Tool voor interactieve analyse van grote datavolumes

De Hilbert Engine

Rob Peters

Data-analisten krijgen met steeds grotere datavolumes te maken. Een telecom-bedrijf bijvoorbeeld wil het effect van promoties op het mobiele telefoon-gebruik onderzoeken. Omdat daarbij verschillende invalshoeken bekeken worden is een interactieve analyse nodig.

Daarvoor moet al snel een paar miljoen gesprekken per dag worden onderzocht. Wanneer een periode van een maand of langer wordt onderzocht dan gaat het om meer dan 100 miljoen gesprekken. Een ander voorbeeld is een supermarktketen die het effect van acties op individuele producten over een langere periode bestudeert en daarvoor meer dan 100 miljoen transacties doorzoekt. Ook een bank analyseert al snel grote volumes aan transacties.

Interactief

Het analyseren van grote datavolumes kost veel tijd. Op een gangbare database, die op een gemiddelde server draait, zal een query over een tabel met meer dan 100 miljoen records tientallen minuten kosten. In een test bij een telecombedrijf werd voor een aantal servicenummers het aantal gesprekken en de duur van die gesprekken onderzocht. Deze query draaide over een bestand met 150 miljoen records, in een Oracle database en op een IBM p660 server met 8 CPU's en 8 GB geheugen. De query duurde 17 minuten.

> De eerste stap is gezet door een systeem te ontwikkelen dat woorden verwerkt als getallen

Een interactieve analyse is dan niet mogelijk, hoewel de vraagstelling dat vaak vereist. Niet alleen de query-duur is daar debet aan, maar ook het feit dat dergelijke zware query's eerder door een DBA dan een analist worden uitgevoerd. Er is dus behoefte aan een database of tool waarmee grote datavolumes interactief kunnen worden geanalyseerd.
De Hilbert Engine is zo'n tool. Het is een database die snel toegang tot grote hoeveelheden data verleent. Een demoversie die op een laptop met een Pentium 1.6 GHz processor en 512 MB intern geheugen draait, selecteert in minder dan een seconde uit een gegevensset met 5 miljoen records die records behorende bij een opgegeven naam. De Hilbert Engine is eind jaren negentig in de Verenigde Staten ontwikkeld. De Hilbert Engine ontleedt haar naam aan de Hilbert-ruimte die door de Duitse wiskundige David Hilbert (1862-1943) is gedefinieerd. De Hilbert-ruimte is een veralgemenisering van het begrip n-dimensionale ruimte: de elementen van een Hilbert-ruimte zijn eindige of oneindige rijtjes reële getallen. De Hilbert Engine doet aan een Hilbert-ruimte denken vanwege haar array-structuur en de conversie van alle data naar getallen.

Achtergrond

Een grondgedachte achter de Hilbert Engine is dat het hart van een computer wordt gevormd door een CPU die bedoeld is om getallen te verwerken. Daarom is het logisch problemen aan de CPU aan te bieden in numerieke en mathematische termen. De eerste stap is gezet door een systeem te ontwikkelen dat woorden verwerkt als getallen. Dat doet de Hilbert Engine door gebruik te maken van verschillende getallenstelsels naast het decimale stelsel. In feite zijn alle woorden te beschouwen als een getal door gebruik te maken van een nieuw getallenstelsel. Wellicht vinden we het vreemd 'WASHINGTON' te zien als een getal maar van '8458803055151383' vinden we dat niet. Toch zijn beiden een getal, en wel hetzelfde getal alleen in verschillende getallenstelsels. 'WASHINGTON' is de TetraDecimale (40-tallenstelsel) weergave en '8458803055151383' is de Decimale weergave (tabel 1). Alle woorden converteren naar een TetraDecimaal getal is een stap waarbij het oorspronkelijke woord nog herkenbaar blijft.
Een TetraDecimaal getal in de Hilbert Engine gebruikt de onderstaande 40 karakters:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^

Tabel: klant

Klantcode	Klantnaam	Plaats
K1	Jansen	Utrecht
K2	Roelofs	Arnhem
K3	Aarts	Goes
...	...	...

Tabellengroep: klant

	Klantcode
1	K1
2	K2
3	K3
4	...

	Klantnaam
1	Jansen
2	Roelofs
3	Aarts
4	...

	Plaats
1	Utrecht
2	Arnhem
3	Goes
4	...

Tabellengroep: klant
Geconverteerd naar numerieke tabellen (tetradecimaal)

	Klantcode
1	K1
2	K2
3	K3
4	...

	Klantnaam
1	JANSEN
2	ROELOFS
3	AARTS
4	...

	Plaats
1	UTRECHT
2	ARNHEM
3	GOES
4	...

Tabellengroep: klant
Gesorteerde numerieke tabellen (tetradecimaal))

	Klantcode	Rij#
1	K1	1
2	K2	2
3	K3	3
4	...	...

	Klantnaam	Rij#
1	BERG	72
2	AARTS	3
3	ANKER	37
4	...	...

Afbeelding 1: Opsplitsing van een tabel met drie kolommen in drie één-kolom tabellen, conversie naar numerieke waarden, en sortering van de numerieke waarden.

Dit 40-talligstelsel gebruikt tien cijfers en 26 hoofdletters en vier extra symbolen. Een andere keuze van cijfers, letters en symbolen is ook mogelijk. Maar dat zal een andere vertaling tot gevolg hebben. Het bovenstaande stelsel vertaalt een woord als 'Washington' in het getal 'WASHINGTON'. De keuze van het getallenstelsel wordt bepaald door de analysevraag. Wanneer bij de vergelijking van woorden het verschil tussen hoofd- en kleine letters geen rol speelt, dan volstaat een 40-talligstelsel, anders is een 66-talligstelsel nodig.

Getallenstelsel	Basis	Getal
Binair	2	1111000001101001111001000111101010 0
		1111001001000101 1
Octaal	8	360323621724744427
Decimaal	10	8458803055151383
Hexadecimal	16	1E0D3C8F53C917
TetraDecimaal	40	WASHINGTON

Tabel 1: Hetzelfde getal in vijf verschillende getallenstelsels.

Nadat een woord is vertaald in een getal wordt de verwerking heel eenvoudig. Twee getallen worden door een CPU vele malen sneller vergeleken dan twee woorden. De volgende stap die voor de Hilbert Engine is genomen betreft de bewerking van tabellen. Een tabel met meerdere kolommen wordt opgesplitst in meerdere tabellen met ieder één kolom (zie afbeelding 1). In deze tabellen is de waarde gekoppeld aan een rijnummer. Omdat alle kolommen uit de oorspronkelijke tabel in dezelfde volgorde worden geladen, kan men aan de hand van het rijnummer het oorspronkelijke record uit de verschillende één-kolom tabellen samenstellen. Dit heeft als voordeel dat men bij een query veel minder data hoeft te benaderen, namelijk alleen die één-kolom tabellen die in de query gevraagd worden.

De één-kolom tabellen die afkomstig zijn van één tabel uit de oude database worden een tabellengroep genoemd. Na de opsplitsing worden de één-kolom tabellen volgens bovenstaand principe geconverteerd naar getallen. Een aantal van de één-kolom tabellen wordt in de toekomstige analyse gebruikt voor het selecteren van data. Van deze één-kolom tabellen wordt daarom nog een gesorteerde versie gemaakt. Deze zijn vergelijkbaar met indexen in de oorspronkelijke databasetabellen. In de gesorteerde versie zijn de numerieke waarde en het oorspronkelijke rijnummer opgenomen (zie afbeelding 1). Let op, in deze gesorteerde tabel komt BERG voor AARTS omdat de laatste met vijf karakters een groter getal is.

Indien meerdere tabelgroepen in een analyse worden betrokken dan moet men de relatie tussen deze tabelgroepen vastleggen in Hilbert. Binnen een tabellengroep verwijzen de rijnummers naar dezelfde instantie. De relatie tussen tabellengroepen wordt

Tabellengroep: orders

	Order
1	1001
2	1002
3	1003
4	...

	Datum
1	3-1-2005
2	5-1-2005
3	6-1-2005
4	...

	Klant
1	K1
2	K9
3	K1
4	...

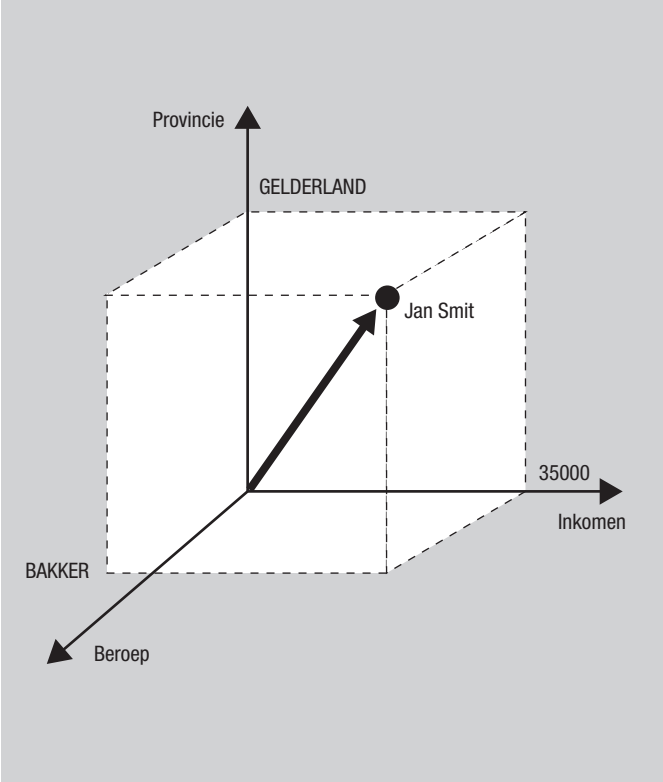
Relatietabel-1

	Order	Klant
1	1	1
2	2	9
3	3	1
4	...	...

Relatietabel-2

	Klant	Order
1	1	1
2	1	3
3	1	49
4	...	...

Afbeelding 2: Relatietabellen tussen tabellengroepen Klant (zie afbeelding1) en Orders. Indien via orders, bijvoorbeeld datum, gegevens worden gezocht dan wordt relatietabel-1 gebruikt om eventueel klantgegevens daarbij te zoeken.



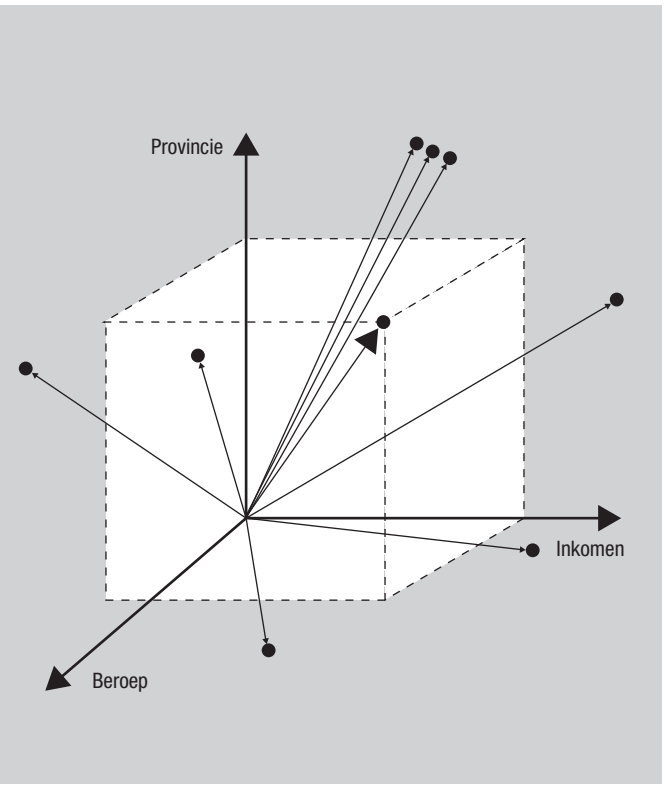
Afbeelding 3: Provincie, inkomen en beroep van Jan Smit als een vector in een driedimensionale ruimte.

vastgelegd in rijnummer combinaties met behulp van twee relatietabellen (zie afbeelding 2). Deze twee relatietabellen zijn gesorteerd op de rijnummers van respectievelijk de eerste en de tweede tabellengroep. Wanneer de selectie vanuit de eerste tabellengroep plaatsvindt dan wordt de ene relatietabel gebruikt, anders de andere. De tabellengroepen zijn vergelijkbaar met de join via sleutels tussen de oorspronkelijke tabellen.
In een aantal stappen is zo de Hilbert Engine ingericht. De attributen zijn als aparte kolommen opgenomen, ze zijn geconverteerd naar numerieke waarden, en ze zijn, waar nodig, gesorteerd. Omdat zo ieder attribuut een aantal keren is opgeslagen, is de totale benodigde ruimte op de harde schijf twee tot drie keer zo groot in de Hilbert Engine als in een plat bestand. Wel kan men selectief omgaan met het aantal attributen door alleen die attributen die voor de analyse nodig zijn in de Hilbert Engine te laden. Later kan men eenvoudig attributen toevoegen.
Het resultaat van de bovengenoemde bewerkingen is een omgeving waarin de data uit meerdere tabellen als vectoren zijn samengebracht. Door de conversie naar numerieke waarden kan men combinaties van attributen – de één-kolom tabellen – als vectoren in een n-dimensionale ruimte zien. Vector-meetkunde is één van de wiskundige toepassingen die nu mogelijke zijn ter ondersteuning van de analyse van de zo gecreëerde omgeving.

Toepassing

Het gebruik van numerieke waarden maakt selecties veel sneller. Omdat een CPU twee getallen veel sneller dan twee woorden vergelijkt, kan de Hilbert Engine zeer snel een resultaatset ophalen, ook al is het argument een naam of een code. Dit is van belang bij zeer grote datasets, zoals een bestand met telefoongesprekken bij een telecom-bedrijf. Interactieve analyses van grote datasets komen binnen handbereik.
Naast selecties zullen ook analyses zeer snel worden uitgevoerd. Het gebruik van numerieke waarden maakt wiskundige toepassingen mogelijk, zodat bepaalde analyses heel eenvoudig gedefinieerd kunnen worden. Dat heeft weer een snelle verwerking tot gevolg. Een voorbeeld is de vergelijking van twee namen die zijn ingevuld in een klantsysteem. Het gaat om dezelfde klant maar bij de ene naam zijn twee letters omgewisseld: Jansen en Jasnen. Dit kan worden vertaald naar een wiskundig probleem.

Vergelijk de getallen 2345 en 2435 waarbij de cijfers 3 en 4 zijn omgewisseld. Het verschil is een veelvoud van 9, het laatste karakter in de decimale reeks 0 1 2 3 4 5 6 7 8 9. Dit geldt voor alle getallen waarbij twee cijfers zijn omgewisseld. Op dezelfde manier kan worden bepaald dat het bij JANSEN en JASNEN om dezelfde naam gaat met twee omgewisselde letters. Het verschil tussen beide TetraDecimale nummers is 4Z00 en dat is een veelvoud van '^' – het laatste karakter in de tetradecimale reeks. Het bepalen van een dergelijke wiskundige berekening is veel sneller dan de vertaling (en het logische probleem van de vergelijking) van letters in een woord naar machinetaal en de uitvoering door de CPU.



Afbeelding 4: Verschillende clusters van personen worden zichtbaar wanneer weergegeven in een driedimensionale ruimte.

Een ander voorbeeld is de toepassing van vector-meetkunde. De numerieke attributen van een entiteit kan men zien als een vector in een n-dimensionale ruimte, bijvoorbeeld de weergave van provincie, inkomen en beroep van een persoon in een driedimensionale ruimte (zie afbeelding 3). Dit is interessant indien men in een verzameling personen de relatie tussen

> Interactieve analyses van grote datasets komen binnen handbereik

provincie, inkomen en beroep wil onderzoeken. Indien alle personen in de verzameling in de driedimensionale ruimte worden geplot ziet men een patroon van clusters ontstaan (zie afbeelding 4). Binnen een cluster wijzen de vectoren ongeveer dezelfde kant op. Aan de hand van de hoek tussen twee vectoren kan men definiëren of twee personen tot dezelfde cluster behoren. Clusters kunnen dan snel worden bepaald. In de Hilbert Engine zijn alle attributen geconverteerd naar numerieke waarden en kunnen ze alle worden gebruikt in een dergelijke vector-analyse. De Hilbert Engine is op verschillende manieren manipuleerbaar. Voor de ontwikkelaars is er de speciaal ontworpen programmeertaal Hilbertscript. De scripts voor het laden van data zijn eenvoudig toe te passen. Voor de gebruikers worden tot nu maatwerk interfaces geleverd. Voor begin volgend jaar is de oplevering van een generieke interface gepland.

Conclusie

De Hilbert Engine biedt een redelijk alternatief voor analisten die interactief grote datasets willen onderzoeken. Door de conversie van woorden naar numerieke waarden kan men selecties en analyses uitvoeren als wiskundige berekeningen en wordt beter gebruik gemaakt van de processor. Door te kiezen voor een nieuwe aanpak van dataverwerking is de Hilbert Engine een geheel ander pad ingeslagen dan de gangbare databases.
Die ontwikkeling was mogelijk omdat de Hilbert Engine vanaf het begin is ontwikkeld vanuit een vraag naar een snelle data-analyse omgeving.

Rob Peters (rob.peters@qnh.nl) is consultant.