# georgeOfChina



# Qt编写自定义控件及插件的使用

在使用Qt Designer设计窗体界面时,我们可以使用Widget Box里的窗体控件非常方便的绘制 界面,比如拖进去一个按钮,一个文本编辑器等。虽然Qt Designer里的控件可以满足我们大部分 的需求,但是有时候,也会产生一些特殊的需要,比如一个输入框,我们要输入的是经纬度,此 时就会有两种输入方式,一种是小数形式,一种是度分秒的形式,此时只使用一个简单的LineEdit 是无法满足需求的。我们设想构造这样一个输入控件,它可以支持浮点数输入,同时它还具有一 个属性,更改这个属性可以使其切换为经纬度输入形式。如果我们的多个窗体上都需要输入经纬 度,那么构造这样一个控件,将会非常方便。下面就以此为例,讲解一下如何创建自定义的窗体 控件。

# 第一步: 创建QtDesigner自定义控件工程

打开Qt Creator,创建一个Qt 设计师自定义控件,如下图所示:

📴 New Project		×
选择一个模板:		所有模板
项目 Application Library 其他项目 Non-Qt Project Import Project 文件和英	<ul> <li>Qt单元测试</li> <li>Qt4 设计师旨定义控件</li> <li>Qt Quick UI</li> <li>Qt Quick Controls UI</li> <li>Auto Test</li> <li>子目录项目</li> <li>Empty qmake Project</li> <li>Code Snippet</li> </ul>	创建一个94设计师自定义控件或者一个自定义控件集合。 <b>支持的平台</b> : 桌面 通用Linux设备
		Choose Candel

根据向导提示,建立一个新类,取名"Test"(注:首字母大写,不能全小写,不然后面会出错),一直点击"下一点",完成工程创建。

### 公告

昵称: georgeOfChina
园龄: 2年8个月
粉丝: 4
关注: 4
+加关注

۲		20	20年6	月		>
日	_	$\equiv$	Ξ	四	五	六
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4
5	6	7	8	9	10	11



常用链接

我的随笔 我的评论 我的参与 最新评论 我的标签

随笔分类

Android(2) C++ 内存(1) Qt(4) 网络通讯(1)

随笔档案

2017年12月(1) 2017年11月(2) 2017年10月(5)

#### 最新评论

1. Re:sdk与ndk的区别 不错

--beautifulzzzz

### 阅读排行榜

1. Qt编写自定义控件及插件的使用(25744) 2. Qt在Windows下的三种编程环境搭建(90 61)

- 3. sdk与ndk的区别(6558)
- 4. Qt多线程和GUI界面假死(6358)
- 5. QT只运行一个实例(1196)

#### 评论排行榜

×

1. sdk与ndk的区别(1)

### 推荐排行榜

- 1. sdk与ndk的区别(2)
- 2. Qt编写自定义控件及插件的使用(2)
- 3. C++中一个class类对象占用多少内字节 (7个例子,很清楚)(1)

Location Kits 📫 Custom Widgets	目定义控件列表 指定自定义控件列表及其属性。 控件类(C):	
Plugin Details 汇总	Test <新类>	<ul> <li>● 植接痒(L)</li> <li>● 包含项目(J)</li> <li>● 创建代码框架(K)</li> <li>● 包含项目(J)</li> <li> <sup>1</sup>控件庫(Y): test</li> <li> <sup>1</sup>控件项目文件(F): test.pri     </li> <li> <sup>1</sup>控件项文件(E): test.h     </li> <li> <sup>1</sup>控件项文件(F): test.opp     </li> <li> <sup>1</sup>控件项文件(F): testplugin     </li> <li> <sup>1</sup>插件项文件(F): testplugin.opp     </li> <li> <sup>1</sup>燃売     </li> </ul>

### 工程创建完毕,如下图所示:



### 第二步:编辑工程代码

在新建的工程中,Test类没有ui界面文件,我们要手动添加上去,如果没有ui界面文件,工程 编译也可以通过,但该控件添加到ctreator后,一打开desinger工具的时候就会直接挂掉。所以, 我们可以把原先的Test类的test.c和test.h删除掉,重新添加一个Test类。





送择一个摄版: 文件和英 C++ Modeling Qt GLSL General Java Python 送择一个摄版: Q (t tem Model Q (t tem Model Q (t designer form Q (t Resource File Q (ML File (Qt Quick 1) Q (Quick UI File J S File 文件和現文件》用于实现,您可以将此箇体文件和相应的类(c+	19 新建文件		
文件和美     ① qt Item Model     創建一个4+设计师面体文件和相应的类(++-4) 件和原文件)用于实现,您可以将此窗体文件 算和原文件)用于实现,您可以将此窗体文件 算和原文件和相应的类(++-4) 件和原文件)用于实现,您可以将此窗体文件 类加入到已经存在的94 控件项目中。       Qt     Qt Designer Form     ① qt Resource File       ① qt Resource File     ② QML File (Qt Quick 1)       ③ qt Quick UI File     ② qt Quick UI File       Python     〕 J S File	选择一个模板:		所有模板
	文件和通 C++ Modeling Qt GLSL General Java Python	Qt Item Model Qt 设计师界面类 Qt Designer Form Qt Resource File QML File (Qt Quick 1) QML File (Qt Quick 2) Qt Quick UI File JS File	创建一个44设计师面体文件和相应的类(C++头文件和现象文件)用于实现。您可以希比面体文件和 类加入到已经存在的94 这件项目中。 支持的平台:桌面通用Linux设备

- Qt设计器界面类			
Form Template む Class Details 汇总	选择类名 类 类名(c):	Test	
	头文件(H): 源文件(S): 界面文件(F): 略忽(P):	test.h test.opp test.ui	كاللاح
	<b>府</b> (r):	L:\VMV\qt_eg_code\lest	浏览

为了减少出错机率,此时不对界面做任何动作。

在工程头文件"test.h"中添加 "#include <QtUiPlugin/QDesignerExportWidget>",在类名前添加"QDESIGNER\_WIDGET\_EXPORT"宏。如下图:

->-		u wau
a 🔇	( > 1	👔 🚹 test. h* 🗾 🔫 📉 🎁 Test
	1	#ifndef TEST H
	2	#define TEST H
	3	
	4.	<pre>#include <qwidget></qwidget></pre>
	5	<pre>#include <qtuiplugin qdesignerexportwidget=""></qtuiplugin></pre>
	6	
	7 🗸	namespace Ui {
	8	class Test;
	9	
	10	
	11	Class QDESIGNER_WIDGET_EXPORT Test : public Qwidget
	12	1 O OBJECT
	14	Q_OBOLCI
	15	public:
	16	explicit Test(OWidget *parent = 0);
	17	~Test();
	18	
	19	private:
	20	Ui::Test *ui;
	21	};
	22	
	23	<pre>#endif // TEST_H</pre>
	24	

### 第三步:编译工程

保存并选择release方式编译。先qmake一下,

	test.cpp	
	test.h	
	test.ui	
	项目: Test 构建舊件: Desktop Qt 5.7.0 IISVC2013 部署: 在本地部署 部署: 古本地部署	32bi t
Test	构建	
Γ.	Debug	
Release	Profile	



点击左下角的"构建"按钮,进行编译。

Test		
Γ.		
Release		
Â		
~		
No.	■ P. Type to locate (Ctrl+K)	1 jā

编译完成。此时会生成testplugin.dll和testplugin.lib两个库。

### 第四步:部署插件

编译完成后,在输出目录下,找到对应的testplugin.dll和testplugin.lib两个文件。分别拷贝到 QT库的designer路径下和QT的IDE工具的designer路径下。

以我本地5.7.0路径为例: (具体以个人的QT安装路径为准)

QT库的designer库路径: C:\software\GRG\Qt\Qt5.7.0\5.7\msvc2013\plugins\designer

QT库的IDE路径: C:\software\GRG\Qt\Qt5.7.0\Tools\QtCreator\bin\plugins\designer

→ 此电脑 → Windows (C:) → software →	$GRG \rightarrow Qt \rightarrow Qt5.7.0 \rightarrow T$	ools > QtCreator >	bin > plugins > des
名称	修改日期	类型	大小
axwidget.dll	2016/6/1 19:53	应用程序扩展	204 KB
gquickwidget.dll	2016/6/1 19:53	应用程序扩展	22 KB
testplugin.dll	2017/11/2 20:00	应用程序扩展	20 KB
testplugin.lib	2017/11/2 20:00	Object File Library	6 KB
			plugins / designer
名称 个	修改日期	类型	大小
名称 函 qaxwidget.dll	修改日期 2016/6/13 4:12	类型 应用程序扩展	大小 204 KB
名称 ④ qaxwidget.dll ④ qquickwidget.dll	修改日期 2016/6/13 4:12 2016/6/13 4:12	类型 应用程序扩展 应用程序扩展	文小 204 KB 22 KB
名称 國 qaxwidget.dll 國 qquickwidget.dll 國 qwebengineview.dll	修改日期 2016/6/13 4:12 2016/6/13 4:12 2016/6/13 2:59	美型 应用程序扩展 应用程序扩展 应用程序扩展	大小 204 KB 22 KB 21 KB
名称 ③ qaxwidget.dll ④ qquickwidget.dll ⑤ qwebengineview.dll ⑤ testplugin.dll	修改日期 2016/6/13 4:12 2016/6/13 4:12 2016/6/13 2:59 2017/11/2 20:00	类型 应用程序扩展 应用程序扩展 应用程序扩展 应用程序扩展	大小 204 KB 22 KB 21 KB 20 KB

### 第五步:测试使用插件

新建一个工程,在工程使用该Test自定义插件。

1、打开\*.ui, 在左下角会出现我们前面编译的自定义插件, 如下图:

	<u></u>		
	- QC	QuickWidget	
-	~	自定义窗口部件	
>	Qt Te	st	

直接拖拽到ui界面。

	对象		类
			The second second second
	× 📷	Widget	QWidget
			Test
		test	Test
<u>.</u> <u>.</u>			
i na manana a kata na manana kata kata na manana manana manana manana manana ma			
e an a taran an ar 🖌 📕 na taran an 📕 🖉 an an ar an			
化氯化 化合金法 化合金法 化合金法 化合金法 化合金法 化合金法 化合金法 化合金			
en a ser			
nanga anananga anangka an			
しんえ ひかええ ひえんえ ひえええ ひえええ ひかええ ひえええ ひえええ ひろええ ひ			

2、在工程所在文件目录中,新建include文件夹,将前面的插件头文件test.h放在该include文件夹中。新建lib文件夹,将前面生成的testplugin.lib静态库放在此lib文件夹中。

名称	修改日期	类型	大小
📙 include	2017/11/3 10:15	文件夹	
📙 lib	2017/11/3 10:15	文件夹	
• main.cpp	2017/11/3 10:10	C++ Source file	1 KI
TestUsePlugin.pro	2017/11/3 10:10	Qt Project file	1 KI
🗟 TestUsePlugin.pro.user	2017/11/3 10:10	Visual Studio Pr	23 KI
widget.cpp	2017/11/3 10:10	C++ Source file	1 K
b widget.h	2017/11/3 10:10	C++ Header file	1 KE
🗋 widget.ui	2017/11/3 10:10	UI 文件	1 KI
脑 > Windows (C:) > GRG > qi	t_eg_code → TestUsePlugin	> include	+ 1
脑 > Windows (C:) > GRG > q 名称	t_eg_code > TestUsePlugin 修改日期	⇒ include 类型	大小
脑 > Windows (C:) > GRG > qi 名称 h test.h	t_eg_code → TestUsePlugin 修改日期 2017/11/2 19:33	include 类型 C++ Header file	大小 1 Ki
脑 > Windows (C:) > GRG > q 名称 for	t_eg_code > TestUsePlugin 修改日期 2017/11/2 19:33	⇒ include 类型 C++ Header file	大小 1 Ki
脑 > Windows (C:) > GRG > qi 名称 h test.h	t_eg_code > TestUsePlugin 修改日期 2017/11/2 19:33	⇒ include 类型 C++ Header file	大小 1 K
脑 > Windows (C:) > GRG > qi 名称 庐 test.h	t_eg_code > TestUsePlugin 修改日期 2017/11/2 19:33	⇒ include 类型 C++ Header file	大小 1 KI
脑 > Windows (C:) > GRG > qi 名称 h test.h	t_eg_code > TestUsePlugin 修改日期 2017/11/2 19:33	⇒ include 类型 C++ Header file	大小 1 Ki
脑 > Windows (C:) > GRG > q 名称 h test.h	t_eg_code > TestUsePlugin 修改日期 2017/11/2 19:33	⇒ include 类型 C++ Header file	大小 1 Ki
脑 > Windows (C:) > GRG > q 名称	t_eg_code > TestUsePlugin 修改日期 2017/11/2 19:33	⇒ include 类型 C++ Header file	大小 1 Ki
脑 > Windows (C:) > GRG > q 名称	t_eg_code > TestUsePlugin 修改日期 2017/11/2 19:33	⇒ include 类型 C++ Header file	大小 1 Ki
脑 > Windows (C:) > GRG > q 名称 h test.h	t_eg_code > TestUsePlugin 修改日期 2017/11/2 19:33 gt eg code > TestUsePlugi	⇒ include 类型 C++ Header file	大小 1 KI

	名称	修改日期	类型	大小	
*	🙀 testplugin.lib	2017/11/2 20:00	Object File Library		6 K
*					

3、打开测试工程的\*.pro文件。添加相应的include路径和lib路径。

### 添加以下内容:

4

1 LIBS += \$\$PWD/lib/testplugin.lib

2 INCLUDEPATH += \$\$PWD/include

### 如下图:

1	±
2	#
3	# Project created by QtCreator 2017-11-03T10:10:40
4	#
5	#
6	
7	QT += core gui
8	
9	<pre>greaterThan(QT_MAJOR_VERSION, 4): QT += widgets</pre>
10	
11	TARGET = TestUsePlugin
12	TEMPLATE = app
13	2020 Suburdina da anti-a de la companya d
14	LIBS += \$\$PWD/lib/testplugin.lib
15	INCLUDEPATH += \$\$PWD/include
16	
17	
18	SOURCES += main.cpp\
19	widget.cpp
20	
21	HEADERS += widget.h
22	
23	FORMS += widget.ui
24	

4、切换到release编译模式,点击qmake,生成相应的makefile文件。并将前面生成的 testplugin.dll动态库添加到release输出目录中。如下图:



手册

E∰V1.5

此时。直接运行\*.exe就可以使用该插件了。自定义插件的新建并使用到此算是OK了。

## 第六步:Qt自定义插件注意事项:

1: 每个Qt库bin目录的designer可执行文件都是和该库同一个编译器编译的,可用,如果想要集成 到Qt Creator中,则需要注意版本,一般在windows上的Qt Creator版本是MSVC的,则需要对应 的Qt库也是MSVC编译的,库版本和编译器版本必须保持一致才能是顺利集成到Qt Creator的重要

前提。

2: 自定义控件的名称不能小写,否则拖过去的控件自动生成的默认名称和类名一样,会编译通不过。这个问题坑了我很久,造成自动生成的UI代码保存,一直没有怀疑,后面才发现自动生成的代码类名和实例名称一样,冲突导致的。

3: 自定义控件类头文件引入, Qt5.7以下版本为#include <QtDesigner/QDesignerExportWidget> 以上版本为#include <QtUiPlugin/QDesignerExportWidget>

4: 类名前必须加入 QDESIGNER\_WIDGET\_EXPORT 宏。否则集成到Qt Creator 中编译会报错。不加的话可以在设计器中加载,但是编译会报错。

5:如果将生成好的dll文件放到Qt库目录下的 plugins\designer 下,可以在 designer 中看到。放到 Qt Creator下的 bin\plugins\designer 则可以集成到Qt Creator中。

6:将自定义控件的头文件、dll文件、lib (mingw编译器为.a)文件复制出来,放到include (可自己随便命名,我这里习惯用include)目录,将include目录放到项目的源码文件下,在使用了自定义控件的项目的pro文件中,增加两行 INCLUDEPATH += \$\$PWD/include LIBS += \$\$PWD/include/\*\*\*.lib(mingw编译器为.a),这样可以正常编译,但是编译完成后不能运行,还需要将对应自定义控件的dll文件复制到可执行文件同一目录即可,至此大功告成。

番外话:大部分文章介绍都是将对应的库文件和头文件放到Qt安装目录对应文件夹下,为什么这 里要放到一个include目录,随着项目一起呢?个人是这么理解的,随项目一起,每次都可以很方 便的将运行库文件复制到可执行文件同一目录,而不会忘记从Qt库对应目录找该运行库。而且发 布代码的时候也可以有个很好的参考。

7: 官网提供的Qt Creator版本基本上是MSVC版本,如果需要在mingw的Qt库对应的Qt Creator中集成自定义控件,需要自己用对应的Qt库编译Qt Creator源码。

参考:

http://www.cnblogs.com/feiyangqingyun/p/6182320.html http://blog.csdn.net/giselite/article/details/12622429





《上一篇: Qt在Windows下的三种编程环境搭建

»下一篇:<u>QT只运行一个实例</u>

posted @ 2017-11-03 10:42 georgeOfChina 阅读(25749) 评论(0) 编辑 收藏

📮 lanmanck:

兄台真是帮我大忙了,谢谢了

刷新评论 刷新页面 返回顶部

### 发表评论

支持Markdowr	ı				
提交	退出	订阅评论			

#### 感谢您的回复:) 服务器端执行耗时36毫秒

### [Ctrl+Enter快捷键提交]

【推荐】了解你才能更懂你,博客园首发问卷调查,助力社区新升级 【推荐】超50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库 【推荐】开放下载!《阿里巴巴大数据及AI实战》深度解析典型场景实践

#### 相关博文:

- 1年天時文:
   4編写Q1 Designer自定义控件(一) ——如何创建并使用Qt自定义控件
   Qu编写自定义控件插件路过的坑及注意事项
   Qt入门之基础篇(二): Qt项目建立、编译、运行和发布过程解析
   Qt入门之基础篇(二): Qt项目建立、编译、运行和发布过程解析

- ·QT5 动态链接库的创建和使用
- » 更多推荐…

### 最新 IT 新闻:

- ·微软宣布7月恢复Windows 10可选更新的发布工作
- · Ripple20漏洞影响数亿设备
- ·Twitter将推出新的语音推文功能旨在创造"更人性化的体验"
- ·MIT科学家研发全新家庭健康设备 利用无线信号监测生命体征
- · Apple Watch第一季度出货量远超华为三星,总出货量即将破1亿块

» 更多新闻…

Copyright © 2020 georgeOfChina Powered by .NET Core on Kubernetes