Subject: Prefix to prevent name clash in SQL: SqlId, variables, tables, ...
Posted by mingodad on Sun, 17 Feb 2008 12:40:01 GMT
View Forum Message <> Reply to Message

Today I was converting one application to upp 2008.1 and found that the actual system of parse database schema can generate name clash:

```
TABLE_(CURRENCIES)
 INT_ (CURRENCY)
END_TABLE
```

The above will generate a name clash:

```
----------
main.cpp
In file included from d:\upp\uppsrc/Sql/sch_model.h:162,
                 from d:\upp\uppsrc/plugin/sqlite3/Sqlite3Schema.h:82,
                 from d:\upp\uppsrc/Sql/sch_header.h:45,
                 from d:\MyApps\DADBiz\/DADBiz.h:17,
                 from d:\MyApps\DADBiz\main.cpp:1:
d:\MyApps/DADBiz/dadbiz.sch:33: error: 'Upp::SqlId CURRENCY' redeclared as different kind of symbol
d:\upp\mingw\include/oaidl.h:113: error: previous declaration of 'typedef union CY CURRENCY'
In file included from d:\upp\uppsrc/Sql/sch_model.h:162,
                 from d:\upp\uppsrc/plugin/sqlite3/Sqlite3Schema.h:82,
                 from d:\upp\uppsrc/Sql/sch_source.h:6,
                 from d:\MyApps\DADBiz\/DADBiz.h:18,
                 from d:\MyApps\DADBiz\main.cpp:1:
d:\MyApps/DADBiz/dadbiz.sch:33: error: 'Upp::SqlId CURRENCY' redeclared as different kind of symbol
d:\upp\mingw\include/oaidl.h:113: error: previous declaration of 'typedef union CY CURRENCY'
----------
```

[FEATURE REQUEST]
So I studied the source code and end up with some modifications that allow us to specify a prefix to the generated symbols with a definition of one preprocessor macro:

```
#define SCHEMA_PREFIX_CPP _ANY_PREFIX_YOU_LIKE_
```

And after that any reference to tables or fields should be done as:

```
 dbList.AddKey(_ANY_PREFIX_YOU_LIKE_CURRENCY);
 dbList.SetTable(_ANY_PREFIX_YOU_LIKE_CURRENCIES);
 dbList.AddColumn(_ANY_PREFIX_YOU_LIKE_CURRENCY, t_("Id"), .SetFormat("%8>d");
    dbList.SetOrderBy(_ANY_PREFIX_YOU_LIKE_CURRENCY).Limit(100);
```

It doesn't break any application that already works, because without defining SCHEMA_PREFIX_CPP all generated symbols remain as before.

In this process I've found a bug in GCC when joining string in macros, in some cases it adds a

space betwen the tokens, that's why the macro ADD_SCHEMA_PREFIX_CPP2 was created.

I've sending attached the files I've modified.

I'll hope it will be usefull to others and probably be included in the official distribution too !

Thanks for all of you that help develop this fantastic tool !

If you need more explanations, ask me !

Ps:
[FEATURE REQUEST]
As well to allow applications made with upp to work with databases, tables generated by other programs that doesn't create table and record fields names in uppercase (commonly lowercase) a patch to sql.cpp to make comparisons after convert field names retrieved fron queries to upper case.

Instead of:

```
// sql.cpp line 254
Value Sql::operator[](SqlId id) const {
 String s = ~id;
 for(int i = 0; i < cn->info.GetCount(); i++)
  if(cn->info[i].name == s)
   return operator[](i);
 return Value();
}
```

Use this :

```
Value Sql::operator[](SqlId id) const {
 String s = ~id;
 for(int i = 0; i < cn->info.GetCount(); i++)
 {
  String sb = ToUpper(cn->info[i].name);
  if(sb == s)
   return operator[](i);
 }
 return Value();
}
```

File Attachments

1) sch_header.h, downloaded 637 times
2) sch_model.h, downloaded 675 times
3) sch_source.h, downloaded 932 times
4) Sql.cpp, downloaded 754 times

Subject: Re: Prefix to prevent name clash in SQL: SqlId, variables, tables, ...
Posted by mirek on Tue, 19 Feb 2008 22:42:05 GMT
View Forum Message <> Reply to Message

OK, why not. Path accepted.

Only I have rather placed ToUpper to Execute -> that way we will avoid ToUpper in each row fetch... (Equality comparison is fast, ToUpper is not...):

```
bool Sql::Execute() {
 SqlSession &session = GetSession();
 if(session.traceslow < INT_MAX)
  cn->starttime = GetTickCount();
 if(session.usrlog)
  UsrLogT(9, cn->statement);
 Stream *s = session.GetTrace();
 if(s) {
#ifndef NOAPPSQL
  if(this == &AppCursor())
   *s << "SQL* ";
#endif
  *s << cn->statement << '\n';
 }
 session.GetStatus().Statement(cn->statement);
 session.PassStatus(ActivityStatus::EXECUTING);
 bool b = cn->Execute();
 session.GetStatus().Time(GetTickCount() - cn->starttime);
 session.PassStatus(ActivityStatus::END_EXECUTING);
 if(!b) {
  if(s) {
   *s << "## ERROR: " << session.GetLastError() << '\n';
  }
  session.GetStatus()
   .Error(session.GetLastError())
   .ErrorCode(session.GetErrorCode())
   .ErrorCode(session.GetErrorCodeString());
  session.PassStatus(ActivityStatus::EXECUTING_ERROR);
 }
 for(int i = 0; i < cn->info.GetCount(); i++)
  cn->info[i].name = ToUpper(cn->info[i].name);
 return b;
}
```

(please check)

BTW, was your suggestion based on real problems or do you just want to play safe?

Mirek

Subject: Re: Prefix to prevent name clash in SQL: SqlId, variables, tables, ...
Posted by WebChaot on Wed, 30 Jul 2008 15:57:39 GMT
View Forum Message <> Reply to Message

Hi Mirek!

I found this article in forum related to my current problem:

I need to create grids (arrayctrl) via database tables. But because of these lines

```
 for(int i = 0; i < cn->info.GetCount(); i++)
  cn->info[i].name = Toupper(cn->info[i].name);
```

i allways get the column headers in upper case. Is it possible to put switch to suppress this auto-uppercase functionality?

WebChaot

Subject: Re: Prefix to prevent name clash in SQL: SqlId, variables, tables, ...
Posted by mirek on Fri, 01 Aug 2008 08:07:51 GMT
View Forum Message <> Reply to Message

WebChaot wrote on Wed, 30 July 2008 11:57Hi Mirek!

I found this article in forum related to my current problem:

I need to create grids (arrayctrl) via database tables. But because of these lines

```
 for(int i = 0; i < cn->info.GetCount(); i++)
  cn->info[i].name = Toupper(cn->info[i].name);
```

i allways get the column headers in upper case. Is it possible to put switch to suppress this auto-uppercase functionality?

WebChaot

Hm, I am afraid it is not the right approach.

Maybe we could add the name in the "original form" to ColumnInfo?

Mirek

Subject: Re: Prefix to prevent name clash in SQL: SqlId, variables, tables, ...
Posted by sevenjay on Mon, 30 Aug 2010 13:45:34 GMT
View Forum Message <> Reply to Message

I have another opinion on this.
When you create tables/columns with lower-case names, set SqlID like
SqlId TITLE("Title");
It will not match the table/column name "Title" because it will be "TITLE" in cn->info[i].name.
That is not convenient to force user to set SqlID all upper case.
SqlId TITLE("TITLE");

I think it could be case insensitive equality comparison.

if(cn->info[i].name == s)
could be
if(0 == s.CompareInsensitive(~(cn->info[i].name)))
The method CompareInsensitive maybe like this

```
template <class B>
int AString<B>::CompareInsensitive(const tchar *b) const
{
 const tchar *a = B::Begin();
 const tchar *ae = End();
 for(;;) {
  if(a >= ae)
   return *b == 0 ? 0 : -1;
  if(*b == 0)
   return 1;
  int q = cmpval__(*a) - cmpval__(*b);
  if( (cmpval__('a') < cmpval__(*a) && cmpval__(*a) < cmpval__('z')) ||
      (cmpval__('A') < cmpval__(*a) && cmpval__(*a) < cmpval__('Z')) )
   if(q == cmpval__('a') - cmpval__('A') || q == cmpval__('A') - cmpval__('a'))
    q = 0;
  if(q)
   return q;

  *a++;
  *b++;
 }
}
```

I tried it's OK.
But I am not sure if it is faster than ToUpper.