
Subject: Constant font size in resizable RichEdit?
Posted by [Mindtraveller](#) on Sun, 27 Dec 2009 12:29:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

There is a task of making RichEdit editing some text. Control itself is resizable. Each time I change its width, it changes font size accordingly. This is suitable for reading books, but writing text while font size heavily depends on width of the window makes serious visual problem for writer. So I want to make font size constant and independent from width of RichEdit control. Is it possible? Could you please tell how to do it?

Subject: Re: Constant font size in resizable RichEdit?
Posted by [mirek](#) on Sun, 27 Dec 2009 18:57:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

I believe it would be possible to achieve it using `ViewBorder` or `SetZoom` - you would have to adjust both dynamically as the size changes.

Mirek

Subject: Re: Constant font size in resizable RichEdit?
Posted by [iST1](#) on Mon, 07 Oct 2013 16:18:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

`SetZoom` doesn't help: with value > 100 text shifted outside left border.

Subject: Re: Constant font size in resizable RichEdit?
Posted by [mirek](#) on Tue, 08 Oct 2013 06:05:55 GMT
[View Forum Message](#) <> [Reply to Message](#)

I am sorry, `SetZoom` indeed does not work, but something like

```
struct MyEdit : RichEdit;

void MyEdit::Layout()
{
    RichEdit::Layout();
    SetPage(Size(minmax(GetSize().cx, 50, 10000), INT_MAX));
}
```

does.

Subject: Re: Constant font size in resizable RichEdit?

Posted by [koldo](#) on Tue, 08 Oct 2013 10:22:03 GMT

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

Please see this nice proposal from Alendar:

http://www.ultimatepp.org/forum/index.php?t=msg&goto=40773&#msg_40773

It includes Mirek solution plus control mouse zoom.

It could be good to include this behavior in RichEdit. It is proposed in Redmine:

<http://www.ultimatepp.org/redmine/issues/518>

Subject: Re: Constant font size in resizable RichEdit?

Posted by [mirek](#) on Tue, 08 Oct 2013 17:59:50 GMT

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

koldo wrote on Tue, 08 October 2013 06:22 Please see this nice proposal from Alendar:

http://www.ultimatepp.org/forum/index.php?t=msg&goto=40773&#msg_40773

It includes Mirek solution plus control mouse zoom.

It could be good to include this behavior in RichEdit. It is proposed in Redmine:

<http://www.ultimatepp.org/redmine/issues/518>

I know, but it is good for specialized variants of RichEdit.

The problem is that RichEdit is intended (and used!) for WYSIWYG. Changing number of characters per line defeats that purpose.

Anyway, perhaps the right solution would be to provide derived class?

Subject: Re: Constant font size in resizable RichEdit?

Posted by [koldo](#) on Wed, 09 Oct 2013 06:15:19 GMT

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

Hello Mirek

In my case in fact it has more sense the derived than the original RichEdit as the expected behavior is:

- to maintain font size when changing control size
 - Ctrl-Wheel resizes the font
-

Actual RichEdit is so rich that with few additional lines of code it can have this new behavior.

This way, I would just add the new behavior to actual RichEdit and I would add a SetXXX() to set the preferred behavior.

Subject: Re: Constant font size in resizable RichEdit?

Posted by [iST1](#) on Fri, 11 Oct 2013 09:02:27 GMT

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

Both solutions do not come:

```
class ExtRichEdit : public RichEdit {
public:
    typedef AppRichEdit CLASSNAME;
    float zoomlevel;

    //=====
    virtual void MouseWheel(Point p, int zdelta, dword keyflags) {
        if (keyflags == K_CTRL) {
            // Zooms font
            float fzdelta = zdelta;
            zoomlevel+= (fzdelta / 240.0); // One bump on the mouse wheel is 120 on my machine
            RefreshLayoutDeep();
        } else {
            // Scrolls down
            RichEdit::MouseWheel(p, zdelta, keyflags);
        }
    }

    //=====
    void Layout() {
        RichEdit::Layout();
    #if 0
        //Mirek's solution: to big font
        SetPage(Size(minmax(GetSize().cx, 50, 10000), INT_MAX));
    #else
        //Alendar's solution: to small font
        long editor_cx = GetSize().cx;
        long adaptive_cx = (editor_cx * zoomlevel); // Smaller the number, the bigger the text
        SetPage(Size(adaptive_cx, INT_MAX));
    #endif
    }
};
```

GUI_APP_MAIN

```
{
  ExtRichEdit edit;
  TopWindow r;
  r.Add(edit.SizePos());

  TopWindow wnd;
  SplitterFrame sp;
  wnd.AddFrame(sp.Right(r, 300));
  wnd.AddFrame(NullFrame());

  wnd.Run();
}
```

Subject: Re: Constant font size in resizable RichEdit?
Posted by [koldo](#) on Fri, 11 Oct 2013 12:21:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello iST1

Mirek's solution permits a fixed zoom and Alendar's one permits variable zoom using mouse wheel or other means.

Subject: Re: Constant font size in resizable RichEdit?
Posted by [iST1](#) on Fri, 11 Oct 2013 12:23:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

Yes, but default behavior must be, i mean, with normal font size

Subject: Re: Constant font size in resizable RichEdit?
Posted by [iST1](#) on Mon, 21 Oct 2013 15:58:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

In example above must be typedef ExtRichEdit CLASSNAME; and i still with troubles in getting a normal font's size

Subject: Re: Constant font size in resizable RichEdit?
Posted by [koldo](#) on Tue, 22 Oct 2013 14:10:07 GMT
[View Forum Message](#) <> [Reply to Message](#)

My proposal:

```
class RichEdit2 : public RichEdit {
```

```
public:
RichEdit2() {
    zoomlevel = 7;
}
virtual void Layout() {
    RichEdit::Layout();
    SetPage(Size(int(zoomlevel*GetSize().cx), INT_MAX)); // Smaller the total, the bigger the text
}
virtual void MouseWheel(Point p, int zdelta, dword keyflags) {
    if (keyflags == K_CTRL) { // Zooms font
        zoomlevel += zdelta/240.;
        if (zoomlevel < 1)
            zoomlevel = 10;
        else if (zoomlevel > 9)
            zoomlevel = 1;
        RefreshLayoutDeep();
    } else // Scrolls down
        RichEdit::MouseWheel(p, zdelta, keyflags);
    }
double zoomlevel;
};
```

Subject: Re: Constant font size in resizable RichEdit?
Posted by [iST1](#) on Tue, 22 Oct 2013 14:28:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

Great!

Subject: Re: Constant font size in resizable RichEdit?
Posted by [mirek](#) on Tue, 29 Oct 2013 17:57:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

RichEdit now has "Floating" method that does exactly this...

Subject: Re: Constant font size in resizable RichEdit?
Posted by [koldo](#) on Wed, 30 Oct 2013 12:04:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Mirek

This is a good addition, although it would be great to modify the zoom with the mouse wheel.

This feature works in many if not all text processing programs even in image editors and CAD programs.

Subject: Re: Constant font size in resizable RichEdit?

Posted by [mirek](#) on Wed, 30 Oct 2013 12:07:58 GMT

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

koldo wrote on Wed, 30 October 2013 08:04>Hello Mirek

This is a good addition, although it would be great to modify the zoom with the mouse wheel.

Have you checked? It is supposed to work... (only in floating mode though)

Subject: Re: Constant font size in resizable RichEdit?

Posted by [koldo](#) on Wed, 30 Oct 2013 22:07:35 GMT

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

Oh yes you are right

I did not see the change in Mouse.cpp .

Subject: Re: Constant font size in resizable RichEdit?

Posted by [Tooraj](#) on Tue, 18 Dec 2018 15:41:31 GMT

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

It seems, that this important change was forgotten for RichTextView:

For RichTextView the override needs to be slightly modified:

```
class RichTextView2 : public RichTextView {
public:
    RichTextView2() {
        zoomlevel = 7;
    }
    virtual void Layout() {
        RichTextView::Layout();
        PageWidth( int(zoomlevel*GetSize().cx) ); // Smaller the total, the bigger the text
    }
    virtual void MouseWheel(Point p, int zdelta, dword keyflags) {
        if (keyflags == K_CTRL) { // Zooms font
            zoomlevel += zdelta/240.;
            if (zoomlevel < 1)
                zoomlevel = 10;
            else if (zoomlevel > 9)
                zoomlevel = 1;
            RefreshLayoutDeep();
        }
    }
};
```

```
} else // Scrolls down
  RichTextView::MouseWheel(p, zdelta, keyflags);
}

double zoomlevel;
};
```
