Postgres OnLine Journal: August / Sept 2009

An in-depth Exploration of the PostgreSQL Open Source Database

Table Of Contents

A Product of Paragon Corporation
http://www.paragoncorporation.com/
http://www.postgresonline.com/

PostGIS changing of the Guards

As David Page already noted, Leo and I are taking over responsibility of building PostGIS windows one-click installers/stack builder from Mark Cave-Ayland. The PostGIS 1.4 windows packaging was a little late in coming this time since it was our first and also some things changed in the PostGIS packaging for 1.4. Even so we made some mistakes such as statically compiling libproj in with the postgis-1.4.dll and forgetting some new images in the packaged html help, which we will fix in 1.4.1 release.

Mark will still be providing a supporting role and helping out when we screw up or helping us if we run into compile issues as we go along so he's not going away; he will be a great safety net. When Mark started his role a long time ago, he was as many would like to say "Very entrenched in the dark side," and over the years, he has seen the light. As a result, these moments of catching issues in the PostGIS release cycle that effect windows users such as troubleshooting the memory bug in the loader files that affected Windows Vista users and testing on various Windows OS, has fallen on us, because well we have access to all windows os.

It also became painful for Mark to walk in the shadow of darkness once he had seen the light. Luckily we are still windows addicts so this having to constantly test on Windows and building for Windows is something we would naturally do anyway and yes as shocking as it sounds we do run some production PostgreSQL apps on windows and it works pretty well, thank you very much. We don't expect this to change any time soon.

As part of this change, we hope to provide more interim windows builds of PostGIS so windows users can experiment with future releases before they come out. Yes compiling on windows is a tad bit more difficult than on Linux. These PostGIS windows experimental builds can be found http://www.postgis.org/download/windows/experimental.php

Main changes in PostGIS

- The PostGIS steering committee has agreed to be good and not be adding new functions between micro releases of PostGIS as we have done in the past and as we've been smacked around for. As part of that change, from PostGIS 1.4 moving forward each micro version will overwrite the previous micro version in the MS Windows registry. E.g. 1.4.1 will overwrite 1.4.0 so no need to uninstall the old and reinstall to get rid of registry junk. Just install on top of your existing 1.4.
- As of PostGIS 1.4 it is possible to run different versions of PostGIS in different databases on teh same PostgreSQL server install since the .so/.dll from minor to minor have unique names (naming is postgis-1.4.so (postgis-1.4.dll), postgis-1.5.so etc). This is mostly useful for testing and comparing different versions of PostGIS before you officially upgrade and if you have several different spatial apps using different databases, you don't risk breaking them all at once.
- PostGIS is now an official incubation project of OSGEO. Things are still being drafted. But I guess that means our PostGIS defacto steering committee composed of Kevin, Paul, Mark, and myself is now more or less official.

On the topic of Writing

We just submitted first revision of chapter 6 of our upcoming book PostGIS in Action and are working on our second revision of that chapter. The second revision will most likely be split into 2 chapters because its too long for a single chapter. It will cover loading/exporting various format types of spatial data and everything you never wanted to know about spatial reference systems.

We also submitted the second revision of the upcoming RefCardz DZone **Essential PostgreSQL**. Not sure when that will be released since we are already a bit behind schedule. I imagine we will have a couple more iterations of that before its done. It is really hard to cram into six pages the most important things that every PostgreSQL user should know and all the exciting stuff in PostgreSQL without feeling like you are leaving something critical out.

I'm always amazed how people manage to write books by themselves. If you think for each chapter you go thru 4-5 revisions and each subsequent revision is about 20-50% of the previous, that's a lot. Leo and I are writing together and its still a struggle though its been fun. I guess writing together has its own challenges because you are constantly cutting each others stuff out and arguing over what is important and what is not. We have managed to not have too many heated debates that would spoil dinner. This is good since Leo does the cooking because he thinks I'm too absent-minded to be capable of complex tasks like cooking.

Back to Table Of Contents

From the Editors

CodePlex Foundation

Just read that Microsoft has formed a new foundation called CodePlex foundation, presumably to spinoff their Code plex site and allow it to stand separately from Microsoft. The mission appears to be to allow an easier avenue for developers working for proprietary software companies to contribute to open source projects.

Monty has some details about this on his blog The CodePlex Foundation: Why is Microsoft founding it?.

The line up of people they have on their advisory board (including Monty) and board of directors is interesting CodePlex About. I'm particularly happy that Miguel De Icaza is on the board since he is one of my favorite people and I believe shares my pragmatic ideals on the synergy between open source and non-open source software. I wonder what it takes to get on this board. It would be really nice if someone in the PostgreSQL community were on this board just to ensure the needs of the PostgreSQL community (especially our growing number of windows users) is well represented.

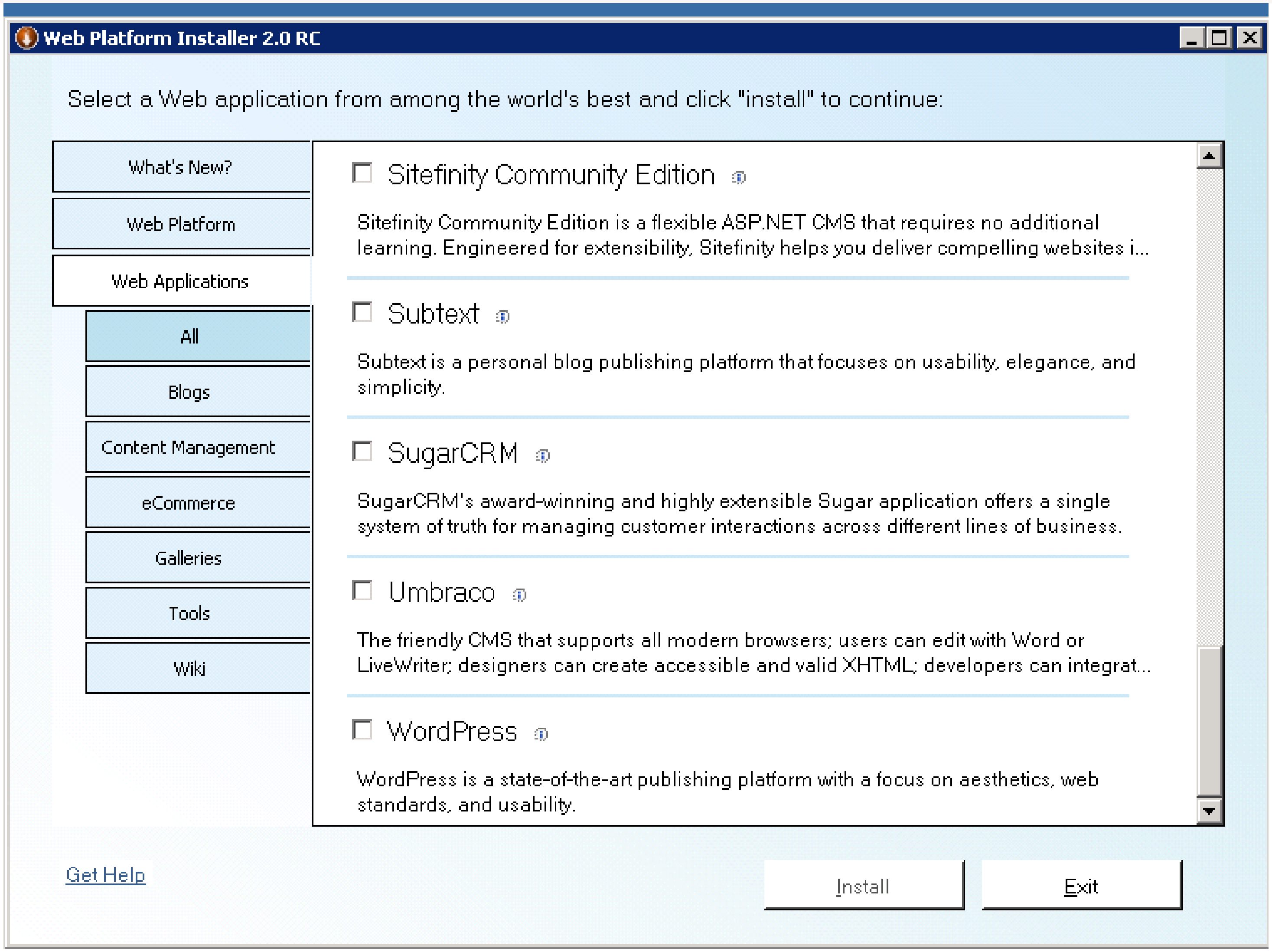
As to the argument of Monty's that software for sell is dying, not sure I quite agree though haven't given it much thought. Certainly I would like to think that service for sell is rising since that's the business we are in and enjoy most. One thing I believe is that software is getting more complicated and people expect more. With that said even as a company that sells software, you would be foolish not to try to leverage on the open source software out there that fits nicely into your codebase. You just won't be able to compete even with the sole proprietor next door who is *with it*.

Microsoft reinventing itself

First I would first like to give this caveat that yes I am a pro Microsofter. Always was, and really can't see myself changing. Over the years I have seen myself change along with the face of Microsoft. I no longer use just Microsoft products, but partake too of all the good things that the world has to offer, much of which thank goodness is free in the sense of not costing as well as not having restrictive uses, and fitting often more nicely with our clients codebase and general requirements than closed source software.

I have observed over the years, that Microsoft has been embracing PHP and basically trying to make the world forget about ASP. This all seemed puzzling to me at first and then realized it makes perfect logistical sense. ASP is a dead product and even as the owner of said dead product, you want to get your customer off of it as quickly as possible so you don't have to waste money supporting it. That is why the PostgreSQL and other communities push people to upgrade. All those die-hard ASP fans hated ASP.NET. ASP.NET was too complicated for their needs. Ironically the transition for many people from ASP to PHP was a much easier one than from ASP to ASP.NET. PHP also had the advantage of running pretty consistently on Linux as it did on Windows. Work still needs to be done to allow ASP.NET that luxury. I happen to like both for very different reasons and workflows.

Last week I noticed what appears to be an acceleration in whatever company Microsoft is turning into. I wanted to try out the new SQL Server 2008 Reporting Services by downloading the free SQL server 2008 Express with Advanced Services. Last time I downloaded it the install of SQL Server 2008 express was a bit painful, but this time I spent a lot of time puzzled. You see I had to download this thing called a Web Platform Installer. It recognized the dependencies I was missing and dowloaded it for me. What puzzled me however were these screens.



Web Platform Installer 2.0 RC

Select a Web application from among the world's best and click "install" to continue:

□ Sitefinity Community Edition

Sitefinity Community Edition is a flexible ASP.NET CMS that requires no additional learning. Engineered for extensibility, Sitefinity helps you deliver compelling websites i...

□ Subtext

Subtext is a personal blog publishing platform that focuses on usability, elegance, and simplicity.

□ SugarCRM

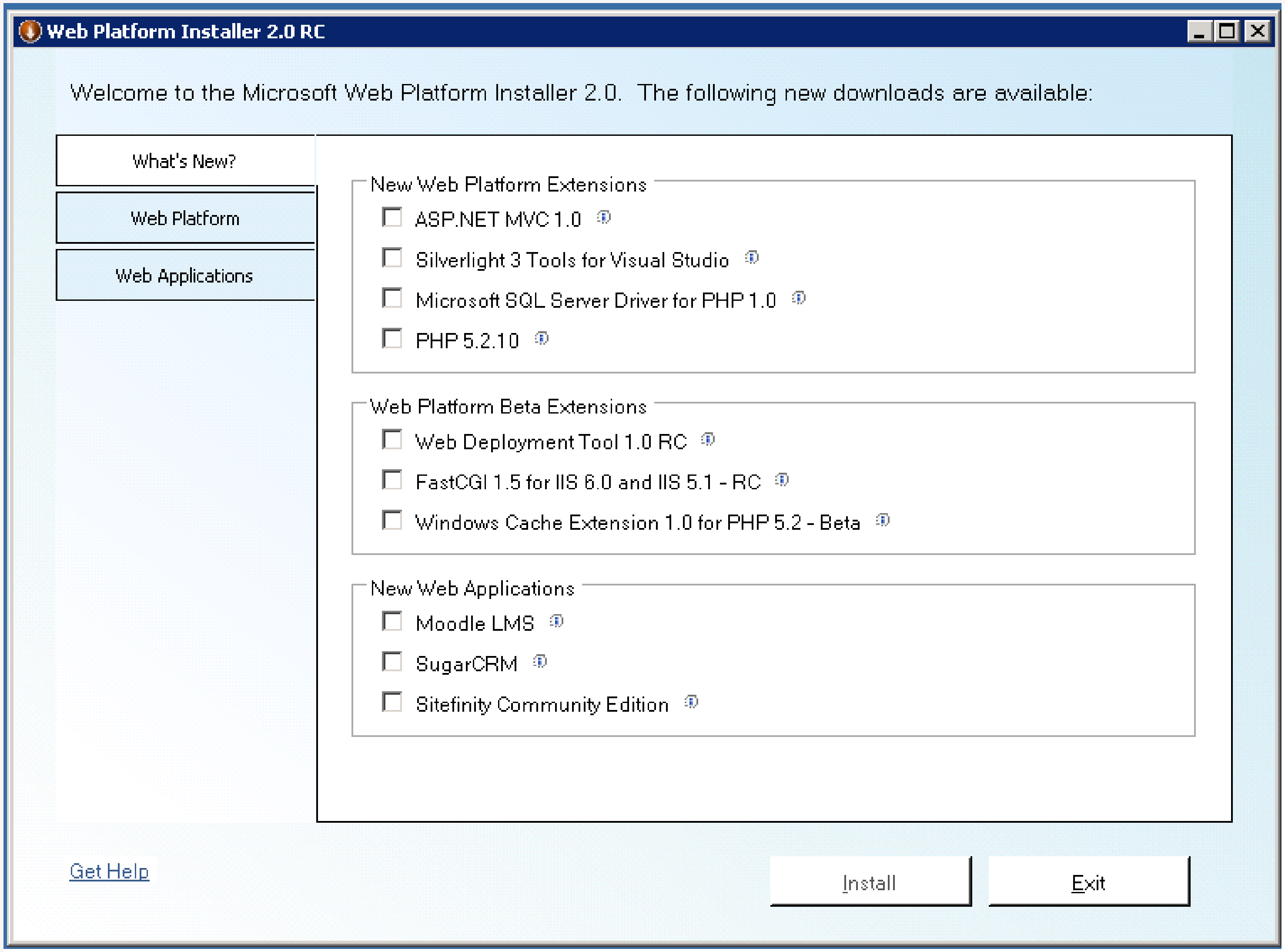
SugarCRM's award-winning and highly extensible Sugar application offers a single system of truth for managing customer interactions across different lines of business.

□ Umbraco

The friendly CMS that supports all modern browsers; users can edit with Word or LiveWriter; designers can create accessible and valid XHTML; developers can integrat...

□ WordPress

WordPress is a state-of-the-art publishing platform with a focus on aesthetics, web standards, and usability.



Welcome to the Microsoft Web Platform Installer 2.0. The following new downloads are available:

New Web Platform Extensions

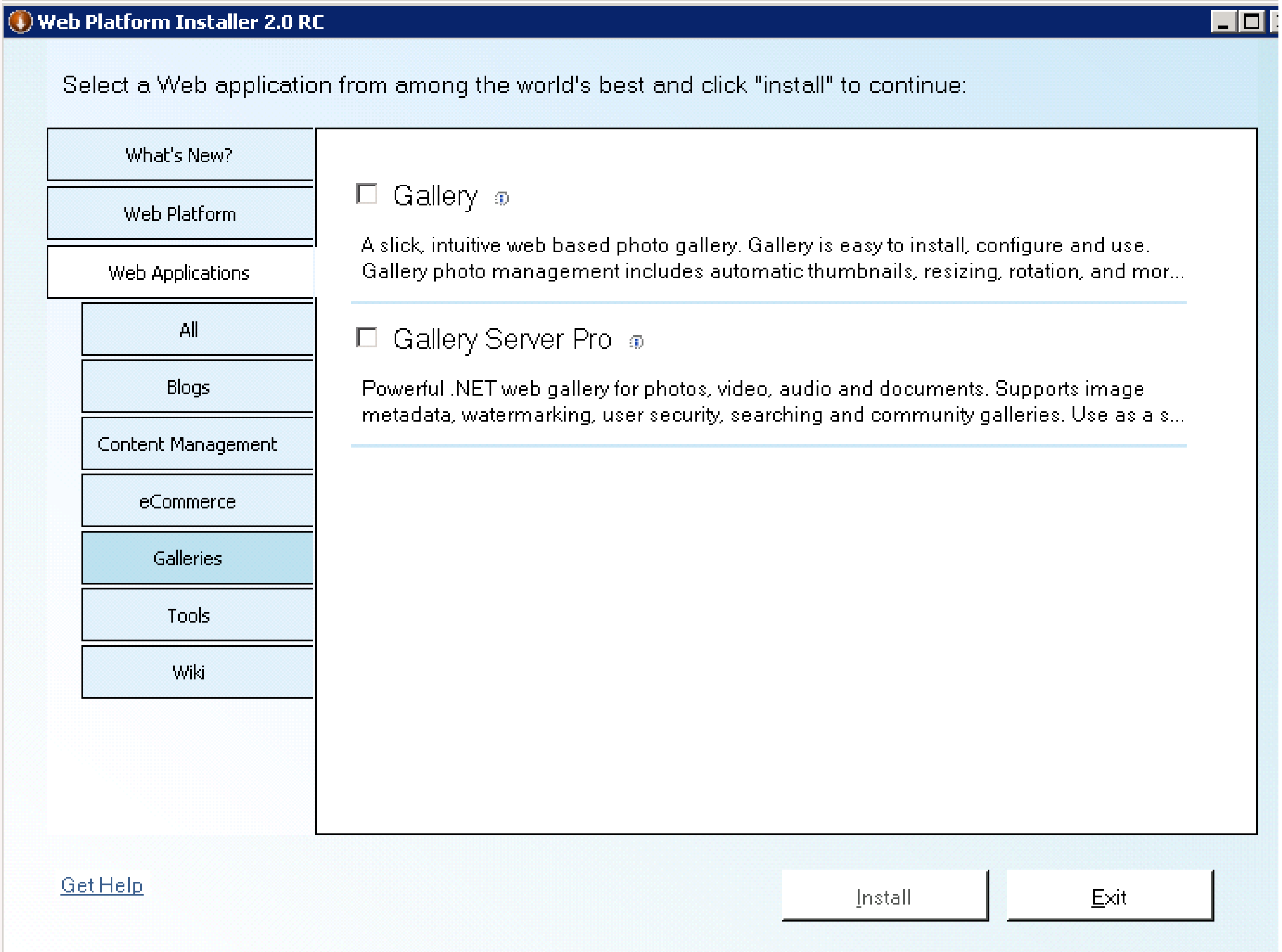
- □ ASP.NET MVC 1.0
- □ Silverlight 3 Tools for Visual Studio
- □ Microsoft SQL Server Driver for PHP 1.0
- □ PHP 5.2.10

Web Platform Beta Extensions

- □ Web Deployment Tool 1.0 RC
- □ FastCGI 1.5 for IIS 6.0 and IIS 5.1 - RC
- □ Windows Cache Extension 1.0 for PHP 5.2 - Beta

New Web Applications

- □ Moodle LMS
- □ SugarCRM
- □ Sitefinity Community Edition



Select a Web application from among the world's best and click "install" to continue:

□ Gallery

A slick, intuitive web based photo gallery. Gallery is easy to install, configure and use. Gallery photo management includes automatic thumbnails, resizing, rotation, and mor...

□ Gallery Server Pro

Powerful .NET web gallery for photos, video, audio and documents. Supports image metadata, watermarking, user security, searching and community galleries. Use as a s...

The strange thing is that a lot of the PHP side of products are MySQL centric and don't even work with SQL Server to my knowledge and much of it doesn't work with PostgreSQL (except for Moodle, Gallery, the variant of Drupal they packaged Acquia Drupal - claims to only support MySQL. What happened to plain Drupal that supports PostgreSQL). So on the one hand I was glad that Microsoft was embracing open source and on the other hand I was irritated by the choice of offerings. I'm not sure if I can blame Microsoft for the lack of applications I can actually use. I'm sure a poll was done and this is what people commonly use and being customer centric as they are that is what they put out.

Back to Table Of Contents CodePlex Foundation Reader Comments

Terminating Annoying Back Ends ***Beginner***

One of the small little treats provided in PostgreSQL 8.4 is the new **pg_terminate_backend** function. In the past when we wanted to kill runaway postgresql queries issued by a database or user or hmm us, we would call the pg_cancel_backend function. The problem with that is it would simply cancel the query in the backend process, but often times the offending application would simply launch the same query again.

In PostgreSQL 8.4 a new function was introduced called **pg_terminate_backend**. This doesn't completely replace pg_cancel_backend, but basically does what you do when you go into say a Windows Task manager and kill the offending postgres process or on Linux, you call a kill command on a postgres process. Its nicer in the sense that you can do it all within PostgreSQL and you can use the pg_stat_activity query to help you out a bit. Also you don't run the risk as easily of killing the root postgres process and killing the postgres service all together.

Below are the ways we commonly use these functions. One of the features I really love about PostgreSQL which I miss when working with SQL Server, is the ability to call a function that does something from within a query. This feature makes SQL so much more powerful.

Cancel/Termination Examples

Slap on Wrist

```
 -- Cancel all queries in an annoying database
    SELECT pg_cancel_backend(procpid)
    FROM pg_stat_activity
    WHERE datname = 'baddatabase';

    -- Cancel all queries by an annoying user
    SELECT pg_cancel_backend(procpid)
    FROM pg_stat_activity
    WHERE usename = 'baduser';
```

Baseball bat to the head

Terminating backends is also useful for freeing up memory from idle postgres processes that for whatever reason were not released or if you need to rename a database and need to kill all live connections to a database to do so.

```
-- terminate process by annoying database
    SELECT pg_terminate_backend(procpid)
    FROM pg_stat_activity
    WHERE datname = 'baddatabase';

-- terminate process by an annoying user
    SELECT pg_terminate_backend(procpid)
    FROM pg_stat_activity
    WHERE usename = 'baduser';
```

One thing we have noticed with the baseball approach to database management is that it doesn't always play well with pooled connection like things. For example in one PHP app we had that uses pooled connections, the connections became stale. Or at least we think this is the culprit. So you sometimes have to restart the app, or it sometimes gives annoying messages to the user until the dead connections are released. With that said, we still try the slap on the wrist before reaching for the baseball bat. Its always nice to have an easy to swing baseball bat handy though.

Back to Table Of Contents

Basics

Cross Compare of PostgreSQL 8.4, SQL Server 2008, MySQL 5.1

Comparison of PostgreSQL 8.4, Microsoft SQL Server 2008, MySQL 5.1

In our May 2008 issue of Postgres OnLine Journal, we cross compared Microsoft SQL Server 2005, MySQL 5, and PostgreSQL 8.3. Some people mentioned well since 8.4 has now come out, shouldn't we go back and update the reference. We deliberated and decided not to. To be fair all 3 products have released new versions, so it would seem unfair to compare a newer PostgreSQL against older versions of MS SQL Server and MySQL. We have therefore decided to repeat our exercise and include parts people felt we should have covered, as well as comparing the latest and greatest stable release of each product.

People ask us time and time again what's the difference why should you care which database you use. We will try to be very fair in our comparison. We will show equally how PostgreSQL sucks compared to the others. These are the items we most care about or think others most care about. There are numerous other differences if you get deep into the trenches of each.

People have been posting comments on Reddit as well

Feature	Microsoft SQL Server 2008	MySQL 5.1	PostgreSQL 8.4
OS - Why is this important? Why would you even dream of not running on Windows? If you decide one day that Microsoft is not your best friend in the whole wide world, you can ditch them or at least on your DB Server (could that ever happen?). On a side note, Microsoft can't compete with Oracle on Linux/Unix anyway. If Microsoft has a non-Microsoft DB running on a customer's box, I wonder which database they would prefer - Oracle, IBM DB2, Sun MySQL or PostgreSQL?	Windows Desktop/Server	Windows Desktop/Server , Linux, Unix, Mac	Windows Desktop/Server, Linux, Unix, Mac
Licensing	Commercial - Closed Source, Various levels of features based on version, Free Crippleware (4 GB limit but free for embedding inclusion)	GPL Open Source, Commercial. Now owned by Oracle, and some of us are still wondering how Oracle will make hay out of its new found treasure. Will they use to upsell. Monty Widenius has an interesting blog entry explaining the MySQL dual licensing and how it has changed.	BSD Open Source. Its the freest of all and many argue about it. You can fork it all you want without giving back and make your own commercial derivative. Though ironically there are not all that many forks. Netezza forked it initially (though its unclear if any PostgreSQL code lives on in their product), GreenPlum forked it, EnterpriseDb forked to give Oracle features, but to their credit gives back a lot of fixes to the community.

Install/Maintenance Process	Most time-consuming to install and most dependencies, but lots of wizards to help you forget everything and mail you when somethng bad happens (non-free versions)	Still Easiest. There are a lot of prepackaged products with it and its install rarely fails.	Medium (I put PostgreSQL at medium because on occasion we have run into issues where we have manually init the db on Windows. Those are rare though. On Linux its still the deal that MySQL often comes pre-installed for you and for PostgreSQL you have to figure it out yourself or they give you some super antiquated version. With PostgreSQL Yum respository its much easier now if you manage your own box.
Drivers already installed on Windows	Yes - when you have a windows shop this is huge especially when you are not allowed to install stuff on client desktops and you need to integrate seamlessly with desktop apps. This is why using SQL Server Linked Server to get at yummy features of PostgreSQL comes in handy.	No	No
ODBC, JDBC, ADO.NET drivers available	Yes	Yes	Yes
Read-Only Views	Yes	Yes	Yes
Open Source products available for it	Few except CodePlex/.NET	Many	Few but ramping up and in PHP more than SQL Server
Commercial	Moderate -- still a lot of commercial stuff hasn't been certified on 2008 because 2008 is so new	Moderate	Moderate -- I think PostgreSQL has improved since last we checked, but probably still lower than MySQL penetration.
Updateable Views	Yes - even for 2 table views will automatically make them updateable if they have keys and update does not involve more than one table. You can write instead of triggers against more complex views to make them updateable	Yes - Single one table views are automatically updateable, some 2 table views are updateable if they don't have left joins and don't involve update of more than one table. If you have more complex views you want to make updateable - good riddance - no support for triggers or rules on views.	Yes, but not automatic. You have to write rules against views to make them updateable but can make very complicated views updateable as a result.
Materialized/Indexable Views	Yes but varies slightly depending on if you are running SQL Express, Workgroup, Standard, Enterprise and numerous restrictions on your views that makes it of limited use	No	No, but there are I think 2 contrib modules e.g. matviews that are simple and basically rebuild the materialized view
Can add columns and change names, data types of views without dropping	Yes	Yes	Yes - sort of - as of 8.4 you can now add columns to the end of view without dropping it.
Can drop tables, (drop, change size, data type of columns), and views used in views - this is a arguably a misfeature but sometimes it comes in handy when you are an EXPERT user :)	Yes - (but if you schema bind your tables and views, you can not drop dependent objects so this does appear to be the best of both worlds)	Yes - yikes!	No

Graphical Query/View Designer (e.g. you can see tables and select fields drag lines to do joins) included no additional charge. As people pointed out there are lots of commercial and free tools that will do this for you. We'll provide a listing and brief summary of features etc. of some of these in our product showcase later.	Yes via SQL Management Studio and Studio Express and pretty nice.	No	Yes. As of PgAdmin III 1.9, but its kind of hockey -- doesn't do JOINS right.
Computed Columns	Yes - but we still like using Views more except when we really need the computed column indexed and often we just do triggers. Computed columns are of very limited use since they can't hold roll-ups.	No - but looks like its slated for future release	No - but PostgreSQL has functional indexes so just use a view.
Functional Indexes - indexes based on a function	No - but you can create a computed column and create an index on it	No	Yes
Partial Indexes - e.g. you want to create a unique index but only consider non-null values	Yes - as of SQL server 2008 See Tom's notes and called Filtered Indexes.	No	Yes
ACID compliance - do I dear say this is sometimes over-rated - not all data is created equal and sometimes bulk-insert speed is more important than ACID	Yes	Some storage engines e.g. InnoDB, PBXT (see comments from Giuseppe Maxia) and (not MyISAM)	Yes
Foreign Key - Cascade Update/Delete	Yes	InnoDB and not MyISAM	Yes
Multi Row value insert	Yes	Yes	Yes
UPSERT logic - where you can simultaneously insert if missing and update if present	Yes via MERGE UPDATE	Yes - via INSERT IGNORE, REPLACE INSERT ON DUPLICATE UPDATE	No
Replication - haven't used much except for SQL Server so this is mostly hear-say	Yes - all sorts - log shipping, mirroring, snapshot, transactional and merge etc. and can even have non-SQL Server windows-based subscribers. Its still a bear to get working the way you want it and makes making structural changes difficult. Built-In	Yes - including master-master (built-in) See comments below and from numerours reports a big selling point of MySQL.	Yes but from reports seems to be the least polished of the bunch, although numerours third-party options to choose from that are both free and non-free. PostgreSQL 8.5 or higher is slated to have built-in replication. Sorry guys this did not make it in. Slony is still used for replication, and many like it but find it finicky and harder to use than MySQL.

Can program stored procs/ functions in multiple languages	Yes - In theory any language that complies with CLR -e.g VB.Net, C#, IronPython - but you need to compile into a dll first and then load the dll into the database. The dll is stored as part of the database and the dependencies registered in the SQL Server GAC - a real PITA if you have lots of these dependencies than are non-standard.	No (except C and Pl/SQL)	Yes - PostgreSQL just does it the cool way (common ones PL/PgSQL, sql, PL/Python, PL/Perl, PL/R) - we like having our code right there where we can see what it is doing. Downside server must host the language environment. It now supports variadic functions similar to Oracle. Neither MySQL nor SQL Server support that.
Can define custom aggregate functions	Yes - any .NET language, but not TRANSACT SQL. Why is Transact-SQL thrown out to dust like this?	Yes but only in C as UDF	Yes - any PL language and built-in C, SQL, PLPgSQL.
Triggers	Yes	Yes	Yes
Table Partitioning	Yes - only Enterprise version - functional, range	Yes	via Table Inheritance, Constraint Exclusion, RULES and Triggers - basically RANGE. Issues with using foreign-key constraints with inherited tables in 8.4 the constraint_exclusion has another option called "partition" which is the new default. Which basically means you can have constraint_exclusion for partitioning and not have your other queries suffer.
Can write Set/Table returning functions that can be used in FROM clause	Yes	No	Yes
Support creation of functions - e.g. CREATE FUNCTION	Yes	Yes	Yes
Support creation of stored procedures - e.g. CREATE PROCEDURE	Yes	Yes	Sort-Of - CREATE FUNCTION serves the same need
Dynamic and action SQL in functions	No - but you can in Stored procedures but you can't call stored procs from SELECT statements so much more limiting than PostgreSQL	No, but can in Stored procedures which aren't callable from SELECT statements so more limiting than PostgreSQL	Yes! - you can do really cool things with action functions in SELECT statements
Graphical Explain Tool - no additional charge	Yes - SQL Management Studio/ Express	No (someone on Reddit mentioned maatkit visual explain for MySQL. This is still a text format though and not quite as pretty as SQL Server or PgAdmin III graphical explain plan. Are there others?	Yes - PgAdmin III
Job Scheduling Agent controllable from DB Manager client, for running batch sql and shell jobs - no additional charge (not CronTab)	Yes - SQL Agent (not for Express), administer via Management Studio. Can do sql, sql maintainence plans, batch scripts, and SSIS work flows. Its still the best. Has wizard for setting up maintenance plans.	Yes - though appears can only use it for MySQL sql calls.	Yes - PgAgent and can run postgresql sql as well as batch scripts. Administrated via PgAdmin III.

Access tables from other databases on same server	Yes - server.db.schema.table, can even access disparate data sources via linked server or open query	Yes - db.table, but not easily across servers. Across servers you need Federated storage engine See Rob Wultsch note	Sort of - via Dblink, but much less elegant than MSSQL and MySQL way and much less efficient. Can also access disparate data sources via DBI Link
Case-Insensitivity - e.g. LIKE 'abc%' and LIKE 'ABC%' mean the same thing	By default its not case sensitive, but can change this down to the column level.	It is not case-sensitive by default but depends on character set (see comments from Giuseppe Maxia)	By default is case-sensitive , but in 8.4 we have newer contrib citext integrated to define case insensitivity fields.
Date Time support	Finally they support plain date - Date, DateTime,DateTimeOffset. (Date and DateTimeOffset are new)	Date and DateTime but none with Timezone, but you can have timezone, see Rob Wultsch comments.	(Seems pretty much on par with SQL Server) - Not much changed - Date, TimeStamp and TimeStamp with Timezone (not to be confused with MySQL's timestamp which autoupdates or SQL Server's deprecated timestamp which is a binary). Has Interval which neither MySQL nor MS SQL Server have.
Authentication	Standard Db security and NT / Active Directory Authentication	Standard Db with table-driven IP like security	Extensive - standard, LDAP, SSPI (can tie in with Active Directory if running on NT server, but still not quite as nice as SQL Server seamless integration), PAM, trust by IP, etc.
Column Level Permissions	Yes	Yes	Yes (introduced in 8.4)
DISTINCT ON	No	No	Yes
WITH ROLLUP	Yes	Yes	No
WITH CUBE	Yes	No	No
Windowing Functions OVER.. PARTITION BY	Yes	No	Yes - and its way better than SQL Server 2008
Common Table Expressions and Recursive queries	Yes	No	Yes
COUNT(DISTINCT), AGGREGATE (DISTINCT)	Yes	Yes	Yes
OGC Spatial Support - for the My dad is better than your dad fight in the GIS world between SQL Server and PostgreSQL/ PostGIS check out A look at PostgreSQL and ArcSDE, Also check out our companion critque of the 3 spatial offerings	Yes - now built-in, but we aren't allowed to provide benchmarks for obvious reasons. If you use it, you'll really want to install the SQL Server 2008 Spatial Extension tools. The upcoming SQL Server 2008 R2 (currently in CTP is supposed to have Report Builder with map integration features which should be interesting). It has geodetic which PostGIS does not. SQL Server 2008 and PostGIS have pretty identical commercial support for spatial, but PostGIS still has a much larger Open source tool belt following.	Yes - MBR mostly and spatial indexes only work under MyISAM. Limited spatial functions. Some commercial (MapDotNet, Manifold.net), Open source GIS tools gaining steam but still more behind PostGIS.	Yes - PostGIS is great and lots of spatial functions and fairly efficient indexing and lots of open source and commercial support - ESRI ArcGIS 9.3, MapInfo, Manifold, CadCorp, FME , no geodetic but expect the first version of geodetic in PostGIS 1.5, and fairly robust geodetic in PostGIS 2.0.
Schemas	Yes	No (technically MySQL is implemented as a single db with schemas -- according the the information_schema schema, though in practice its not quite as clear cut as Oracle)	Yes

CROSS APPLY	Yes	No	No but can for the most part simulate by putting set returning functions in SELECT clause. As of 8.4, all set returning functions can be used in the SELECT regardless of language it is written in.
LIMIT .. OFFSET	No - has TOP and ansi compliant ROW_NUMBER() OVER (ORDER BY somefield) As Row --- where ..Row >= ... AND Row <= ... which is much more cumbersome to use	Yes (no ansi compliant way)	Yes, and also supports the ansi compliant ROW_NUMBER() OVER (ORDER BY somefield)
Advanced Database Tuning Wizard	Yes - SQL Management Studio recommends indexes to put in etc. Very sweet. NOT available for Express or Workgroup.	No	No
Maintenance Plan Wizard	Yes via SQL Management Studio - Workgroup and above. Very sweet. Will walk you thru creating backup plan, reindexing plan, error checking and schedule these for you via SQL Agent	No	No
Pluggable Storage Engine	No	Yes	No
Correlated Subqueries	Yes	Yes	Yes
Query Planner for complex queries (like doing correlated joins, lots of joins, lots of aggregates etc)-- the thing that figures out how to navigate data based on SQL Statement and histograms and stuff. This is off the cuff rating and varies based on kinds of queries you write. For the joe blow blog or CMS or plain read SELECt ... FROM, this is probably not important and all 3 will perform adequately. We do a lot of statistical and financial apps where ability to run complex queries against millions of records in under 5 seconds is important.	Moderate (but supports parallel processing out of the box).	Sucks Okay this was a spatial analysis and we can argue why beat a dead horse. But this is just a bounding box query.	Best. PostgreSQL doesn't support parallel processing out of the box, but supports shared reads and with GridSQL (which we haven't tried), you do get parallelism
FullText Engine - all 3 have it, but we don't feel right comparing since we haven't used each enough to make an authoritative comparison. Its annoying there is no set standard for doing Full Text SQL queries	Yes	Yes	Yes

Sequences /Auto Number	Yes - via IDENTITY property of int field	Yes - via AUTO_INCREMENT of int field	Yes - via serial data type or defaulting to next Sequence of existing sequence object - this is better than MySQL and SQL Server simple auto_increment feature. The reason it is better is that you can use the same sequence object for multiple tables and you can have more than one per table. In the past PostgreSQL sequence was a pain but now you just create it with data type serial if you want it to behave like SQL Server and MySQL and it will automatically drop the sequence if you drop the table it is bound to.
Transactional DDL - ability to rollback CREATE, ALTER etc statements	Yes (I couldn't find any documentation on this, but I tested it and it correctly rolls back). There is a caveat that can't roll back DDL within a DDL trigger.	No	Yes - see this comparative analysis

Back to Table Of Contents Cross Compare of PostgreSQL 8.4, SQL Server 2008, MySQL 5.1 Reader Comments

Basics

Using Recursive Common table expressions to represent Tree structures *Intermediate*

A very long time ago, we wrote an article on how to use PostgreSQL to show the fully qualified name of an item in an inventory tree. Basically we were modeling a paper products tree. The original article can be found here Using PostgreSQL User-Defined Functions to solve the Tree Problem and was based on PostgreSQL 7.4 technology.

We'll repeat the text here for completeness and demonstrate the PostgreSQL 8.4 that solves this and more efficiently.

The Problem

Suppose you are tracking supplies and have a field called si_item and another called si_parentid. The parent keeps track of what subclass a supply item belongs to. E.g. you have paper parent that has subclasses such as recycled, non-recycled. When someone takes supplies, you want to return the fully qualified name e.g. Paper->Recycled->20 Lb

Below is what the structure of your table looks like.

si_id int, si_parentid int, si_item. In your table are the following entries

si_id	si_parentid	si_item
1		**Paper**
2	1	Recycled
3	2	20 lb
4	2	40 lb
5	1	Non-Recycled
6	5	20 lb
7	5	40 lb
8	5	Scraps

Solution

```
CREATE TABLE supplyitem(si_id integer PRIMARY KEY, si_parentid integer, si_item varchar(100));

--load up the table (multirow constructor introduced in 8.2)
INSERT INTO supplyitem(si_id,si_parentid, si_item)
VALUES (1, NULL, 'Paper'),
(2,1, 'Recycled'),
(3,2, '20 lb'),
(4,2, '40 lb'),
(5,1, 'Non-Recycled'),
(6,5, '20 lb'),
(7,5, '40 lb'),
(8,5, 'Scraps');

--Recursive query (introduced in 8.4 returns fully qualified name)
WITH RECURSIVE supplytree AS
(SELECT si_id, si_item, si_parentid, CAST(si_item As varchar(1000)) As si_item_fullname
FROM supplyitem
WHERE si_parentid IS NULL
UNION ALL
SELECT si.si_id,si.si_item,
     si.si_parentid,
     CAST(sp.si_item_fullname || '->' || si.si_item As varchar(1000)) As si_item_fullname
FROM supplyitem As si
     INNER JOIN supplytree AS sp
     ON (si.si_parentid = sp.si_id)
)
SELECT si_id, si_item_fullname
FROM supplytree
ORDER BY si_item_fullname;
```

Result looks like

```
si_id |      si_item_fullname
------+----------------------------
 1    | Paper
 5    | Paper->Non-Recycled
 6    | Paper->Non-Recycled->20 lb
 7    | Paper->Non-Recycled->40 lb
 8    | Paper->Non-Recycled->Scraps
 2    | Paper->Recycled
 3    | Paper->Recycled->20 lb
 4    | Paper->Recycled->40 lb
```

Back to Table Of Contents Using Recursive Common table expressions to represent Tree structures Reader Comments

Database Administration, Reporting, and Light application development *Beginner*

One of the most common questions people ask is **Which tools work with PostgreSQL**. In a sense the measure of a database's maturity/popularity are the number of vendors willing to produce management and development tools for it. Luckily there are a lot of vendors producing tools for PostgreSQL and the list is growing. One set of tools people are interested in are Database administration, ER diagramming, Query tools, and quickie application generators (RAD).

For this issue of our product showcase, we will not talk about one product, but several that fit in the aforementioned category. All the listed products work with PostgreSQL and can be used for database administration and/or architecting or provide some sort of light reporting/rapid application building suite. By light reporting/application building, we mean a tool with a simple wizard that a novice can use to build somewhat functional applications in minutes or days. This rules out all-purpose development things like raw PHP, .NET, Visual Studio, database drivers etc. Things we consider in this realm are things like OpenOffice Base and MS Access. Most of these tools are either free or have 30-day try before you buy options.

You can't really say one tool is absolutely better than another since each has its own strengths and caters to slightly different audiences and also you may like the way one tool does one important thing really well, though it may be mediocre in other respects. We also left out a lot of products we are not familiar with and may have gotten some things wrong.

If we left out your favorite product and you feel it meets these criteria, or you feel we made any errors, please let us know, and we'll add or correct it. We will be including Free open source as well as proprietary products in this mix. If we left out what you consider an important criteria, please let us know and we'll try to squeeze it in somewhere.

UPDATE: We have added more entries since we initially published this. In playing around with some of these, we discovered that more than we thought sport a drag and drop table draw line join kind of query builder, but some do it better than others. The same holds true with ER modeling relational foreign key gui dialogs within the relational diagram. Since our list is now bigger, we went back to test drive some of these to see how well they do it for PostgreSQL. So in these cases, you'll see more than a Yes/No answer and some of our personal prefences such as JOIN syntax implementation may show more than we would like.

The query builder part is probably something we are more critical of, not because we care that much for them, but that's the first thing new database users need badly. From an expert database user stance, this is only really useful if you can toggle back and forth between design and SQL view without losing your changes.

There were some also added features we noticed such as code completion and plpgsql debugger which we didn't test, but tried to put in the general admin/edit description section if we know it does it or someone commented on it.

We may have also incorrectly tagged some things No if it wasn't intuitively obvious how to do it in our 5-10 minutes of testing, so please correct us in those cases and how you do it as well.

RAD stands for Rapid Application Development - we are only considering tools Yes if they include a wizard to build.

Product	OS Support	Price (license)	Administration / Data Edit / Database Programming	DB Support	WYSIWIG Query Builder	Graphical Explain	RAD	Reporting	Data Import/ Export	Relational Designer / ER Modeler

Aqua Datastudio 7.5	Windows, Linux, MacOSX, Solaris	Proprietary ($400) check feature matrix feature matrix	Yes - also has database compare/ synchronize, backup and integration with subversion, general admin + edit and procedural debugger	PostgreSQL, MySQL, Oracle, SQL Server, Sybase, Informix, Apache Derby (others with JDBC driver)	Yes	Yes	No	Yes - both meta data and adhoc on screen charts (only viewable in tool?)	Yes - Export (XML, HTML, Excel), Import - Delimited	Yes
Database .NET	Windows no install required (aside from .NET) self-standing 5 MB exe	Freeware	Sort Of - ability to script table structures, browse table structure and data (no wysiwig for table design, or editing data)	PostgreSQL, Firebird, MySQL, SQLite, SQL Server, SQL Server CE, Oracle, MS Access, Dbase, OLEDb, ODBC	No	No	No	No	Yes - Export (CSV, TXT, XML)	No
Database Master	Windows	Proprietary ($49)	Yes - general user management, ability to edit data including blob, ability to add tables, but datatype options impoverished.	PostgreSQL, MySQL, SQLite, ODBC, OLEDb	No	No	No	Yes - simple query result with ability to export to PDF. In-built reports for server config, process etc.	Yes - Import/ Export (XML, HTML, Excel, CSV)	Yes, but doesn't seem to be able to read foreign key relationships of pg (or couldn't get that to work) and can't edit from diagram.
DBTools Manager Enterprise 3.4	Windows	Proprietary ($100-$500) depending if you get add-on packs for non-OS dbs (also Freeware version and standard) - check feature matrix	Yes - include edit, also has database compare/ synchronize, backup (in sql format), Task wizard for ETL (interface intuitive)	PostgreSQL, MySQL, Interbase, Firebird, SQLite packaged (extra purchase packs for SQL Server, Oracle, Sybase, ODBC, MS Access)	Yes - you have to launch the sql query designer icon (not intuitively obvious). Also automatically shows defined relationships, but doesn't seem to allow LEFT, RIGHT JOIN or toggle between SQL/ Design view.	Yes	Yes but can currently only use within DBTools. A redistributable runtime is expected for later versions.	Yes - both meta data and data reports	Yes (also database migration wizard) - MS Access, Excel / Open Office spread sheet, CSV, DBF,HTML, XML - ADO/ DAO/ODBC (Paradox, Foxpro, DBase) data sources	Yes

DbVisualizer 6.5	Windows, Linux, MacOSX	Proprietary ($150) (also a freeware version) check feature matrix feature matrix	Yes - general admin + edit and blob editing show in chart	PostgreSQL, Oracle, Sybase, SQL Server, DB2, Mimer, Neoview, MySQL, Informix, JavaDB/ Derby	Yes - supports ANSI LEFT, RIGHT, FULL joins, and toggle between SQL and Design, changes in SQL view can be seen in design by clicking load to query design view.	Yes	No	No	Yes - CSV, HTML, XLS, XML	No? or at least couldn't find it
EMS SQL Management Studio for PostgreSQL (2007) include Manager and the whole bundle of export, backup etc. tools.	Windows	Proprietary ($250 for full-version) (also Freeware version of Manager with subset of features of full) - check feature matrix	Yes - eidt, table designer (supports full array of data types including array of types), also has database compare/ synchronize, backup	PostgreSQL (similar products for MySQL, SQL Server, FireBird/ Interbase, Oracle, DB2)	Yes can do views too. Nice toggle feature between query builder and sql view. Supports inner/left/ outer/full joins, however adding in GROUP BY in sql view confuses the designer though can make group by changes in designer. Making simple join changes in SQL view or adding columns can toggle back and forth to design view.	Yes	No	Yes - both meta data and data reports	Yes - MS Access, Excel / Open Office spread sheet, CSV, DBF,HTML, XML - ADO data sources	Yes including ability to edit foreign keys and add columns from diagrammer.
MicroOlap Database Designer/SQL for PostgreSQL	Windows (can run in Linux under WINE)	Proprietary ($400)	Yes (Designer)	PostgreSQL (similar products for MySQL and SQL Server)	Yes	No	No	Yes -- just meta data/ relational designer/ reverse engineer reports.	Import structures from ODBC supported dbs	Yes (Designer) - fairly sophisticated and reverse engineering

Navicat 8 for PostgreSQL	Windows, Mac OSX, Linux	Proprietary (~$200 with report viewer) and Free Lite version (only for non-proprietary use)	Yes include job agent, backup, report scheduling, DDL, data synch of data and/or structure compare etc.	PostgreSQL (similar product for MySQL, Oracle)	Yes	No	No	Yes -- focused on end user reports with charts and ability to schedule and email reports.	dbf, access, excel, html, xml and 10 other flat file formats	No
MS Access 2000-2007	Windows	Proprietary ($200 or part of MS Office Professional +)	No DDL but can edit data	Any database with an ODBC/ OLEDB driver	Yes (pretty good), can't create views	No	Yes, Desktop App and relies on MS Access full or free runtime download.	Yes - has subreports, charting fairly advanced	Yes - from and to any ODBC data source or excel with wizards and ability to save import specs	Yes but will not make any changes to PostgreSQL. It will just look pretty and can print. Also doesn't read relationships from PostgreSQL aside from primary key, but you can draw them in the relationship diagrammer.
Once:Radix	Mac OSX/ Windows / Linux	Free (GPL)	No	PostgreSQL	No	No	Yes - webapps based on Java Servlets (Apache/ Tomcat)	Yes - uses Jasper Reports	No	No
Open Office Base 3.1	Windows / Mac / Linux / Unix	Free (LGPL) - (also Star Office proprietary)	Yes (can define tables/etc. but sometimes flaky for DDL, fine for editing data)	Any database with an JDBC/SDBC/ ODBC driver	Yes (pretty good) - can also design database views graphically,	No	Yes - end app needs OpenOffice base to run	Yes can also do more sophisticated reporting if you download Sun Report	Yes - via the Paste Special/ Paste commands - supports CSV, HTML Table, OOBase/Excel copy pase. The wizard is nice but tempermental when importing into	Yes. Will also read Primary/ Foreign key constraints and allow you to edit them. Kind of flaky for creating relationships (sometimes creates duplicate foreign keys if you reedit a relationship)

					but can't reedit :(			builder	PostgreSQL. We'll write up another article about tricks of getting around its idiosyncracies.	but great for looking at them and setting layout. Still seems to lack a print option in this version. We use PgAdmin to create them and OO to look at them.
Pentaho BI Suite Enterprise	Windows, Linux	Commercial Open Source (comes in priced and free community editions). You can also download pieces separately like Report Designer. Pricing for commercial is not stated. Licensing is a mix of (GPL/LGPL/ Mozilla PL) see feature matrix pg 3	This is focused on BI, so BI administration, report design, OLAP, report scheduler, dashboards - web based reporting. built on java/ tomcat/jetty/ mondrian. Seems like a fairly hefty product (600MB download for full), not designed for beginners. Note we did not test it so can't speak for its merits.	While you can report on PostgreSQL data and any source with a JDBC driver, insists on you installing MySQL to store BI metadata so its evidentally very MySQL centric.	?	?	?	Yes - that's what its for. Web-based reporting in category of MS Reporting Services, Cognos,SAP BI Objects	Appears to have Advanced ETL features to pull data from any data source	No
PhpPgAdmin 4+	Windows, Linux, Mac OSX, Unix any webserver supporting PHP (any client with a web browser)	Free (GPL)	Yes DDL builder, user management, data editing in grid view, database object browser	PostgreSQL	Sort of	No	No	Advanced reporting and dashboards. Web-based	CSV,tab, xml	No

PgAdmin III 1.10	Windows, Linux, Mac OSX, Unix	Free Open Source (BSD)	Yes DDL builder, plpgsql debugger as plugin, user management, data editing in grid view, database object browser, job schedule interface to pgAgent	PostgreSQL, EnterpriseDb, GreenPlum	Yes (not great)	Yes (great)	No	Administration specific - Db objects	Limited to CSV. No direct import (but you can use psql/ PostgreSQL native COPY	No
PostgreSQL Maestro 9.5	Windows	Proprietary ($300 for the full PostgreSQL bundle, $200 for just Maestro)	Yes DDL builder, pl/pgsql debugger, user management, data editing in grid view with blob edit, database object browser	PostgreSQL (similar products for MySQL, MS SQL Server, Oracle, SQLite, Firebird, Sybase, DB2)	Yes	No	Yes (packaged separately as a free tool - a PHP page generator for query or table)	Yes, includes quick charts	Yes	Yes
Power*Architect 0.9	Java 1.6 (Windows, Linux, Mac OSX, Unix)	Free open source (GPL)	Sort Of (can browse, but no edit/create except via import/export and model generation scripts), ability to compare data models	PostgreSQL, SQL Server, MySQL, Oracle, Derby, DB2, HSQL, SQLStream	No	No	No	Yes but just modeling reports. Ability to export data model to PDF and XML.	Yes - Export - CSV/HTML, Import -- appears to have ability to import and export between database connections, but had trouble getting it to work.	Yes. This is its main focus. Lots of ER features and ability to generate logical data models and generate scripts to make them physical for desired database platform.
SQL Workbench/J	Java 6 (JDK 1.6) based - Windows, Linux, Mac OSX, Unix	Free Open Source (custom license) free for commercial but no right to change source	Geared toward querying, editing, and data import/ export. Sports a nice SQL formatter, nice SQL field/table code completion, schema diff/data diff, ability to see DDL of all tables, triggers etc and script, and a really nice data pumper. It is also very light-weight	PostgreSQL, EnterpriseDb, Oracle, H2, Firebird, Apache Derby, IBM DB2, Ingres, SQLite, MySQL, MS SQL Server though appears can support most JDBC driver based dbs. Have to download JDBC driver separately	No -just query editor with data browser/ basic data edit (autogenerate of UPDATE/ insert sql)	No	No	No	Yes -XML/txt but also sports an impressive looking Data Pumper that allows you to map fields between two JDBC datasources and copy data across.	No

			with a very sleek clean interface.	and specify its location via the driver template interfacee						
SQuirreL SQL 3.0	Java 1.6 (Windows, Linux, Mac OSX, Unix)	Free open source (LGPL)	Yes most of the additional features are available via plugins included as options in the install or for free download	PostgreSQL (plug-in based architecture), specific ones also for Firebird, MySQL, MS SQL Server, H2, Oracle, Informix, Sybase, DB2. Although you can check to install the PostgreSQL plugin, you still need to download the jdbc driver from http://jdbc.postgresql.org/download.html and dump in the lib folder of SQuirrel install for the PostgreSQL plugin to become active. Same for other databases.	No	No	No	No	Yes - Import/ Export CSV/ Excel	Yes via Graph plugin. Renders relationships but doesn't allow adding/ editing relationships.

Back to Table Of Contents Database Administration, Reporting, and Light application development Reader Comments

Reader Comments

CodePlex Foundation

gregj

this is all wishful thinking.
Microsoft won't ever do anything good either for customers or opensource. They are in business of making cash. But care only about large customers, and themselves.
So thinking that microsoft wants to make something good, and contribute in this way is at least naive wishful thinking.

Regina

Greg,

I think all companies are out to make money and maximize on that. So of course I don't expect Microsoft to be any different. I am arguing its in their best interest to do so. Because they have a large developer following pushing them to do so. I honestly don't know why everyone is so against Microsoft.

I bet if you talk to customers both large and small vs. the customers of Oracle -- you will find that most Microsoft customers are much happier with Microsoft than they are with Oracle. Well at least in the arenas I've been in (both large and small).

In my mind Oracle fits your description better -- a company that pretends to be on the side of open source but really isn't.

gregj

you are right. My description probably describes likes of Oracle as well.
What I am trying to say, is - Microsoft in no way is trying to make 'world better'. They are forced to try support some opensource base. It is really simple. Opensource 'stuff' currently runs mainly on linux, or other BSDs. So obviously, to change that pattern, microsoft has to do something for the opensource world.
But that's again, not 'pro opensource' move. It is more along the lines of leaning towards what people do at the moment.
They didn't get they idea, that they are still quite evil at what they do, and until they change that - (stop stealing ideas, stop making stupid 'innovations' that aren't really any innovations at all, stop abusing 'network effect', etc, etc).
The fact that microsoft creates opensource portal, or whatever that thing really is going to be - doesn't mean they changed their policies, or attitude.
Until that happens, any that sort of 'news' is as funny and propostrus as Chinese democratical humanitarian aid force in africa.

Regina

Greg,

Okay we are not in disagreement. My point
is it really doesn't matter if Microsoft is evil or not. The market forces if pointing in the right direction will force them to do the right thing.

When a company gets beyond say 200 people, I think the judgement of morality etc. goes out the door (you have to judge them as a machine and see what is feeding that machine). Google is no longer "NOT EVIL". they were not evil when they were 20 buy they are a machine now.

Oracle ironically I would sense is less pressured to do right by Open Source -- simply because they have high margin business, higher number of sales force and a lower developer customer base. To them open source is a means to upsell. Microsoft makes much of their money on lower margins and ISVs reselling their wares as part of their things. They can't afford to piss off that group as much as Oracle can get away with.

of course I could be wrong with all these observations, but that is my general non-statistically proven sense of the way the wind blows.

gregj

I am not trying to say that microsoft is evil.
It looks to me, like we are in agreement here, really.
I am trying to show the reasons behind Microsoft's steps, which clearly shows you - that believing in their honesty and generosity is just wrong.

Basically, they don't make anything new here. They don't innovate, they don't help people to create and grow. They try to take a slice of the 'open source' pie, grab it and hold it.

That just defines any microsoft action. It shows you how great is the 'network effect', and how really easily fooled people are.

On a side note, any corporation is evil, both if you work for it, or if you're their customer.

Josh Berkus

Leo, Regina,

Sorry, that's wishful thinking. If you look the CodePlex board is composed entirely of Microsoft Staff, a community-hater, and one genuine open source person (Monty) who is too busy to participate in any decision-making. Further, if you read their Contributer Agreement, it's an "all your code are belong to us" agreement, which says that Microsoft can use your code in any way they please.

So, Microsoft has done some good things in the recent past with OSS. However, Codeplex isn't one of them; it's nothing more than developer entrapment.

gregj
I second Josh here. He is fortunate enough, to be able to put it into nice words, and explain it better than I do.
I double checked Josh's article sources: http://it.toolbox.com/blogs/database-soup/codeplex-stay-away-34128?rss=1
And yes, This is pretty much my feeling in the guts.
Microsoft will never do anything good. The fact, that few ex-opensource activists (can't really call Miguel an opensource friend anymore) got fooled, and paid little something to belive, well.
As they say, Lie changes depending on the level. In corporate environment, you have to convince common people that they do good, in order to achieve a/the goal. So you lie. And lie changes, depending on the level.
It is no different for Microsoft, and other corps.

Regina
Josh and Greg,

Only time will tell. You may or may not be right, but I don't see the harm in welcoming with half-open arms. Of course Microsoft wouldn't join an existing org, look at the animosity in such orgs at the name of Microsoft.

I provide a link to Miguel's blog in contrast.

http://tirania.org/blog/archive/2009/Sep-10.html

I think there is a chasm here -- and it seems to fall between .NET programmers and non .NET programmers. I happen to sit on both camps. But I do consider Miguel someone I hold in high-regard - he isn't militant and egoistic like some people I can name in the OSS community (not you 2 so don't think I'm talking about you).

Is it true that FSF threw Miguel out because he refused to call Linux GNU\Linux? or were there other reasons. I would be interested in knowing your take on that. Because if that is what it is then the whole discussion of "Why wasn't the FSF contacted about CodePlex" is shall I say one-sided BS.

Cross Compare of PostgreSQL 8.4, SQL Server 2008, MySQL 5.1

Rob Wultsch
For whatever it is worth...

Date Time support in MySQL I think exists better than you think in the datatype TIMESTAMP. It is stored in gmt and converted automatically to the tz of the session of the db connections. It is also possible to do conversion. Is this the same as storing a tz, no. I think it is better.

Table Partitioning is very much in 5.1 (and it is very much ugly). So I suggest that you revise "Yes?".

Access tables from other databases on same server (but not easily across servers): on MySQL this exists in Federated (and Federated-X) and is easy, though not recommended for many work loads.

Thomas
The TIMESTAMP data type of MySQL is very limited in its date range: it cannot store dates before 1970 and not after 2038...

Vincenzo Romano
Comparing DBs is not an easy task, indeed.
Nonetheless, I'd have divided the comparison in sections: costs (money, human, operational and maintenance), features, performance, reliability.
Then I'd have added Oracle.
Finally I'd have cut away all those topics with the same values (like readonly views or stored procedures.
Finally, the comments on the MySQL column seem too skinny when compared to with the ones in PG and MS columns. This could lead the reader to perceive some kind of bias, which is human anyway.
A referecne to the online documentation for each technical topic would moreover give more confidence in the comparison.

Giuseppe Maxia
Some remarks, for the sake of completeness:

"Partial Indexes - e.g. you want to create a unique index but only consider non-null values"
Yes. MySQL supports it. It's called UNIQUE, and differs from PRIMARY because it allows NULL values.

"ACID compliance": you mention MariaDB, which is not an engine but a fork of MySQL that includes the "maria" engine. The "Maria" engine is like MyISAM with crash recovery, but it is far from being ACID. So, only InnoDB at the moment is ACID. You may instead mention PBXT, a community developed engine, which is ACID.

"UPSERT logic". MySQL also supports INSERT ... ON DUPLICATE KEY UPDATE ...

"Table partitioning" is supported in MySQL 5.1, which has been GA since November 2008

"case insensitivity" in MySQL it depends on the character set used.

Cheers

Giuseppe

older
Can you add firebird sql to comparison?

Chris
Can you compare perfomance against all three? I always feel that mysql joins are extremly slow and would love to see real world comparisons against the other two. I've always been happy with ms-sql but would like to see how postgres performs.

Dave
You missed "GUI Front-end that does 90% of what you need to do without you having to remember the f-in SQL"
2008 - YES, and it's fantastic

Everything else - what's a GUI? Command-lines-roolz.

Regina
Vincenzo,

good ideas -- we'll change to include links to the online docs where relevant.

You are right its hard not to be biased in these things. Though we work with all 3, we have worked more intimately with PostgreSQL and SQL Server.

Ryan Bair
What makes you claim PostgreSQL's planner is the best? Personally, I get far better plans with Microsoft SQL's planner than Postgres' for complex queries.

I'm pretty sure IronPython cannot be used for stored procedures because it relies on runtime code generation which is prohibited, even in unsafe mode.

You should add in a row for Index Organized Tables/Clustered Indexes.

Andrew Dunstan
One of the coolest Postgres features that you don't mention is that in Postgres almost all DDL is transactional, so you can roll it back safely. This is terrific for things like upgrade scripts - if the script fails somewhere you are not left with the database in a half upgraded state. It's also a great boon for developers. Most databases don't have this.

There are also several other replication methods for Postgres that you didn't mention. One I like is Londiste, from the Skype people, and the soon to be (I hope) new release of which looks very cool indeed.

Regina
Andrew -- good points. The transaction one I think we should include too. Its important feature for people packaging applications. For the replication thing. I think that may deserve another article since there are many replication products out there and many that will help with cross replication against disparate products too.

So it seems a bit unfair to talk about things that aren't built into the core product in this article.
Granted part of what makes Microsoft SQL server much longer install -- is that its really 5 or 6 products (and some that you can't just not check) and if you are like most who just check most everything (I want reporting services, ssis, database server, management tools, notification this and that) -- Its like installing a whole OS, but at least you can say its part of the same product package even though a lot of that stuff has not much to do with what makes a database good.

Regina
Ryan,

I guess Best wasn't quite right. We should have said Best for us.

Which version of SQL Server are you running? With Standard for example you don't get partitioning -- so can't talk about how well SQL Server handles that without a big disclaimer.

With PostgreSQL you get simple partitioning you get that and it works pretty well with constraint_exclusion -- so that we are generally able to run more complex bigger db workloads on PostgreSQL without shelling out a lot of money.

The other thing that makes PostgreSQL planner better in many cases is the IN LINING of SQL functions. SQL Server sees all functions as Opaque so doesn't take advantage of indexes that can be used within a function that can be folded into the overall plan. You can wrap pretty complicated constructs in an sql function.

PostgreSQL also allows you to cost functions differently which I don't think SQL Server has yet so in conditions where the planner has to decide which function to test first -- it can look a the cost of each and use the less costly first. Lets just say when I started using PostgreSQL, I was very impressed with its speed and ease with which I could use my SQL Server knowledge right away.

For our spatial queries, PostGIS is just better. The SQL Server 2008 spatial selectivity still seems pretty weak and it often brain-deadly doesn't use a spatial index when it obviously should, but even when it does it is still a bit slower (unless you fiddle around with gridding and so forth). This all may change in the SQL Server 2008 R2 (which the CTP1 has just been released). So of course we are a bit biased in this regard.

IronPython. I haven't tried writing a stored proc in IronPython, but I gathered it was possible by this
http://ironpython.codeplex.com/WorkItem/View.aspx?WorkItemId=21740. I should probably give it a spin since it seems rot with problems from above.
As far as PostgreSQl, well its almost brain dead easy to write a python stored procs and lots of people do.

-- Ah yes clustered indexes -- we'll add that to our list.

Regina
Chris,

Its hard to give these tests without coming up with some example that everyone will claim is biased against one database or that you didn't optimize the database right. So I tend not to like to do that.

So my off the cuff answer -- which is biased based on working with all 3.

Yes Mysql join is kind of slow compared to SQL Server and PostgreSQL especially when you have a lot of joins. I think many people agree with me on that.

In terms of the speed difference with complex queries between SQL Server and PostgreSQL. I don't think there is a clear winner. See my note to Ryan -- who says he has better luck with SQL Server. My gut sense is that just looking at the architecture and the fact that SQL Server has a more complex planner strategy, that if you have say 4 or more cpus and have license for all 4 your sql server, SQL Server will perform better on an individual query because of its parallelism. However my general impression (which I haven't timed) is if you have more people hitting the server with complex queries, PostgreSQL seems to perform better (not sure if its because of its free threading model or if that makes a difference), but it seems to handle more queries at a time a bit better than SQL Server. I'll need to experiment with that a bit to turn feeling into hard numbers.
In the multi case the processors are used to server different users. In the parallelism the processors are used to divide the work of one query.

Regina
To add we wanted to include all the Yes Yes Yes for 2 reasons
1) these answers are a bit different from last time.

2) This is also meant for people migrating from some other db, who aren't sure if the other covers this important feature. So even though in some cases the answers are yes in all, we thought it important to include.

The discussion about maintenance is difficult. I don't think there is a clear cut answer. If you are a unix user and someone slaps you with a choice of SQL Server or PostgreSQL or MySQL. Pg or MySQL are obviously more comfortable and easier for you to deploy.

In case of Microsoft users, its even less clear cut. If you talk about hard-core database users who are very into security, programming in the db and so forth. They would find MySQL extemely cumbersome. But for someone who knows little about databases and just needs one -- MySQL's simplicity is a feature -- as all the features in SQL Server and PostgreSQL would just be noise. Then there is the issue of ISP support and so forth. I'm not sure how to categorize all those things into a cohesive chart.

Leo Hsu
We'll think about it. Its on our to do to test

BostonGIS Blog
One of the interesting things to come out this week is the SQL Server 2008 R2 CTP
One of the enhancements we are really looking forward to and interested in experimenting with is the Report Builder 3.0/Reporting Services support for Geospatial data. We

Tom
I believe SQL Server has Partial Indexes in 2008 - SQL Server calls them Filtered Indexes. You can index on non-null values, or numbers <> 1, etc http://sqlfool.com/2009/04/filtered-indexes-what-you-need-to-know/

Devrim GÜNDÜZ

Partial index is not described in that way. Just a FYI.

Leo
Thanks we'll correct that line. Most have missed that one.

Cameron Eure
PostgreSQL does support Materialized Views, but they're not automatic... You have to write your own Triggers maintain the table(s).

Thomas
The cool thing about Oracle's MViews is that they are used even if the original SQL doesn't include a reference to them. If Oracle's optimizer sees that a SQL statement (or even part of it) could be replaced with a select to a MV it does that in the background.

That is something that is not available with either "manual" MVs in Postgres (or any other DBMS) nor the indexed views of SQL Server.

Using Recursive Common table expressions to represent Tree structures

David Fetter
Great topic!

A couple of observations:

* Unless the length 1000 has some significance, use TEXT instead of
VARCHAR(1000).

* It might well be both faster and more correct to push items into an array
and use array_to_string() in the outer SELECT, and it won't be subject to
sorting anomalies.

```
WITH RECURSIVE supplytree AS
(
SELECT
si_id,
si_item,
si_parentid,
ARRAY[si_item] AS si_item_array
FROM supplyitem
WHERE si_parentid IS NULL
UNION ALL
SELECT
si.si_id,si.si_item,
si.si_parentid,
sp.si_item_array || si.si_item As si_item_array
FROM
supplyitem As si
JOIN
supplytree AS sp
ON (si.si_parentid = sp.si_id)
)
SELECT
si_id,
array_to_string(si_item_array, '->') AS si_item_fullname
FROM supplytree
ORDER BY si_item_array;
```

Arek
Have thought about using ltree ?

http://www.postgresql.org/docs/current/static/ltree.html

I'am not saying than WITH RECURSIVE is bad .. just that, there are simpler solution sometimes ;-)

sabra

How do you use it to find the parent path for just a single item?

Leo
Good point. We haven't explored the use of ltree so will have to give it a test drive sometime. I think the only thing against it is that its a PostgreSQL specific feature where as the CTE is more ANSI portable (except for possiblyt the word RECURSIVE)

Leo
Sabra,

Couple of ways -- you could write a function as we demonstrated in linked article, but that is not as suitable for multiple sets since it would probably do a subquery for each record.

You coulde also take our example and limit with a WHERE clause but that is much slower than it could be.

The other way would be to recurse backward from the child to the parent. So instead of starting at parent nodes -- you start at the child node and keep on unioning until you hit a parent with no parent. Will have to write that up sometime.

Database Administration, Reporting, and Light application development

Scott Bailey
One of my favorite tools is Aqua Datastudio. It supports all of the databases I use on a daily basis; runs on Mac, Linux and Window; logs all of your queries; and is fairly feature complete. If it did code completion for Oracle packages, it would be the only tool I'd use.

http://www.aquafold.com/

Greg Siems
Nice writeup!

One tool that I use a lot is DbVisualizer. It supports several database engines using JDBC and runs on Linux, Mac, Unix and Windows. There is a free version and the paid for version is reasonably priced.

http://www.dbvis.com

Balázs Bárány
Don't forget the huge Java ecosystem of Open Source database and Business Intelligence tools that work over JDBC so database compatibility is excellent. Just a few examples:

SquirrelSQL, a SQL browser and frontend:
http://squirrel-sql.sourceforge.net/

Power*Architect for database design (includes forward and reverse engineering), with special support for special database dialects including Postgres:
http://www.sqlpower.ca/page/architect

Everything from Pentaho (e.g. data integration, reporting, OLAP):
http://community.pentaho.com/

Those are the ones I use on a daily basis. I seldom need other tools.

Leo
Thanks all for the feedback. We'll add these to the list.

FYI - pgAdmin III includes an excellent PL/pgSQL debugger.

Greg Smith
I just linked your great guide into the less well maintained page covering this on the PG wiki: http://wiki.postgresql.org/wiki/Community_Guide_to_PostgreSQL_GUI_Tools

Thomas
You might want to consider SQL Workbench/J as well, which has strong export/import capabilities, schema diff, data diff and can be used in batch files as well.

David Fetter
Any chance you could change "commercial," to "proprietary" where appropriate? FLOSS has been used in commerce for some decades now.

Regina

Done

Leo
Thanks Greg

John
Database Master is my favorit. It is a awesome tool and awesome user friendly. I can visualize relations, import csv,sql files. But most important is editing and reporting data (inluding blob fields).

Great Tool and thank you for this great product ;-)
http://www.nucleonsoftware.com

Regina
Thomas,

Thanks for the heads up. We've added to our list. I really love the light-weight slick interface, sql formatter and the data pump. I'll have to play with the data pump some more since its something I can forsee using a lot and I really love the way you have implemented the screen. So far been impressed.

John
You missed out Servoy. http://www.servoy.com - (It's proprietary, but not sure on latest prices - should be on the website somewhere.)

I use it regularly as a RAD environment (it's simple enough that I can use it) and it works well with Postgres.