

Re: Extended Stored Procedure: Get the current db of the client

Source:

<http://www.tech-archive.net/Archive/SOL-Server/microsoft.public.sqlserver.odbc/2006-05/msg00021.html>

- *From:* Michael C <MichaelC@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Mon, 15 May 2006 07:01:02 -0700
-

"Gert E.R. Drapers" wrote:

No, you are incorrect; for an extended stored procedure you have to pass in the database context as a parameter if you need it, that is the only thing that works. Did you ever write an extended stored procedure?

I have written several, several, several extended stored procedures. In fact, I just publicly released about 3 dozen that cover everything from AES, Blowfish, Twofish, DES and TripleDES encryption to regular expressions to recursively reading a local subdirectory listing.

In fact, here's a little experiment for you extended procedure maestro: Put this regular stored procedure in the Master database:

```
CREATE PROCEDURE dbo.Test1
AS
SELECT db_Name()
GO
```

Now run it from within the Model database. Or the Northwind database. What database name comes up? Master, that's what. According to your solution, you need to recreate this exact same stored procedure in every single database you own in order to get the current database context out of it.

As I said: changing the name to "sp_..." and marking it as a system object will allow you to use JUST ONE copy of the stored procedure in Master. It will run in the context of the CURRENT DATABASE, no matter what database you invoke it from. But I'm sure you're well aware of that.

Besides that it does not make sense to call the DB-Lib function dbname() until you established a loopback connection over DB-Library, which would default to the default database for the user which is not the same as the database context. See the attached example which shows this behavior.

Re: Extended Stored Procedure: Get the current db of the client

And that's all well and good. I was simply pointing out some things that might be tried, and you pointed out that it wouldn't work in your own little snide way.

The `srv_rpc*` class methods in the `OPENDS60.LIB` file are obsolete since they are gateway calls and not longer supported; `srv_rpcdb()` only gave you a database context when you where a remote procedure, which is something different than an extended stored procedure, so that is not giving you what you want either.

I know `srv_rpcdb` doesn't work, and didn't suggest it as a solution. I'm sure whoever didn't know that will be happy to hear it from you, however.

So Mike C#, the ONLY solution is to pass it in as a parameter!

Which is fine, and perfectly acceptable. The difference is simply this, if you refer back to my original post: Your method requires the same stored procedure be copied to all 28 of my databases. Alternatively I can put a single copy in the Master database and be done with it.

BTW: Next time you are calling somebody names you might want to check your facts before replying an making a fool out of yourself.

BTW: You should check your facts before you accuse someone of not having any experience in your little domain over there before making a fool of yourself.

<http://www.sqlservercentral.com/columnists/mcoles/sql2000dbatoolkitpart1.asp>
<http://www.sqlservercentral.com/columnists/mcoles/sql2000dbatoolkitpart2.asp>
<http://www.sqlservercentral.com/columnists/mcoles/sql2000dbatoolkitpart3.asp>
<http://www.sqlservercentral.com/columnists/mcoles/sql2000dbatoolkitpart4.asp>

Of course I'd love an opportunity to learn at the master's feet. So where does Master Gert keep his extended procedures, that I may immerse myself in the knowledge to be gained?

.