

---

Subject: Drag and Drop between instances [FEATURE REQUEST]

Posted by [nixnixnix](#) on Wed, 10 Nov 2010 23:03:22 GMT

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

---

Hi,

I can't find anything on this but I really suspect it already exists.

My users want to be able to pick up an object from my TreeCtrl and drop it into another instance of my software. The items in my TreeCtrl are all serializable and can already be transferred from one instance to another by writing to file, but D&D would be very slick. However, they don't exist outside the software at the time when the D&D is done.

Is this already possible in UPP or would it be a new feature?

Nick

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [mirek](#) on Wed, 17 Nov 2010 10:12:59 GMT

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

---

nixnixnix wrote on Wed, 10 November 2010 18:03Hi,

I can't find anything on this but I really suspect it already exists.

My users want to be able to pick up an object from my TreeCtrl and drop it into another instance of my software. The items in my TreeCtrl are all serializable and can already be transferred from one instance to another by writing to file, but D&D would be very slick. However, they don't exist outside the software at the time when the D&D is done.

Is this already possible in UPP or would it be a new feature?

Nick

It is already possible. You will have to implement some data format to store what you want to d&d for doing that...

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [nixnixnix](#) on Thu, 02 Dec 2010 19:28:03 GMT

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

---

Cool! Is there an example of this? I already have serialisation functions to copy to and from the clipboard (if that is how it is done).

I get that I need to override DragLeave() in my TreeCtrl and I see there is a member called Clipboard that is of type PasteClip. I imagine that I need to create a StringStream and serialize to it. However, connecting the StringStream to the clipboard is where I am missing a piece.

Hmmm, the more I look the less I think I have a clue. There are no reference examples or documentation for this and when I look at the objects involved (like PasteClip) there are no functions which would let me write to the clipboard.

Nick

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [mirek](#) on Sat, 11 Dec 2010 09:34:22 GMT

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

---

I have added reference/DragAndDrop example:

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

using namespace Upp;

const char *MyDnDName = "MyReferenceDragAndDropExample";

struct MyApp : TopWindow {
    TreeCtrl tree;

    typedef MyApp CLASSNAME;

    void DropInsert(int parent, int ii, PasteClip& d)
    {
        tree.AdjustAction(parent, d);
        if(d.Accept(MyDnDName)) {
            tree.SetCursor(tree.Insert(parent, ii, Image(), ~d));
            tree.SetFocus();
        }
    }

    void Drag()
    {
        if(!tree.IsCursor())
            return;
        int id = tree.GetCursor();
        String text = tree.Get();
        Size isz = GetTextSize(text.ToWString(), StdFont());
        ImageDraw iw(isz);
        iw.DrawRect(isz, White);
    }
}
```

```

iw.DrawText(0, 0, text);

VectorMap<String, ClipData> clip;
clip.Add(MyDnDName, text);

if(DoDragAndDrop(clip, iw) == DND_MOVE)
    tree.Remove(id);
}

MyApp() {
    Add(tree.SizePos());
    Vector<int> parent, parent2;
    parent.Add(0);
    tree.SetRoot(Image(), "The Tree");
    for(int i = 1; i < 10000; i++) {
        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);
    Sizeable();
}
};

GUI_APP_MAIN
{
    MyApp().Run();
}

```

Frankly, there are more details to care about, so we might want yet another example with even more low-level D&D (and some fine dics about D&D would be fine as well, right?), but as first iteration this might be helpful.

Mirek

---



---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]  
 Posted by [nixnixnix](#) on Wed, 12 Jan 2011 01:37:52 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Thanks Mirek,

That all works easy enough. However, an extra complication is that sometimes I DnD between instances but most of the time within the same instance. My code looks like this:

```

void LayerTree::DropInsert(int parent, int ii, PasteClip& d)
{
    AdjustAction(parent, d);
    if(AcceptInternal<LayerTree>(d, "mytreedrag")) {
        const TreeCtrl &src = GetInternal<LayerTree>(d);
        Vector<int> sel = src.GetSel();
        SaveStateToLayers(); // we need to rebuild the tree
        for(int i=0;i<sel.GetCount();i++)
        {
            Drop(parent,sel[i],ii);
        }
        m_ptr->SetTree();
        SetFocus();
        return;
    }
    else if(d.Accept("externalInstance"))
    {
        m_ptr->SetStatus("dropped");

        StringStream ss(d.Get());

        Layer* pLayer = Layer::Load(ss);

        if(pLayer==NULL)
            return;

        m_ptr->AddLayer(pLayer);

    }

    // if(AcceptText(d))
    // {
    //     SetCursor(Insert(parent, ii, Image(), GetString(d)));
    //     SetFocus();
    //     return;
    // }
    // }

void LayerTree::Drag()
{
    if(m_ptr->AreWeBusy())
        return;

    if(DoDragAndDrop(InternalClip(*this, "mytreedrag"),
                    this->GetDragSample()) == DND_MOVE)
    {

```

```

    RemoveSelection();
}
}

void LayerTree::DragLeave()
{
    int id;

    id = GetCursor();
    if(id<=0) // if id==0 then its the root node which is not a layer
        return;

    // copy to clipboard
    TreeCtrl::Node node = GetNode(id);
    LayerOption* ptr = (LayerOption*)~node.ctrl;
    Layer* pLayer = ptr->GetLayer();

    stringstream ss;

    ss.SetStoring();

    ss.Put("layer");
    int type = int(pLayer->GetType());
    ss.Put32(type);

    pLayer->Serialize(ss);

    String sLayer(ss);

    String text = pLayer->GetName();

    Size isz = GetTextSize(text.ToWString(), StdFont());

    ImageDraw iw(isz);

    iw.DrawRect(isz, White);

    iw.DrawText(0, 0, text);

    VectorMap<String, ClipData> clip;

    clip.Add("externalInstance", sLayer);

    if(DoDragAndDrop(clip, iw) == DND_MOVE)
    {

    }
}

```

```
TreeCtrl::DragLeave();
```

```
}
```

My questions are: do I need to treat my internal drag and drops the same way as my external ones? Is there an external and an internal DnD working simultaneously? Is there a way to know which instance a DnD is coming from?

It all seems to work fine so long as my cursor doesn't leave and return to the same instance.

Cheers,

Nick

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [mirek](#) on Wed, 12 Jan 2011 08:38:37 GMT

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

---

nixnixnix wrote on Tue, 11 January 2011 20:37

My questions are: do I need to treat my internal drag and drops the same way as my external ones?

In principle, yes. There is some "internal drop" stuff in U++, but it is there only to simplify internal drops (that then do not work 'external') - the thing is that for internal drops you do not have to define external format.

Quote:

Is there an external and an internal DnD working simultaneously?

The code is the same, at least client code. I would have to check whether there are some internal - external code pathways in U++ DnD implementation.... but I doubt it, except for name of internal clips.

Quote:

Is there a way to know which instance a DnD is coming from?

Ctrl::

```
bool IsDragAndDropSource();
```

```
bool IsDragAndDropTarget();
```

Mirek

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [mirek](#) on Wed, 12 Jan 2011 08:43:01 GMT

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

---

nixnixnix wrote on Tue, 11 January 2011 20:37

My questions are: do I need to treat my internal drag and drops the same way as my external ones? Is there an external and an internal DnD working simultaneously? Is there a way to know which instance a DnD is coming from?

Sorry, bad reply in previous post.

No, you can treat internal and external as you wish fit.

Internals are just another format. The only difference is that internal does not move data to the clipboard (keeps them in local memory), so it does not need to define external format.

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [nixnixnix](#) on Wed, 12 Jan 2011 20:54:59 GMT

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

---

Thanks Mirek,

I think I am still missing something fundamental about how this works.

I can drag and drop between instances fine.

I can drag and drop within the same instance using internal clip so long as I don't trigger DragLeave() (see below)

If I trigger DragLeave() which I use to drag and drop between instances, then if I drop onto the same instance I get a crash. If I try to do both internal and external from inside Drag() then the external one doesn't work.

Is there any documentation on how this framework works please or would it be possible to outline how one would do internal and external together?

Nick

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [mirek](#) on Wed, 12 Jan 2011 21:26:01 GMT

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

---

nixnixnix wrote on Wed, 12 January 2011 15:54 Thanks Mirek,

I think I am still missing something fundamental about how this works.

I can drag and drop between instances fine.

I can drag and drop within the same instance using internal clip so long as I don't trigger DragLeave() (see below)

If I trigger DragLeave() which I use to drag and drop between instances

Well, now this sound quite suspicious. Can you provide your DrawLeave method so that I can get a clue?

Are we speaking about

`void Ctrl::DragLeave()`

here?

Quote:

Is there any documentation on how this framework works please or would it be possible to outline how one would do internal and external together?

I am sorry, not yet. I guess it will be my homework over weekend...

Mirek

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [nixnixnix](#) on Thu, 13 Jan 2011 00:53:54 GMT

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

---

Hi Mirek,

Yes that was a typo and should have been DragLeave()



This is my code as it stands just now.

```
void LayerTree::DropInsert(int parent, int ii, PasteClip& d)
{
    AdjustAction(parent, d);
    if(!IsDragAndDropSource() && d.Accept("externalInstance"))
    {
        m_ptr->SetStatus("dropped");

        stringstream ss(d.Get());

        Layer* pLayer = Layer::Load(ss);

        if(pLayer==NULL)
            return;

        if(parent==0)
        {
            m_ptr->AddLayer(pLayer);
            return;
        }

        TreeCtrl::Node node = GetNode(parent);
        LayerOption* ptr = (LayerOption*)~node.ctrl;
        Layer* pParent = ptr->GetLayer();

        if(pParent)
        {
            pParent->AddChild(pLayer);
            m_ptr->SetTree();
            m_ptr->RefreshNow();
        }
        else
        {
            m_ptr->AddLayer(pLayer);
        }
    }
    else if(AcceptInternal<LayerTree>(d, "mytreedrag") )
    {
        const TreeCtrl &src = GetInternal<LayerTree>(d);
        Vector<int> sel = src.GetSel();
        SaveStateToLayers(); // we need to rebuild the tree
        for(int i=0;i<sel.GetCount();i++)
        {
            Drop(parent,sel[i],ii);
        }
        m_ptr->SetTree();
    }
}
```

```

    SetFocus();
    return;
}

}

void LayerTree::Drag()
{
    if(m_ptr->AreWeBusy())
        return;

    if(DoDragAndDrop(InternalClip(*this, "mytreedrag"),
                     this->GetDragSample()) == DND_MOVE)
    {
        RemoveSelection();
    }
}

void LayerTree::DragLeave()
{

    int id;

    id = GetCursor();
    if(id<=0) // if id==0 then its the root node which is not a layer
        return;

    // copy to clipboard
    TreeCtrl::Node node = GetNode(id);
    LayerOption* ptr = (LayerOption*)~node.ctrl;
    Layer* pLayer = ptr->GetLayer();

    stringstream ss;

    ss.SetStoring();

    ss.Put("layer");
    int type = int(pLayer->GetType());
    ss.Put32(type);

    pLayer->Serialize(ss);

    String sLayer(ss);

    String text = pLayer->GetName();

```

```

Size isz = GetTextSize(text.ToWString(), StdFont());

ImageDraw iw(isz);

iw.DrawRect(isz, White);

iw.DrawText(0, 0, text);

VectorMap<String, ClipData> clip;

clip.Add("externalInstance", sLayer);

if(DoDragAndDrop(clip, iw) == DND_MOVE)
{

}

TreeCtrl::DragLeave();

}

```

I am definitely missing something in my understanding of how these methods are triggered and what they do and so of how to get them to do what I want. I would like to leave the internal clip intact and to only trigger the external clip once I know it is being dropped on another instance although I expect that is not possible.

Thanks,

Nick

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [mirek](#) on Thu, 13 Jan 2011 08:47:33 GMT

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

---

nixnix wrote on Wed, 12 January 2011 19:53Hi Mirek,

Yes that was a typo and should have been DragLeave()

This is my code as it stands just now.

```

void LayerTree::DropInsert(int parent, int ii, PasteClip& d)
{
    AdjustAction(parent, d);
    if(!IsDragAndDropSource() && d.Accept("externalInstance"))

```

```

{
    m_ptr->SetStatus("dropped");

    stringstream ss(d.Get());

    Layer* pLayer = Layer::Load(ss);

    if(pLayer==NULL)
        return;

    if(parent==0)
    {
        m_ptr->AddLayer(pLayer);
        return;
    }

    TreeCtrl::Node node = GetNode(parent);
    LayerOption* ptr = (LayerOption*)~node.ctrl;
    Layer* pParent = ptr->GetLayer();

    if(pParent)
    {
        pParent->AddChild(pLayer);
        m_ptr->SetTree();
        m_ptr->RefreshNow();
    }
    else
    {
        m_ptr->AddLayer(pLayer);
    }
}

else if(AcceptInternal<LayerTree>(d, "mytreedrag") )
{
    const TreeCtrl &src = GetInternal<LayerTree>(d);
    Vector<int> sel = src.GetSel();
    SaveStateToLayers(); // we need to rebuild the tree
    for(int i=0;i<sel.GetCount();i++)
    {
        Drop(parent,sel[i],ii);
    }
    m_ptr->SetTree();
    SetFocus();
    return;
}

}

void LayerTree::Drag()

```

```

{
    if(m_ptr->AreWeBusy())
        return;

    if(DoDragAndDrop(InternalClip(*this, "mytreedrag"),
        this->GetDragSample()) == DND_MOVE)
    {
        RemoveSelection();
    }
}

void LayerTree::DragLeave()
{

    int id;

    id = GetCursor();
    if(id<=0) // if id==0 then its the root node which is not a layer
        return;

    // copy to clipboard
    TreeCtrl::Node node = GetNode(id);
    LayerOption* ptr = (LayerOption*)~node.ctrl;
    Layer* pLayer = ptr->GetLayer();

    stringstream ss;

    ss.SetStoring();

    ss.Put("layer");
    int type = int(pLayer->GetType());
    ss.Put32(type);

    pLayer->Serialize(ss);

    String sLayer(ss);

    String text = pLayer->GetName();

    Size isz = GetTextSize(text.ToWString(), StdFont());

    ImageDraw iw(isz);

    iw.DrawRect(isz, White);

    iw.DrawText(0, 0, text);

```

```

VectorMap<String, ClipData> clip;

clip.Add("externalInstance", sLayer);

if(DoDragAndDrop(clip, iw) == DND_MOVE)
{

}

TreeCtrl::DragLeave();

}

```

I am definitely missing something in my understanding of how these methods are triggered and what they do and so of how to get them to do what I want. I would like to leave the internal clip intact and to only trigger the external clip once I know it is being dropped on another instance although I expect that is not possible.

Thanks,

Nick

Ah, I see. Well, without going into details, I can imagine that this can cause great havoc

DragLeave is intended only for visual feedback in dnd target. It is sort of MouseLeave equivalent (actually, all 'Drag' functions are...)

In any case, all dragged formats have to be known at the moment drag starts. If you do not want to create binary data at that moment, you should rather use

```
ClipData(const Value& data, String (*render)(const Value& data));
```

constructor - render is supposed to 'convert' data (which can be basically anything) into your binary data, in essence exactly what you are trying to do in DragLeave.

---

**Subject: Re: Drag and Drop between instances [FEATURE REQUEST]**

Posted by [nixnixnix](#) on Thu, 20 Jan 2011 01:15:23 GMT

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

---

OK so it looks like it is too complicated to do both methods and I don't want to compromise my internal drag and drop.

How does one go about implementing copy and paste as a method for transferring data objects

from one instance to another please?

Is there an example of this?

Nick

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [mirek](#) on Thu, 20 Jan 2011 13:25:39 GMT

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

---

nixnixnix wrote on Wed, 19 January 2011 20:15OK so it looks like it is too complicated to do both methods and I don't want to compromise my internal drag and drop.

I am not sure where you got that impression. The only problem I can see is that you are doing in DragLeave what you should have done before starting Drag operation.

If you are afraid about speed (I mean, if you believe it would take too long to pack all data to transferable format), you just need to use that "Render" option.

Quote:

How does one go about implementing copy and paste as a method for transferring data objects from one instance to another please?

Copy&paste is using the exactly same mechanism as Draw&Drop, only you put ClipData onto clipboard instead...

I am sorry about docs - I really believed I will do that over weekend, but other issues took precedence.

Mirek

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [mirek](#) on Thu, 20 Jan 2011 15:36:11 GMT

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

---

nixnixnix wrote on Wed, 19 January 2011 20:15OK so it looks like it is too complicated to do both methods and I don't want to compromise my internal drag and drop.

How does one go about implementing copy and paste as a method for transferring data objects from one instance to another please?

Is there an example of this?

I guess a good example can be seen in RichEdit.

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [nixnixnix](#) on Fri, 28 Jan 2011 18:04:42 GMT

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

---

Hey Mirek,

I ran into problems trying to implement both side by side. If you do have time to do that example at some point, my users and I would be very grateful

Nick

---

---

Subject: Re: Drag and Drop between instances [FEATURE REQUEST]

Posted by [luluxiu](#) on Fri, 01 Jul 2011 01:52:13 GMT

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

---

You can put internal and external as you want to adapt. House is just another format. The only difference is that no internal moving data to the clipboard , so it does not define an external format.

---

#### File Attachments

1) [huh.gif](#), downloaded 336 times

---