Subject: howto force evaluation after <CR>
Posted by ebojd on Wed, 01 Aug 2007 19:16:51 GMT
View Forum Message <> Reply to Message

I have come across a trivial problem while developing an interface to drive our LiDAR (think of it as a telescope with a 3000W laser pointer attached). When I tell the telescopic mount to go to a particular elevation or azimuth via an EditDoubleNotNull field, it calls the callback function for every character typed in instead after a <CR> or moving the mousse out of the focus area. What is the easiest way to force the evaluation after a <CR>?

Follows is the prototype code:

```
void telescope_ctl::btn_move_el_ctl()
 {El = (double)~jog.jog_pos_el;}


telescope_ctl::telescope_ctl()
{
  ...
  jog.jog_pos_el <<= THISBACK(btn_move_el_ctl);
  ...
}
```

Thanks,

EBo --

Subject: Re: howto force evaluation after <CR>
Posted by mrjt on Wed, 01 Aug 2007 20:46:26 GMT
View Forum Message <> Reply to Message

I have had the same problem and solved it in two different ways depending on the situation:

1) This is the way I think it was intended to be done. Because EditField doesn't actually handle K_RETURN it is passed up to the parent window's Key function, and see if that field has focus:

```
if (key == K_RETURN && FocusCtrl() == &jog.jog_pos_el)
    // Do whatever
```

2) If you really want a callback you can do:

```
template<class T>
struct WithReturnCallback : public T
{
    Callback WhenReturnKey;
    virtual bool Key(dword key, int count)
```

```
    {
        if (key == K_RETURN) {
            WhenReturnKey();
            return true;
        }
        return T::Key(key, count);
    }
}
```
And then us it as a wrapper for the editfield.

Apologies in advance for any errors in the the above, Idon't have access to a compiler this minute.

Quote:our LiDAR (think of it as a telescope with a 3000W laser pointer attached)
Awesome!

James.

Subject: Re: howto force evaluation after <CR>
Posted by ebojd on Thu, 02 Aug 2007 12:25:53 GMT
View Forum Message <> Reply to Message

James, thanks for your reply.

mrjt wrote on Wed, 01 August 2007 15:46
I have had the same problem and solved it in two different ways depending on the situation:

...

I poked at it a little and was unable to get it to compile and/or work (but I'm sure that it is simply a function of my inexperience programming U++). Anyway, I came up with a workaround by adding a "Move To" button. I'll look into this again when I get-a-round-to-it.

Quote:Quote:our LiDAR (think of it as a telescope with a 3000W laser pointer attached)
Awesome!

I cannot wait until I get it all working together and fire the puppy up in the field.

Thanks again,

EBo --

Subject: Re: howto force evaluation after <CR>

Posted by mirek on Thu, 02 Aug 2007 17:35:30 GMT
View Forum Message <> Reply to Message

mrjt wrote on Wed, 01 August 2007 16:46
if (key == K_RETURN && FocusCtrl() == &jog.jog_pos_el)
    // Do whatever

Or just jog.HasFocus();

Mirek

Subject: Re: howto force evaluation after <CR>
Posted by ebojd on Thu, 02 Aug 2007 20:19:55 GMT
View Forum Message <> Reply to Message

luzr wrote on Thu, 02 August 2007 12:35mrjt wrote on Wed, 01 August 2007 16:46
if (key == K_RETURN && FocusCtrl() == &jog.jog_pos_el)
    // Do whatever

Or just jog.HasFocus();

Thanks Mirek, but that did not work. I'll play with it more tonight.

EBo --

Subject: Re: howto force evaluation after <CR>
Posted by mirek on Tue, 07 Aug 2007 17:23:56 GMT
View Forum Message <> Reply to Message

Well given the HasFocus implementation:

bool   HasFocus() const   { return FocusCtrl() == this; }

it is really strange that your version should work and HasFocus not

Subject: Re: howto force evaluation after <CR>
Posted by ebojd on Tue, 07 Aug 2007 21:55:06 GMT
View Forum Message <> Reply to Message

I'll look at this again later. I decided that typing the number in then punching the "Go" button made more intuitive sense -- also I lost one of the power supplies on my test bench and one-thing-lead-to-another and I fried the output drivers on my embedded development system, so I have had other things on my mind...

Thanks for following up on this though...

EBo --

ps: IIRC the problem was not with HasFocus but it did not behave like I wanted -- wait to return until after 1) typeda <CR>, or 2) left focus. The "Go" button makes intuitive sense without violating the law of minimal surprise...