

Firebird als 'enterprise' DBMS

# Lessen in integriteit

Marco Bommeljé

**Database management systemen die geschikt zijn voor grote databases worden in de vakpers getooid met het toevoegsel 'enterprise'. In de academische wereld hanteert men veeleer de term VLDB-technologie. VLDB staat voor Very Large Database, hetgeen duidelijker is, maar nogal oubollig klinkt. Oubollig, dat klopt, want academische onderzoeken en symposia over VLDB-technologie gaan terug tot ver in de jaren tachtig van de vorige eeuw. VLDB-technologie werd en wordt als een aparte tak van sport gezien.**

Het spreekt vanzelf dat ook het ontwerpen, programmeren, bedienen en beheren, kortom, het laten functioneren van Very Large Databases een vak apart is. Dat blijkt klip en klaar uit het zesde artikel van René Veldwijk over de Polisadministratie. Hij beschrijft de technische en organisatorische maatregelen die nodig zijn om de steeds maar groeiende Polisdatabase performant te houden. Het meest extreme voorbeeld daarvan is dat schrijvende programma's moeten worden stilgelegd om langlopende bevragingen te kunnen uitvoeren. Hoewel zulke productieplanning van programma's doet denken aan de *mainframes* uit de jaren tachtig, gaat het hier toch om moderne 'enterprise' DBMS-technologie, zou men denken.

## Geschied en suitable

Niet ieder DBMS kwalificeert zich als 'enterprise' DBMS. In de praktijk blijkt slechts een handvol DBMS'en beschouwd te worden als 'enterprise' DBMS. Het Forrester Wave rapport over 'Enterprise Database Management Systems' van juli 2009 analyseert zeven DBMS'en en concludeert dat Oracle, IBM, Microsoft en Sybase de beste 'enterprise-class features and strategy' bieden. De drie open source DBMS'en MySQL, Ingres en PostgreSQL worden door Forrester afgeserveerd als op z'n best interessante nicheproducten voor specifieke toepassingen.<sup>1</sup> In juli 2009, enige dagen na publicatie van het Forrester rapport, werd de winnaar van de SourceForge Award voor 'Best Project for the Enterprise' uitgeroepen. Dat was, net als in 2007, Firebird DBMS.

Oplettende lezers van Database Magazine weten dat het Firebird-project in het jaar 2000 is opgericht om zich te ontfemen over de broncode van Interbase DBMS, nadat daarvan een open source versie was vrijgegeven door eigenaar Borland.<sup>2</sup> De DBMS-technologie van Firebird gaat dus al veel verder terug,

namelijk tot halverwege de jaren tachtig (alweer die jaren tachtig!) toen Jim Starkey het ontwierp en ontwikkelde voor Digital Equipment Corporation.

Over de vraag of Firebird als 'enterprise' DBMS kan worden betiteld lopen de meningen kennelijk uiteen. Dit artikel beschrijft enige kenmerken van Firebird waarmee de lezer een oordeel kan vellen over de mogelijke inzetbaarheid van Firebird als 'enterprise DBMS'.

## Wat is een 'enterprise' DBMS?

Wat nu maakt een DBMS tot een 'enterprise' DBMS? De term 'enterprise' roept vooral beelden op van groot en stoer en duur, maar is moeilijk in meetbare criteria te vangen. Behalve producteigenschappen, spelen ook zachte factoren zoals toekomstvastheid en ondersteuning een rol.

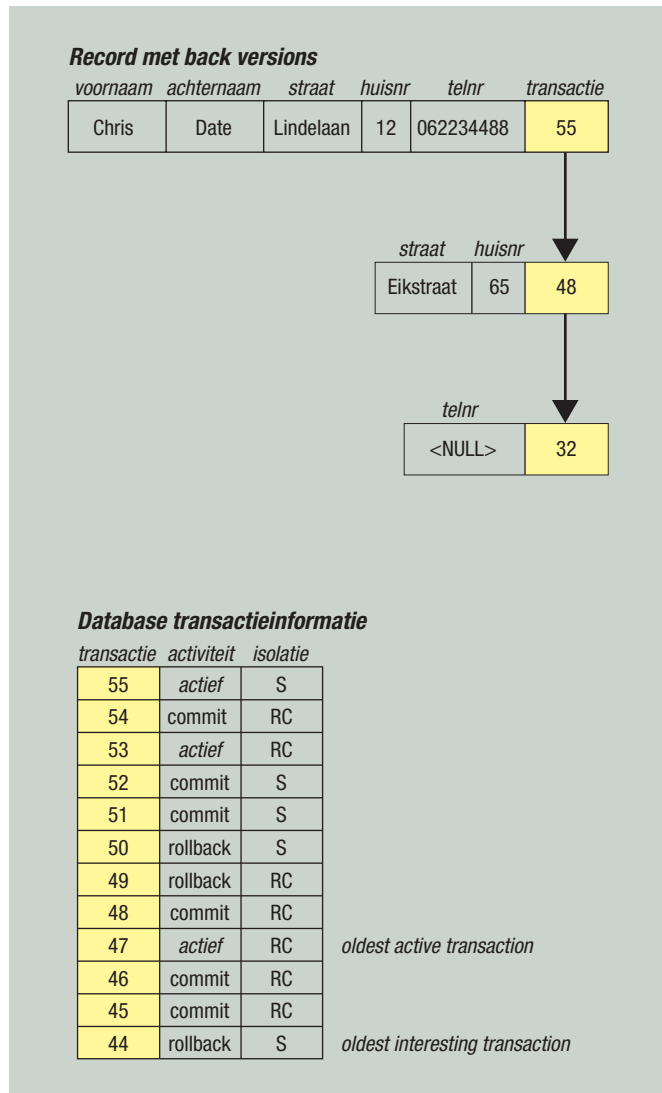
### Integriteit.

Een 'enterprise' DBMS is in de eerste plaats een DBMS dat zijn meest essentiële taken goed vervult. Het gaat dan om het verzorgen van gelijktijdige toegang tot gegevens voor meerdere gebruikers (concurrent access) waarbij gegevensintegriteit (*data integrity*) gewaarborgd wordt.<sup>3</sup>

Bij 'enterprise' databases is de zorg om gegevensintegriteit extra groot, omdat deze vaak een centrale plaats innemen in een gegevenslogistiek netwerk. Problemen op een knooppunt vermenigvuldigen zich over het netwerk.

### Vermogen.

Naast deze basale *functionele* producteigenschappen worden van een 'enterprise' DBMS specifieke *operationele* eigenschappen gevraagd die het werken met Very Large Databases met zich meebrengt. Dat zijn capaciteit, schaalbaarheid en beschikbaarheid.



**Afbeelding 1:** Welke versie krijgt lezende transactie Nr 47?

### Bruikbaarheid.

Of een DBMS succesvol grootschalig kan worden ingezet, hangt in hoge mate af van programmeerbaarheid en beheerbaarheid. Programmeerbaarheid staat voor efficiënt ontwikkelen van toepassingen. Beheerbaarheid betekent dat een DBA de database op eenvoudige manier kan beheren. In wezen gaat het hier om een kostenoverweging, namelijk kosten om expertise te verwerven door opleiding dan wel inhuur, om het DBMS naar behoren te laten werken.

### Integriteit

Firebird heeft vanaf zijn eerste oorsprong een eigen en bijzonder mechanisme om *concurrent access* mogelijk te maken en daarbij gegevensintegriteit te waarborgen; de zogenaamde *multi-generational architecture*.

### Locks en logs.

Van oudsher gebruiken database management systemen een combinatie van *locks* en *logs*. Locks dienen om schrijfacties van

verschillende gebruikers van elkaar te isoleren en leesacties een consistent beeld van de database te bieden. Logs dienen om gegevens veilig te stellen.

Bij wijzigingen worden oorspronkelijke gegevens in een *before-image journal* (ook wel transactielog of undo log) geschreven, zodat een rollback de database kan terugbrengen in de vorige consistente staat. Een *after-image journal* (ook wel write-ahead log of redo log) dient om een database na een crash te kunnen herstellen tot en met de laatst voltooide transactie. Zo'n herstelactie heet *rollforward recovery*.

### Record versioning.

MGA bestaat eruit dat Firebird in de database zelf een administratie van transacties bijhoudt en oorspronkelijke recordversies veilig stelt. Firebird zet in ieder record het identificatienummer van de transactie die het record heeft geschreven, zie afbeelding 1. Bij wijzigen of verwijderen van een record maakt Firebird een *back version* met daarin de oorspronkelijke waarden van gewijzigde velden, waaronder het transactie-id. Wanneer de transactie wordt teruggedraaid, worden de waarden uit de back version weer teruggezet in het record met de actuele versie.<sup>4</sup>

### MGA versus locks 'n logs.

MGA is een buitengewoon efficiënte en robuuste architectuur gebleken. Vergelijking met de traditionele locks 'n logs architectuur laat de volgende voordelen van MGA zien:

- MGA biedt betere *concurrency*. Door het ontbreken van read locks kunnen leesoperaties nooit een schrijfactie blokkeren;
- MGA is I/O efficiënt. Een *commit* is niet meer dan het omzetten van de status in de transactieadministratie. Back versions worden meestal op dezelfde database page geschreven en gelezen.
- MGA biedt vrijwel instantane recovery. Firebird DBMS zet bij het starten van de server alle actieve transacties in zijn administratie op 'rollback'. Daarna is het *business as usual*. Een traditionele roll forward recovery vanuit een log is een tijdrovende operatie.
- MGA heeft minder points of failure. Door het ontbreken van logs is alleen de database zelf een *point of failure*.

*Interbase was lange tijd het enige DBMS met MGA. Oracle introduceerde in versie 7 een vorm van multi-versioning. Helaas houdt Oracle geen adequate administratie van transacties bij, waardoor back versions zomaar kunnen verdwijnen en langlopende leesoperaties falen met een 'Snapshot too old' foutmelding. Productieplanning is dan de enige oplossing.*

*PostgreSQL introduceerde MGA op basis van de Firebird broncode. Ook Microsoft schijnt nu in SQL Server een vorm van versioning geïmplementeerd te hebben.*

### Careful write strategy.

Dichterbij het metaal van de harde schijf staat de integriteit van de interne database-structuur op het spel. Firebird hanteert een

*careful write strategy*, waardoor de zogenaamde *on disk structure* altijd consistent is. Simpel uitgelegd komt het neer op het volgende. Een Firebird database is intern opgebouwd uit vaste blokken die database pages heten. De informatie op een database page kan naar informatie op een andere database page verwijzen. Een actuele recordversie kan verwijzen naar een back version op een andere database page.

Careful write strategy houdt in dat Firebird een database page waarnaar verwezen wordt altijd eerder schrijft dan de database page met de verwijzing. Een soft crash levert daardoor nooit een database op met ongeldige verwijzingen. Hooguit kunnen *orphaned pages* ontstaan, die door een herstelprogramma (GFIX) of door backup/restore weer bruikbaar gemaakt worden.<sup>5</sup>

## Schaalbaarheid

In de oertijd, toen Firebird nog zelfs geen Interbase maar Groton Database heette, was de karakteristieke kantoorcomputer van die tijd: een multi-user minicomputer met een 16-bits CPU en met 64 KB RAM geheugen. Om op dergelijke hardware te kunnen werken is Firebird ontworpen om efficiënt met computer resources om te springen. Die efficiency wordt tot op heden beloond met schaalbaarheid en performance.

De keerzijde van deze erfenis is lange tijd geweest dat een Firebird database een aantal ingebouwde beperkingen had. Met

de introductie van Firebird versie 2 in november 2006 zijn die limieten drastisch opgerekt, waardoor dat niet een beperking kan zijn voor inzet als 'enterprise' DBMS.

Meer dan 16 miljard rijen per tabel, 32 TB data per tabel en 64.000 tabellen per database zou toch for all practical purposes genoeg moeten zijn, althans voor de komende paar jaren.<sup>6</sup>

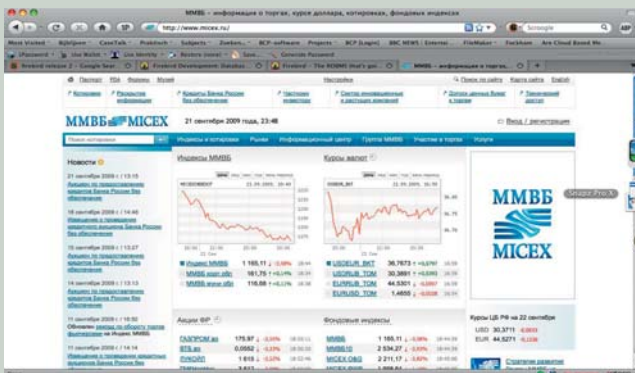
## Architectuur in drie smaken.

De grote onbekende onder de capaciteitslimieten is het mogelijke aantal actieve *concurrent connections*. Firebird kent met dezelfde code base drie verschillende architecturen, elk voor een bepaalde schaalgrootte van aantallen concurrent connections. *Superserver* is de meest gangbare architectuur. SuperServer beheert zelf de databaseconnecties en verdeelt die over threads. Alle connecties delen een gezamenlijk database cachegeheugen. Van oudsher was de capaciteit begrensd door het adresseerbaar geheugen, waardoor een maximum van ongeveer 500 gelijktijdige gebruikers verondersteld werd. Met de ondersteuning van 64-bits CPU's is deze beperking weggefallen.

*Classic* start een onafhankelijk serverproces voor elke user connection. In Classic kan een connectie niet profiteren van een shared cache, maar juist wel van parallele verwerking op SMP computers wanneer grote aantallen gelijktijdige connecties bediend moeten worden.

## Geld verdienen

MICEX ([www.micex.ru](http://www.micex.ru)) is een voorbeeld van Firebird als grootschalig 'enterprise' DBMS. MICEX is de grootste aandelen-, valuta- en derivatenbeurs in Rusland. De Firebird database wordt gebruikt als back-end voor het MICEX Trading System, waarin alle marktdata, orders en transacties worden opgeslagen. Het systeem bedient meer dan 2000 gebruikers verspreid over de acht tijdzones van de Russische Federatie. Dagelijks worden op real-time basis meer dan 250.000 orders verhandeld en 180.000 financiële transacties verwerkt. Daarnaast krijgt Firebird massieve financiële clearing operaties als batch-verwerking aangeboden. Gartner stelde vast dat het MICEX Trading System voldeed aan het Five Nines reliability criterium van 99,999 procent beschikbaarheid.

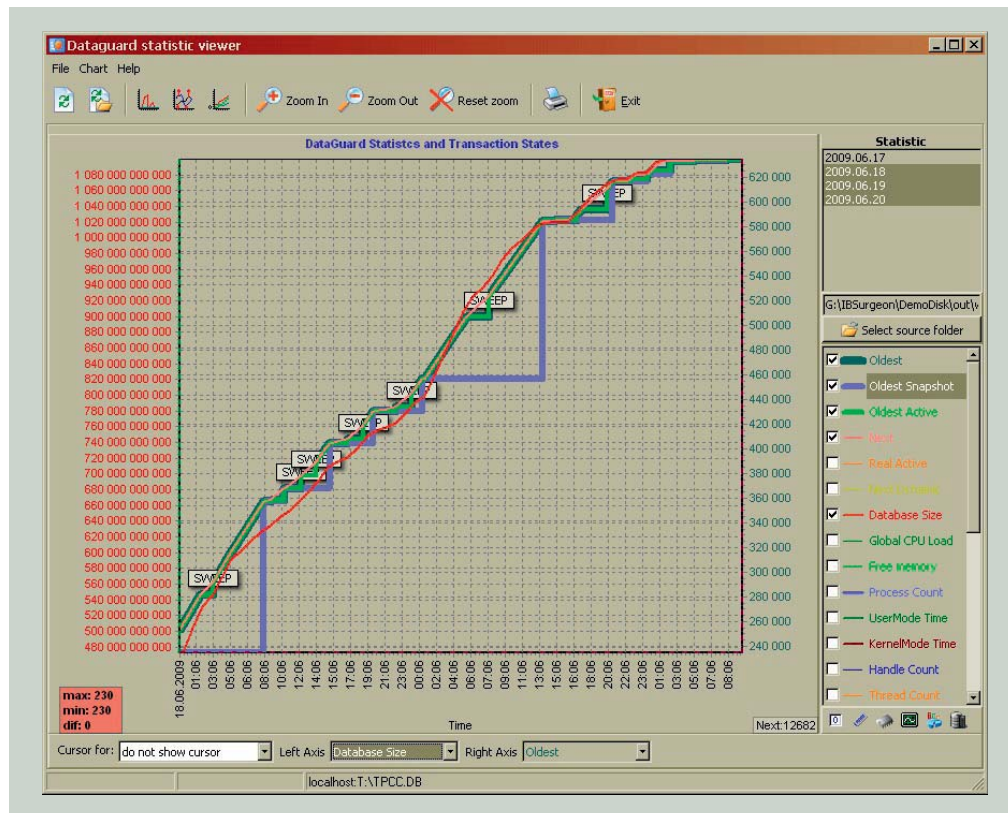


Bas-X ([www.basx.com.au](http://www.basx.com.au), Australia) levert Software-as-a-Service toepassingen voor de retailbranche. Enkele grotere klanten hebben databases van 200 GB tot 450 GB. Firebird DBMS bedient in een cloud computing opstelling duizenden gebruikers.

Van zeer grootschalig 'enterprise' gebruik is ook sprake aan de kleine kant van de schaalbaarheid: Embedded Firebird. De Brits/Spaanse telefonieaanbieder Telefónica O2 installeert het Mobile Internet programma op draagbare telefoons met Firebird als lichtgewicht embedded database.

Een voorbeeld van een succesvolle open source toepassing die werkt met Firebird DBMS is TurboCash ([www.turboCash.net](http://www.turboCash.net)). Het is een boekhoudpakket dat zowel vanaf een memory stick kan werken als in een bedrijfsbrede multi-user opstelling. TurboCash heeft wereldwijd meer dan 60.000 gebruikers.

Distributel ([www.distributel.com](http://www.distributel.com)) is een Canadese Telco Services Provider waar Firebird tot het hart van de infrastructuur behoort. Firebird bedient letterlijk de elektronische schakelkasten die de telefoonverbindingen tot standbrengen, registreert de tikken van de telefoongesprekken en bewaart de normale administratieve gegevens. Bij het opnemen van de telefoon wordt de kiestoon geproduceerd, nadat de database de telefoon heeft gekoppeld aan de klantgegevens. Volcontinu betekent hier 24/7 met een wettelijk vereiste beschikbaarheid van 'five nines' van de telefoongesprekken.



**Afbeelding 2:** 1 Terabyte laden blijkt lineair schaalbaar.

*Embedded* biedt de andere, kleine kant van schaalbaarheid, bedoeld voor toepassingen die als enige de database benaderen. De database is – zoals de naam doet vermoeden – ingebed in het programma.

### Performance.

Schaalbaarheid gaat niet zozeer over groot en veel, als wel over het behoud van performance bij toenemende groei of belasting van de database. Maar ja, wat is performance? In de jaren van de *database wars* bestreden de nu gevestigde 'enterprise' DBMS'en elkaar met benchmark-resultaten. Sinds deze DBMS-leveranciers gezamenlijk het TPC benchmarking-instituut opgericht hebben, hebben de benchmarks geen spannende resultaten meer opgeleverd. De licentievoorzwaarden verbieden inmiddels aan buitenstaanders om benchmarks te publiceren. Iedere uitspraak over performance van Firebird is daardoor onvergelijkbaar. Wel worden er op dit moment interessante testen uitgevoerd op een 1 TB grote Firebird database. Uit de eerste resultaten blijkt dat de throughput, de hoeveelheid data die per tijdseenheid naar de database geschreven kan worden, niet waarneembaar achteruit gaat wanneer de database groeit, zie afbeelding 2.

### Programmeerbaarheid en beheerbaarheid

De SourceForge Community Choice Award die door Firebird is gewonnen, is vooral een *beauty contest* voor open source projecten. Zij die open source software ophalen bij SourceForge stemmen op hun favoriete project. Dat zijn in het geval van Firebird ongetwijfeld ontwikkelaars van toepassingen die met Firebird

werken. Zij geven met hun massale stem een blijk van waardering aan de mensen van het Firebird project. Het gaat hier zeker niet om een handjevol *geeks* die enthousiast zijn. Firebird passeerde vorig jaar de grens van vijf miljoen downloads. Programmeurs waarderen Firebird om het *just works* principe. Het ontbreken van elke vorm van *featuritis* is een groot voordeel, doordat het niet in de weg zit van de core business van de ontwikkelaar: toepassingen maken. Firebird's SQL variant volgt akelig precies de SQL 2003 standaard. Daarbuiten is het niet nodig om een hele rambam van technische details te kennen.

### Garbage Collection en Sweeping.

Een zichzelf respecterend 'enterprise' DBMS wordt geleverd met een legertje van experts aan wie de geheimen geopenbaard zijn van *space allocation, clustering, partitioning, grids, blades, process global memory, high water marks, procedure cache, spin locks, latches* en zo meer.

De dagelijkse praktijk van het gemiddelde 'enterprise' DBMS doet vergeten dat een van de essenties van DBMS-technologie gaat om data independence, abstractie van de fysieke opslag. Veel 'enterprise' DBA's hebben een dagtaak aan *space management* en fungeren in feite als 'oppasmoeder' voor de database om de performance op peil te houden.

Firebird voorziet zelf in zijn *housekeeping* en *tuning*. Beide hebben te maken met de Multi-Generational Architecture. Back versions van records zijn op een gegeven moment niet meer nodig. Dat is omdat de transactie van een actuele recordversie teruggedraaid is of omdat de transactie die het *snapshot* nodig

had, is voltooid. Het opruimen van back versions en het bijwerken van de transactieadministratie gebeurt door *garbage collection* en *sweeping*, twee activiteiten die de Firebird engine zelf voor zijn rekening neemt.

### Tuning.

Door deze housekeeping ontstaan open plekken in de database die weer beschikbaar zijn voor nuttig gebruik. De sponzige databasestructuur die dan ontstaat, biedt een optimale performance. Weinig contention op database pages, nieuwe ruimte voor nieuwe data en back versions en een evenwichtig beeld van responstijden voor uiteenlopende soorten query's. Ditzelfde mechanisme verklaart waarom Firebird DBMS zonder verdere fratsen al volcontinue ('24x7') beschikbaarheid kan bieden.

### Opnieuw: integriteit

Technische kwaliteiten alleen zijn niet voldoende om het predicaat 'enterprise' te verdienen. Rapporten van industry analysts beoordelen steevast ook marktpositie en strategie van de leverancier. De 'marktpositie' van Firebird is in dit artikel al enigzins toegelicht aan de hand van enkele voorbeelden. Het blijkt dat Firebird juist in de 'opkomende' markten, zoals Rusland en Brazilië al een belangrijke plaats inneemt in de 'enterprise' IT-infrastructuur.

## Ook diep technische software vertegenwoordigt in essentie menselijke waarden

Firebird heeft in de afgelopen negen jaar wortel geschoten in de wereldgemeenschap van softwareontwikkelaars en wordt in toenemende mate ingezet voor het grote werk. Dit komt de levensverwachting van het Firebird project zeer ten goede. De FirebirdSQL Foundation, de stichting die de softwareontwikkeling financiert en logistiek ondersteunt, trekt gestaag steeds meer sponsors aan.

Niettemin zal ook in de toekomst de belangrijkste pijler van succes zijn dat Firebird vasthoudt aan de grondslagen van het ontwerp: *simplicity and performance*. De eigenschappen die maken dat Firebird als 'enterprise' DBMS kan worden ingezet, zijn alle hiervan afgeleid.

Wanneer programmatuur in zijn ontwerp en implementatie laat zien op welke *core values* het is gebaseerd, spreekt men van conceptuele integriteit. Dat nu is de belangrijkste kwaliteit van een softwareproduct (aldus Brooks, auteur van het klassieke boek *The Mythical Man Month*). En zo blijkt dat ook diep technische software in essentie menselijke waarden vertegenwoordigt.

### Noten

1. Het Forrester Wave rapport kan voor 1749 dollar gekocht worden, of gratis worden opgehaald van de websites van Oracle en Microsoft. Forrester licht de gehanteerde methode enigzins toe. Het onderzoek bestaat uit vier stappen met elk een eigen aanpak: 1. selecteren van te vergelijken producten; 2. bepalen van evaluatiecriteria; 3. bepalen van scores op deze criteria; 4. wegen van evaluatiecriteria; 5. rapporteren. In de loop van het onderzoek worden telkens klanten van Forrester en de DBMS leveranciers geraadpleegd. Door deze werkwijze weerspiegelt het onderzoek in belangrijke mate de mening bij de grote bedrijven in Forresters eigen klantenkring en bij de leveranciers van de DBMS'en die daar in gebruik zijn. Deze evaluatie van DBMS'en kan dus op z'n best beschouwd worden als een opiniepeiling binnen een zeer beperkte kring van users en vendors. De onderzoeksmethode zorgt ervoor dat sommige evaluatiecriteria (zoals TCO) geheel buiten beeld blijven, terwijl andere evaluatiecriteria zoals 'market presence' (gemeten door het aantal betalende 'enterprise' klanten) zwaar meetellen. Onderzoeken die op deze manier worden uitgevoerd en gepubliceerd, doen weinig meer dan een reeds bestaande perceptie bevestigen als ware het de werkelijkheid. Informatie van daarbuiten blijft ongezien. Zo ontstaat een spiegelpaleis voor IT managers. Ziehier één van de redenen waardoor Firebird onbekend is in de wereld van de grote IT-projecten.
2. Jos van Dongen heeft Firebird een aantal malen genoemd in zijn DB/M artikelen over OS BI-tools. Schrijver dezes heeft eerder, in 2003, in DB/M een artikel aan Firebird gewijd dat uit online archief van Array kan worden opgehaald.
3. Gegevensintegriteit betekent dat gegevens voldoen aan de regels van de database en dat wijzigingen de database altijd in consistente staat achterlaten. Er zijn nog enkele andere essentiële eigenschappen voor een DBMS waarop het artikel niet ingaat: data security en data independence, de abstractie van de fysieke opslag.
4. Bij het wijzigen of verwijderen van een record zoekt Firebird in de transactieadministratie op of de actuele versie van het record al is ge-commit. Is de transactie nog actief, dan is sprake van een update conflict dat, zoals gebruikelijk, leidt tot wachten of falen van de transactie (afhankelijk van een opgegeven instelling). Bij het lezen van een record ziet Firebird aan transactie-id en -status van het actuele record of dat gebruikt kan worden. Voor snapshot consistency wordt de lijst van back versions doorzocht totdat een versie gevonden is van een transactie die eerder is gestart dan de lezende transactie. Voor read committed consistency hoeft alleen een back version geraadpleegd te worden, indien de actuele versie nog niet ge-commit is.
5. Careful Writing is een bijzonder robuust mechanisme gebleken. In een geval uit 1998 bleek een database (dat was toen nog Interbase) jarenlang ongemerkt probleemloos te hebben gewerkt. De laatste medewerker die iets van Interbase wist, had zes jaar daarvoor het bedrijf verlaten. Sindsdien was de database stilletjes aan tot meerdere Gigabytes gegroeid zonder dat ooit een database backup was gemaakt. De informatie in de database bleek nog volledig intact.
6. Limieten Firebird 2.x
  - a. Max grootte van 1 DB-bestand: beperkt door bestandssysteem.
  - b. Max aantal DB-bestanden per database: 64.535.
  - c. Max grootte van een database: (a x b).
  - d. Max aantal tabellen: 64.535.
  - e. Max grootte van 1 tabel: 32 Terabyte.
  - f. Max aantal rijen per tabel: >16 miljard (grootste test).
  - g. Max aantal bytes per rij: 65.536.
  - h. Max database page size: 16 KB.
  - i. Max aantal kolommen per tabel: hangt af van datatypes
  - j. Max aantal indexen per tabel: 65.535.
  - k. Max grootte van index key: 1/4 van page size.
  - l. Max aantal indexen per database: > 4 miljard.
  - m. Max aantal user connections: verschilt per architectuur.

Marco Bommeljé is consultant bij Ockham Consulting Partners.