
Subject: Re: Using SQL databases from IDL?

Posted by [lecacheux.alain](#) on Tue, 27 Nov 2012 09:04:28 GMT

[View Forum Message](#) <> [Reply to Message](#)

Le lundi 26 novembre 2012 00:47:00 UTC+1, Bogdanovist a écrit :

> Does anyone have experience using SQL databases from IDL? Note that I am referring to vanilla IDL without additional license requirements such as Dataminer. I have found, for instance, a package idl-sql (<http://code.google.com/p/idl-sql/>) that allows SQL queries from IDL, but I also would need to be able to write to the database, as well as read.

>

>

>

> I currently use simple CSV files to store data however the kind of data queries I need to do means that I am effectively re-coding a relational database within my IDL project in order to be able to provide myself a simple interface to grab the relevant data at any point that is likely to be distributed across multiple files. Can I use an SQL database to avoid re-engineering that wheel or does IDL not play nice with this?

A general solution for accessing SQL databases may consist in using ODBC (or JDBC) universal drivers via IDL COM (or Java) bridges. That is likely the solution adopted in the "idl-sql" software. You can develop a more powerful solution, still by using IDL bridges, but by bypassing ODBC and directly accessing the driver specific to your particular database. In MS world, I am personally happy with the combination IDL(COM)bridge/Delphi from Embarcadero for accessing various SQL databases like Firebird, SQLserver or Oracle.

Another plain IDL solution would consist in accessing some Web service, if any does exist for your database. I mean a Web server able to transmit SQL query and deliver HTML data to and from a remote database server. In this case the IDLnetURL and IDLffXMLSAX IDL objects, as well as the new (8.2) JSON_PARSE and JSON_SERIALIZE functions are very appropriate.

alain.

Subject: Re: Using SQL databases from IDL?

Posted by [John Correia](#) on Tue, 27 Nov 2012 13:55:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

On 11/25/2012 06:46 PM, Bogdanovist wrote:

> Does anyone have experience using SQL databases from IDL? Note that I
> am referring to vanilla IDL without additional license requirements
> such as Dataminer. I have found, for instance, a package idl-sql
> (<http://code.google.com/p/idl-sql/>) that allows SQL queries from IDL,
> but I also would need to be able to write to the database, as well as
> read.

>

> I currently use simple CSV files to store data however the kind of
> data queries I need to do means that I am effectively re-coding a
> relational database within my IDL project in order to be able to
> provide myself a simple interface to grab the relevant data at any
> point that is likely to be distributed across multiple files. Can I

- > use an SQL database to avoid re-engineering that wheel or does IDL
- > not play nice with this?

I've been using `pgsql_query` (https://github.com/segasai/pg_idl) on a Linux box to (mostly) read from a PostgreSQL database and have been very pleased with it.

John

Subject: Re: Using SQL databases from IDL?
Posted by [Matt Francis](#) on Tue, 27 Nov 2012 21:27:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks for the replies. Indeed it looks like using the IDL Java bridge is the best bet. This is actually convenient as I have a colleague who works on the same data using Java code. Does anyone have a sense of the computational overheads involved in this? I have some tasks that run in near real time and need to stay lean in order to 'keep up' with the walltime. If I replace text file I/O using standard IDL `readf/printf` with DB access using the Java bridge, is this likely to be significantly slower?

Subject: Re: Using SQL databases from IDL?
Posted by [lecacheux.alain](#) on Wed, 28 Nov 2012 09:15:21 GMT
[View Forum Message](#) <> [Reply to Message](#)

Le mardi 27 novembre 2012 22:27:34 UTC+1, Bogdanovist a écrit :

> Thanks for the replies. Indeed it looks like using the IDL Java bridge is the best bet. This is actually convenient as I have a colleague who works on the same data using Java code. Does anyone have a sense of the computational overheads involved in this? I have some tasks that run in near real time and need to stay lean in order to 'keep up' with the walltime. If I replace text file I/O using standard IDL `readf/printf` with DB access using the Java bridge, is this likely to be significantly slower?

I think that the overhead due to using IDL bridge should be negligible with respect to writing over the network (remote database) or to disk (in case of local database).
But some issue can arise from the fact that usual databases do not guarantee **real-time** accessibility: at a given time, even in case of non concurrent access (single user) some database engines can temporarily get busy by demanding tasks like garbage collecting or index reorganizing.
alain.
