Subject: Knob control
Posted by koldo on Wed, 11 Aug 2010 20:30:13 GMT
View Forum Message <> Reply to Message

Hello all

It is included in Controls4U a new Knob control. You can see samples in the picture.

The interface is preliminar. If you have any advice please tell me asap.

It includes:

- 4 color types: SimpleWhiteType, SimpleBlackType, WhiteType (preliminar), BlackType (preliminar)

- 2 marks: Line and Circle

Angle begin, angle end, min value, max value, step value are programmable.

In addition it is possible to set numbers in marks (SetNumber) and add the possibility of knob locks in marks (SetInterlocking).

The status is:

- Visual style: 70%
- Class methods: 90%
- User input: 95%
- .usc layout designer: 20%
- Documentation: 0%

File Attachments

1) dib.PNG, downloaded 1105 times

Subject: Re: Knob control
Posted by jerson on Thu, 12 Aug 2010 00:44:30 GMT
View Forum Message <> Reply to Message

Hi Koldo

Looks good. I shall 'try' it and 'catch' some points to discuss

I'm not sure if you can add fine marks between the major marks. Could be set to 0 if a person does not want them.

Thank you very much

Subject: Re: Knob control
Posted by koldo on Thu, 12 Aug 2010 06:13:04 GMT
View Forum Message <> Reply to Message

Quote:I'm not sure if you can add fine marks between the major marks. Could be set to 0 if a person does not want them.
No problem .

Two questions:
- How would you call the marks between major marks?
- How would you call the mark locking knob behavior?

Subject: Re: Knob control
Posted by jerson on Thu, 12 Aug 2010 08:11:49 GMT
View Forum Message <> Reply to Message

koldo wrote on Thu, 12 August 2010 11:43
Two questions:
- How would you call the marks between major marks?
- How would you call the mark locking knob behavior?
I am thinking more in terms of the slider code I just modified. In that code, the ticks are marked for every minorticks and thick one for every majorticks. The increment is by minorticks.

The knob locks on to minortick increments. Majorticks is just a special case of minorticks as I see it.

This is the relevant part of that code

```
 // draw gradations
 for(  int i = Min();
   ( m_nMajorTicks > 0 ) && ( i <= Max() ) ;
   i += ( m_nMinorTicks == 0 ? m_nMajorTicks : m_nMinorTicks ) ) {

  int nPos = SliderToClient( i );

  if( ( m_nMajorTicks != 0 ) && ( i % m_nMajorTicks ) == 0 )
   DrawTick( w, MAJOR, (HOVE)HoVe( HORZ, VERT ), nPos, i );
  else if( ( m_nMinorTicks != 0 ) && ( i % m_nMinorTicks ) == 0 )
   DrawTick( w, MINOR, (HOVE)HoVe( HORZ, VERT ), nPos, i );
 }
```

Regards

Subject: Re: Knob control
Posted by koldo on Thu, 12 Aug 2010 08:23:16 GMT
View Forum Message <> Reply to Message

Quote:the ticks are marked for every minorticks and thick one for every majorticks. The increment is by minorticks.Ok

A question: The Knob sensibility and behavior when using it with mouse and keyboard is right for you?. When selected and/or clicked it changes slightly its shape.

Subject: Re: Knob control
Posted by jerson on Thu, 12 Aug 2010 11:29:26 GMT
View Forum Message <> Reply to Message

Hi Koldo

Frankly, I haven't been able to check it out as yet. My internet connection is on the blink and downloading 27MB is painful. If I have the changed files log, I can manually download the relevant files and update my src directory.

Cheers

Subject: Re: Knob control
Posted by jerson on Fri, 13 Aug 2010 04:29:58 GMT
View Forum Message <> Reply to Message

Hi Koldo

I've been able to get the complete package and checked the knob. It looks good. As for the focus behaviour, I think it is perfectly fine to know that the knob is captured.

Best Regards

Subject: Re: Knob control
Posted by koldo on Fri, 13 Aug 2010 23:55:22 GMT
View Forum Message <> Reply to Message

Hello Jerson

I have removed SetStep and added SetMinorMarks and SetMajorMarks.

These include what you want and are much safer.

Be careful. SetMajorMarks set the number of major marks excluded the initial and the end. SetMinorMarks includes the number of minor marks between every pair of major marks.

I have changed also how to show the selected Knob. Here the results:

File Attachments

1) dib.PNG, downloaded 942 times

Subject: Re: Knob control
Posted by jerson on Sat, 14 Aug 2010 02:04:22 GMT
View Forum Message <> Reply to Message

Hi Koldo

This looks very good. Thank you

Since it is a work in progress, I guess, the USC does not link to the knob properly and I get errors if I try to SetFineTicks(IIRC) and some other parameters too.

Regards

Subject: Re: Knob control
Posted by koldo on Sat, 14 Aug 2010 14:09:14 GMT
View Forum Message <> Reply to Message

Hello Jerson

Quote:Since it is a work in progress, I guess, the USC does not link to the knob properly and I get errors if I try to SetFineTicks(IIRC) and some other parameters too.
Yes, .usc is not updated yet, but I do not know what is "SetFineTicks".

Subject: Re: Knob control
Posted by jerson on Sat, 14 Aug 2010 14:58:28 GMT
View Forum Message <> Reply to Message

koldo wrote on Sat, 14 August 2010 19:39
Yes, .usc is not updated yet, but I do not know what is "SetFineTicks".
Hi Koldo

You won't believe it, my mind doesn't know anymore what my hand is doing The original name I wanted to quote is SetFineStep from the USC file. I renamed it to SetMinorMarks in the .USC file,

but then I realised it is not implemented in the CPP file.

Regards

Subject: Re: Knob control
Posted by koldo on Sat, 14 Aug 2010 19:10:28 GMT
View Forum Message <> Reply to Message

Hello Jerson

Sorry. You will have today a version with the control look and feel at 100% for the styles implemented today, and .usc coherent with C++.

Subject: Re: Knob control
Posted by koldo on Tue, 17 Aug 2010 08:10:52 GMT
View Forum Message <> Reply to Message

Hello all (and specially Jerson)

C++ and .usc will be updated this night so the controls in Layout Designer look similar to real ones.

Now there is a new SetStyle(), with Rugged as one of the options.

You can try it all with the demo in Controls4U_Demo.

Please check them and propose other names it you do not like actual ones.

File Attachments
1) dib.PNG, downloaded 895 times

Subject: Re: Knob control
Posted by unodgs on Tue, 17 Aug 2010 08:36:17 GMT
View Forum Message <> Reply to Message

Hi Koldo!
Respect man I remember doing usc paint for GridCtrl.. Esc language is ok but the main trouble is you have to write paint code twice and then maintain it. It would be great if it was possible to use real widgets in designer. I guess FormEditor follows this way.

Subject: Re: Knob control
Posted by jerson on Tue, 17 Aug 2010 12:16:49 GMT
View Forum Message <> Reply to Message

Hi Koldo

Thanks very much for your efforts. The knobs really look professional grade now. Especially the ones on the row above. I haven't got down to running it on my machine yet as I am severely behind schedule on almost everything. I will definitely add to this post when I check this control. It is very exciting.

Subject: Re: Knob control
Posted by koldo on Tue, 17 Aug 2010 14:01:08 GMT
View Forum Message <> Reply to Message

unodgs wrote on Tue, 17 August 2010 10:36Hi Koldo!
Respect man I remember doing usc paint for GridCtrl.. Esc language is ok but the main trouble is you have to write paint code twice and then maintain it. It would be great if it was possible to use real widgets in designer. I guess FormEditor follows this way.
Yes Daniel

In fact my focus use to be to do the Paint() function first in .usc and after that in C++. Both are almost the same, but as there is no Painter in .usc I have to emulate some Painter capabilities. The .usc widgets are fully resizable as the Painter ones.

So it would be great to use real widgets in layout designer.

Meanwhile I strongly encourage all contributors to:

- Use Painter
Visual experience is much better

- Create .usc widgets for your classes
If a visual class is not in .usc, it does not exist

Subject: Re: Knob control
Posted by dolik.rce on Tue, 17 Aug 2010 16:22:53 GMT
View Forum Message <> Reply to Message

Hi,
I apologize for engaging into slightly off-topic discussion, but I couldn't help it

unodgs wrote on Tue, 17 August 2010 10:36It would be great if it was possible to use real widgets in designer. I guess FormEditor follows this way.

Unfortunately FormEditor can AFAIK use only widgets that are compiled in. Compiling theide every time you change something in a widget would be tedious. That might or might not be solvable in the future using some plugin-like solution to let the widget to be compiled separately. But in the meantime, I guess it should not be that hard to learn Esc how to use Painter, at least the basic functions.

koldo wrote on Tue, 17 August 2010 16:01Meanwhile I strongly encourage all contributors to:

- Use Painter
Visual experience is much better

- Create .usc widgets for your classes
If a visual class is not in .usc, it does not exist
I strongly agree with both and join your encouragement

Best regards,
Honza

Subject: Re: Knob control
Posted by koldo on Tue, 17 Aug 2010 19:17:52 GMT
View Forum Message <> Reply to Message

Hello all (and specially Jerson)

Files uploaded and documentation included.

I have improved painting speed by using virtual Layout() to do most of the Painter process, and doing just DrawImage() calls when the user interacts with the control.

Subject: Re: Knob control
Posted by jerson on Wed, 18 Aug 2010 01:30:45 GMT
View Forum Message <> Reply to Message

Hi Koldo

I had a run of the knob in my project. It looks and feels really great. Thank you

Regards

Subject: Re: Knob control
Posted by koldo on Wed, 18 Aug 2010 06:15:08 GMT
View Forum Message <> Reply to Message

Subject: Re: Knob control
Posted by jerson on Wed, 18 Aug 2010 13:00:12 GMT
View Forum Message <> Reply to Message

Hi Koldo

I think I caught some bugs on the knob USC. SetMajorMarks to 3, SetMinorMarks to 4 and the widget display is wrong. However, the CPP paint does its job well. Same way, set SetMajorMarks to 4, SetMinorMarks to 5 and the USC code crashes.

Another minor suggestion, I would define major and minor marks differently like this; but, that's my opinion.

MajorTickInterval and MinorTickInterval so user can decide the tick interval in terms of a quantum of the range. MajorTickinterval of 25 means a major tick every 25 counts of the range and minortick of 5 means a dot every 5 counts of the range

I've done something like that in the slider.cpp file I uploaded on the slider thermometer thread. this is the relevant paint code

```
 // draw gradations
 for(  int i = Min();
   ( m_nMajorTicks > 0 ) && ( i <= Max() ) ;
   i += ( m_nMinorTicks == 0 ? m_nMajorTicks : m_nMinorTicks ) ) {

  int nPos = SliderToClient( i );

  if( ( m_nMajorTicks != 0 ) && ( i % m_nMajorTicks ) == 0 )
   DrawTick( w, MAJOR, (HOVE)HoVe( HORZ, VERT ), nPos, i );
  else if( ( m_nMinorTicks != 0 ) && ( i % m_nMinorTicks ) == 0 )
   DrawTick( w, MINOR, (HOVE)HoVe( HORZ, VERT ), nPos, i );
 }
```

Best Regards

Subject: Re: Knob control
Posted by koldo on Wed, 18 Aug 2010 15:01:43 GMT
View Forum Message <> Reply to Message

Hello Jerson

Quote:I think I caught some bugs on the knob USC. SetMajorMarks to 3, SetMinorMarks to 4 and the widget display is wrong. However, the CPP paint does its job well. Same way, set SetMajorMarks to 4, SetMinorMarks to 5 and the USC code crashes.
Solved.

Quote:MajorTickInterval and MinorTickInterval so user can decide the tick interval in terms of a quantum of the range. MajorTickinterval of 25 means a major tick every 25 counts of the range and minortick of 5 means a dot every 5 counts of the range
I understand the algorithm but I do not know the reason. Why to have MajorTickinterval, MinorTickInterval and range instead of simply MajorTickinterval and MinorTickInterval?

Subject: Re: Knob control
Posted by jerson on Wed, 18 Aug 2010 15:09:59 GMT
View Forum Message <> Reply to Message

koldo wrote on Wed, 18 August 2010 20:31
I understand the algorithm but I do not know the reason. Why to have MajorTickinterval, MinorTickInterval and range instead of simply MajorTickinterval and MinorTickInterval?

Simply MajorTickInterval and MinorTickInterval should be fine. Range is nothing more than Min & Max and stays as it is.

An example. If I have Max = 100, MinorTickInterval = 5 and MajorTickInterval = 20, I will get minor marks at 5,10,15, major mark at 20, minor at 25,30,35, major at 40 and so on. Only reason I suggested this arrangement is it is easier to understand while designing the UI.

Subject: Re: Knob control
Posted by koldo on Wed, 18 Aug 2010 20:27:30 GMT
View Forum Message <> Reply to Message

jerson wrote on Wed, 18 August 2010 17:09koldo wrote on Wed, 18 August 2010 20:31
I understand the algorithm but I do not know the reason. Why to have MajorTickinterval, MinorTickInterval and range instead of simply MajorTickinterval and MinorTickInterval?

Simply MajorTickInterval and MinorTickInterval should be fine. Range is nothing more than Min & Max and stays as it is.

An example. If I have Max = 100, MinorTickInterval = 5 and MajorTickInterval = 20, I will get minor marks at 5,10,15, major mark at 20, minor at 25,30,35, major at 40 and so on. Only reason I suggested this arrangement is it is easier to understand while designing the UI.

Oh. In fact the first version was like this, like in Meter.

If everything agrees I will implement it tomorrow.

Subject: Re: Knob control
Posted by koldo on Thu, 19 Aug 2010 21:17:29 GMT
View Forum Message <> Reply to Message

Changes uploaded.

Included SetMajorStep() and SetMinorStep(). If the minor step is not multiple of major step, it is rounded.

Subject: Re: Knob control
Posted by jerson on Fri, 20 Aug 2010 01:22:49 GMT
View Forum Message <> Reply to Message

thanks Koldo. Perfect