Subject: ClearMultisort being called after total_cols reduced to 1 [BUG?]
Posted by alendar on Thu, 30 Dec 2010 23:00:21 GMT
View Forum Message <> Reply to Message

Happy New Year, all!

I'm trying to reuse a GridCtrl in a custom Sql tool that resets the grid according to different Sql result sets, like SqlCommander does (except with an ArrayCtrl).

I'm getting an out-of-bounds error when I do a Reset() if I've set the grid for MultiSort and sort any of the columns.

The error occurs in GridSort.cpp at line 127 at "hitems[c].sortmode = 0;". c is -1 and out of range.

```
void GridCtrl::ClearMultisort(int n)
{
 for(int i = n; i < sortOrder.GetCount(); i++)
 {
==>  int c = GetIdCol(sortOrder[i], true);
  hitems[c].sortmode = 0;
  hitems[c].sortcol = 0;
 }
 sortOrder.Clear();
}
```

GetIdCol is returning a -1;

```
int GridCtrl::GetIdCol(int id, bool checkall) const
{
 for(int i = checkall ? 1 : fixed_cols; i < total_cols; i++)
 {
  if(id == hitems[i].id)
   return i;
 }
 return -1;
}
```

total_cols = 1, which is incorrect. There are 2 or 3 actually, if you count the hidden first column.

It looks like, in "void GridCtrl::Clear(bool columns)", it should call ClearMultisort() (line 5684) befor it sets total_cols to 1. If any other changes have been made to fixed/hidden columns, the ClearMultisort may still have problems.

```
void GridCtrl::Clear(bool columns)
{
```

```
 doscroll = false;

 UpdateCtrls(UC_HIDE | UC_CTRLS);

 int nrows = columns ? 1 : fixed_rows;
 items.Remove(nrows, items.GetCount() - nrows);
 vitems.Remove(nrows, vitems.GetCount() - nrows);

 if(columns)
 {
  hitems.Remove(1, hitems.GetCount() - 1);
  items[0].Remove(1, items[0].GetCount() - 1);
  rowbkp.Remove(1, rowbkp.GetCount() - 1);
  edits.Remove(1, edits.GetCount() - 1);
  total_cols = 1; <== lost my 2 columns here
  total_width = 0;
  total_height = 0;
  firstVisCol = 0;
  lastVisCol = -1;
  firstCol = -1;
  lastCol = -1;
  fixed_cols = 1;
  firstVisRow = -1;
  lastVisRow = -1;
  coluid = 0;
  hcol = -1;
  sortCol = -1;
  ClearMultisort(); <== Trying to clear after it lost all its columns
 }
```

I tried this:

```
 if(columns)
 {
  ClearMultisort(); // <== Moved up here before range reduced from 3 to 1
  hitems.Remove(1, hitems.GetCount() - 1);
  items[0].Remove(1, items[0].GetCount() - 1);
  rowbkp.Remove(1, rowbkp.GetCount() - 1);
  edits.Remove(1, edits.GetCount() - 1);
  total_cols = 1; // <== Now columns can be flushed (except for hidden indicator)
  total_width = 0;
  total_height = 0;
  firstVisCol = 0;
  lastVisCol = -1;
  firstCol = -1;
  lastCol = -1;
  fixed_cols = 1;
```

```
  firstVisRow = -1;
  lastVisRow = -1;
  coluid = 0;
  hcol = -1;
  sortCol = -1;
//  ClearMultisort(); // <== Crashes
```

This fixes the bug, but then new columns from AddColumn() act a bit funny. The first column is squeezed shut.

Here's my test code:

```
GridCtrlMultiSortResetErr::GridCtrlMultiSortResetErr()
{
 typedef GridCtrlMultiSortResetErr CLASSNAME;
 CtrlLayout(*this, "Window title");
 maingrid.AddColumn(col1, "column 1", 100); // ready flag = 0, so RefreshLayout is not called
 maingrid.AddColumn(col2, "column 2", 100);
 maingrid.MultiSorting(true);
 maingrid.Add("Test val 1a", "Test value 1b");
 btnResetAfterSortCol <<= THISBACK(ResetAfterSortCol);
}

void GridCtrlMultiSortResetErr::ResetAfterSortCol() {
 maingrid.Reset(); // Crashes here if user has sorted any column (fix identified)
 maingrid.AddColumn(col1, "column 1", 100); // ready flag = 1, so RefreshLayout is called, and
column is scrunched
 maingrid.AddColumn(col2, "column 2", 100);
 maingrid.MultiSorting(true);
 maingrid.Add("Test val 1a", "Test value 1b");
}
```

So the first patch is to move ClearMultisort call before any column information is destroyed.

To correct the column shrinking, I just added some lines to my code:

```
void GridCtrlMultiSortResetErr::ResetAfterSortCol() {
 maingrid.Reset(); // Crashes here if user has sorted any column
==> maingrid.Ready(false);
 maingrid.AddColumn(col1, "column 1", 100);
 maingrid.AddColumn(col2, "column 2", 100);
 maingrid.MultiSorting(true);
 maingrid.Add("Test val 1a", "Test value 1b");
==> maingrid.Ready(true);
}
```

Now columns resize identically regardless of whether initial creation or after a Reset().

Whew! Fun stuff.

Thanks,
Jeff

<edit> I added the [BUG?] tag to the Title.

File Attachments

1) GridCtrlMultiSortResetErr.zip, downloaded 271 times

Subject: Re: ClearMultisort being called after total_cols reduced to 1 [BUG?]
Posted by unodgs on Fri, 07 Jan 2011 13:24:31 GMT
View Forum Message <> Reply to Message

You're absolutely right about the fix. I only simplified your solution, because calling ClearMultisort in Reset() is only about clearing sortOrder member. There is no sense to clear flags in particular columns if few lines later those columns are removed. So i removed calling ClearMultisort and replaced it with sortOrder.Clear(). Now in svn. Thank you for great testcase and pointing the solution!