

Xizhi Zhu的博客

[登录](#) [注册](#) [博客首页](#)

Known as Steven

全站

搜索

[空间](#)[博客](#)[好友](#)[相册](#)[留言](#)[用户操作](#)[\[留言\]](#) [\[发消息\]](#) [\[加为好友\]](#)

Steven ID : zhu_xz



共 13317 次访问，排名 12378，好友 18 人，关注者 31 人。

说得好听叫工程师，其实就是个软件民工

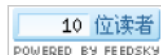
Steven 的文章

原创 23 篇

翻译 0 篇

转载 3 篇

评论 18 篇

[订阅我的博客](#)

zhu_xz 的公告

神爱世人，甚至将他的独生子赐给他们，叫一切信他的，不至灭亡，反得永生。（约翰福音 3:16）

[文章分类](#)

- [Cucumber](#)
- [Linux](#)
- [Maemo](#)
- [Qt](#)
- [Ruby](#)
- [W3C Widget](#)
- [Web Runtime](#)
- [网络安全](#)
- [杂七杂八](#)

[原](#) [用Qt开发Web和本地混合的应用](#) [收藏](#)

QtWebkit 模块使得Qt widget能够通过HTML的object标签嵌入到web页面中，并通过JavaScript代码进行访问，而Qt对象也能相应的访问web页面元素。

将Qt对象插入到web页面中

首先继承并实现QWebPluginFactory类：

```
1. class MyPlugin: public QWebPluginFactory
2. {
3.     Q_OBJECT
4. public:
5.     MyPlugin(QObject* parent = 0);
6.     // 当QtWebkit模块解析到HTML中的object标签时被调用
7.     virtual QObject* create(const QString& mimeType,
8.                             const QUrl& url,
9.                             const QStringList& argumentNames,
10.                            const QStringList& argumentValues) const;
11.     // 返回该factory支持的plugin
12.     virtual QList<Plugin> plugins() const;
13. };
14. MyPlugin::MyPlugin(QObject* parent)
15.     : QWebPluginFactory(parent)
16. {
17.     // do nothing
18. }
19. QObject* MyPlugin::create(const QString& mimeType,
20.                            const QUrl& url,
21.                            const QStringList& argumentNames,
22.                            const QStringList& argumentValues) const
23. {
24.     QLabel* label = new QLabel(url.toString());
25.     return label;
26. }
27. QList MyPlugin::plugins() const
28. {
29.     QList list;
30.     QWebPluginFactory::Plugin entry;
31.     entry.name = "plugin名";
32.     entry.description = "描述";
33.     list.push_back(entry);
34.     return list;
35. }
```

然后将其嵌入到HTML页面中：

```
1. <html>
```

6. </html>

最后，在Qt代码中加载并显示 该页面：

```
1. QWebView view;
2. view.settings()->setAttribute(QWebSettings::PluginsEnabled, true);
3. view.page()->setPluginFactory(new MyPlugin);
4. view.load(QUrl("test.html"));
5. view.show();
```

这里，当QtWebkit模块解析该HTML文件，遇到object标签时，会自动调用QWebPluginFactory::create()函数，并传递以下参数：

contentType：application/xz-plugin

url：http://blog.csdn.net/zhu_xz

此外，还可以通过object标签的param子标签传递参数。

在web页面中访问Qt对象

在web页面中可以通过类似于下的JavaScript代码访问Qt对象：

```
1. <a href=javascript:document.getElementById("myLabel").setText("通过JavaScript访问Qt对象");" mce_href="javascript:document.getElementById("myLabel").setText("通过JavaScript访问Qt对象");">点击访问Qt对象</a>
```


在Qt对象中访问web页面元素

在QWebPluginFactory::create()函数中添加以下代码：

```
1. // myPlugin指向的对象可在HTML中用名字myPluginObject进行访问
2. webView->page()->mainFrame()->addToJavaScriptWindowObject("myPluginObject", myPlugin);
3. // 当信号signalEmitted被触发时，调用JavaScript的functionToCall函数
4. webView->page()->mainFrame()->evaluateJavaScript("myPluginObject.signalEmitted.connect(functionToCall);");
```

发表于 @ 2009年12月25日 02:21:00 | [评论 \(2 \)](#) | [举报](#) | [收藏](#)

旧一篇:Ubuntu下删除不用的内核和配置文件 | 新一篇:Qt单元测试框架

tingsking18  发表于2009年12月25日 上午 10:21:36 [举报](#) [回复](#)



不错！感觉这个很有局限，不如Qt 提供的browser plugin用起来方便

zhu_xz 发表于2009年12月25日 下午 03:56:04 [举报](#) [回复](#)



你说的是QtBrowserPlugin？我觉得是各有所长吧

发表评论

表情：



评论内容：

用户名： 匿名用户

[登录](#) [注册](#)

发表评论

热门招聘职位

广州金确度高薪诚聘 VC 项目主管

【惠普】研发中心诚聘 .NET Tech Lead(Quality Center)

【惠普】研发中心诚聘 研发经理 (Quality Center) (猎头职位)

【惠普】研发中心诚聘 C++资深开发工程师

北京明天高软科技 高级需求分析师

北京明天高软科技有限公司 高级 java工程师

北京明天高软科技 JAVA开发工程师

北京明天高软科技 测试工程师

【惠普】研发中心诚聘 QA Tech Lead(LoadRunner)

【惠普】研发中心诚聘 .NET Senior Developer (Quality Center)

【惠普】研发中心诚聘 R&D Tech Lead(LoadRunner or QTP)

【CSDN】招聘:Ruby 工程师、Java搜索工程师、运营编辑、兼职编辑

广州金确度高薪诚聘 VC 项目主管

【惠普】研发中心诚聘 .NET Tech Lead(Quality Center)

Java搜索工程师、运营编辑、兼
职编辑

[公司简介](#) | [招贤纳士](#) | [广告服务](#) | [银行汇款帐号](#) | [联系方式](#) | [版权声明](#) | [法律顾问](#) | [问题报告](#)

北京创新乐知广告有限公司 版权所有，京 ICP 证 070598 号

世纪乐知(北京)网络技术有限公司 提供技术支持

 Email:webmaster@csdn.net

Copyright © 1999-2010, CSDN.NET, All Rights Reserved



.

.