Subject: Re: How to catch and overwrite child control event?
Posted by Oblivion on Mon, 21 Jan 2008 01:00:36 GMT

One more general (maybe application wide) way is to use a "keyhook".
I assume you derive your class from Ctrl or a Ctrl derived class (as usual in U++):

Define a KeyHook.

Below is the code of a key hook for intercepting Ctrl+Home key:

```
class foo : public Ctrl
{
// some other declarations...
// some other declarations...

 static bool CtrlKeyHook(Ctrl* ctrl, dword key, int count) { if(key == K_CTRL_HOME && ctrl ==
&myctrl) // here goes the intercepting function or method... // ; return true; }
}
```

As You can see in the above declaration, you can identify the "Ctrl" which sends the message too.

in the class implementation:

```
foo::foo()
{
 InstallKeyHook(&foo::CtrlKeyHook);
}

foo::~foo()
{
 DeinstallKeyHook(&foo::CtrlKeyHook);
}
```

(Two things: don't forget to return true (and dont forget to "Deinstall" hook...)

You can find more info about other hook types (mouse, state) in CtrlCore/Src/Ctrl section of the Help Topics of TheIDE

Ps: I don't know why your code didn't work (I should work), but I uploaded a Key() virtual function version below. I Hope It'll help...

File Attachments

1) KeyTest.rar, downloaded 399 times