
Subject: Re: Q: howto incorporate a native console window in GUI

Posted by [dolik.rce](#) on Thu, 10 Feb 2011 13:54:56 GMT

[View Forum Message](#) <> [Reply to Message](#)

I think you are taking it too seriously Here is a working example (change the paths if you are on windows): #include <CtrlLib/CtrlLib.h>
using namespace Upp;

```
class ConsoleCtrl:public LineEdit{
typedef ConsoleCtrl CLASSNAME;
LocalProcess p;
int last;
public:
virtual bool Key(dword key,int count){
switch(key) {
case K_ENTER:{
String cmd=line.Top().text.Right(GetLength()-last)+'\n';
Insert(GetLength(),"\n");
p.Write(cmd);
}
return true;
case K_UP: //might implement history browsing...
case K_DOWN:
case K_PAGEUP:
case K_PAGEDOWN:
case K_CTRL_PAGEUP:
case K_CTRL_HOME:
case K_CTRL_PAGEDOWN:
case K_CTRL_END:
return true;
}
return LineEdit::Key(key,count);
}
void Read(){
String s;
p.Read(s);
if(s.IsEmpty()) return;
Insert(GetLength(),s);
last=GetLength();
SetCursor(last);
}
ConsoleCtrl(const String& cmd="/bin/bash"){
p.Start(cmd);
SetTimeCallback(-50,THISBACK(Read));
}
~ConsoleCtrl(){
p.Kill();
}
```

```
};

class App :public TopWindow {
    typedef App CLASSNAME;
    ConsoleCtrl c;
public:
    App():c("/usr/bin/python -i"){ // -i is required to use it interactively
        Add(c.SizePos());
    }
};
```

```
GUI_APP_MAIN{
    App().Sizeable().Run();
}
```

It is just quickly hacked to gather, but the general idea seems to be working just fine. I tested it with bash and python and it feels almost like a terminal, except for missing features like history or tab completion... but those could be probably implemented too with a bit of work.

Honza
