
Subject: Is Upp:: output necessary in *.lay files?

Posted by [luoganda](#) on Sun, 10 Oct 2021 20:35:24 GMT

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

Once one opens older layout files with new layout designer(after upp-13664),

"Upp::" is prepended/added to all layout-files controls.

This is many times undesirable since not all control may be a part of Upp:: namespace.

For now, i have to use older upp-ver(13664:2019.2 or lower) to correctly compile code without errors.

At least some option somewhere would be ok to prepend this or not.

By the way, where has bazaar folder went in a few new versions?

I also don't find it on github.

Subject: Re: Is Upp:: output necessary in *.lay files?

Posted by [zsolt](#) on Fri, 06 May 2022 22:14:15 GMT

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

Put a

namespace;

line at the top of your .usc file.

Or

namespace MyNamespace;

if your control is in MyNamespace.

Bazaar now is UppHub

Subject: Re: Is Upp:: output necessary in *.lay files?

Posted by [Alboni](#) on Mon, 13 Jun 2022 13:18:01 GMT

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

Why was this done? It also changes the order of the attributes of the elements and if my colleague and me use different version we're constantly committing order changes to svn for no reason.

And then I get this:

```
ITEM(LabelBox, dv___0, SetLabel(t_("Kies aub een betaalmethode")).SetInk(Color(28, 0, 200)).SetFont(StdFontZ(14).Bold()).LeftPosZ(13, 388).TopPosZ(7, 349))
```

changed to

```
ITEM(Upp::LabelBox, dv___0, SetInk(Color(28, 0, 200)).SetFont(StdFontZ(14).Bold()).SetLabel(t_("Kies aub een betaalmethode")).LeftPosZ(13, 388).TopPosZ(7, 349))
```

which doesn't even compile.

Error: /home/ruben/MyApps/DisProject/something.lay (580): error: 'class Upp::StaticText' has no member named 'SetLabel'

Subject: Re: Is Upp:: output necessary in *.lay files?

Posted by [luoganda](#) on Tue, 14 Jun 2022 19:27:01 GMT

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

"Why was this done?"

I don't have a clue, but probably because larger code is more organized this way, one can have same name classes in different namespaces, so that the same name classes don't mix(wrongly) - which is in the long run for new - larger apps - probably better.

Alboni has a point here also, but this is an old thing probably from always mostly concerning Label...

When one starts to rearranging controls, some get rearranged and don't compile properly.

But i think this was fixed in new versions with subclassing Label... from control also for chaining purposes,

so even if controls gets rearranged - it should compile properly.

The old code still might not compile because opening it with new layout editor would throw Upp:: prefix back again.

One possibility for older non-compatible code with new Upp:: thing would probably be compiling it with command line tool

or the thing like zsoft suggests.

Aha... for bazaar.
