Subject: Possible SqlOption Problem [Solved]
Posted by rbmatt on Fri, 04 Aug 2006 21:58:49 GMT

View Forum Message <> Reply to Message

My application is currently using two SqlOption checkboxes which are stored into the database (SqLite or MySQL). I've been having some problems though when retrieving the values.
The following code crashes with an assert:
if(list.CListView.Get(SERVICE))
Assertion failed in C:\uppsrc\Core/Value.h, line 417
dynamic_cast<const RichValueRep *>(p)
It should be returning int(0,1) bool(true,false) or NULL. It looks like it's returning a RichValue? I don't know enough about U++ yet to track it down.
BTW Mirek, this is probably the same thing that made me put in an AsString.

Subject: Re: Possible SqlOption Problem
Posted by mirek on Sat, 05 Aug 2006 04:54:00 GMT

View Forum Message <> Reply to Message

I am sorry, it is really not very intuitive, but to conserve the space in DB, scheme BOOL is represented by string with values "0" and "1". That is also the difference between Option and SqlOption (regular Option uses bool).

Therefore you should be using

if((String)list.CListView.Get(SERVICE) == "1")

Mirek

Subject: Re: Possible SqlOption Problem
Posted by rbmatt on Sat, 05 Aug 2006 05:24:30 GMT

View Forum Message <> Reply to Message

Makes sense. Thanks.

Subject: Re: Possible SqlOption Problem
Posted by marroyo on Fri, 14 Aug 2009 19:39:05 GMT

View Forum Message <> Reply to Message

I've same problem when using a SqlOption in a PostgreSQL application. The column is a bool type PGAdmin III show column values as true or false.

Assertion failed in C:\uppsrc\Core/Value.h, line 417
dynamic_cast<const RichValueRep *>(p)

Assertion fails on sql_array.Query();

Using Option doesn't work because it return a int and then insert fails.

Any help?

Subject: Re: Possible SqlOption Problem
Posted by mirek on Sun, 16 Aug 2009 21:58:55 GMT
View Forum Message <> Reply to Message

marroyo wrote on Fri, 14 August 2009 15:39I've same problem when using a SqlOption in a PostgreSQL application. The column is a bool type PGAdmin III show column values as true or false.

Assertion failed in C:\uppsrc\Core/Value.h, line 417
dynamic_cast<const RichValueRep *>(p)

Assertion fails on sql_array.Query();

Using Option doesn't work because it return a int and then insert fails.

Any help?

Well, I do not really have enough info.

Anyway, SqlOption provides String - because some DB do not support "bool" type.

If the "bool" in DB is real bool, I believe normal Option should work fine.

Mirek

Subject: Re: Possible SqlOption Problem
Posted by marroyo on Mon, 17 Aug 2009 13:39:25 GMT
View Forum Message <> Reply to Message

Hi Mirek. I will be more explicit now.
I have a PostgreSQL table SOCIO with a activo (active) boolean field.

I've defined a window with a SqlArray object.

SqlArray show's me the boolean column as value 1.

I want edit fields, so I add a column to SqlArray by doing

socios_array.AddColumn(ACTIVO, "Activo").Edit(activo);

If I declare activo of SqlOption type, when I try to do socios_array.Query(), I get the following assertion failure:

Assertion failed in C:\uppsrc\Core/Value.h, line 417
dynamic_cast<const RichValueRep *>(p)

If I declare activo as Option type, I can't edit (or insert) rows because the SQL statement generated is

insert into SOCIO(OBJID, RAZON_SOCIAL, ACTIVO, CATEG_IVA) values (801, 'MARCELO', 1, 'C')

You can see that ACTIVO column value is an integer (not a boolean), getting a database error from PostgreSQL.

I could fix it doing a Convert from Option, but I think it is the goal of SqlOption.

Thank you.

Subject: Re: Possible SqlOption Problem
Posted by mirek on Mon, 17 Aug 2009 18:18:30 GMT
View Forum Message <> Reply to Message

Hm, so does it mean it does not accept 1 0 as boolean values?

That is pretty bad... In that case we might have to develop a special type for this purpose...

Mirek

Subject: Re: Possible SqlOption Problem
Posted by marroyo on Tue, 18 Aug 2009 18:25:41 GMT
View Forum Message <> Reply to Message

PostgreSQL doesn't accept 0, 1 as boolean values.

Extract from PostgreSQL manual:

Valid literal values for the "true" state are:

TRUE
't'
'true'

'y'
'yes'
'1'

For the "false" state, the following values can be used:

FALSE
'f'
'false'
'n'
'no'
'0'

For now, I've to define a Convert for Option.

Thank you, Mirek.

Subject: Re: Possible SqlOption Problem
Posted by mirek on Fri, 21 Aug 2009 12:27:16 GMT

marroyo wrote on Tue, 18 August 2009 14:25PostgreSQL doesn't accept 0, 1 as boolean values.

Extract from PostgreSQL manual:

Valid literal values for the "true" state are:

TRUE
't'
'true'
'y'
'yes'
'1'

For the "false" state, the following values can be used:

FALSE
'f'
'false'
'n'
'no'
'0'

For now, I've to define a Convert for Option.

Thank you, Mirek.

Hm, now looking at THIS, maybe the correct solution would be to convert Postgre bool (in to "0" or "1" (now it is getting converted to numeric value). In that case, SqlOption would still be working and Insert would accept "0" "1" values just fine. What do you think?

Something like:

```
void PostgreSQLConnection::GetColumn(int i, Ref f) const
{
.....
  case BOOL_V:
   f.SetValue(*s == 't' ? true : false);
   break;
```

to

```
void PostgreSQLConnection::GetColumn(int i, Ref f) const
{
.....
  case BOOL_V:
   f.SetValue(*s == 't' ? "1" : "0");
   break;
```

Subject: Re: Possible SqlOption Problem
Posted by marroyo on Fri, 21 Aug 2009 17:12:34 GMT
View Forum Message <> Reply to Message

I think that it should be the best solution. I'll try it and let you know.

Thanks again Mirek, very much.

Subject: Re: Possible SqlOption Problem
Posted by marroyo on Wed, 26 Aug 2009 21:37:01 GMT
View Forum Message <> Reply to Message

Mirek, it works!

I think PostgreSQL users will have the same problem. It should be a permanent change in the PostgreSQL package.

Thanks, again.

Subject: Re: Possible SqlOption Problem
Posted by mirek on Fri, 28 Aug 2009 07:54:56 GMT
View Forum Message <> Reply to Message

marroyo wrote on Wed, 26 August 2009 17:37Mirek, it works!

I think PostgreSQL users will have the same problem. It should be a permanent change in the PostgreSQL package.

Thanks, again.

It is now

Mirek