
Subject: Re: Ctrls<>() Callback / Disable some rows
Posted by [mirek](#) on Wed, 13 Jun 2007 07:33:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

kcabobert wrote on Tue, 12 June 2007 16:31

2) My ArrayCtrl has an Index column (via `AddIndex("primary_key","id")`), a column of options (via `AddColumn("use","Use").Ctrls<Option>()`), and a name column (via `AddColumn("name","Name")`).

Just a note: This was supposed to use those `Id` (or `SqllId`) things, like:

```
Id PRIMARY_KEY("PRIMARY_KEY"); // possibly global, "Id constant"
```

```
....  
AddIndex(PRIMARY_KEY, "id");
```

Using `char *` there works as well, but it is a but more error prone (you can mistype the string) and U++ has to perform `char * -> Id` conversion. That is especially not very helpful if you are looping through the `ArrayCtrl` values.

3.a) I need to SOMETIMES assign a callback to the option column that will uncheck all other options (like a radio ctrl), other times multiple selections are OK. I can do this if I can reliably figure out in which row the option that triggered the callback resides (I need people to be able to select via mouse or keyboard).

3.b) In some cases I may want one option completely disabled (in other words one row disabled). However, I still need the row to list. In these cases removing the option ctrl is acceptable.

Well, `Ctrls` part of `ArrayCtrl` seems to be really deficient.

I was thinking about the issue a lot and really the problem seems to be that we have stepped out of U++ paradigm here, introducing class factory, which is nice to solve simple problems, but fails with more complicated ones.

I plan to add some other interface for embedding widgets into `ArrayCtrl` that will rather use widgets that belong outside the `ArrayCtrl` (e.g. in the dialog). Something like:

```
struct MyDialog {  
    Array<Option> option;  
  
    void OptionColumn(Ptr<Ctrl>& ctrl, int index)  
    {  
        ctrl = &option.At(index);  
    }  
}
```

```
MyDialog() {  
    list.AddColumn("Options").Ctrls(THISBACK(OptionColumn));  
}  
};
```

This way, you would have the full access to the list of options.

What do you think?

Mirek
