



Open Source- Datenbanken



Das Angebot ist groß, doch welche Datenbank passt am besten zu welchem Zweck? Für die Antwort auf diese Frage fehlen oft passende Kriterien und nachprüfbare Fakten. Dieser Beitrag stellt vier bekannte freie Datenbanken im Vergleich vor. [Jens-Christoph Brendel](#)

Wer eine Datenbank braucht, hat die Qual der Wahl. Selbst wer sich bereits für freie Software entschieden hat und darum Oracle, Informix, DB2 oder Sybase ausklammert – die teils ebenfalls kostenlose Einsteiger-Pakete bieten – kann noch aus dem Vollem schöpfen. Um die Übersicht zu wahren, lässt dieser Artikel deshalb die zahlreichen kleineren Datenbanken außen vor, die sich besonders als eingebettete Komponenten von Applikationen eignen, etwa SQLite, Berkeley DB, HSQLDB oder Apache Derby. Stattdessen konzentriert er sich auf vier freie, voll ausgebaute relationale Allzweck-Datenbankmanagementsysteme: MySQL, PostgreSQL, Ingres und Firebird.

Mittel zum Zweck

Datenbanken sind Mittel zum Zweck. Selten stehen sie für sich allein und werden von Anwendern direkt per SQL befragt. So gut wie immer ist es eine Applikation, die sich der Datensilos bedient und mit ihrem Benutzerinterface die Speicher hinter den Kulissen verbirgt. Solche Applikationen stellen Ansprüche an Ihr Backend – genau die sind in den meisten Fällen auch gleich die entscheidenden Auswahlkriterien.

Ein zweiter wesentlicher Gesichtspunkt ist das im Einsatzumfeld verfügbare Know-how. Wer erfahrene MySQL-DBAs beschäftigt, wird nicht ohne Not zu PostgreSQL greifen und umgekehrt. Die Materie ist nun einmal einigermaßen komplex und kann eine gehörige Portion Wissen und Können vertragen.

Gerade dort, wo man sich vielleicht keinen Datenbank-Guru leisten mag, ist drittens der Support ein kritisches Thema. Für MySQL und Ingres spricht da zweifellos, dass kommerzieller Support direkt vom Hersteller zu haben ist. Zwar findet man auch für PostgreSQL und Firebird Systemhäuser, die die Verantwortung für eine Datenbankinstallation übernehmen, allerdings kann hier die räumliche Distanz ein hinderlicher Faktor sein. Community-Support ist für alle in guter Qualität verfügbar, allerdings ohne garantierte Reaktionszeiten.

Viertens lohnt ein Blick auf die Tools und Zusätze, die für die Datenbank der Wahl verfügbar sind. Wo auch die fortgeschrittenen Grundfunktionen kein trennscharfes Kriterium mehr sind, gewinnt die Usability an Bedeutung. Nicht immer, aber oft kommt es auch auf die Verfügbarkeit und Performance an.

Fünftens ist die mitgelieferte Dokumentation – häufig ein Stiefkind freier Software – ein Thema. Dabei offenbart sich ein sich selbst verstärkender Kreislauf: MySQL ist die meist benutzte Datenbank und daher auch die, zu der es mit Abstand die meiste Sekundärliteratur gibt. Seltenere eingesetzte Systeme haben oft nicht nur eine lückenhaftere Dokumentation, es verlegt auch niemand Bücher darüber, die das kompensieren könnten.

Unvergleichlich?

Beim Abgleich der benötigten Features mit den Angeboten hilft **Tabelle 1**. Allerdings ist dabei Augenmaß geboten. Einerseits limitiert der verfügbare Platz für die Tabelle die Darstellungstiefe. Andererseits ist eine bestimmte Eigenschaft oft nicht von vorneherein gut oder schlecht, sondern meist nur für konkrete Einsatzszenarien besser oder schlechter geeignet. Gerade MySQL zeichnet sich dadurch aus, die Feature-Wahl pro Tabelle zu bieten.

Erschwerend kommt auch hinzu, dass ähnliche Features verschieden implementiert sein können oder erst im Verbund ihren vollen Nutzen entfalten. Beispielsweise gewinnen Sperren auf Zeilenebene (Row Level Locks), wenn die Datenbank zugleich in der Lage ist, sie in den Fällen automatisch zu Sperren auf Tabellenebene zu verdichten (Lock-Eskalation), wo andernfalls ihr Overhead überproportional ansteigt.

Dennoch hilft sicher der Überblick über die wesentlichsten Features, den die Tabelle bietet. ►

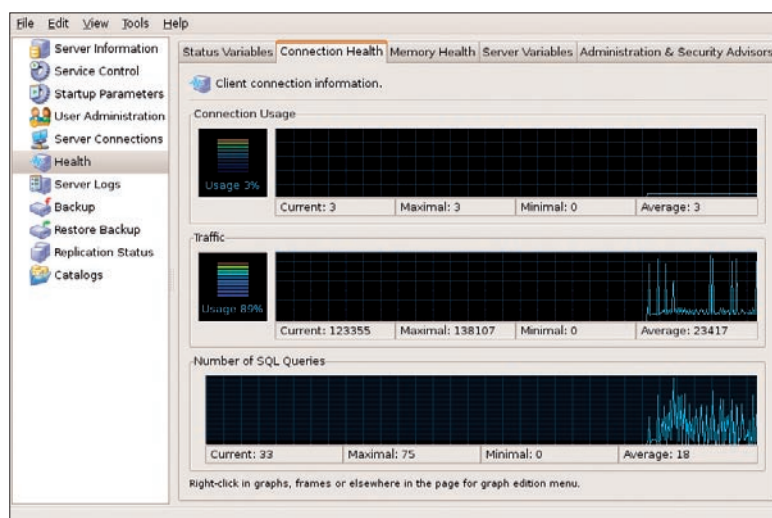
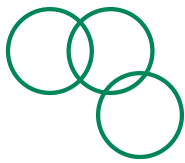


Abbildung 1: Zahlreiche GUI-Tools, hier der MySQL-Administrator, erleichtern die Arbeit mit der freien Datenbank.



■ MySQL

Die bekannteste und verbreitetste Open-Source-Datenbank ist MySQL (1). Der Hersteller – der Anfang des Jahres unter den Fittichen von Sun Microsystems gelandet ist – rechnet mit 11 Millionen Installationen weltweit und zählt bis zu 50 000 Downloads am Tag. Eine Studie der Evans Data Communication (2, 3) weist einen Bekanntheitsgrad unter Entwicklern von rund 40 Prozent aus. Auch gehört MySQL zur Standardsoftware der meisten Linux-Distributionen.

Lange Zeit als die Web-Datenbank gehandelt, die sich nur bedingt für komplexere Aufgaben eignet, legen die letzte stabile Version 5.0 und die bereits als Release Candidate verfügbare 5.1 nun das Ruder um. Neuer Kurs: Universelle Datenbank für Geschäftsanwendungen. Dafür hat das letzte Release etliche Funktionen nachgerüstet, die bisher eher dem kommerziellen Lager zuzurechnen waren: Stored Procedures, Cursor, Trigger, Views und verteilte Transaktionen etwa. Zwar fehlen noch immer wünschenswerte

Feature-Vergleich Open-Source-Datenbanken

Name	MySQL	PostgreSQL	Ingres	Firebird
Hersteller, Version, Lizenz und Support				
URL	(http://www.mysql.de)	(http://www.postgresql.org)	(http://www.ingres.com)	(http://www.firebirdsql.org)
entwickelt seit	1996	1989	1974	2000
aktuelle Version	5.0.51 (stable) 5.1.24 (RC) 6.0 (Alpha)	8.3.3	Ingres 2006 Release 3	2.1 (stable) 2.1.1 (RC) 2.5 (Alpha)
Lizenz	GPL und kommerziell	BSD	GPL 2	Interbase Public License Mozilla Public License
Plattformen	Unix, Linux, Windows, MacOS X, IBM iSeries, Symbian, OpenVMS	Unix, Linux, Windows, Netware, OS/2	Unix, Linux, Windows, OpenVMS, MacOS X	Unix, Linux, Windows, MacOS X
Kommerz. Support	Sun Microsystems	(http://www.postgresql.org/support/professional_support_europe)	Ingres Corp.	u.a. IBPhoenix (http://www.ibphoenix.com/main.nfs?a=ibphoenix&page=ibp_services)
Community Support	User Foren, Mailing Lists, FAQ	User Foren, Mailing Lists, FAQ (http://www.postgresql.org/community/resources)	User Foren, Mailing Lists, FAQ	User Foren, Mailing Lists, FAQ (http://www.ibphoenix.com)
Basic Database Features				
SQL	teils SQL99, teils 2003, teils proprietär	SQL92/99	SQL92/99	SQL92/99/2003 in Teilen
ACID-Transaktionen	ja, enginespezifisch	ja	ja	ja
Schemas,/Name-spaces für Cross-Data-base-Queries	ja	ja, seit 7.3	ja	nein
Locking	ja, Tabellen-Level, und MVCC Row-Level enginespezifisch	ja, Tabellen-Level und MVCC, Advisory Locks	Row Level mit Eskalation, MVCC geplant für Ingres 10 (2011)	Multi Generational Architecture (MGA), ähnlich MVCC; Optimistic Row-Level-Locking
Sequences	nur AUTO_INCREMENT	ja, frei definierbar	ja, frei definierbar	ja, aber nur mit Inkrement 1
Unions	ja, seit 4.0	ja	ja	ja
User Defined Functions (UDF)	ja	ja	nein	ja
Stored Procedures (SP)	ja, seit 5.0 (nur SQL)	ja, PL/pgSQL, Perl, Python, Tcl	ja (nur SQL)	ja (nur PSQL)
Volltext-Index	ja, aber kein Unicode	ja, integriert Tsearch2	nein	nein, aber diverse 3rd-Party-Tools verfügbar



Eigenschaften – beispielsweise durchgängiger Unicode-Support, geplant für MySQL 6 –, aber die größten Lücken sind geschlossen.

Bei vielen Anwendern besonders geschätzt, ist die einfache, integrierte Replikation und die eingebaute Cluster-Fähigkeit. Hier kann MySQL vor allem gegen Konkurrenten aus dem Open-Source-Lager punkten (PostgreSQL, Firebird) wo sich ähnliche Funktionen nur über umständlicher zu bedienende und schlechter integrierte 3rd-Party-Komponenten nachrüsten lassen.

Daneben unterscheidet sich MySQL durch eine ganz besondere Eigenart von allen anderen Datenbanken: Gemeint sind die Storage Engines, die zuweilen auch Tabellentypen heißen. Dabei handelt es sich um Module, die einen Teil der Datenablage übernehmen. Das Connection Management, die Security, das Caching und das Parsen der SQL-Statements verbleiben dabei im Datenbank-Kern, alles andere jedoch realisiert jeweils eine Storage Engine auf spezifische Weise. ►

Featurevergleich Open-Source-Datenbanken

Name	MySQL	PostgreSQL	Ingres	Firebird
Basic Database Features (Fortsetzung)				
Cursors	ja, seit 5.0	ja, seit 6.4	ja	ja
Triggers	ja, seit 5.0	ja	hier unter dem Namen »Rules«	ja, seit 2.1
Views	ja, seit 5.0	ja	ja	ja
Subqueries	ja, seit 4.1	ja	ja	ja
Foreign Keys	nur mit InnoDB seit 3.23, mehr ab 6.x geplant	ja	ja	ja
Advanced Database Features				
Partitionierung	ja, aber erst mit 5.1: Range-, List-, Hash- und Key-Partitionen	indirekt via Table Inheritance, entspricht RANGE-Partitionierung	ja, AUTOMATIC, HASH, LIST, and RANGE-Partitionen	nein
Replikation	ja, Row-based (Log Shifting, erst mit 5.1) und Statement-based, beides Single Master	nein, aber 3rd Party-Lösungen: Slony, PgPool, PgCluster, ... geplant für späteres Release	ja, asynchron, transaktionsbasiert	nein, aber diverse 3rd-Party-Lösungen wie Fibre, FBReplicator, ...
Clustering	ja	ja, Warm Standby	ja, setzt aber Shared Storage, Cluster-Filesystem und Distributed Lock Mgr. voraus	nein
Event Scheduler	ja, aber erst mit 5.1	nein	nein, geplant für Release 4	nein
XML Funktionen	Basic XPath-Support	XML/SQL-Support	nein	nein
User Defined Data Types (UDT)	nein	ja	nein	nein
Plugin-API	ja, aber erst mit 5.1	nein	nein	nein
External Functions (außer UDFs in SQL)	nein, nur rudimentär und undokumentiert	ja, in C und C++	nein	nein
XA Transactions	ja, seit 5.0	ja, seit 8.1	ja, für JDBC-Transaktionen	nein
Unicode-Support	nein, geplant für 6.x	ja	ja, seit Release 3: UTF-8	ja, seit 2.1
Technische Limits				
Max. DB-Größe	unbegrenzt	unbegrenzt	unbegrenzt	30 TByte
Max. Tabellengröße	4096 Columns (InnoDB 100 Columns)	32 TByte	unbegrenzt	ca. 30 GByte
Max. Zeilenlänge	64 KByte, BLOG 2 GByte	1,6 TByte	256 KByte	64 KByte
Max. BLOB/CLOB-Größe	4 GByte	1 GByte	2 GByte	2 GByte
Max. Größe Zahl	64 Bits	unbegrenzt	64 Bits	64 Bits

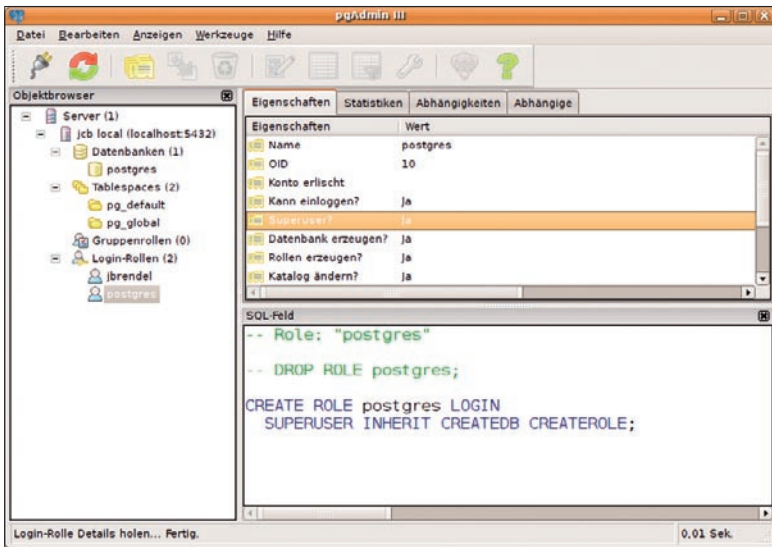


Abbildung 2: Ein beliebtes Administrationswerkzeug für PostgreSQL ist PgAdmin3. Es unterstützt Unicode, sämtliche PostgreSQL-Datentypen und ist für 30 Sprachen lokalisiert.

Das hat den Vorteil, dass der Anwender pro Tabelle festlegen kann, ob etwa der Overhead von Transaktionen hier gerechtfertigt ist, oder ob er zugunsten der Performance darauf verzichten kann. Der Nachteil ist: Keine Engine kann alles und die Feature-Kombinationen liegen fest. Wer etwa die clusterfähige NDB-Engine einsetzen möchte, der muss auf Transaktionen verzichten, die es nur zusammen mit der InnoDB-Engine gibt (beziehungsweise später mit den Neuentwicklungen Falcon und Maria geben wird).

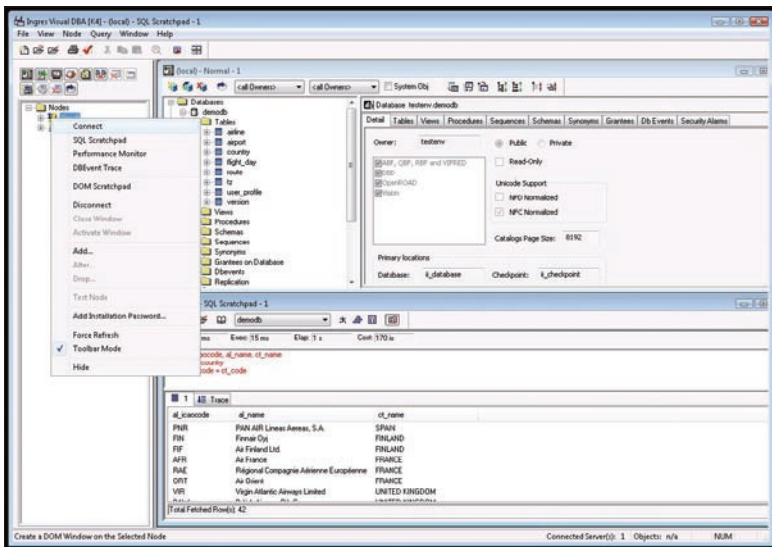


Abbildung 3: Auch Ingres kann selbstverständlich mit grafischen Werkzeugen für die Datenbankadministration aufwarten. Daneben steht dem geübten DBA aber auch die Kommandozeile offen.

■ PostgreSQL

PostgreSQL (4) entstammt einem Projekt, das in den späteren 1980er-Jahren an der Universität Berkeley einen Ingres-Nachfolger entwickeln wollte, worauf der Name anspielt (post ingres). Mitte der Neunzigerjahre gaben die Entwickler ihre Datenbank dann frei.

PostgreSQL verfügte im Vergleich zu MySQL lange über mehr Features, wie sie im professionellen Umfeld gefragt sind. Darunter Two-Phase-Commit für verteilte Transaktionen oder Multi Version Concurrency Control (MVCC) für mehr Effizienz bei konkurrierenden Zugriffen. Hier hat MySQL aufgeholt.

Auf manchen Gebieten hat PostgreSQL aber auch heute noch die Nase vor allen anderen Open-Source-Datenbanken, beispielsweise bei der Verarbeitung von Geodaten mit dem fast konkurrenzlosen Postgis (hier hält noch Ingres mit), bei besonderen und auch selbstdefinierbaren Datentypen, bei der Einbindung von Prozeduren fast beliebiger Programmiersprachen oder bei der Flexibilität der Indizierung sowie der Konformität zum SQL-Standard. Der Bekanntheitsgrad ist ebenfalls sehr hoch.

PostgreSQL ist auf vielen Plattformen zu Hause. Unter Solaris supported es offiziell sogar Sun, das den Konkurrenten MySQL kaufte. Auf einem Sun Server T2000 Firebird stellte PostgreSQL auch einen Performance-Weltrekord für Open-Source-Datenbanken in einem Java-Benchmark auf (5) Es erreichte im letzten Jahr 814 Operationen pro Sekunde nach den Standard SPECjAppServer 2004 (MySQL: 721).

Im Laufe der Zeit haben sich auf der Grundlage von PostgreSQL eine Reihe freier und kommerzieller Datenbanken etabliert, die das Ausgangsprodukt um interessante Features erweitern. Dazu gehört beispielsweise Pervasive SQL (6) oder die freie EnterpriseDB Postgres Plus (7) mit eingebauter Replikation, eigenem GUI-Tool, MySQL-Konverter und vorbereitet für Geo-daten. Das kommerzielle Pendant EnterpriseDB Postgres Plus Advanced Server (8) wirbt darüber hinaus vor allem mit seiner Kompatibilität zu Oracle und einem Werkzeug für automatisches Tuning.

Ebenfalls mit integrierter Replikation, die dem Original fehlt, wartet Mammoth Postgres + Replikator (9) auf. Auf den Bereich Data Warehousing und Business Intelligence zielt dagegen die kommerzielle Variante Bizgres (10) von Greenplum.

■ Ingres

Ingres (11) blickt auf eine lange und wechselvolle Geschichte zurück. Unter der Regie des späteren Postgres-Schöpfers Mike Stonebreaker entstand es bereits in den Siebzigerjahren des letzten Jahrhunderts als beispielhafte Implementation des damals von Edgar F. Codd neu entwickelten Konzepts einer relationalen Datenbank.

Ingres war zunächst kommerziell erfolgreich und inspirierte zahlreiche weitere Datenbankprojekte (NonStop SQL, Sybase, Informix, Microsoft SQL Server). Im Jahr 1990 kaufte zunächst die ASK Corporation Ingres, die Gründer verließen die Firma, 1994 übernahm dann Computer Associates (CA) das Projekt und bot Ingres unter verschiedenen Markennamen an.

Der Stern der Datenbank aber sank, sie verlor mehr und mehr Marktanteile, schließlich wanderten auch die Kernentwickler ab. So war die Freigabe unter einer eigenen Open-Source-Lizenz (CATOSL) im Mai 2004 eher als letzter Rettungsversuch zu interpretieren.

Er hatte Erfolg. Ingres zählte zu den ersten freien Datenbanken mit Enterprise-Features wie Partitionierung, parallelen Abfragen oder Replikation. Inzwischen haben viele Mitbewerber aufgeschlossen und solche Fähigkeiten sind kein Alleinstellungsmerkmal mehr. Ein Gebiet aber, die Verarbeitung von Geodaten, muss sich Ingres bislang nur mit PostgreSQL teilen.

■ Firebird

Firebird (12) ist Nachfahre des kommerziellen Interbase (13) und entsprang einer kurzen Liaison seiner Mutter Borland mit der Open-Source-Welt. Bei Borland hatte man Interbase 6.0 im Jahr 2000 bereits so gut wie abgeschlossen und gab es deshalb frei. Aus dieser Version entstand Firebird 1.0, noch bevor es sich Borland wieder anders überlegte. Heute existieren die Halbgeschwister nebeneinander.

Der Interbase-Vater Jim Starkey schloss sich 2006 MySQL an, um dort die neue Storage-Engine Falcon zu entwickeln. Kürzlich verließ er MySQL aber vor Projektende wieder.

Firebird ist eine für den professionellen Einsatz auf vielen Plattformen geeignete Datenbank. Ein Blick in die Feature-Vergleichstabelle zeigt allerdings, dass ihr einige der fortgeschrittenen Funktionen fehlen – etwa eine eingebaute Replikation, Clustering, XML-Fähigkeiten, verteilte Transaktionen, eigene Volltextsuche und Geo-Datentypen.

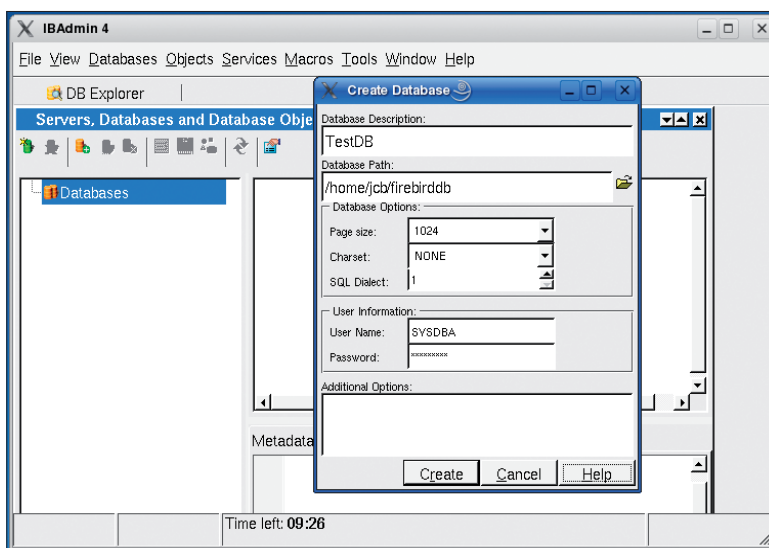


Abbildung 4: Auch Firebird kennt ein grafisches Administrationstool namens IBAdmin, das den Admin bei vielen Routinejobs unterstützt.

Dass muss kein Beinbruch sein, solange man auf keines dieser Feature angewiesen ist. Die Entwicklung trägt eine spendenfinanzierte Firebird Foundation. Support leisten die Community und eine Reihe Systemhäuser, allen voran IB-Phoenix (14).

Infos

- (1) MySQL: (<http://www.mysql.de>)
- (2) Evans Data Communication: (<http://www.evansdata.com>)
- (3) MySQL Marktanteil: (<http://www.mysql.de/why-mysql/marketshare/>)
- (4) PostgreSQL; (<http://www.postgresql.org>)
- (5) Benchmark-Resultate: (<http://www.spec.org/jAppServer2004/results/jAppServer2004.html>)
- (6) Pervasive SQL: (<http://ww2.pervasive.com/Database/Products/Pages/PSQLOverview.aspx>)
- (7) EnterpriseDB Postgres Plus: (http://www.enterprisedb.com/products/postgres_plus.do)
- (8) EnterpriseDB Postgres Plus Advanced Server: (http://www.enterprisedb.com/products/postgres_plus_as.do)
- (9) Mammoth Postgres + Replikator: (<http://www.commandprompt.com/products/mammothreplicator/>)
- (10) Bizgres: (<http://www.bizgres.org/home.php>)
- (11) Ingres: (<http://www.ingres.com>)
- (12) Firebird: (<http://www.firebirdsql.org/>)
- (13) Interbase: (<http://www.codegear.com/products/interbase>)
- (14) IBPhoenix: (<http://www.ibphoenix.com>)

