
Subject: Re: Wrap EditString Rect to text
Posted by [mirek](#) on Wed, 16 Oct 2013 06:35:59 GMT
[View Forum Message](#) <> [Reply to Message](#)

NoInternalMargin fixed, the method definition you seek is

```
Size GetStdSize(const Value& q) const {  
    return GetTextSize((String)q, StdFont().Height(fntSize));  
}
```

However, there are still some minor problems (selected node is vertically shifting); will fix those soon.

Mirek

P.S.: Complete testcase (for my reference):

```
#include <CtrlLib/CtrlLib.h>  
  
using namespace Upp;  
  
enum {fntSize = 19};  
  
struct AppNodeSize : public Display {  
    virtual void Paint(Draw& w,const Rect& r, const Value& q,Color ink,Color paper,dword style)  
    const {  
        w.DrawRect(r, paper);  
        w.DrawText(r.left, r.top, (String)q, StdFont().Height(fntSize), ink);  
    }  
  
    Size GetStdSize(const Value& q) const {  
        return GetTextSize((String)q, StdFont().Height(fntSize));  
    }  
};  
  
class EditableTree : public TreeCtrl {  
public:  
    typedef EditableTree CLASSNAME;  
  
    EditableTree() : selId_(0) {  
        NoRoot();  
  
        editNode_.SetCtrl(edit_);  
        edit_.NoInternalMargin().SetFrame(NullFrame());  
        edit_.SetFont(StdFont().Height(fntSize));  
  
        WhenSel = THISBACK(OnSelect);
```

```

}

void AddNode() {
    CloseEdit();

    //with some ico in /**/ offset lines more visible
    selNode_ = Node(/*espudImg::secExpand(), */"NODE" + AsString(GetChildCount(0) + 1));
    selNode_.SetDisplay(Single<AppNodeSize>()); //(2)

    selId_ = Add(0, selNode_);
    BeginEdit();
}

void OnSelect() {
    CloseEdit();

    selId_ = GetSel()[0];
    selNode_ = GetNode(selId_);

    BeginEdit();
}

private:
int selId_;
    Node selNode_;

    EditString edit_;
    Node editNode_;

void BeginEdit() {
    editNode_.image = selNode_.image;
    SetNode(selId_, editNode_);

    String val = selNode_.value;
    edit_.SetData(val);
    edit_.AutoSize();
    edit_.Show();
    edit_.SetFocus();
    edit_.SetSelection(val.GetCount());
}

void CloseEdit() {
    if (selId_ > 0) {
        selNode_.value = ~edit_;
        SetNode(selId_, selNode_);
        edit_.Hide();
    }
}

```

```
};

GUI_APP_MAIN
{
    TopWindow wnd;
    Button btn;
    wnd.Add(btn.SetLabel("Add").LeftPos(10).TopPos(10));

    EditableTree tree;
    wnd.Add(tree.VSizePos(40).HSizePos());

    btn <<= callback(&tree, &EditableTree::AddNode);
    wnd.Run();
}
```
