
Subject: Help needed with OptionTree

Posted by [Tom1](#) on Fri, 10 Nov 2023 21:33:09 GMT

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

Hi,

I just cannot figure out how to do these in OptionTree:

- Have Option operating only from the checkbox, not the text label
- Select items and show the selection (at least one at a time) from the text label
- Support WhenLeftDouble from the text label
- Support WhenBar from the text label
- Support Internal DnD from the text label (to change ordering of items)

This is what I tried, but the checkbox seems to own the text label and all events get stuck there:

```
#include "CtrlLib/CtrlLib.h"

using namespace Upp;

struct App : TopWindow {
    OptionTree otree;

    typedef App CLASSNAME;

    void DropInsert(int parent, int ii, PasteClip& d){
        otree.AdjustAction(parent, d);
        if(AcceptInternal<OptionTree>(d, "mytreedrag")) otree.InsertDrop(parent, ii, d);
    }

    void Drag(){
        otree.DoDragAndDrop(InternalClip(otree, "mytreedrag"), otree.GetDragSample());
    }

    Array<Option> opts;

    App() {
        Add(otree.SizePos());

        otree.NoRoot();
        otree.ManualMode();

        for(int i=0;i<5;i++){
            otree.Add(0, Image(), opts.Add(), Format("Item %d",i+1));
            if(i==0){
                otree.Add(1, Image(), opts.Add(), String("SubItem 1"));
                otree.Add(1, Image(), opts.Add(), String("SubItem 2"));
            }
        }
    }
}
```

```

}

otree.WhenBar = [&](Bar &bar){ bar.Add(Format("Menu for item %d", otree.GetCursor()), [&](){
PromptOK("Menu item selected");}); };
otree.WhenLeftDouble = [&](){ PromptOK("Left double-click"); };
otree.WhenDropInsert = THISBACK(DropInsert);
otree.WhenDrag = THISBACK(Drag);
otree.NoCursor(false);
Sizeable();
}
};

GUI_APP_MAIN
{
App().Run();
}

```

Any ideas how to go around this issue?

Thanks and best regards,

Tom

Subject: Re: Help needed with OptionTree
Posted by [Tom1](#) **on** Sun, 12 Nov 2023 18:26:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Helping myself: Almost there... Just Drag-n-drop no longer working. For some reason, the items are not reconstructed at the target drop location, but still disappear from their origin:

```
class OptionText : public ParentCtrl{
public:
int id;
Option o;
String label;
```

```
void SetLabel(const String &text){
label = text;
}
```

```
void Layout(){
Size sz = GetSize();
o.SetRect(0,0,sz.cy,sz.cy);
}
```

```

OptionText(){
    id = -1;
    Add(o);
    SizePos();
}

void Paint(Draw &w){

    TreeCtrl *pc = dynamic_cast<TreeCtrl*>(GetParent());
    if(pc && pc->IsSel(id)){
        Rect r(GetTextSize(label, StdFont()));
        r.Offset(GetSize().cy,0);
        w.DrawRect(r, SColorHighlight());
        w.DrawTextA(GetSize().cy, 0, label, StdFont(), SColorHighlightText());
        return;
    }
    w.DrawRect(GetRect(), SColorPaper());
    w.DrawTextA(GetSize().cy, 0, label, StdFont(), Blue());
}

bool Key(dword key, int count){
    Ctrl *pc = GetParent();
    if(pc) return pc->Key(key, count);
    return false;
}

void LeftDown(Point p, dword keyflags){
    Ctrl *pc = GetParent();
    if(pc) pc->LeftDown(p + GetRect().TopLeft(), keyflags);
}

void LeftUp(Point p, dword keyflags){
    Ctrl *pc = GetParent();
    if(pc) pc->LeftUp(p + GetRect().TopLeft(), keyflags);
}

void LeftDrag(Point p, dword keyflags){
    Ctrl *pc = GetParent();
    if(pc) pc->LeftDrag(p + GetRect().TopLeft(), keyflags);
}

void LeftDouble(Point p, dword keyflags){
    Ctrl *pc = GetParent();
    if(pc) pc->LeftDouble(p + GetRect().TopLeft(), keyflags);
}

void RightDown(Point p, dword keyflags){
    Ctrl *pc = GetParent();
}

```

```

if(pc) pc->RightDown(p + GetRect().TopLeft(), keyflags);
}

void RightUp(Point p, dword keyflags){
Ctrl *pc = GetParent();
if(pc) pc->RightUp(p + GetRect().TopLeft(), keyflags);
}

void RightDrag(Point p, dword keyflags){
Ctrl *pc = GetParent();
if(pc) pc->RightDrag(p + GetRect().TopLeft(), keyflags);
}

void RightDouble(Point p, dword keyflags){
Ctrl *pc = GetParent();
if(pc) pc->RightDouble(p + GetRect().TopLeft(), keyflags);
}
};

struct App : TopWindow {
TreeCtrl tree;

typedef App CLASSNAME;

int dragparent; // Use this to ensure the target only moves within its own parent list

void DropInsert(int parent, int ii, PasteClip& d) {
tree.AdjustAction(parent, d);
if(parent == dragparent && AcceptInternal<TreeCtrl>(d, "mytreedrag")) {
tree.InsertDrop(parent, ii, d);
tree.SetFocus();
return;
}
}

void Drag()
{
dragparent = tree.GetParent(tree.GetCursor());
tree.DoDragAndDrop(InternalClip(tree, "mytreedrag"), tree.GetDragSample());
}

Array<OptionText> ot;

App() {
Add(tree.SizePos());
Vector<int> parent;
parent.Add(0);
tree.SetRoot(Image(), "The Tree");
}

```

```

for(int i = 1; i < 10000; i++) {
    OptionText &o = ot.Add();
    o.SetLabel(FormatIntRoman(i, true));
    o.id = tree.Add(parent[rand() % parent.GetCount()], Image(), o, 2000);
    parent.Add(o.id);
    if((rand() & 3) == 0)
        tree.Open(parent.Top());
}
tree.Open(0);
tree.WhenBar = [&](Bar &bar){ bar.Add(Format("Menu for item %d", tree.GetCursor()), [&](){
PromptOK("Menu item selected");}); };
tree.WhenLeftDouble = [&](){
PromptOK("Left double-click");
};
tree.WhenDropInsert = THISBACK(DropInsert);
tree.WhenDrag = THISBACK(Drag);
tree.MultiSelect();
Sizeable();
}
};

GUI_APP_MAIN
{
App().Run();
}

```

Any idea what I'm doing wrong?

Best regards,

Tom

Subject: Re: Help needed with OptionTree
Posted by Tom1 on Sun, 12 Nov 2023 18:53:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi again,

My above solution would work if Copy() in TreeCtrl.cpp did not clear the ctrl with "x.ctrl = NULL;".

See below:

```

int Copy(TreeCtrl& dst, int did, int i, const TreeCtrl& src, int id)
{
    TreeCtrl::Node x = src.GetNode(id);
    x.ctrl = NULL;
    did = dst.Insert(did, i, x);
    dst.Open(did, src.IsOpen(id));
    for(int i = 0; i < src.GetChildCount(id); i++)
        Copy(dst, did, i, src, src.GetChild(id, i));
    return did;
}

```

}

Well, I can probably go around this issue by duplicating and modifying the code for TreeCtrl::InsertDrop() and Copy(), in order to get my drop to cleanly move the item.

Best regards,

Tom
