

Qt patch 必合修改点

zorowk

<2020-08-25 Tue>

在代码中寻找修改点

- 通过此方法寻找的大概还有 30 多个修改点。
- 目前的技巧就是各个贡献者存在约定俗成的 Todo 修改标记点，是一些需要改进的修改地方，但是修改后可能会涉及到 test, example 工程的联动修改需要便编译来避免错误。
- 找到修改点的方法就是在 qtbase 工程中搜索 // ### Qt6: 就是 qt5 向 qt6 升级的需要升级的修改点，一般之前的贡献者会在 commit 信息或者注释里面留下为什么需要修改的解释。
- 观察qt/base 上关于其他贡献者的提交是一个好主意，上次的提交也是先看了其他提交者的提交知道了那些地方是值得修改的。
- 其实在上次的浏览历史提交记录中已经知道了 Qt5 正在做大版本的升级到 Qt6. 所以上次其实有查找 Qt5 到 Qt6 由那些特性会升级从而有贡献代码的机会。
- 这次的浏览提交历史中有一个提交引起了我的注意,QAbstractSocket: Resolve remaining Qt6 TODOs 提交内容就是 qt5 到 qt6 需要合入修改的地方
- 这里的 Qt 6 Todo 的意思，有对代码比较熟悉的贡献者觉得这个地方需要改进就在这里加入了标记，方便自己也方便其他贡献者去继续完成代码的修改一般这种修改就是重构或者代码局部类或者函数的改进，可能写 Todo 的作者比较繁忙，应该是代码量太大了，打个标记以后更改。
- 当有很多人查找// ### Qt6: 修改点的时候可能需要注意的是避免有多人同时修改一个点。

如何使用 qtcreator 打开 qtbase

- [kde build openource qtbase](#)
- [how build the qt project itself using qtcreator](#)
- 按顺序执行一下命令：
 1. Run the configure script (and add the -developer-build option)
`./configure -prefix $PWD/qtbase -opensource -confirm-license -nomake tests -nomake examples -dbus -no-separate-debug-info -developer-build`
 2. Open QtCreator and go to "Tools > Options > Kits"
 3. Go to "Qt Versions" add press "Add" - select the qmake executable generated by the configure script. Then hit "Apply"
 4. Go to "Kits" and press "Add" - Make shure to select the correct compilers and debugger and select the previously create "Qt Version". Press "Ok"
 5. Open the top level .pro file in QtCreator and choose the previously created Kit. QtCreator will now use the correct qmake executable

遇到的坑

- 修改 `// ### Qt6:` 这样的修改点有可能会影响到 qtbase 里面的 test, example 工程所以一定需要找到自己修改点引用的工程都修改全了
- 之前提交过一个 patch 编译后发现, example 的例子编译不过去了
- 一次提交一个修改点有助于合入。