propositielogica. We zullen in de volgende twee paragrafen de basisprincipes van deze hoofdtakken beschrijven.

## 3.2 Propositielogica

De propositielogica heeft betrekking op *proposities*. Proposities zijn *beweringen* die *waar* of *onwaar* zijn. Een voorbeeld van een bewering is ‘Diane woont in Amsterdam’. In de propositielogica wordt een dergelijke bewering meestal met een enkel symbool, bijvoorbeeld *p*, weergegeven. Buiten symbolen voor beweringen bevat de propositielogica tevens logische operatoren, zoals *en*, *of*, *niet* en *als-dan*. Deze operatoren worden ook wel *boolean-operatoren* genoemd naar de ontdekker ervan, de Engelse wiskundige George Boole (1815-1864). Met deze operatoren kunnen beweringen gecombineerd worden tot andere beweringen. Laten we uitgaan van de beweringen p ‘Diane woont in Amsterdam’ en q ‘Diane is artiest’. De meest gebruikte en bekende operatoren zijn:

- conjunctie (en): $p \cap q$ stelt de bewering voor ‘Diane woont in Amsterdam en Diane is artiest’
- disjunctie (of): p V q stelt de bewering voor ‘Diane woont in Amsterdam of Diane is artiest’
- ontkenning (niet): $\neg p$ stelt de bewering voor ‘Diane woont niet in Amsterdam’
- implicatie (als-dan): $p \rightarrow q$ stelt de bewering voor ‘Als Diane in Amsterdam woont, dan is Diane artiest’

Beweringen die met een enkel symbool worden weergegeven, noemen we *elementaire beweringen*. Beweringen die naast één of meer symbolen ook operatoren bevatten, worden *samengestelde beweringen* genoemd. Ze zijn als het ware ‘samengesteld’ uit meerdere elementaire beweringen.

Een bewering (of de waarde van een bewering) is *waar* of *onwaar*. Het wel of niet waar zijn van een elementaire bewering kan alleen bepaald worden door de realiteit te bestuderen. De bewering ‘Diane woont in Amsterdam’ kan alleen getoetst worden door bijvoorbeeld in het bevolkingsregister van Amsterdam te kijken. Echter, de propositielogica laat dit buiten beschouwing. Er wordt gewoon van uitgegaan dat een bewering *waar* of *onwaar* is, ongeacht de betekenis van de gebruikte woorden en zinnen. Er wordt alleen gekeken of de redenering correct is. Vandaar dat beweringen als abstracte symbolen worden weergegeven.

De waarde (*waar* of *onwaar*) van een samengestelde bewering wordt afgeleid van de waarde van alle elementaire beweringen waaruit de samengestelde bewering is opgebouwd. In tabel 3.1 staat een zogenaamde *waarheidstafel*. Hierin wordt aangegeven wat de waarde van een samengestelde bewering is bij gegeven waarden van de elementaire beweringen.

[3] Whitehead A.N. en Russell B.A.W., *Principia Mathematica*, Cambridge University Press, Cambridge, delen I, II en III, 1910 - 1913.

| p | q | p ^ q | p ∨ q | ¬p | p → q |
|---|---|---|---|---|---|
| waar | waar | waar | waar | onwaar | waar |
| waar | onwaar | onwaar | waar | onwaar | onwaar |
| onwaar | waar | onwaar | waar | waar | waar |
| onwaar | onwaar | onwaar | onwaar | waar | waar |

**Tabel 3.1** *Een waarheidstafel*

Hoe moet zo'n waarheidstafel gelezen worden? Veronderstel dat u moet weten wat de waarde van de bewering p ∨ q is. U kunt de waarde van deze bewering alleen bepalen als u weet wat de waarden van de elementaire beweringen p en q zijn. Laten we aannemen dat deze respectievelijk *waar* en *onwaar* zijn. De waarde van de samengestelde bewering is nu te bepalen door op die regel in de waarheidstafel te kijken waar de bewering p *waar* en de bewering q *onwaar* is. Dit is dus de tweede regel. Vervolgens kijkt u op die regel en in de kolom waarboven p ∨ q staat, en daar staat het antwoord: *waar*.

Indien een bewering meer dan één operator bevat, wordt de evaluatie beïnvloed door de prioriteiten van de operatoren. De operator met de hoogste prioriteit is ¬ (niet), dan ^ (en), vervolgens ∨ (of) en tenslotte → (als-dan). De operator met de hoogste prioriteit wordt dan als eerste geëvalueerd. Dus in de volgende bewering wordt eerst $b_2$ ^ $b_3$ geëvalueerd.

$$b_1 \vee b_2 \wedge b_3 \vee b_4$$

Veronderstel dat $r_1$ het resultaat is van $b_2 \wedge b_3$. Hierna wordt $b_1 \vee r_1$ geëvalueerd; het resultaat is $r_2$. Het uiteindelijke resultaat wordt de waarde van $r_2 \vee b_4$. We kunnen dit ook als volgt weergeven:

$b_2 \wedge b_3 \rightarrow r_1$
$b_1 \vee r_1 \rightarrow r_2$
$r_2 \vee b_4 \rightarrow$ eindresultaat

Met haakjes kan de evaluatie van de waarde van een bewering beïnvloed worden. Stel, we hebben de volgende bewering:

$$(b_1 \vee b_2) \wedge (b_3 \vee b_4)$$

de volgorde van verwerking wordt nu:

$b_1 \vee \mathrm{b}_2 \rightarrow r_1$
$b_3 \vee b_4 \rightarrow r_2$
$r_1 \wedge r_2 \rightarrow$ eindresultaat

Bij gegeven waarden voor $b_1$, $b_2$, $b_3$ en $b_4$ kan de waarde van de eerste bewering anders zijn dan die van de tweede. Veronderstel bijvoorbeeld dat $b_1$ en $b_2$ *waar* zijn en $b_3$ en $b_4$ *onwaar*. De waarde van de eerste bewering is dan *waar* en die van de tweede *onwaar* (ga dit zelf na).

Per definitie kunnen met één of twee operatoren alle andere operatoren uitgedrukt worden. De gekozen operator(en) noemen we dan de *elementaire* of *primitieve operatoren* en de andere de *afgeleide operatoren*. Een veelgebruikte keuze is het ¬ en ^. Bewijs:

$p \vee q \Leftrightarrow \neg(\neg p \wedge \neg q)$
$p \rightarrow q \Leftrightarrow \neg(p \wedge \neg q)$

## 3.3 Predikatenlogica

De propositielogica is niet geschikt voor het formuleren van alle mogelijke beweringen. Beweringen als 'Alle werknemers verdienen meer dan € 50.000,-' of 'Enkele werknemers zijn secretaresse' zijn met de propositielogica alleen als enkel symbool weer te geven. Om aan deze tekortkoming tegemoet te komen, is de propositielogica uitgebreid tot de *predikatenlogica*. In de predikatenlogica wordt een elementaire bewering niet abstract weergegeven, maar wordt uitgewerkt. Bijvoorbeeld de bewering 'Diane woont in Amsterdam' wordt in de propositielogica met een enkel symbool weergegeven. De predikatenlogica geeft deze bewering als volgt weer:

WONEN(Diane, Amsterdam)

Deze bewering moet gelezen worden als: Diane heeft een relatie met Amsterdam en die relatie heeft betrekking op wonen. De bewering 'Ruud woont in Rotterdam' wordt dan als volgt weergegeven:

WONEN(Ruud, Rotterdam)

En tenslotte, de bewering 'Diane is artiest':

ARTIEST(Diane)

De samengestelde bewering 'Diane woont in Amsterdam en Diane is artiest':

WONEN(Diane, Amsterdam) ^ ARTIEST(Diane)

Beweringen in de predikatenlogica hebben echter meestal een meer algemene vorm dan de voorbeelden die tot nu toe zijn gegeven. De bewering 'Alle artiesten wonen in Amsterdam' bijvoorbeeld, wordt:

$\forall x$(ARTIEST($x$) $\rightarrow$ WONEN($x$, Amsterdam))

Het $\forall$ symbool wordt de *alquantor* of de *universele quantor* genoemd en moet gelezen worden als 'Voor alle'. Het symbool $x$ stelt een *variabele* voor, in tegenstelling tot Diane en Amsterdam dat constanten zijn. Voor $x$ kunnen we elke andere letter kiezen. De bovenstaande bewering moet gelezen worden als: 'Voor alle $x$ geldt dat, als $x$ een artiest is, dan woont $x$ in Amsterdam'. Deze bewering is dus *waar* als er geen enkele $x$ te vinden is die wel artiest is, maar niet in Amsterdam woont. Een ander voorbeeld: 'Er is een artiest die in Amsterdam woont'.

$\exists x$(ARTIEST($x$) ^ WONEN($x$, Amsterdam))

Het $\exists$ symbool wordt de *existentiequantor* genoemd en moet gelezen worden als 'Er bestaat een'. De bovenstaande bewering moet gelezen worden als: 'Er bestaat een $x$ waarvoor geldt dat $x$ een artiest is en $x$ in Amsterdam woont'.

In de predikatenlogica worden beweringen meestal *formules* genoemd. Formules dienen aan een aantal regels te voldoen. Niet elk samenraapsel van operatoren en beweringen geldt als een correcte formule. De syntaxis voor correcte formules is in figuur 3.3 weergegeven (zie de bijlage in *Het SQL Leerboek* voor een beschrijving van de gebruikte syntaxissymbolen). Een formule die aan deze regels voldoet, wordt in de literatuur een *well-formed formula*, of kortweg *wff* genoemd.

```
<formule> ::=
   <atoom> |
   ¬ <formule> |
   ( <formule> <boolean-operator> <formule> ) |
   <quantor> <variabele> ( <formule> )

<boolean-operator> ::= ^ | ∨ | →

<quantor> ::= ∀ | ∃
```

**Figuur 3.3** *Syntaxisregels voor formules*

We geven een aantal voorbeelden van formules met twee of meer quantoren:

1. Alle werknemers behoren tot een afdeling.

$$\begin{array}{l}\forall x(\exists y(\text{WERKNEMER}(x) \rightarrow \\ \quad ((\text{AFDELING}(y) \wedge \text{WERKT-VOOR}(x,y))))\end{array}$$

2. Een werknemer is programmeur of analist; maar niet beide.

$$\begin{array}{l}\forall x\ (\text{WERKNEMER}(x) \rightarrow \\ \quad ((\text{PROGRAMMEUR}(x) \vee \text{ANALIST}(x)) \wedge \\ \quad \neg(\text{PROGRAMMEUR}(x) \wedge \text{ANALIST}(x)))\end{array}$$

3. Niet elke werknemer behoort tot een afdeling.

$$\begin{array}{l}\neg\forall x\ (\exists y(\text{WERKNEMER}(x) \rightarrow \\ (\text{AFDELING}(y) \wedge \text{WERKT-VOOR}(x,y))))\end{array}$$

4. Elke afdeling bevat tenminste twee werknemers.

$$\begin{array}{l}\forall x\ \exists y \exists z(\text{AFDELING}(x) \rightarrow \\ (\text{WERKNEMER}(y) \wedge \text{WERKNEMER}(z) \wedge \\ \text{WERKT-VOOR}(y,x) \wedge \text{WERKT-VOOR}(z,x) \wedge y \neq z))\end{array}$$

Voor beide quantoren geldt dat met die quantor en met de operator ¬ de andere quantor uitgedrukt kan worden:

$$\begin{array}{l}\exists \text{x}(\text{formule}) \Leftrightarrow \neg\forall x(\neg\text{formule}) \\ \forall x(\text{formule}) \Leftrightarrow \neg\exists x(\neg\text{formule})\end{array}$$

We zullen deze twee stellingen met twee voorbeelden duidelijk maken. We geven de formule voor de bewering ‘Er bestaat tenminste één werknemer’:

$$\exists \text{x}(\text{WERKNEMER}(x))$$

Deze formule is gelijkwaardig aan:

$$\neg\forall x(\neg\text{WERKNEMER}(x))$$

Deze laatste formule moet als volgt gelezen worden: ‘Niet voor alle $x$ geldt dat $x$ geen werknemer is’. En dit is inderdaad een omslachtige, maar gelijkwaardige bewering voor ‘Er bestaat tenminste één $x$ die werknemer is’. De volgende bewering ‘Alle werknemers wonen in Amsterdam’:

$$(\forall x(\text{WERKNEMER}(x) \rightarrow \text{WONEN}(\text{Amsterdam},x))$$

is gelijkwaardig aan:

$\neg\exists x(\neg(\text{WERKNEMER}(x) \rightarrow \text{WONEN}(\text{Amsterdam},x)))$

'Er bestaat geen *x* waarvoor geldt dat x geen werknemer is die in Amsterdam woont'.

Opmerking: Eigenlijk moeten we voor de volledigheid spreken over de *predikatenlogica van de eerste orde,* want achter de quantoren staan we alleen variabelen toe en geen complete verzamelingen.

## De Auteur

**Rick F. van der Lans** is auteur van vele boeken over SQL. Naast dit SQL Leerboek dat in diverse talen vertaald is, waaronder Engels, Duits, Chinees enItaliaans, heeft hij SQL boeken geschreven voor producten als MySQL, Oracle, SQLite, Ingres en Pervasive PSQL.

Hij is onafhankelijk adviesur, auteur en docent gespecialiseerd in databasetechnologie, datawarehousing en applicatie-integratie. Hij is oprichter en directeur van R20/Consultancy. Door de jaren heen heeft hij veel organisaties geadviseerd op het gebied van IT-architecturen.

Als spreker op conferenties en seminars wordt hij internationaal gerespecteerd. Al meer dan vijfentwintig jaar geeft hij over de gehele wereld lezingen, inclusief in de meeste Europese landen, Noord- en Zuid-Amerika en Australië. Hij is voorzitter van het jaarlijkse European Data Warehouse and Business Intelligence Conference. Hij schrijft een column voor Database Magazine en voor het internationale BeyeNetwork.com. Zeven jaar lang was hij lid van de Nederlandse ISO commissie verantwoordelijk voor ISO SQL Standaard.

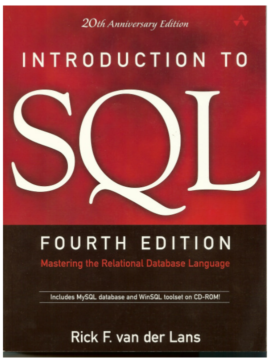
Rick kan via de volgende kanalen bereikt worden:

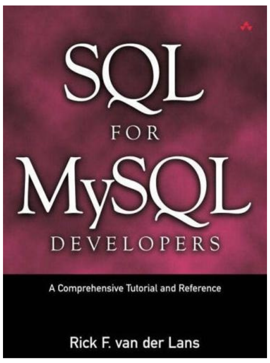
| | |
|---|---|
| Email: | rick@r20.nl |
| Twitter: | http://twitter.com/Rick_vanderlans |
| LinkedIn: | http://www.linkedin.com/pub/rick-van-der-lans/9/207/223 |

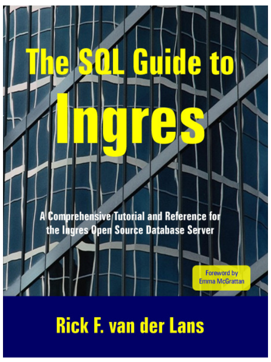
## Cursussen over de volgende onderwerpen kunnen door Rick F. van der Lans verzorgd worden

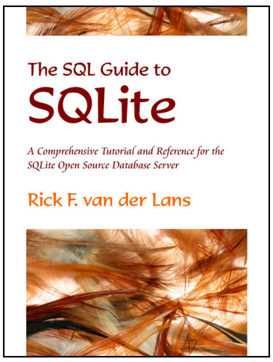
- Database-ontwerp en informatiemodellering
- De basis van SQL
- Het ontwikkelen van geavanceerde SQL queries
- Datawarehousing en business intelligence
- Data virtualisatie

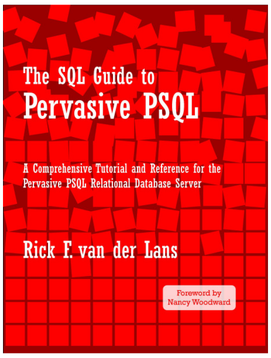
## Andere boeken geschreven door Rick F. van der Lans

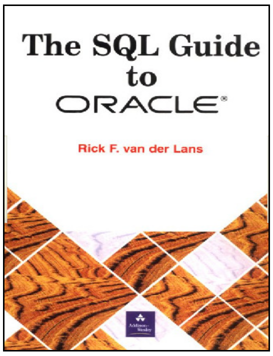



The SQL Guide to
Pervasive PSQL
A Comprehensive Tutorial and Reference for the Pervasive PSQL Relational Database Server
Rick F. van der Lans
Foreword by
Nancy Woodward


The SQL Guide
to
ORACLE
Rick F. van der Lans