Subject: Tabctrl and RightDown.
Posted by witriol on Wed, 14 Nov 2012 14:32:59 GMT
View Forum Message <> Reply to Message

First!
I'm sorry my English is poor.

I have a problem with RightDown(and similar Leftdown).

This is the class definition:

```
class Gun : public WithGunLayout<TopWindow>
{
.....;
Withkep_lay <ParentCtrl> mj;
.....;
}
```

I use this layoutfile in a TabCtrl:

```
Gun::Gun()
{
.....
CtrlLayout(mj);
summa_vievver.Add(mj,"Image");//suma_vievver is a TabCtrl in the Gun.lay.
.....
}
```

Should I put a picture of mj.
Right click choose the interesting part but right click does not work. This is logical RightDown of mj overdefine/overwrite Gun::RigtDown. How to do call the right click function of mj?.

Thank you for help.

Subject: Re: Tabctrl and RightDown.
Posted by koldo on Wed, 14 Nov 2012 14:50:46 GMT
View Forum Message <> Reply to Message

Hello witriol

Welcome to the forum .

Personally I would do this:

```
class Gun : public WithGunLayout<TopWindow>
{
.....;
```

```
Withkep_lay <StaticRect> mj;
.....;
}

Gun::Gun()
{
.....
CtrlLayout(mj);
summa_vievver.Add(mj.SizePos(),"Image"); // So mj will fill all suma_vievver available area when resizing
.....
}
```

Quote:Right click choose the interesting part but right click does not work. This is logical RightDown of mj overdefine/overwrite Gun::RigtDown. How to do call the right click function of mj?.

Do you mean that you want to call a function when the user right-clicks when hovering over mj?

Subject: Re: Tabctrl and RightDown.
Posted by witriol on Wed, 14 Nov 2012 15:41:19 GMT
View Forum Message <> Reply to Message

Subject: Re: Tabctrl and RightDown.
Posted by koldo on Wed, 14 Nov 2012 20:11:07 GMT
View Forum Message <> Reply to Message

Hello Witriol

That is easy. Instead of calling directly

Withkep_lay <StaticRect> mj;

you can create explicitly the class adding the mouse handler

```
class Kep_lay : public Withkep_lay<StaticRect> {
public:
 typedef Kep_lay CLASSNAME;

 virtual void RightDown(Point p, dword keyFlags) {
  Exclamation(Format("Clicked on %d:%d", p.x, p.y));
 }
```

};

and including Kep_lay mj; inside Gun class.

Subject: Re: Tabctrl and RightDown.
Posted by witriol on Thu, 15 Nov 2012 11:28:18 GMT
View Forum Message <> Reply to Message

Thanks, but is not the Gun::RightDown function. I want to run Gun::RightDown, when the user click on the mj. This function uses x, and y coordinates.

Subject: Re: Tabctrl and RightDown.
Posted by koldo on Thu, 15 Nov 2012 15:07:10 GMT
View Forum Message <> Reply to Message

Hello Witriol

Do you mean that when moving the mouse over mj, if the user right clicks the mouse a function has to be called in Gun with the coordinates?

Subject: Re: Tabctrl and RightDown.
Posted by witriol on Fri, 16 Nov 2012 08:57:29 GMT
View Forum Message <> Reply to Message

If the user use right cilck over mj, I would like to use the Gun::RightDown, not the mj::rightdown.

Subject: Re: Tabctrl and RightDown.
Posted by koldo on Fri, 16 Nov 2012 12:40:08 GMT
View Forum Message <> Reply to Message

Hello Witriol

A simple way would be to add a reference to Gun in Kep_lay class constructor and call Gun::RightDown handler from Kep_lay::RightDown handler.

Subject: Re: Tabctrl and RightDown.
Posted by witriol on Sun, 18 Nov 2012 15:59:54 GMT
View Forum Message <> Reply to Message

I definied this class.

```
class Kep_lay : public Withkep_lay<ParentCtrl> {
public:
 typedef Kep_lay CLASSNAME;

 virtual void RightDown(Point p, dword keyFlags) ;
};
```

And included in the Gun class:

```
class Gun : public WithGunLayout<TopWindow> {
....
public:
....
 Kep_lay mj;
....
```

And I wrote the function:

```
void Kep_lay::RightDown(Point p, dword keyFlags)
{
....
Exclamation("OK");
....
}
```

It's work. BUT!

How do I call, Gun::RightDon (or other Gun::I_definied)?
I tried like atomic thread callback, but not work.

Subject: Re: Tabctrl and RightDown.
Posted by koldo on Sun, 18 Nov 2012 20:59:47 GMT
View Forum Message <> Reply to Message

Hello Witriol

Perhaps it is not the nicest way to do it but you may do something like this:

```
class Gun : public WithGunLayout<TopWindow> {
....
public:
....
```

```
 Gun() {
  mj.Init(this);
 }
 Kep_lay mj;
 void RightMouseHandler(Point p, dword keyFlags) {
  // Do it.
 }
....
};

class Kep_lay : public Withkep_lay<StaticRect> {
public:
 typedef Kep_lay CLASSNAME;

 void Init(Gun *_parent) {parent = _parent;}
 virtual void RightDown(Point p, dword keyFlags) {
  parent->RightMouseHandler(p, keyFlags);
private:
 Gun *parent;
};
```