

Re: What so special about PostgreSQL and other RDBMS?

Source: <http://coding.derkeiler.com/Archive/PHP/comp.lang.php/2004-05/1172.html>

From: Volker Hetzer (volker.hetzer_at_ieee.org)

Date: 05/17/04

Date: Mon, 17 May 2004 21:54:58 +0200

"Quirk" <quirk@syntac.net> schrieb im Newsbeitrag
news:4e20d3f.0405121614.28ecad7b@posting.google.com...

> "Volker Hetzer" <volker.hetzer@ieee.org> wrote in message
news:<c7t0v0\$sbv\$1@nntp.fujitsu-siemens.com>...

> > > And what was your reply?

>

> > I asked first.

>

> > Is this grade school?

>

> > > Really, care to quote the part of the Contract that Gaurantees you any

> > > rights?

>

> > <http://oracle.com/support/index.html?policies.html>

>

> > I asked you to QUOTE the part of the Contract that Guarantees you any

> > rights, not post a link to a description of support options and what

> > they cost.

That IS the licence agreement. Or at least tke part that deals with after sales stuff.

That's exactly the link the licence agreement for the database points to when it
comes to what wecan expect for paying support.

>

> > And even so, if you bother to read that page you would have noticed

> > that it is mostly about protecting Oracle's rights from you, not

> > granting you any.

>

> > For example:

>

> > "Oracle may provide additional releases or versions of its programs

> > in the form of an Update as part of our technical support services. It

> > may become necessary as a part of Oracle's product lifecycle to

> > desupport the programs and, therefore, Oracle reserves the right to

> > desupport its programs."

So? That's what desupport notices are for. The good thing is that

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

you find out beforehand, not after you start asking around because the latest ftp download is from 1996.

- > *If my application required a cucumber, I wouldn't sign a deal with a*
- > *cucumber vendor that insisted I could only buy cucumbers from them,*
- > *for ever, even if their cucumbers no longer work for me, while they*
- > *could stop providing cucumbers any time they feel like it and still*
- > *forbid me to use my own, proprietary cucumber dependant, application.*
- > *I would, at least, make my application work with any cucumber.*

But what if you required a cucumber with special properties, like a Monsanto engineered one which contains some drug you want to sell?

What if you required support in case someone chokes on it and the support company requires you only to buy certified cucumbers?

- >
- > *This conversation has gotten ridiculous, can it be that you really*
- > *don't know the difference between a cucumber and an application*
- > *dependency?*

It's a buying decision. You invest money and get a ware. Open source is only different in that you can go behind the stall and see whether you can make something of the stuff they dropped. That's where the "for free" stuff comes in.

- >
 - > > *Have you tested alternatives?*
 - >
 - > > *The other example was buying gcc support from cygnus.*
 - > > *One bug, never got resolved in one year, consequently*
 - > > *we cancelled support.*
 - >
 - > *Yet in this case, you could have purchased gcc support from another*
 - > *company, however, without source, you would not have this option.*
- No, I couldn't because no one else was selling it.

- >
 - > > > *Are so so stupid that you actually expect a serious answer that was*
 - > > > *obviously a*
 - > > > *hostile attempt to insult by way of a rhetorical question?*
 - >
 - > > *Ok, so for you explicitly: That was not a rhetorical question. Your response*
 - > > *indicated you didn't read my posting, or at least not the relevant part, so > I wanted to check whether it*
 - > > *was worth posting*
- any more.
- >
 - > *What nonsense, please demonstrate this by comparison, I have clearly*
 - > *responded to all your arguments, regardless of how little sense they*
 - > *made.*

">>>>> The right to modify is a red herring.

>>>>> *Not if your application and the permanency of your data is important."*

And this after I just detailed how a migration is made way more difficult by all the other bits that change and that the underlying database is really the least of

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

your worries when migrating to another app or the same app ten years later. I brought up the chip design example because we've been through this. I could give you a board design example we are going through right now. System 1 used informix, system 2 uses oracle. Guess what's the only part of the migration that doesn't give us any problems.

> *Worth posting what? Your great advice that developers should *NOT* abstract their code?*

That they should think before they abstract.

> > *I start to repeat myself here.*

>

> *Too bad you have no actual argument to repeat, you are merely repeating your empty rhetoric and unsubstantiated bunk.*

That seems so to you because you can't read back more than one quote.

>

> > *The right to the source code does not mean anything useful, see the part you quoted below.*

>

> *Yes it does, it's too bad you don't understand it.*

>

> *If I have the source code, I know I can rely on a product for ever, and never talk to the original developer again if I so chose. Without source, the developer holds all the cards.*

You know what, you do that. Use any open source database, and ten years after the project has been abandoned you go and port it to another platform, and try to get customers to use it okay? Then we'll talk again about "forever".

>

> *Let's take a simple case, say you hired a consultant to write you a simple*

> *application, say a specialized contact manager.*

>

> *When the project was over, would you let the consultant leave your office, only turning over a compiled binary of the application? Or would you insist that he provide the source?*

Depends on how simple and how frequent my requirements change. If he's the only guy that understands it I'd insist on maintenance and a customizing possibility (like ActiveX or an integrated tcl interpreter). If the requirements are volatile I'd do a long term contract detailing what money he has to pay for getting out of it.

By the way, we did a bit of chip design before and had a tool made by a small company situated here near Augsburg. Great tool. There we did what I said (with the customizing) and we also got the source code. And you know what? It was much too much bother even to get it to compile, much less change. It was always cheaper to let those guys do the work. And what do you think would have happened if we had wanted them to support code *we* modified?

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

>
>> *Unsubstantiated bunk, if you have the source code, it is not magic to*
>> *fix it, or extend it, just normal programming.*
>
>> *Right. So, if I do CAD programming, why should I learn database programming*
>> *only to support a dead database? It's much easier to migrate to another one.*
>
> *Why are you struggling so hard with such simple logic?*
Because it's erroneous.

>
> – *If a Dead Database means your application is also dead, if*
> *migration is impossible; having source code can save the day.*
My customers won't want a dead database so I'd have to
migrate or die anyway.

>
> – *If migration is possible, migrating is easier with abstraction.*
Weakening the case for the rights of the source code.

>
> – *If you have source *AND* you have abstracted, whoa nelly, you are*
> *in *really* good shape.*

As I said before, database migration isn't hard. For our new board design
tool we developed a database interface from scratch. 4 weeks, including
testing.

And yes, we went proprietary because otherwise we'd have spent ages
doing all the generic layers. And if oracle dies, so what? If our app
really still exists then we'll spend another 4 weeks interfacing to
wherever we want!

>
> – *If your data is archived in a self contained, self describing,*
> *human readable format, why, you are all but invincible.*

Another case on not reading what I write.

>
> *Thus my advice.*

>
>> *Besides, have you considered that quite a few open source projects get abandoned*
>> *because they have become unmaintainable?*

>
> *And closed-source applications have never been abandoned???*

Which database do you have in mind?

>
> *Another simple question: If your application is abandoned, are you in*
> *better shape with, or without source code?*

I assume, you mean if my database is abandoned.

It doesn't matter because I'll be migrating anyway.

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

>

> > *Anyone remembers hurd? Groff?*

>

> *Yeah, what about them?*

They are dead.

>

> > *What was the last gmake improvement? And if the authors throw up their hands,*

> > *what can I do? Ask my boss to form a department for the beating of dead*

> > *horses?*

>

> *If you are dependent on them, at least you always have the source code*

> *and can thus continue to use the product,*

I can do that with oracle.

> *even have it modified if you*

> *need to.*

I can't because it's unmaintainable. At least groff is, to stay with that example. It already was so in 1993.

> > > *Simple calling something*

> > > *an illusion does not explain why you consider it impossible to*

> > > *actually change a program. Perhaps you should consider a different*

> > > *line of work.*

>

> > *Oh, it's pretty easy to change a program. Working through millions*

> > *of lines of code and repairing it with less time or money than it would*

> > *cost to migrate to another database is the trick.*

>

> *Reminder: I am the one advocating Abstraction, which would make it*

> *easier to migrate to another database. What the hell are you talking*

> *about?*

No, you are the one talking rights to the source code. Abstraction is just a side tracking because the right to the source code should, in your world nullify the need to abstract.

>

> *And If, for some reason, you *must* repair the database, say the bug*

> *is simple and is easier to fix than to migrate a large working*

> *implemtation, at least with the source, you can, without the source*

> *you can not.*

If the program is fixable it wouldn't get abandoned.

>

> > *Convincing the customer to*

> > *install *my* database version is another, particularly if three or four*

> > *developers do this.*

>

> *Leaving the customer stranded because your application is hosed by an*

> *obsoleted dependency is even a harder sell.*

So, we need the original maintainer org to do the maintenance because only in

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

that case you can get one consistent version hosting all apps, right?
And if the db is dead *all* apps would be dead too, right?
Where's the difference to a commercial app then?

>
> > > *Same question: Did you read what I wrote?*
> > >
> > > *A better question: What kind of an idiot are you that, in the face of*
> > > *good sense, the best you can do is attempt insulting, evasive*
> > > *rehetoric?*
>
> > *It's not a better question. You keep bringing up that stupid*
> > *source code argument totally ignoring the fact that it simply doesn't*
> > *work, at least not for the money a normal support contract costs.*
>
> *You keep basing your entire argument on nonsensical out-of-hand*
> *dismissals, like 'it simply doesn't work.'*

The difference is we've been through this and we are using open source software and, to go back to the subject line of this thread, open source databases for mission critical apps don't work better because of the right to the source code. If I find a bug in tcl I'll put it in the activestate and let them sort it out because I'm not going to maintain my own test suite and adapt it to every new version. This is even more important for databases where I might even not have a means of recovering the data if my fix results in corrupted data half a year later.

Btw, do you know what ARM paid for gcc support?
Over half a million bucks if I remember correctly. This for a supposedly free product they could have fixed/ported themselves.

>
> *It does work, let me let you into a little secret: programmers modify*
> *source code, that's how programs are made and fixed.*
Or broken.

> *Without source*
> *code you can not fix a program.*
I can fix a program by telling the vendor to fix it, remember?
At least I can do that with all the programs we are currently introducing, oracle and the said board development tool included.

>
> > *And if support doesn't work, I still won't support it on my own.*
>
> *You can do what you want, my advice is just that, advice, many people*
> *are in different situations from you, and have a different point of*
> *view.*
Typically they sit at universities and have access to plenty of cheap and skilled labour. Sometimes I still envy the TeX group at my old university.

>
> > > *As I said, my comments were meant *FOR DEVELOPERS* that is those who*

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

> > > *are developing *NEW* applications, and my advice is simple enough,
> > > despite your contortions: If your application is important to you, do
> > > not engineer a dependency on code you do not have access to.'*

>

> > *Do you develop for platforms other than linux?*

>

> *Yes, I have and do develop for many platforms, but *I* am not the
> topic of this thread, despite your desperation. Once again, you only
> attack the arguer because you have no argument.*

So, if windows or MacOS is among them I guess you will be dependent on some libraries and kernel properties that you don't have access to, right? Only I'm trying to learn from you.

>

> *The assertion you quote remains true, and your response, as usual, is
> not a response at all.*

Just you stating it doesn't make it true. People are creative, that means they can do things you can't. So to use them you become dependent on them.

>

> > > *More unsubstantiated bunk, first of all, in many cases you can hire
> > > the original developers,*

>

> > *Yeah, exactly. A man year here costs about USD200000,-. A support
> > contract with oracle costs me about a tenth of that.*

>

> *In many cases you can acquire a support contract from corporations that
> have the original developers working for them.*

Right. At which we are back to the point where open source doesn't give me an advantage. Remember, if I want informix support today I can go to IBM too.

>

> > *And even if I buy some incident based support contract, there is still
> > no difference from an incident based support contract with oracle.*

>

> *Yes there is, since you value the original developers so highly, we'll
> try this example.*

>

> *The best original developer of Oracle, the one with the greatest
> knowledge of the system and code, quits Oracle and goes to work for
> Databases-R-Us, since you have no source, you must continue to deal
> with Oracle, the copyright holder, and can not hire Databases-R-Us,
> who employ the developer.*

You are mixing something up here. Oracle doesn't depend on a single freak but on a well maintained turnover process.

What you mean is what open source software is famous for.

>

> *The best original developer of MySQL, the one with the greatest
> knowledge of the system and code, quits MySQL AB and goes to work for*

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

- > *Databases–R–Us, since you do have source, you no longer need to deal*
- > *with MySQL AB, the copyright holder, and can instead, choose*
- > *Databases–R–Us, who employ the developer.*

Fortunately commercial software rarely depends on one individual.

- > > *As long as that guy exists and I can sue him into doing his job I don't*
- > > *need the source code (he needs) and otherwise I have no one to*
- > > *replace him.*

>

- > *Suing him is a red herring. Your applicaion is not powered by law*
- > *suits, but rather by compiled source code.*

But it is powered by lawsuits. At least by the threat of it. Just like boeing or airbus are when it comes to engineering security in. They buy insurance against lawsuits and, with the insurance company work out a process of ensuring quality which keeps the premiums down and the insurance from losing lawsuits.

>

- > > *But thanks for acknowledging that reliable support costs money.*
 - > *If stating the obvious is somehow of help to you, you're welcome.*
- Whatever.

>

- > > > *regardless of your right to the source code,*
- > > > *secondly, by hiring the "Copyright Holders" you *ARE NOT NECESSARLIY*
- > > > *HIRING THE DEVELEORS*, who may not even be with the company anymore,*
- > > > *in fact you are often hiring some peon who they scooped of the*
- > > > *consulting market 5 minutes before sending him to your office as an*
- > > > *certified solutions prodiver or whatever idiotic buzzword whey have*
- > > > *for their unskilled labour.*

>

- > > *Try it.*

>

- > *Try what?*

Doing a maintenance contract and then measuring the few minutes per week spending kicking their asses compared with all the programming you'd have to do for yourself.

- > > *Besides, remember, the company has an interest in providing*
- > > *support because they live off it.*

>

- > *They also have an interest in dumping relationships that are no longer*
- > *profitable, and may not be interested in your obscure problem or*
- > *implemation, but rather more interested in selling you (or someone*
- > *else) something new.*

In that case they'd offer me a migration path too, of course. Also, if you ever manage to spend a few dollars on a oracle standard version you might want to look at their bug site. Then you'll see to what lengths they go. Personally I haven't hit any of those "too obscure" problems yet. I've hit a couple real bugs, and once or twice they have nicely told me that the problem wasn't what I thought but that my shitty code (first

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

steps as a PL/SQL programmer/first steps oracle spatial) crashed the parser and that this was fixed in the next release the patch of which was available already for download.

>
> *Other organisations may be quite interested in helping you, but are
> unable to because you have no source code for them to fix.*

If the main developer company abandons the product how long do you think the others will survive? Besides, who tells me that the others aren't just passing the service requests on to the real one?

>
> > *And finally, it is a falacy to say that someone will do a worse job
> > simply because they are not the original developer.*

>
> > *So, if I pick some average application programmer off the street,*

>
> > *how long do you think it takes before he can start smoothing
> > out bugs in the postgres optimizer?*

>
> *I would not recomed you 'pick some average application programmer off
> the street' if you want to sort a bug in the postgres optimizer.*

>
> *Many developers could do whatever you want, for instance: PostgreSQL,
> Inc (not to be confused with PostgreSQL Org), Cybertec Geschwinde &
> Schoenig, NuSphere, or many others which know the system well.*

Nusphere takes one branch and supports that.

Cybertec can't even proper spell the stuff on their homepage.

So, PostgreSQL Inc would be the one we'd be dealing with.

I didn't find Nusphere of Cybertec on PostgreSQL Inc's homepage, at least not in the partner section. Can't be much love then that's lost between them. Which makes me wary about relying on any of those other two as a failsafe.

The fact that it's worth for nusphere to support one port also makes me hesitant about PostgreSQL Incs policy re platforms/versions.

>
> *However when Oracle lets you talk to a programmer, that is just who
> they let you talk to, some average programmer they picked off the
> street, the good programmers in their organisations to not work in the
> support department, but rather on new features for new versions and
> products to sell.*

Actually, that depends on how long the bug stays unresolved.

they escalate bugs. Of course the first guy is there to make sure you've read the manual but then you get the development guys.

> > *But it stops short of guaranting that your aplication will actualy
> > work,*

>
> > *Of course they don't offer that. But they offer to put effort*

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

> > *in it.*

>

> *Only as long as it is profitable for them and no more, then you get*

> *'Desupported'*

No, not "I" get desupported, a product gets desupported. For all users.

>

> > *And they are dependent from me for my money.*

>

> *Just you?*

All it takes is a poster session at the next oracle world and it won't be "just me". At least not if I had anything to complain about.

>

> > > *or that your existing version of the software will be supported.*

>

> > *They provide upgrades and desupport dates. Ok, they do*

> > *what I pay for.*

>

> *Only as long as you pay, and only on their terms, if you have source, you need not change a working system just because it is not supported by Oracle anymore.*

I don't need to do that for oracle either. At least not as long as it runs. And not, even for a open source software I would wait with the migration for the first bug after desupport.

> > *Just look at informix to see how*

> > *it goes when a db disappears from the market:*

> > *They had a big market share, market share dwindled, they got weak and sold themselves to ibm because that's better than going bankrupt.*

> > *Now IBM handles the migration to db2 and supports me as application*

> > *developer in porting my app to db2. This is much better than handing*

> > *me the source code and telling me that from now on I have to develop*

> > *all the new features and fix bugs on my own or simply buy a new db*

> > *and do the migration on my own.*

>

> *Or instead of IBM they could have been bought by CA, and fucked up royaly. Or just been allowed to disappear.*

Databases have customers which are worth a lot. Do you think IBM bought them for their marvellous technology?

Who is CA?

> *Again, you are depending on*

> *good luck and good graces, if you have source, you know for sure, but*

> *as I've said many times, it's even better to have an abstracted*

> *application.*

I'm not repeating again the problems of abstraction. Nor the easyness you can migrate between databases today.

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

- > *And by the way, don't think that IBM is above squeezing these newly*
- > *acquired hostages for every penny they are worth, and tossing aside the*
- > *ones who helping would not be profitable. You dont become a 100*
- > *billion dollar company by being stupid.*

And you can't "toss aside" paying customers. Not sure if you are russian and live in russia but if you are and do) you haven't got a particularly nice capitalism working over there. Maybe you view problems as ordinary that are viewed as pretty exotic in western europe or the US because your country really works that way. Rest assured it's not so in the rest of the world.

- > > > *I have no idea why you are insisting on jumping up and down like this*
- > > > *is crazy talk, the only plausible theory is that you get some kind of*
- > > > *thrill out of embarassing yourself.*

>

- > > *Where do I jump up and down?*

>

- > *When you stoop to making ridiculous, incoherent, awkward stretches of*
- > *logic to keep this conversation going on and on in the face of clearly*
- > *explained, good advice.*

Ah, but I don't. Besides if I were you I'd leave the evaluation of your advice to others.

- > > > *This is just stupid, elegnt coding is hardly as unatainable an ideal*
- > > > *as you seem to be conviced, in fact in this specific case it's a*
- > > > *simply matter of using a standard wrapper function throughtout your*
- > > > *aplication to access your data rather than using proprietary bindings*
- > > > *throughout your application, if your application is sufficently*
- > > > *complicated, perhaps a data abstaction object might be usefull for*
- > > > *this function, perhaps not, if you use any non standard features of*
- > > > *your database server, then write some additional functions as wrappers*
- > > > *for these. It is anything but rocket science.*

>

- > > *So you have defined "elegant" as "abstraction" and expect the rest*
- > > *of the programmers to agree that that's it?*
- > > *Thanks for solving that problem for the rest of the world.*

>

- > *Se here is a good example of your jumping up and down waving around a*
- > *fallacy a s if it was a point.*

>

- > *I did no such thing, I only explain what an elegant solution might be*
- > *//in this specific case// just as it says.*

>

- > *I never claimed to solve the general problem of elegant coding for the*
- > *rest world, this is just you wildly contorting yet again.*

"you can solve this with your own wrappers through elegant coding".

- > > > *What about the human and financial load? As in the load on the DBA,*
- > > > *inhouse developers, consulting budgets and application support staff?*

>

- > > *The load on the DBA depends on the problems the application makes.*

Re: What so special about PostgreSQL and other RDBMS?

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

> > *That typical increases if the application ignores load reducing features for > the sake of being generic*
>

> *And so does constantly changing everything to support different
> databases when he finds your unstarcted application does not use the
> database that all his other applications do.*

Believe me, it's easier to maintain three supported commercial databases than three unsupported (because app-vendor patched) mysql databases. there goes again your argument in favour of the private source code modification.

> > *They could have done half the app in PL/SQL and saved 90%
> > of the network and client load.*

>
> *And locked themselves out of the portion of the market which does not
> use PL/SQL, but rather something else, or simply does not want to
> bear the cost that using PL/SQL adds to the product not only on
> implementation, but also in anual licencing and support costs.*
The cool thing is they *still* required oracle. The other cool thing is that somehow all the big customers (i.e. the ones with real money) aren't really interested in mysql because of all the missing backup/recovery and standby tools.

>
> > *Also, if the database is not the standard one (because you have
> > fixed/improved it) I have, at the worst, maintain two independent
> > installations,*
>
> *No, you only have to maintain the one you actuall have in production.*
Per app. With a commercial db *I* tell them which version to use.

> *In most cases it does not make sence to build your application to
> depend on Oracle, or any thing else, exclusively. However there are
> certainly worse products to be dependant on, MS SQL for example.*
Even this depends. If you are programming exclusively for windows it's not a bad choice.

> > *We are talking about open source versus commercial databases.*
>
> *Again, if by 'We' you mean some imaginary person the rest of can't see
> or hear, please ignore my intrusion, however if you mean You and I, we
> are not.*

Please read the first article of that thread.

>
> *We are talking about two different things, the advantages of source,
> and the advangates of abstarction of access,*
No. You are trying to change the topic.

> *I have made no comments
> in this thread regarding commercial versus open source databases
> except to agree that the commercial ones _do_ have more features, that*

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

> *alone however does not always*
> *make them the best choice.*

See above.

>
> > *I picked*
> > *those two as examples because I have worked with both of them.*

>
> *Great, sadly however, not relevant.*

Which ones have you worked with?

> *However, a closed source contract is designed to hold you hostage, and*
> *to keep competitors away.*

Forget about this okay? A contract gets designed by both parties, even if the other side in this case is represented by oracles marketing department that actually wants to sell the contracts. To big companies with big legal departments.

And no one forbids me to run oracle and db2 and sqlserver on the same machine.

>
> > *So far no one has complained.*

>
> *No one you know is not no one.*

>
> > > *because Database security can only depend on it, not being able to*
> > > *actually protect devices, which is the burden on the OS and networking*
> > > *environment.*

>
> > *The os protects devices, not the network. Or, daring to think the*
> > *unthinkable,*

>
> *The OS is a part of Network security, what manages user priviledges?*
the OS.

> *The Switch? What controls device permissions? Your ethernet cables?*
The OS.

>
> *Your network security is a product of the collection of OSes that make*
> *up the nodes of your network. And the network is exactly as secure as*
> *the weakest node.*

Actually for a database it does not matter what the switch or anyone else says. What comes through the listener is what counts. Look up the oracle architecture, or any other commercial one.

>
> > *do you mean that you consider it ok to have database data on nfs mounts?*

>
> *See, you have just provided an example of how bad network security can*
> *undermine good database security, there are plenty of others as well.*

No I haven't. Because having nfs tablespaces isn't good database security,

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

because it makes a database dependent on the security of the file server. Which is not a network issue.

>
> *My point, once again, is that you can only have Database security,*
> **IF* you have a secure network, which means that the nodes on it are*
> *secure.*

You can make a database so insecure that it depends on network security. It's a totally different thing.

>
> > > *What does reading text files have to do with Chip design?*
>
> > *Because some tool will have to parse the text and create the chip out of it.*
>
> *Yes, that tool being the Application, the very thing following my*
> *advice will help you protect.*
> *Also, not all data is about creating*
> *chips, in many cases the data is the purpose of the application, and*
> *can outlive it, sometimes it must, by law, be accessible for a really*
> *really long time,*

Yes, and do you know what people do then?

They convert the .prn or .ps files to TIFF or
keep one machine with the original software somewhere in a
climatized room and hope like hell the contacts don't rot.

We do have seven years liability here and storing bitmaps
of everything is what we do. And recopying them.

And no, we can't feed them into any current system, if we need to
deal with any seven year old problem we just print out the stuff
and look at it.

If you have that kind of advice to dispense, go and
ask for a job at Lufthansa or Siemens Medical. They are just
waiting for people like you.

> *like in the case of public data, as I said. In this*
> *case in particular, keeping your data in a self contained, self*
> *describing, human readable file format is good sense. That is why*
> *things like XML and dublin core get invented.*

See above. Just accept it. You really don't know what you are talking
about here.

> > *This tool typically costs in the range of USD100000–200000 for a synopsys*
> > *ASIC compiler. You need the same tool because any other tool creates*
> > *totally different designs, ignores the original constraints and rules and*
> > *uses a different library which may even force a complete redesign.*
> > *Compared to that, a database migration is truly a breeze.*

>
> *Then your data does not have a long life span, so why are you*
> *presenting it as an argument, when my advice was specifically qualified*
> *to "ensure the permanency and portability of your important data?"*

See above.

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

>

> *If your data does not need to be either permanent nor portable, why are you discussing this, do you really imagine that because your data does not need to be permanent or portable, that therefore no data needs to be?*

I'm saying is that regardless of the need, data simply *isn't* permanent or portable. Go, ask a bank why they still have the old mainframes running. They should have the money to do things proper, right?

> > *I can read*

> > *text files I created on my Apple][, and no, I do not have the original hardware (well maybe my mom does somewhere in her basement).*

>

> > *Not all textfiles are notices for you to read.*

>

> *Yet some are, and for this data my advice holds true, I have never implied that all data must be kept accessible forever, rather advising on what to consider when it does.*

>

> > *Which ones? That abstracting access to suspect dependencies is a good idea?*

>

> > *That elegance is abstraction.*

>

> *The quote says "That abstracting access to suspect dependencies is a good idea" not "elegance is abstraction"*

Yes, you changed that one from your original posting.

>

> *Here you are jumping up and down again.*

>

> > *That database security is secondary to network security?*

>

> > *Yes.*

>

> *It is, if you ask a security expert you will find they agree with me.*

>

> > *That*

> > *one should keep archives in a format that is likely to be readable*

> > *forever?*

>

> > *Yes.*

>

> *Instead, archives should be kept in a format that can not be readable forever? What do you think archives are for? I don't mean simple backups.*

"This program has 400 features. No one will be able to use it."

"Ok, we'll add "ease of use" to the list."

Data isn't permanent out of choice but because no one has found a good solution yet. Go, do your own, get rich. You've got my blessing and me as a paying customer if it works.

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

> *You mean the same SAP that developed the Open Source SAP DB*
It was old and they released it because they've milked it for what it was worth.

> *and is now*

> *working with MySQL DB in making it MaxDB?*

Yes. And all the big installations run oracle.

If they get mysql to their feet support like, we'll have another look.

Besides, THEY are big enough to buy mysql ab if they want to, right to the source code or no right to the source code.

Also, they invest money to spite oracle, which goes into SAPs market by trying to buy peoplesoft. (I'm not saying that SAP's decision isn't right, or oracle is blessed, ok?)

So, access didn't have much to do with it and it certainly isn't cheap for them.

>

> *In anycase, I'm not intersted in what you are working on. It's*

> *irrelevent and it sounds banal. Nor does it in anyway strengthen your*
> *arguments.*

I have put up evidence in terms of real world experience. What you think of as banal or not is absolutely irrelevant to me.

> > *whereas you are the only one who thinks my arguments are rubbish.*

>

> *How do you know what everybody thinks? you think what is posted in*
> *this thread represent what everyone thinks?*

Ok, you are the only one who posts this. OTOH I know the people in a lot of other groups and have been corrected by them before plenty of times .

> *Thanks, from now on I will never abstract my database access, ignore*

> *network security, refuse to accept source code for any dependency of*

> *my applications, insist on being locked in to single source for all my*

> *support contracts and always, always keep my archives in an*

> *incoprehensible filesystem blob that I can only access by way of a*

> *third party, closed-source daemon.*

You know, now that you say it it doesn't sound too bad?

You get a fast, efficient application, can draw on the development and support expertise of other large successful enterprises and if your customer wants permanency you can actually *ask him* how he would have it done without trying to force your solution on him.

> > > *that the very concept of good programming is an illussion,*

>

> > *No, it's just that so far no one has found out what it is, because*

> > *despite all the attempts software still is not substantially more stable*

> > *than software written 30 years ago.*

>

> *So we should not try to write good programmes then? Quick, someone tell*

> *Don Knuth.*

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

The guy who still writes in TECO assembler? He might know algorithms but he sure doesn't know coding.

> *Again: Not having source is a guarantee that one CAN NOT fix bugs.*

Again: I don't have to fix them. I have to have someone fix them. Namely the guys I've got the support contract with.

> > *And if you are worrying about expiring licences, for many products*

> > *(purify and our oracle installation spring to mind) you get permanent*

> > *licences and pay yearly for support, so I can still use the app when the*

> > *vendor goes bust.*

>

> *Who will fix the bugs when the vendor goes bust? Or compile it for*

> *your new OS, or your new CPU? Or to link a updated library for which*

> *there is a security patch?*

No one. Who ports mysql to my cellphone?

> *And trim your posts better, you don't need to quote every line in the*

> *previous post, only the ones you actually respond to.*

The way you misunderstand me I'm tempted to leave everything in.

> *Sometimes it's best to change companies and keep the product,*

Hm. Support from boeing for an airbus.

> *And you get what you pay for, do not imagine they will consent to*

> *losing money on you for long if their costs go above your flat rate.*

I don't care how much they win or lose as long as they fix the bugs.

Which they do. If they don't, I can still change.

>

> > > *not that anything will be accomplished.*

>

> > *Then they lose money if they don't accomplish anything.*

>

> *Right, if fixing it costs them more that what you are paying them,*

> *then they desupport you,*

No. See my other post.

> > > *Many large companies, and profesional develpoers provide source*

> > > *licences and/or support open source products, including the largest*

> > > *computer company in the world, IBM.*

>

> > *Yep, so I can buy support, mess up the code I've access to and let*

> > *IBM sort it out, is this what I get by using a IBM supported mysql?*

>

> *Who is the developer, you or IBM? If you are hiring IBM, why are you*

> *messing with the code?*

If not, why do I need the source code?

> *I'm sure, if you are willing to pay them*

> *enough, IBM corporate services will indulge this crazy plan of yours,*

Re: What so special about PostgreSQL and other RDBMS?

comp.lang.php: Re: What so special about PostgreSQL and other RDBMS?

- > *but they will probably at least suggest you decide wether it is you*
- > **OR* them who are developing the code, and if you already have screwed*
- > *it up, perhaps they might prefer to start with a fresh copy from MySQL*
- > *AB.*

Great. We're making headway here.

Greetings!

Volker