Csdn：idea4good/b\*\*\*e

简单出奇迹，早晚用大力（最简单，风骚的跨平台UI框架）

超轻量/开源UI框架，支持Android/Linux/Windows/MR/VR

超轻量/开源UI框架，一旦拥有，别无所求

开发板的绝配 -- 超轻量/开源UI框架

买不起开发板的，看过来

买不起VR/MR头盔的，看过来

一次编写，到处运行（Android/Linux/Windows/MR/VR）

照这个作，C/C++通吃Android/Linux/Windows/MR/VR

C/C++的跨平台之旅，支持Android/Linux/Windows/MR/VR

通吃Android/Linux/Windows/MR/VR， C/C++程序员最光荣

C/C++的逆袭，通吃Android/Linux/Windows/MR/VR

照这个作，嵌入式软件跑在手机上

照这个作，Windows程序跑在手机上

照这个作，嵌入式软件跑在VR/MR上

别人已上天，你还在MFC/QT

# 炫酷代码分享 --- 教你让程序同时运行在Android, Windows, Linux，VR上

----------------------------------------------------------------------------------------------------------------

UI框架到处有，但能同时在Android, Linux, Windows, MR（混合现实）, VR（虚拟现实）设备上运行的UI就寥寥无几了；即是有，也庞大无比，中看不中学~~

笔者认为造成这些的原因，只有一个：代码不够简单。记得一位东北的大师说过：“简单出奇迹，必须用大力”

好吧，真心给大家推荐这款“大力”—超轻量UI框架

首先，核心完全使用C/C++编写，代码量仅仅数千行，尽管只有数千行还是分门别类，将这数千行的代码均匀的分布在约30个文件里面，平均每个文件的代码量约200行左右。

这么简单？搞得学会了，都没有成就感~~~~

其次，为了解决跨平台问题，提供了完整的封装方法，代码链接如下：

<https://github.com/idea4good>

简单解释一下代码哈：

NoodleLinuxSample: 是Linux的封装方法，非常普通的封装，但令人发指的是，它居然可以适应各种尺寸的显示屏；而且还能在PC上面预览运行效果，嵌入式工程师再也不用编译一遍，烧写一遍了。。。

NoodleAndroidSample: 是Android的封装方法，跟笔者一样穷苦，买不起开发板的老铁们，记住了，你的手机就是开发板！老铁，在你滑动屏幕的时候，不要以为是Android在发功哈，那是“大力”在顶你！

什么？手机也没有？好吧，往下看。。。

NoodleWinSample: 是Windows的封装方法，靠！这也值得一讲？这可不是普通的Windows！首先它可以是普通的Win32程序，也可以是UWP（universal windows platform）程序；UWP可以运行在MR（混合现实）/VR（虚拟现实）设备上，老实说，没有UWP，我真不知道怎么圆这个MR/VR的场；

最后，最重要的一点！特别是想通过APP赚钱的老铁们，UWP只能通过微软的store发布，彻底消灭盗版，老铁，你的血汗钱有着落了。

福利时间到，眼球经济，没有福利，老铁也会懈怠；下面的视频展示了基本的运行效果，虚拟现实的老铁们，戳呀~~~

[http://v.youku.com/v\_show/id\_XMjc2MjA4MDgzMg==.html](http://v.youku.com/v_show/id_XMjc2MjA4MDgzMg==.html?spm=a2h0k.8191407.0.0&from=s1.8-1-1.2)

-------------------------------------------------------------------------------

感谢楼上老铁的支持！！！

其实3个代码库里面的readme.md文档已经说的比较清楚了，一方面老铁可以耐心读一下，一方面我做个简单的翻译，希望能帮到你

3个readme.md文档都是在说如何编译，运行。

其中android，windows的没啥好说的，老铁需要安装android studio和visual studio，然后点击build/run就可以了。

值得注意android程序不要在模拟器上面运行，要在手机上面运行。因为UI框架是基于arm编译的。

其中略微复杂的是linux下面的编译，运行方法。

先说运行吧：

在prebuild里面已经给大家编译好了可以直接执行的程序，但由于下载后，文件属性可能发生变化，所以保险起见，需要老铁先把文件属性改为可读可写可运行，命令如下：

chmod 777 \*

然后，就可以直接运行程序了，命令如下：

./sample\_native 1 8

为了能看见运行的啥东西，你可以继续输入命令:

ss

这是，你就会在该目录发现9个bitmap文件，这就是界面的预览快照了。

是不是感觉没有动态效果？为了能够动态的看运行界面，你需要安装网页浏览器，并继续输入命令：

sslp

然后用浏览器打开Display.html，你就可以动态的观察界面变化了:)

是不是感觉动态效果比较卡顿，没有实时性？为了改善，你需要做下面的工作：

1. 确认你的linux PC安装了QT creator，如果没有可以通过这个命令安装: sudo apt-get install qtcreator

2. 32位的linux PC运行命令: ./display-x86 | ./sample\_native-x86 shared-fb

3. 64位的linux PC运行命令: ./display-x64 | ./sample\_native-x64 shared-fb

然后，你就可以看到实时的界面效果了~~~~

至于编译，需要安装一个cmake工具，命令是: sudo apt-get install cmake

其他的就跟上面大同小异了。

最后，百闻不如一见，百见不如一试！老铁不妨试试，有问题，随时交流，谢谢!

# [超轻量/跨平台UI框架(目前已经支持: Windows, Android, Linux, VR)](http://blog.csdn.net/idea4good/article/details/77677992)

今天的大部分的软件，都被三大平台牢牢占据：PC上面的Windows， 嵌入式设备上的Linux，手机设备的Android。

如果分别为这三个平台开发软件，工作量和维护量可想而知。特别是界面部分，每次更换平台几乎以为着重写界面！

当然强大的QT也能实现跨平台，但问题是代价太大，这里说的代价不仅仅是cpu，内存的运行代价，更指的是学习代价，请大家试想一下，自己定制一个稍微特殊点的界面，是一个什么样的过程？程序完成后，cpu的使用率和内存的使用量是什么情况？

这里不是说这些UI框架不强大，恰恰相反，他们太强大了，乃至于98%的功能不是我们需要的，而我们的cpu，内存还必须为这98%的无效功能买单。

这里想跟大家分享一款超轻量的UI框架，代码在github上面，目前也还在逐步完善阶段，希望有更多的同学加入其中：

www.github.com/idea4good

运行效果可以参看这个视频：

[http://v.youku.com/v\_show/id\_XMjc2MjA4MDgzMg==.html](http://v.youku.com/v_show/id_XMjc2MjA4MDgzMg==.html?spm=a2h0k.8191407.0.0&from=s1.8-1-1.2)

核心完全使用C/C++编写，代码量仅仅数千行，文件约30个，平均每个文件的代码量约200行左右。以最小的代价实现了界面布局，文字，图片，常规控件，及界面滑动功能。我们认为UI代码不应该是主角，具体的应用才是重中之重，才更有资格获得更多的资源。

为了保证高效率，最大缩短了UI元素到操作底层显示设备的路径，力图用更短的代码，更浅的逻辑，来降低学习曲线。

简单介绍一下目前的代码库状况:

NoodleLinuxSample: 是Linux的封装方法，它可以适应各种尺寸的显示屏；而且还能在PC上面预览运行效果，嵌入式工程师再也不用编译一遍，烧写一遍了。。。

NoodleAndroidSample: 是Android的封装方法，你的手机也可以当开发板使用了！注意：在你滑动屏幕的时候，不要以为是Android在帮忙，那是UI框架独立完成的！

NoodleWinSample: 是Windows的封装方法，首先它可以是普通的Win32程序，也可以是UWP（universal windows platform）程序；UWP可以运行在MR（混合现实）/VR（虚拟现实）设备上，上面的视频有关于VR运行效果，有兴趣的同学可以看一下。

最后，希望有志之士加入我们的行列，qq群: 278694300

# 是我们技不如人？还是你故意炫技？

作为程序员，可能都要跟UI框架打交道，有位老兄就是死活不跟图形界面打交道，把他解释翻译成中文就是：“我不想把生命浪费在界面上”这哥们就是Linux，Git的作者。在下并不是反对图形界面，只是强调了一个问题：选择了图形界面，就意味选择了与细节打交道，是个脏活，累活。

于是，UI框架应运而生，什么MFC，WPF，ucgui，minigui，android，ios。功能上，可以说是应有尽有。但几乎所有的UI框架都是在强迫你接受他们的api，无理由的接受它的各种规则，但是缺拒绝向你透露一切细节！而且一些不务正业的UI框架（没错，我说的就是QT），还自己发明编译器，恨不得把我们牢牢的控制在它的怀抱中。。。

举个例子吧，你知道图形是如何渲染的吗？你知道多窗口是按什么顺序堆叠起来的吗？你知道按钮消息是如何传送的吗？讽刺吧？搞图形界面的你我从来没有真正跟图形打过交道，更多的是在纠结：怎么搭建编译环境，object如何继承，宏定义的格式， 咋有编译不过了………

所有这些图形问题，大牛肯定都知道；但问题是大牛花了多少精力？我们这些小白，怎么办？项目进度，还管不管？今晚，还要不要加班？倘若这些问题，我们研究的清楚，运用这些UI框架必定如鱼得水，否则，只能选择原谅UI框架，感叹自己技不如人~~~

我们能不能有另一个选择？比如，告诉我们一个简单的绘制原则，一个简单的控件管理办法，一个简单的贴图，贴文字的方法，一段2千行左右的核心及实例代码。然后我们各安天命，如果没有达到预期的界面效果，都是我自己的责任。

选择是有的。在github上面已经存在这个开源项目了：

[www.github.com/idea4good](http://www.github.com/idea4good)

基本介绍可以参看：

http://blog.csdn.net/idea4good/article/details/77677992

这是一个超轻量的UI框架，由于太清凉，所以很容易就实现了跨平台，目前支持windows（包括VR, MR平台）, linux, android; 大家在代码里面可以看到详细的实现方法；当然因为是开源项目，你也可以把自己的代码往上推。

最后，由于PC太难用，于是有了智能手机；由于UI框架太晦涩，会不会诞生一种更简易的框架呢？

请大家一起交流，学习（QQ群: 527251257）

# [扪心自问，强大的UI框架，给我们带来了什么？](http://blog.csdn.net/idea4good/article/details/77689050)

MFC, QT, WPF, UcGUI, MINIGUI, [**Android**](http://lib.csdn.net/base/15), [**iOS**](http://lib.csdn.net/base/1)，如果这些平台，你都使用过，在下真心佩服，也相信你对这篇文章最具有发言权，真心期待你的回复和建议。

这些著名的UI说出来都让人如雷贯耳，如果没用过几个，都不好意思打招呼~~

但不夸张的说，很多用过这些框架的同学，已经郁闷的不会打招呼了。在下可能水平不较低，但更不想穿这件隐形的裤衩，也人云亦云的说这框架多么牛逼，多么强大~因为，跟所有苦逼的程序员，我们选择这些框架的初心是：简化工作，为自己的应用程序加一个简单的界面就好。

但现实呢？UI框架的代码动辄上百万行~~NND，比自己的应用程序大几个数量级；编译麻烦，消耗时间不说，光加个简单的按钮，就要倒腾半天，还得从基本类学起；如果你需要的控件，没有现成的（不客气的说，UI框架中自带的控件，没有一个是可以直接拿来用的），那你算是遇到挑战了，慢慢玩吧~~

好吧，因为你足够的聪明，以上的问题都难不倒你，你三下五除二都搞定了，现在可以运行了。看看你的cpu使用率和内存使用量吧，是不是很酸爽？NND，明明只请一个人吃饭，他居然把所有的亲戚朋友都叫上了~~~~ 强大的UI框架就是这么任性，你想用我一个控件，你就得出一个全家桶的钱。

在下的水平很低，所以也就不怕再错一次。在下以为：强大的UI框架从来没有解决过你的问题！不信你看，测试组又报UI的bug了~~~

说了这么多，在下绝没有想辱没这些著名的UI框架；事实相反，在下经常被它们辱没。我想表达的意思是，他们很强，但确实太大了。在寸土寸金的嵌入式平台上，这是抢劫；再富有的硬件，这经不起这样拔毛。我喜欢它们的鸡翅，但绝对买不起，吃不下它们的全家桶。

我们不就是想画几个按钮，然后想做个跟手机类似的滑屏界面嘛！为啥不能直接一点？非拿一堆class，graphic，driver来唬人？今天不忍了，自己干，有兴趣的同学，一起来《超轻量UI框架》。

代码

http://blog.csdn.net/idea4good/article/details/77677992

www.github.com/idea4good

qq群: 527251257