Subject: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Sat, 18 Aug 2007 20:18:55 GMT
View Forum Message <> Reply to Message

it would be nice if it worked with images and options too

I saw the example and was thinking of replacing my drag and drop tree ctrl with yours as yours has some advantages over mine such as full selection and dropping into the middle of the hierarchy which mine doesn't do.

However, I tried to modify the example in the following way

```
App() {
 Add(tree.SizePos());
 Vector<int> parent, parent2;
 Option* option = new Option[10000]; // mine
 parent.Add(0);
 tree.SetRoot(Image(), "The Tree");
 for(int i = 1; i < 10000; i++) {
  option[i].SetLabel(FormatIntRoman(i, true)); // mine

  TreeCtrl::Node node(CtrlImg::File(),option[i],300); // mine

  parent.Add(tree.Add(parent[rand() % parent.GetCount()], node)); // mine

//   parent.Add(tree.Add(parent[rand() % parent.GetCount()], Image(),
//              FormatIntRoman(i, true)));
   if((rand() & 3) == 0)
    tree.Open(parent.Top());
 }
 tree.Open(0);
 tree.WhenDropInsert = THISBACK(DropInsert);
 tree.WhenDrag = THISBACK(Drag);
 tree.MultiSelect();
 Sizeable();
}
```

and it comes up good but when you try to drop a node it crashes

oh and the node selection is weak like in my version.

Is this something that might get added to the UPP TreeCtrl?

Nick

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Sun, 19 Aug 2007 13:01:29 GMT
View Forum Message <> Reply to Message

Thanks for spotting a bug in TreeCtrl...

Please try the fix.

Mirek

File Attachments
1) CtrlLib.zip, downloaded 659 times

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Thu, 23 Aug 2007 13:55:33 GMT
View Forum Message <> Reply to Message

Yep that works fine thanks. I suspect this would have to be integrated into a D&D OptionTree but the problem with both our trees is that selection is very week - its just a focus rect.

Also, its kinda hard to pickup and drag the options in your version as grabbing them by the label doesn't work.

I prefer the way your tree works but I need to be able to pickup nodes by the label and select them by clicking on their label.

Nick

p.s. I would do the selection thing myself in my own version of the tree except there doesnt appear to be a way to replace the label in the Option Ctrl with a custom Display.

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Thu, 23 Aug 2007 15:35:17 GMT
View Forum Message <> Reply to Message

Well, I guess the only option here is to make the label separated from the "option part" - otherwise

there is not way how the mouse click be passed down to the treectrl...

Mirek

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Mon, 27 Aug 2007 22:40:41 GMT
View Forum Message <> Reply to Message

If I replace the option in the modified reference example with the derived option control below

```
class LayerOption : public Option {
private:
 virtual void LeftDown(Point p, dword keyflags)
 {
  if(p.x<this->GetSize().cy)
   Option::LeftDown(p, keyflags);
  if(this->GetData()==true)
  {
   SetFocus();
  }
  // now send pass on this event to the parent for possible drag and drop
  TreeCtrl* pCtrl = (TreeCtrl*)this->GetParent();
  Point pt = p + GetRect().TopLeft(); // transform event into parent coordinates
  if(p.x>this->GetSize().cy)
   pCtrl->LeftDown(pt,keyflags);

  Refresh();
 }

};
```

I can select as well as using the option. The code above appears to change the selection registered by the tree control (although it fails to generate the WhenSel or WhenCursor events). I say it appears to register the selection cos when I ask the tree for the cursor position it reports it as being the node whose option that I just clicked on.

An alternative to coding an option tree with drag and drop would be if the Option control could be "selected".

Another problem with my code above is that it does not kick off the WhenDrag event even though I pass the event to the parent.

Nick

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Tue, 28 Aug 2007 11:13:22 GMT
View Forum Message <> Reply to Message

I am afraid I will have to resolve this problem in TreeCtrl... In fact, what we need is to

a) Ctrl occupies only a fraction of the line
b) In the rest, normal item text is used

This way, we would be able to Option with empty label and everything will be (finally...) OK...

Mirek

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Tue, 28 Aug 2007 12:05:00 GMT
View Forum Message <> Reply to Message

Yes! Of course. So in fact the option control is just the square checkbox.

I found what might be a bug whilst playing with the D&D tree control. If you drag and drop a node onto one of its children or granchildren etc the child accepts the parent node and in effect a copy of the parent and all its children appears to be made. However, there are a finite number of option controls and so a crash eventually results.

In my opinion, a child node should always refuse its parent (or grandparent etc). If a user wants to rearrange their tree so radically, they should explicitly copy the parent node or do two or more separate operations to achieve their goal.

Subject: Re: Like the new Drag and Drop TreeCtrl but...

Posted by nixnixnix on Tue, 28 Aug 2007 14:35:59 GMT
View Forum Message <> Reply to Message

I have a question.

Looking at the example code it appears that: parent is the id of the node onto which the dragged node is dropped; ii is the id that the dragged node will have once it is dropped; how do I find out the id of the node which is being dragged please?

Nick

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mrjt on Tue, 28 Aug 2007 15:46:19 GMT
View Forum Message <> Reply to Message

First you retrieve the source control, then see what was selected:

```
 if(AcceptInternal<TreeCtrl>(d, "mytreedrag")) {
  const TreeCtrl &src = GetInternal<TreeCtrl>(d); // JT
  Vector<int> sel = src.GetSel(); // JT
  // Etc.
  tree.InsertDrop(parent, ii, d);
  tree.SetFocus();
  return;
 }
```

James

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Wed, 29 Aug 2007 09:30:06 GMT
View Forum Message <> Reply to Message

nixnixnix wrote on Tue, 28 August 2007 08:05Yes! Of course. So in fact the option control is just the square checkbox.

I found what might be a bug whilst playing with the D&D tree control. If you drag and drop a node onto one of its children or granchildren etc the child accepts the parent node and in effect a copy of the parent and all its children appears to be made. However, there are a finite number of option controls and so a crash eventually results.

In my opinion, a child node should always refuse its parent (or grandparent etc). If a user wants to rearrange their tree so radically, they should explicitly copy the parent node or do two or more separate operations to achieve their goal.

Do not quite understand this; I believe I have correctly managed this in the code.

What do you mean by "finite number of option controls" ?

(BTW, obviously, embedded ctrls and copy is a trouble - default D&D does not account for that).

Mirek

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Fri, 31 Aug 2007 01:31:04 GMT
View Forum Message <> Reply to Message

Sorry, Mirek, that was just my best guess as to the reason for the crash. I just tested again with code exactly like in my first post in this thread.

You can easily recreate the behaviour by modifying the example as I posted above and then dragging a node onto one of its children or grandchildren.

The first symptom is that a lot of nodes do not have their option control anymore. Drag and drop the broken nodes and the interface stops responding and sometimes crashes. I tried to debug but my system grinds to a halt and even my quad-core is useless (hey Bill, how about writing a multithreaded OS?)

Nick

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Fri, 31 Aug 2007 06:08:24 GMT
View Forum Message <> Reply to Message

Well, what I wanted to say is that when there are embedded widgets, you cannot use prepared d&d routines - at least not yet.

Will look into the issue soon, you are right this should work for move (when you do not really need to create any new widgets).

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Fri, 31 Aug 2007 16:15:11 GMT
View Forum Message <> Reply to Message

For me, what we talked about above is way more important - i.e. disabling the option control's label and using the label from the TreeCtrl with the checkbox from the option control so that node selection will work properly.

I think its up to people to reject when a user drops nodes onto themselves or onto their own

descendents.

Nick

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Sun, 02 Sep 2007 22:15:59 GMT
View Forum Message <> Reply to Message

Well, it is "in the queue". But please, remind me soon

Mirek

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Tue, 18 Sep 2007 02:15:29 GMT
View Forum Message <> Reply to Message

Hey Mirek,

Just a quick nudge

I just installed 709dev1 and I see something about TreeCtrl has changed as the drag and drop that I implemented myself no longer works (none of the controls draw whilst one is being dragged using tracker - could it be the clipping you fixed?).

Anyhow, I don't want to fix my drag and drop if I will be able to abandon it and use yours (which is admittedly way better) in the near future.

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Thu, 20 Sep 2007 21:29:57 GMT
View Forum Message <> Reply to Message

OK, finally got to it...

Fixed it here and there and it now seems to do what I wanted. Please check whether it does what YOU want

Mirek

File Attachments
1) xxx.zip, downloaded 660 times

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Mon, 24 Sep 2007 14:29:22 GMT
View Forum Message <> Reply to Message

Hey Mirek,

Yes that looks great. However, when I drag and drop a node, after it is dropped it loses its option control. I am using the exact code included in the xxx.zip file.

Is this an error or is it up to me to make sure the option control is copied to the right place?

Nick

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Mon, 24 Sep 2007 21:59:15 GMT
View Forum Message <> Reply to Message

nixnixnix wrote on Mon, 24 September 2007 10:29Hey Mirek,

Yes that looks great. However, when I drag and drop a node, after it is dropped it loses its option control. I am using the exact code included in the xxx.zip file.

Is this an error or is it up to me to make sure the option control is copied to the right place?

Nick

Not an error, expected behaviour.

You would need some form of class factory or clone method to solve this problem as you need a copy of widget there....

So I have rather left that to implementor. I did not have too much time, but I believe that we will find a way how to streamline this process (e.g. something like callback in the copy routine to create missing widgets?).

Mirek

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Tue, 25 Sep 2007 15:46:32 GMT
View Forum Message <> Reply to Message

Ok, sounds fair.

However I have another question: why is it that the code inside Drag() never gets executed?

```
void Drag()
{
 if(tree.DoDragAndDrop(InternalClip(tree, "mytreedrag"),
                       tree.GetDragSample()) == DND_MOVE)
 {
  PromptOK("hey");
  tree.RemoveSelection();
 }
}
```

I added a PromptOK() as you can see above and it never pops up (I made sure to drag the selection to merely a different position on the same parent so there was no logical conflict).

Does that seems right to you?

I'm guessing that if that function did something it would copy the options across as its a MOVE rather than a copy right?

Nick

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by mirek on Tue, 25 Sep 2007 22:14:40 GMT
View Forum Message <> Reply to Message

nixnixnix wrote on Tue, 25 September 2007 11:46Ok, sounds fair.

However I have another question: why is it that the code inside Drag() never gets executed?


```
void Drag()
{
 if(tree.DoDragAndDrop(InternalClip(tree, "mytreedrag"),
                       tree.GetDragSample()) == DND_MOVE)
 {
  PromptOK("hey");
  tree.RemoveSelection();
 }
}
```

I added a PromptOK() as you can see above and it never pops up (I made sure to drag the selection to merely a different position on the same parent so there was no logical conflict).

Does that seems right to you?

I'm guessing that if that function did something it would copy the options across as its a MOVE rather than a copy right?

Nick

Well, this is a little bit tricky situation. And maybe existing solution is not right.

The problem is that if DND returns MOVE, you are in fact supposed to delete the source selection. Which would be wrong here - that is why TreeCtrl::InsertDrop changes the action to COPY if the operation is performed withing single widget.

Subject: Re: Like the new Drag and Drop TreeCtrl but...
Posted by nixnixnix on Wed, 26 Sep 2007 16:13:41 GMT
View Forum Message <> Reply to Message

Yeah no worries. I've got to say I am very very happy with the DnD implementation exactly as it is. It works perfectly for me. I see what you do in the TreeCtrl.cpp and its definitely the simplest and safest route to go as default behaviour and extremely easy to override by not calling InsertDrop() and taking the target, position and selection information and customising from there.

In reality you can't know how we want it to behave so InsertDrop will almost always be there just as demo behaviour so that DnD works straight out of the box. Its important that we see it working before we customise it.

Good job! and thanks. Its been a few months but TreeCtrl drag and drop now works better than I could have hoped

Nick