
Subject: inconsistency causes crash

Posted by [nixnixnix](#) on Tue, 29 Jun 2010 18:46:32 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Daniel,

```
GridCtrl::ItemRect& GridCtrl::ItemRect::Editable(bool b)
{
    editable = b;
    parent->SyncCtrls(id);
    return *this;
}
```

.....

```
void GridCtrl::SyncCtrls(int row)
{
    //if(!HasCtrls())
    // return;
```

```
    Size sz = GetSize();
    genr_ctrls = false;
```

```
    Vector<int> cols;
```

```
    for(int i = 1; i < total_cols; i++)
        if(edits[hitems[i].id].factory && hitems[i].editable)
            cols.Add(i);
```

```
    if(cols.IsEmpty())
        return;
```

```
    int dy = sby + fixed_height;
    int dx = sbx + fixed_width;
    Rect r;
```

```
    int js = row < 0 ? 0 : row;
    int je = row < 0 ? total_rows : row + 1;
```

```
    for(int j = js; j < je; j++)
    {
        int idy = vitems[j].id;
        bool fixed = j < fixed_rows;
        bool create = !fixed && vitems[j].editable;
```

```

for(int i = 0; i < cols.GetCount(); i++)
{
    int c = cols[i];
    int idx = hitems[c].id;
    int oid = idx;

```

```

Item *it = &items[idy][idx];

```

```

if(it->isjoined)
{
    it = &items[it->idy][it->idx];
    idx = it->idx;
}

```

.....

the id in the first function becomes the row number in the second. This causes crashes in my code which I need to workaround (SetRowCount(10) then do my shit then SetRowCount(actual number of rows I want).

Just wanted to be dutiful and report the bug.

Nick

Subject: Re: inconsistency causes crash
 Posted by [unodgs](#) on Thu, 01 Jul 2010 10:49:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

nixnixnix wrote on Tue, 29 June 2010 14:46Hi Daniel,

```

GridCtrl::ItemRect& GridCtrl::ItemRect::Editable(bool b)
{
    editable = b;
    parent->SyncCtrls(id);
    return *this;
}

```

.....

```

void GridCtrl::SyncCtrls(int row)
{
    //if(!HasCtrls())

```

```

// return;

Size sz = GetSize();
genr_ctrls = false;

Vector<int> cols;

for(int i = 1; i < total_cols; i++)
    if(edits[hitems[i].id].factory && hitems[i].editable)
        cols.Add(i);

if(cols.IsEmpty())
    return;

int dy = sby + fixed_height;
int dx = sbx + fixed_width;
Rect r;

int js = row < 0 ? 0 : row;
int je = row < 0 ? total_rows : row + 1;

for(int j = js; j < je; j++)
{
    int idy = vitems[j].id;
    bool fixed = j < fixed_rows;
    bool create = !fixed && vitems[j].editable;

    for(int i = 0; i < cols.GetCount(); i++)
    {
        int c = cols[i];
        int idx = hitems[c].id;
        int oid = idx;

        Item *it = &items[idy][idx];

        if(it->isjoined)
        {
            it = &items[it->idy][it->idx];
            idx = it->idx;
        }

        .....
    }
}

```

the id in the first function becomes the row number in the second. This causes crashes in my code which I need to workaround (SetRowCount(10) then do my shit then SetRowCount(actual number of rows I want).

Just wanted to be dutiful and report the bug.

Nick

Thanks! I'll try to fix it ASAP
