
Subject: Init a ctrl inside INITBLOCK

Posted by [koldo](#) on Wed, 11 Jan 2012 08:19:40 GMT

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

Hello all

I wanted to init a Ctrl inside INITBLOCK:

```
struct Tab1 : StaticRect {  
    ...  
}
```

```
INITBLOCK {  
    RegisterExample("Basic", new Tab1);  
}
```

It works but program look is weird:

Is there a safe way to init a Ctrl inside INITBLOCK?

File Attachments

1) [dib.PNG](#), downloaded 430 times

Subject: Re: Init a ctrl inside INITBLOCK

Posted by [dolik.rce](#) on Wed, 11 Jan 2012 09:39:46 GMT

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

I think it is because you are using some GUI stuff before it is initialized, like font sizes, styles etc. Safe solution should be to just make the RegisterExample() store the list of things to be initialized into some container. The actual initialization can then be done by separate function called in GUI_APP_MAIN or in the constructor of the application window.

Best regards,
Honza

Subject: Re: Init a ctrl inside INITBLOCK

Posted by [mr_ped](#) on Thu, 12 Jan 2012 12:16:37 GMT

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

"Is there a safe way to init a Ctrl inside INITBLOCK?"
no. Only after GUI is initialized.

Subject: Re: Init a ctrl inside INITBLOCK
Posted by [koldo](#) on Thu, 12 Jan 2012 14:30:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

Ok. Thank you.

Subject: Re: Init a ctrl inside INITBLOCK
Posted by [koldo](#) on Thu, 12 Jan 2012 14:50:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Honza

I have successfully applied this:

Quote:Safe solution should be to just make the RegisterExample() store the list of things to be initialized into some container. The actual initialization can then be done by separate function called in GUI_APP_MAIN or in the constructor of the application window.

Now code is like this:

```
struct MyExample : ParentCtrl {
    virtual void Init() = 0;
};

struct Tab1 : WithTab1<MyExample> {
    Tab1() {...} // No GUI here
    virtual void Init() {
        CtrlLayout(*this);
        ... // GUI here
    }
    ...
}

INITBLOCK {
    RegisterExample("Basic", new Tab1);
}

...

GUI_APP_MAIN
{
    for (int i = 0; i < Examples().GetCount(); ++i)
        Examples()[i].ctrl->Init(); // GUI loaded here

    ScatterCtrl_Demo().Run();
}
```

Subject: Re: Init a ctrl inside INITBLOCK
Posted by [dolik.rce](#) on Thu, 12 Jan 2012 15:32:03 GMT
[View Forum Message](#) <> [Reply to Message](#)

Well, that looks like a valid solution too

It never stops to amaze me how a simple task can be done in so many ways

Honza

Subject: Re: Init a ctrl inside INITBLOCK
Posted by [dolik.rce](#) on Thu, 12 Jan 2012 17:23:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

Just for the sake of completeness, here is another solution, based on templates:

```
template<class T>
inline Ctrl& Create() {
    return Single<T>();
};
```

```
typedef Ctrl& (*CreateFunc)();
GLOBAL_VAR(Vector<CreateFunc>,sRegisteredObjects);
```

```
template<class T>
void Register(){
    sRegisteredObjects().Add(Create<T>());
};
```

```
GUI_APP_MAIN{
    App a;
    for(int i=0;i<sRegisteredObjects().GetCount();i++){
        Ctrl& c = sRegisteredObjects()[i]();
        //do something with it...
    }
    a.Run();
};
```

```
INITBLOCK{
    Register<Button>();
    Register<EditString>();
};
```

It is not as simple as yours, but it is less demanding on the registered objects. Also, I'm pretty sure there is many other ways to do this

Honza

Subject: Re: Init a ctrl inside INITBLOCK
Posted by [koldo](#) on Thu, 12 Jan 2012 20:27:35 GMT
[View Forum Message](#) <> [Reply to Message](#)
