Subject: Re: building argument... [proposal for new DataMiner-functionality]
Posted by Olaf Stetzer on Mon, 15 Oct 2001 09:58:20 GMT

View Forum Message <> Reply to Message

Tim Williams schrieb:

>
> I have the rows of a table widget that I want to insert to a database
> table via Dataminer. For example, if I have an array of structures
> called values from a table widget, then I would like to use addRecord
> for each row in the table. I thought I could just do
>
> ors->addRecord, values[i]
>
> but values[i] is a structure, so that isn't right, and addRecord is
> expecting a list of fields. I tried building up a string containing
> the fields and then called addRecord with
>
> ors->addRecord, arglist
>
> but only the first field was written. (It turned out that the first
> field was also a string, but other fields weren't) I believe what I
> need to do is get this structure into a list of arguments and call
> addRecord with the correct number of arguments, but I don't know how
> to build this. I'm think of something like 'apply' in Python.
>
> One other idea I haven't tried yet. Should I build up a SQL command
> and call sqlExecute?
>

I suggest using the latter if you are comfortable with it. I do a lot
of connections directly with IDL since I believe that the other
Dataminer-routines lack a lot of basic functionality. Maybe we should
propose some suggestions to improve Dataminer? I had the feeling that
a lot of people that tried (or intended to try) Dataminer are not happy
with the functionality....

Here are some thoughts I already had:

Given a recordset all "communication" to/from the recordset should be
possible by using a structure reflecting the contents of the recordset
(the struct you get when calling ->GetRecord)

Using this struct one should be able to set a NULL-mask containing the
values that should be returned instead of NULL when ->GetRecord is
called.

As you already pointed out the preferred way to add a record should
go via a struct not a arglist.

Some more comments:

I only tried to create a recordset with table=... once, this was so awfully slow that I refuse to try this again. This must be enhanced, using the same table via ODBC from within Access takes less than a second!

There should be some error handling added, for example to catch errors that occur using executeSQL.

This list ist open for discussion, maybe we can collect some more suggstions and then send them to RSI?

Olaf

--
Dr. Olaf Stetzer
Forschungszentrum Karlsruhe
Institut fï¿½r Meterologie und Klimaforschung
Atmosphï¿½rische Aerosole (IMK III) - http://imk-aida.fzk.de
Tel.: +49(0)7247-82-3249 (FAX: -4332)