Subject: Lost focus in edit fields
Posted by Zbych on Thu, 30 Jul 2009 08:17:13 GMT
View Forum Message <> Reply to Message

Hi,

Is there a way to trigger a callback when edit filed loses its focus? Right now I added a callback (WhenLostFocus) to EditField::LostFocus, but maybe there is already some other method to do it?

Subject: Re: Lost focus in edit fields
Posted by mirek on Sat, 01 Aug 2009 06:56:53 GMT
View Forum Message <> Reply to Message

ChildLostFocus can be used sometimes... (but you have to store the pointer in ChildGotFocus).

Mirek

Subject: Re: Lost focus in edit fields
Posted by Didier on Sat, 01 Aug 2009 08:43:39 GMT
View Forum Message <> Reply to Message

HI,

Quote:is already some other method to do it

I had the same problem some time ago and did something this:

```
template < class T >
class EnhancedEditField : public T
{
 public:
  typedef T BaseClass;
  EnhancedEditField() {};
  virtual ~EnhancedEditField() {};


  Callback whenFocusLost;

  virtual void LostFocus()
  {
   T::LostFocus();
   whenFocusLost();
  }
```

```
};
```

```
typedef EnhancedEditField<EditString> EnhancedEditString;
typedef EnhancedEditField<EditInt> EnhancedEditInt;
typedef EnhancedEditField<EditDouble> EnhancedEditDouble;
```

It adds a callback to the type <T> and does what you want while avoiding modifying upp src.

Since nobody asked for this feature before maybe our way of doing things is a little clumsy ? ? ?

Subject: Re: Lost focus in edit fields
Posted by mirek on Sat, 01 Aug 2009 09:16:11 GMT
View Forum Message <> Reply to Message

It is fine. I believe there is similar WithLostFocus template in the "extended universe". Note that the issue is not EditField related..

Mirek

Subject: Re: Lost focus in edit fields
Posted by Zbych on Sat, 01 Aug 2009 09:41:36 GMT
View Forum Message <> Reply to Message

Didier wrote on Sat, 01 August 2009 10:43It adds a callback to the type <T> and does what you want while avoiding modifying upp src.

I don't really mind modyfing upp src , esspecially when it is just two lines (one in *.cpp and one in *.h):

EditField.cpp :

```
void EditField::LostFocus()
{
 if(autoformat && IsEditable() && !IsNull(text)) {
  Value v = convert->Scan(text);
  if(!v.IsError()) {
   const Convert * cv = inactive_convert ? inactive_convert : convert;
   WString s = cv->Format(v);
   if(s != text) text = s;
  }
```

```
 }
 if(!keep_selection) {
  anchor = -1;
  cursor = sc = 0;
  WhenLostFocus();//<--here
 }
 Refresh();
 SyncEdge();
}
```

Subject: Re: Lost focus in edit fields
Posted by Didier on Sat, 01 Aug 2009 10:25:32 GMT
View Forum Message <> Reply to Message

I can't find the 'WithLostFocus' template.

What is the "extended universe" ???

Subject: Re: Lost focus in edit fields
Posted by mirek on Sat, 01 Aug 2009 12:28:03 GMT
View Forum Message <> Reply to Message

Zbych wrote on Sat, 01 August 2009 05:41Didier wrote on Sat, 01 August 2009 10:43It adds a callback to the type <T> and does what you want while avoiding modifying upp src.

I don't really mind modyfing upp src , esspecially when it is just two lines (one in *.cpp and one in *.h):

That is really bad idea. Over long time, you will not be able to keep your U++ sources merged..

Mirek

Subject: Re: Lost focus in edit fields
Posted by mirek on Sat, 01 Aug 2009 12:30:22 GMT
View Forum Message <> Reply to Message

Didier wrote on Sat, 01 August 2009 06:25I can't find the 'WithLostFocus' template.

What is the "extended universe" ???

Well, what is in upp.src is a selection of sources usually originated in development of some "real" apps.

I know that WithLostFocus exists in some of Tom's sources, which are not part of uppsrc now (yet?).

Subject: Re: Lost focus in edit fields
Posted by Didier on Sat, 01 Aug 2009 14:23:50 GMT
View Forum Message <> Reply to Message

Hi Milek,

Maybe this 'WithLostFocus' template could be put int 'CtrlUtil.h'

Subject: Re: Lost focus in edit fields
Posted by jeremy_c on Thu, 29 Jul 2010 09:42:02 GMT
View Forum Message <> Reply to Message

What's the status of WithLostFocus? I also want to trigger a few actions when a widget (EditField in my case) looses focus.

Jeremy

Subject: Re: Lost focus in edit fields
Posted by mrjt on Thu, 29 Jul 2010 10:57:49 GMT
View Forum Message <> Reply to Message

Just use the one from this thread in your own code, though it doesn't need as much code as above:

```
template <class T>
struct WithLostFocus : public T {
  Callback WhenLostFocus;

  virtual void LostFocus() {
   T::LostFocus();
   WhenFocusLost();
  }
```

```
};
```

Subject: Re: Lost focus in edit fields
Posted by jeremy_c on Thu, 29 Jul 2010 12:54:45 GMT
View Forum Message <> Reply to Message

Thanks. A slight typo fix though:

```
template <class T>
struct WithLostFocus : public T {
  Callback WhenLostFocus;

  virtual void LostFocus() {
   T::LostFocus();
   WhenLostFocus();
  }
};
```

i.e. WhenFocusLost() changed to WhenLostFocus();

Jeremy

Subject: Re: Lost focus in edit fields
Posted by Giorgio on Thu, 27 Aug 2015 07:05:23 GMT
View Forum Message <> Reply to Message

Hi there,
my very first message so excuse the dumb question.

I am a little noob and not really experienced with C++ programming. I am trying to do the same as in this thread: add an action when focus is lost on an EditString field. I think that your solution is what I am looking for, but I do not know how to make it work. I could really use an example i.e. where to put your code and how to "connect" the lost focus event with my procedure.

Thanks in Advance.
Giorgio