

## Re: Database Connectivity

**Source:** <http://coding.derkeiler.com/Archive/Python/comp.lang.python/2004-10/1863.html>

---

**From:** R Baumann (rynt\_at\_9yahoo.com)

**Date:** 10/11/04

Date: Mon, 11 Oct 2004 11:17:30 -0700

"Anna Martelli Ravenscroft" <anna@aleax.it> wrote in message  
news:Bdhad.18298\$b5.899971@news3.tin.it...  
> *Anna Martelli Ravenscroft wrote:*  
> > *Alex Martelli wrote:*  
> >  
> >> *Dennis Lee Bieber <wlfraed@ix.netcom.com> wrote:*  
> >> ...  
> >>  
> >>>> *I want to develop a simple Database connectivity program with  
python.*  
> >>>> *I want to use MS Access as a backend.*  
> >>>  
> >>>  
> >>> *Access, itself, is merely a GUI builder front-end. You are  
> >>> likely using M\$ JET as the database engine.*  
> >>  
> >>  
> >> *Amen! People keep making this mistake over and over and over again.*  
> >> *Access is a product Microsoft sells; Jet is one they give away for  
free;*  
> >> *etc, etc -- how CAN it be so hard to distinguish?!*  
> >  
> >  
> > *How can it be so hard to distinguish?*  
> >  
> > *Gee - how about - cuz M\$ doesn't bother telling you? It \*says\* M\$ Access  
> > - right there on the package and on the computer. It doesn't say M\$  
> > Jet... I used M\$ Access for years, even took classes in it, did some  
> > pretty kewl stuff with it. Never was Jet mentioned anywhere at anytime.  
> > I found out from a geek buddy by accident one day...*  
> >  
> > *BTW: I just did a google check:*  
> >  
> > *the Product Overview page for M\$ Access 2003 doesn't mention Jet once!*  
> > <http://www.microsoft.com/office/access/prodinfo/overview.mspx>  
> >

> > *Nor does the Data Access and Storage Developer Center page!*  
> > <http://msdn.microsoft.com/data/Default.aspx>  
> >  
> > *and it ain't just M\$. I found a lovely page called AccesSolutions:*  
> > *Microsoft Access Database Solutions, with not a single mention of Jet.*  
> >  
> > *In other words, if you didn't already \*know\*, you're damned unlikely to*  
> > *find out until some "real" database geek gives you a hard time about not*  
> > *knowing.*  
> >  
> >  
> > *Anna*  
>  
> *Just to follow up on this – I think this is another case of M\$ evility.*  
> *They \*like\* that people think of Access as a database – cuz if people*  
> *realized that they could use another database, they might realize they*  
> *could also use \*another\* front end.*  
>  
> *So, I really don't think it's deliberate blurring of the lines by the*  
> *\*users\* of M\$ Access, so much as by M\$, and the poor users simply don't*  
> *know any better.*  
>  
> *I want to emphasize that I understand how frustrating it is to keep*  
> *coming across the same myths and misunderstandings, and that c.l.py is*  
> *great for gently pointing out the real information to folks.*  
>  
> *Anna*

I'm a database geek. I make damn good money developing and maintaining database apps. And in the corporate world, a database of some sort is pretty well THE application they're concerned with. And, since c.l.py is so great for GENTLY pointing out the real information to folks here's the real scoop on ms-access...

1. Since version 1, ms-access has ALWAYS been two products, the Access front end, and the underlying database engine, JET. Since Access has always been sold as a single product, the JET database engine has just naturally been referred to as the "Access database engine". Technically wrong, but it's just easier to think of it that way. BTW, JET technology is the current underlying database for windows itself when it needs to use a db. (XP only I think. This may change when longhorn is released).

The original purpose of Access, and in many cases still is, was to be a Desktop Database that made it easy for a non-programmer user to create and maintain their own simple contact lists, address books, etc and be able to create and run reports against those db's. And that worked and still works really well for that aspect. Developers got hold of the product, realized they had a real good opportunity, and for years have been making a good living from Access. Many of the upgrades that have come from Access through the years have been due to developer efforts working with Microsoft. Thousands and thousands of custom data base apps have been written using

Access, some using the JET engine, some using another DB engine altogether, like SQL Server, Sybase, MySQL, etc.

2. Access is a total RAD programming environment that is optimized to do ONE thing very well, build and manage databases, although it can be used to do other things as well. Access has ALWAYS had Forms, A query writer, table management, database creation, a report writer(MUCH better than Crystal IMO), and as a built-in language for getting down to the "metal" when you have to, Visual Basic for Applications.

Yeah, yeah, I've heard it all before. BASIC ugh!, Not a real language, etc., etc., etc. If BASIC is not a real language, why do so many languages, including Python have more similarities than differences to it? This is a fair question.

And this is MY OPINION! MY answer is that BASIC, with all it's perceived bad aspects is very easy to learn and understand. An experienced BASIC professional programmer can do one hell'uva good job, and the code won't look like spaghetti, and probably the app will be released into production with fewer errors, and much faster than a C/C++ programmer could make it happen. And as far as writing spaghetti code, huh!, you can do that in ANY language, goto's or not, structured programming techniques or not.

3. When Access 2000 was released, MS included a version of MS-SQL Server called the MSDE(Microsoft Database Engine) that was a deliberately watered-down version of MS-SQL Server 7. By watered down, I mean that MSDE could handle up to 5 concurrent PROCESSES before a governor kicked in to deliberately slow things down. This was to prevent MSDE being used in areas where an industrial strength DB Server needs to be used. A marketing issue? You bet. But, MSDE has been know to work very reliably in small user LANS, and workgroups of up to about 25 users with a medium to somewhat heavy workload. And, if the workload is fairly light up into the hundreds. If the workload is light, OR, ESPECIALLY if the developer knows what they're doing, the governor won't kick in that often. Processes can be made to be efficient. Unless you're a huge corporation with hundreds or thousands of users, this is more than good enough to handle most small companies, and if those companies grow larger, the upgrade path to full blown SQL Server is just a simple step to attach the tables from MSDE to SQL Server.

<GASP!> M\$ sells stuff for money! Why not? I like to eat, you like to eat. Don't like Microsoft?, sounds like a personal problem to me. I don't like everything they do either, but to pay my salary, the company I work for has to make money, not live on indignant or righteous attitudes. I use whatever works best for the particular customer I have to work with. If MS has a product that does what I need it to, I'll use it. If Python is a better way to do things, I'll use it, if QBasic, if Delphi, if...

4. Don't like SQL Server? use ODBC. I've done Access apps using Btrieve, MySql, Interbase/Firebird and others as the backend data engine, and Access as the GUI front end. My development time in Access is 1/3 to 1/2 the time, and with fewer errors.

5. JET is no longer being actively enhanced. It IS being maintained. Eventually it will fade away, but the MSDE will still be there. Access is now up to version 2003 just in case anyone is interested. MSDE is up to version 2000 or perhaps more by now. It's a free download from the MS website. There ARE some licensing issues, so pay attention to them if you download. Not everything in the world has to be or should be "free" or "open source".

Don't get me wrong. I love the idea of Open Source. I make use of it quite often. My problem with it is when the primary developer gets tired of the project, which often happens after a year or two, it generally dies because no-one is interested in taking over, or the project stagnates. MS never said they were in business to produce free or open source software, and at least they don't let their products stagnate. It takes money to build and maintain a Word, or an Excel, or an Access, and sometimes I absolutely need to use a product I know is going to be around a while, and be supported.

6. Finally, MS is positioning Access to be a GUI front-end to any database, and the hooks are already there. Look for a .NET version in the future.

Sorry about the long rant, but so much mis-information has been passed along in this thread and in past threads, mainly due to so much MS hatred, deserved or not, that I felt I had to speak up. I get the impression that most of the info comes from people parroting other people, and not really knowing about the products that they are dis'ing. If you're really interested in seeing what Access CAN do, take a look in particular at the following newsgroup:

comp.databases.ms-access.

Also look at the Microsoft Public Access newsgroups. They all offer help and information, some better, some worse, and sometimes it's even done as gently and correctly as in c.l.py.

RB