Subject: [solved] Can we get the private members of TreeCtrl made protected?
Posted by cbpporter on Tue, 16 Jun 2015 13:59:57 GMT

View Forum Message <> Reply to Message

I created a custom TreeCtrl than only overrides Paint for prettier display of a list of trees and some members used by Paint are private.

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by koldo on Wed, 17 Jun 2015 07:04:57 GMT

View Forum Message <> Reply to Message

Could you show some pictures of it?

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by cbpporter on Wed, 17 Jun 2015 09:10:30 GMT

View Forum Message <> Reply to Message

Sure!

I created a custom TreeCtrl. Normally, the tree control is designed for a single root kind of structure. Even when not displaying the root node, there is still a line pointing upwards and it looks like a tree.

My change is about making it behave like a list of trees. Each entry in that list is a root of a tree, but they do not have a parent. I also changed the way lines are displayed for children nodes, enhanced IMHO for minor aesthetics.

In my particular case, I use this tree control to display folders from a classpath. That's why some folders are amrked with an "x", to show that they are invalid.

I implemented this only changing a few lines in the TreeCtrl::Paint override.

If you wish, I can give you the exact members than need to be made protected.

```
File Attachments
1) tree.png, downloaded 893 times
```

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by cbpporter on Thu, 02 Jul 2015 12:01:01 GMT

View Forum Message <> Reply to Message

Here is the list of things I need protected for my Paint:

```
protected:
 struct Item : Node {
  union {
   int        parent;
   int        freelink;
  };

  bool        free;
  bool        isopen;
  bool        sel;
  Vector<int>    child;
  int        linei;

  Size GetValueSize(const Display *treedisplay) const;
  Size GetCtrlSize() const;
  Size GetSize(const Display *treedisplay) const;

  Item() { isopen = false; linei = -1; parent = -1; canselect = true; sel = false; free = false; }
 };
 struct Line : Moveable<Line> {
  int  level;
  int  itemi;
  int  ll;
  int  y;
 };

 ScrollBars   sb;
 Scroller     scroller;
 bool        nobg;
```

```
int       levelcx;
bool      noroot;
Vector<Line> line;
Array<Item>  item;
int       dropitem, dropinsert;
bool      highlight_ctrl;
int       cursor;
bool      nocursor;
bool      multiselect;

const Display *display;

int   FindLine(int y) const;
const Display *GetStyle(int i, Color& fg, Color& bg, dword& st);
```

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by mirek on Sun, 12 Jul 2015 07:51:08 GMT

View Forum Message <> Reply to Message

I do not know. Generally, things are private to maintain narrow interface. If they become protected, they become part of interface.

OTOH, I like the functionality you propose. I suggest adding this to regular TreeCtrl as option... (files please?)

Mirek

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by cbpporter on Thu, 16 Jul 2015 08:37:04 GMT

View Forum Message <> Reply to Message

How about this?

```
File Attachments
1) TreeCtrl.cpp, downloaded 382 times
2) TreeCtrl.h, downloaded 672 times
```

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by mirek on Fri, 28 Aug 2015 06:33:32 GMT

View Forum Message <> Reply to Message

Sorry for the delay... applied.

However:

```
op -= CtrlImg::cross().GetSize() / 2;
w.DrawImage(op.x, op.y, imgEmpty);
```

Should not that be rather

```
op -= imgEmpty.GetSize() / 2;
w.DrawImage(op.x, op.y, imgEmpty);
```

?

Also, perhaps RenderMultiRoot should call NoRoot?

I have added docs, please check them too...

Mirek

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by cbpporter on Fri, 28 Aug 2015 10:51:08 GMT
View Forum Message <> Reply to Message

Thanks!

Use any name that you think is best. I can adapt. Now if we resolve the TimeStopHR, I can use stock U++ again .

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by mirek on Sat, 29 Aug 2015 05:15:04 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 28 August 2015 12:51Thanks!

Use any name that you think is best. I can adapt. Now if we resolve the TimeStopHR, I can use stock U++ again .

Ops, looks like I have not explained myself well... (english not being my native language).

Please, what I ask about is not about names, but semantics:

- using imgEmpty dimension for calculation in TreeCtrl::Paint instead of surrogate 'cross'

- calling ('executing') NoRoot in RenderMultiRoot (becuse I believe that you need to call NoRoot anyway to make it active).

Mirek

Subject: Re: Can we get the private members of TreeCtrl made protected?
Posted by cbpporter on Mon, 31 Aug 2015 10:35:16 GMT
View Forum Message <> Reply to Message

Oh, I see now.

Using "imgEmpty" for size calculations is indeed better.

As for calling NoRoot, this will have the side effect of forcing the display of the empty node icon for all users who call NoRoot right? That's partially why I added a new flag.