### **平台软件界面学习**

##### **1.项目生成顺序**

BIMGraphicsUIComponent （UI组件模块）

↓

BIMGraphicsUI（UI模块）

↓

BIMGraphicsUICommon（通用模块，应用程序实例）

↓

BIMGraphicsCommand（命令模块）

↓

BIMGraphicsREModeling（AI模块）

↓

BIMGraphicsMainWindow(主窗口)

##### **２.依赖项**

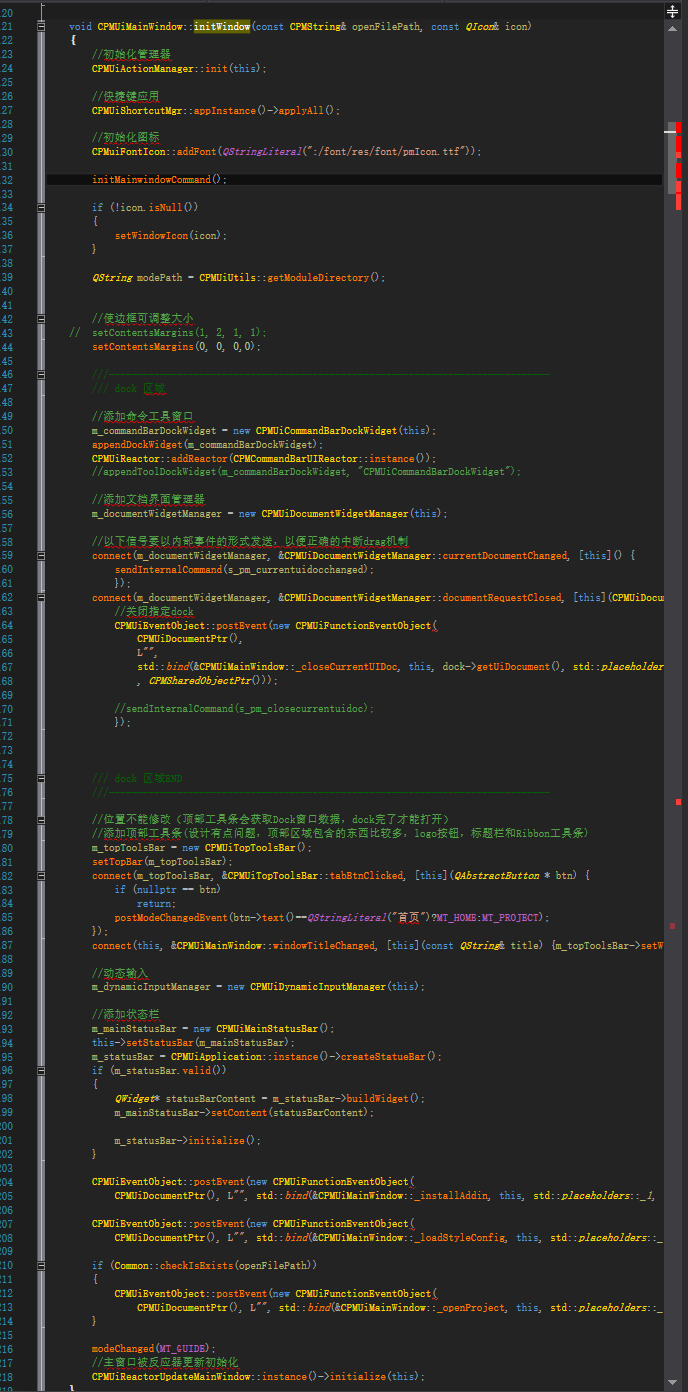
当一个项目使用另一个项目生成的可执行代码时，生成代码的项目被称为使用该代码的项目的项目依赖项。只要某个项目依赖项尚未生成，或自最近生成以来已被修改，则应首先生成该项目，以便生成和更新要使用的代码。

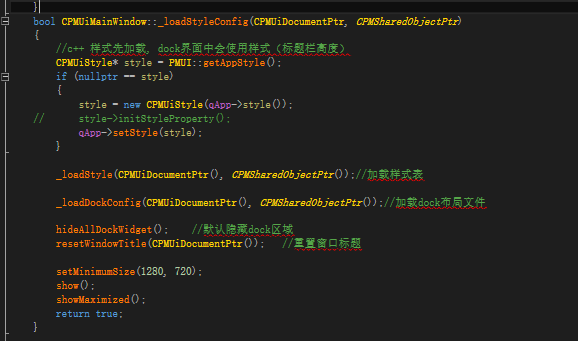
例如，如果项目 B 需要由其项目依赖项项目 A 生成的可执行代码，并且自上次生成以来已对项目 A 进行了修改，则在生成项目 B 之前应重新生成项目 A。这使得可以先对项目 A 提供的可执行代码进行修改，然后再让项目 B 使用它。如果自上次生成以来未对项目 A 进行修改，则不需要重新生成新的项目 A。仅生成项目 B。

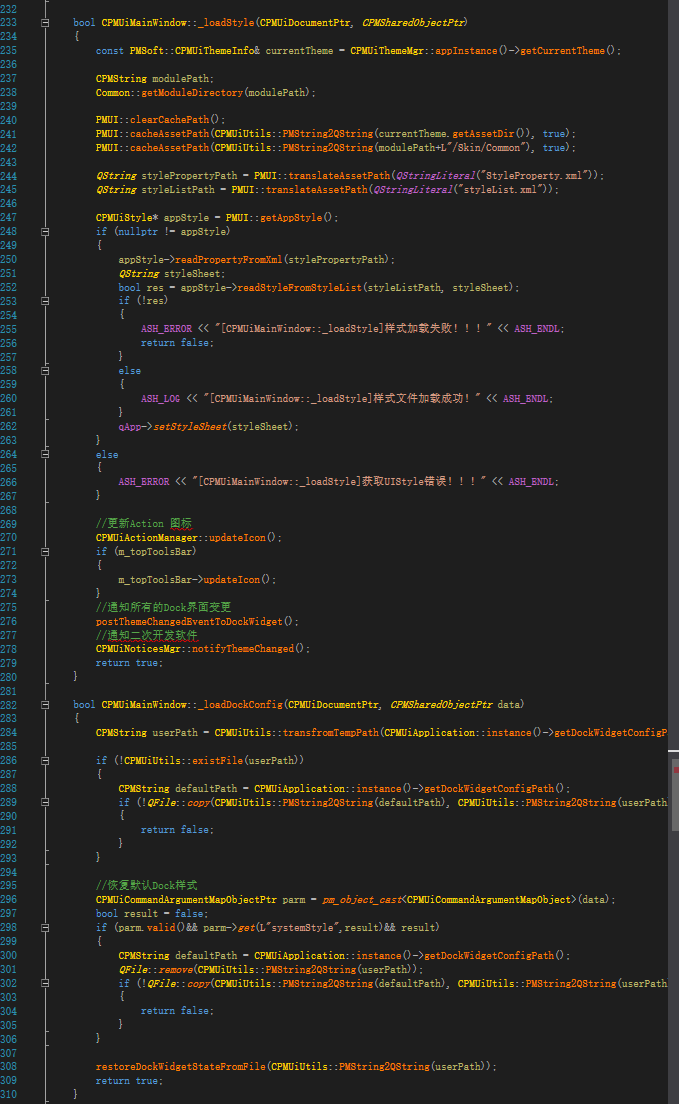
##### **３.样式是怎么加载到界面的？**

主界面初始化:

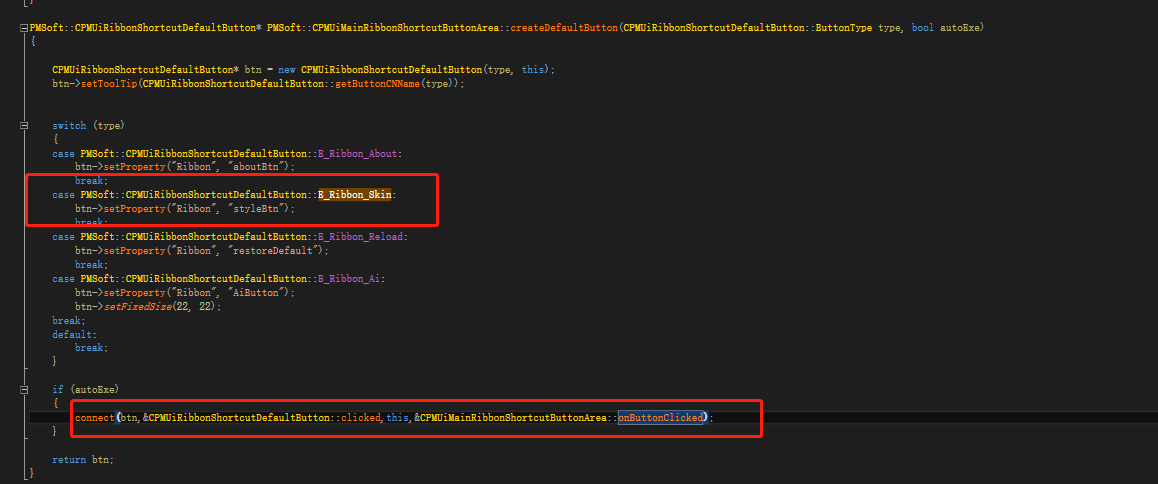
界面组装

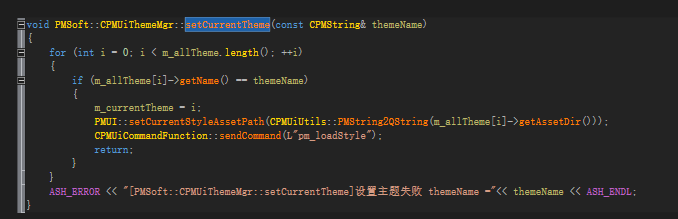
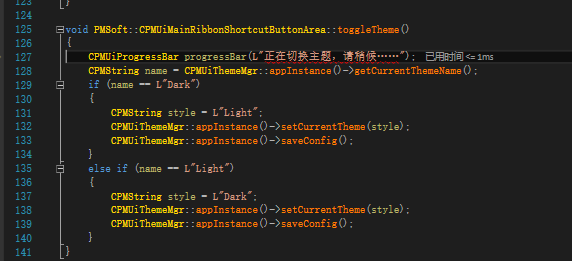
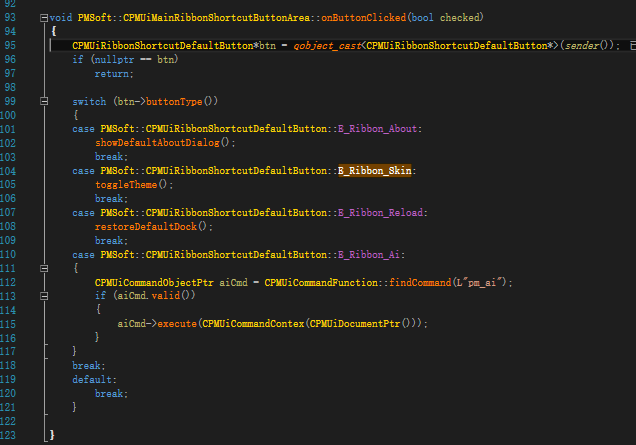




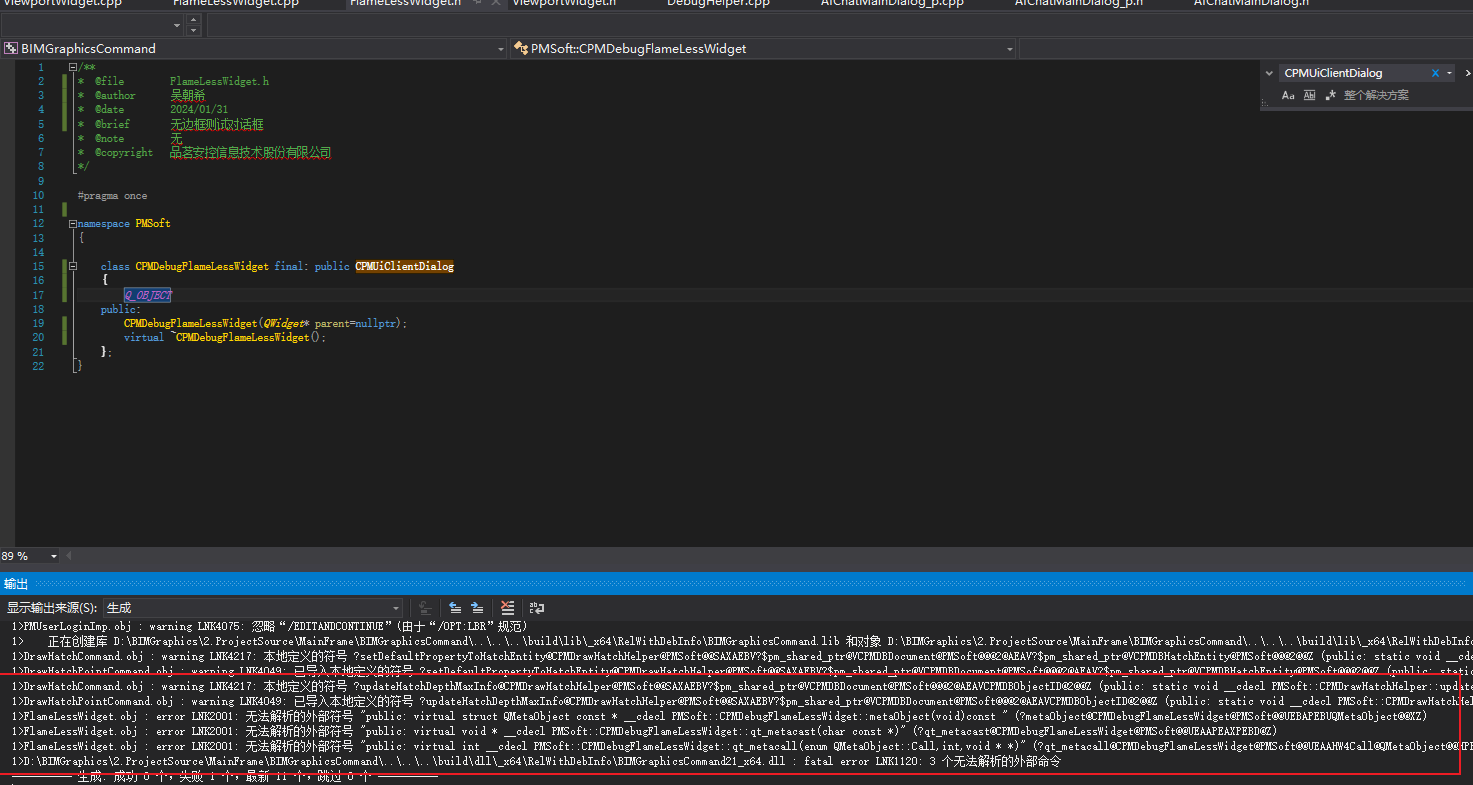


##### **4.样式是怎么切换的？**



设置为当前的主题，并且获取当前主题路径。

2024.1.31遇到的编译问题



加了Q\_OBJECT编译不过，去掉之后就可以编译了。

错误原因：编译器错误，应该改成moc(元对象编译器，这是因为信号和槽机制是QT自行定义的一种通信机制，它独立于标准的 C++ 语言，是标准C++的扩展。因此要正确的处理信号和槽，必须借助一个称为 moc（Meta Object Compiler）的QT工具，一般叫做元对象编辑器，该工具是一个C++ 预处理程序，它为高层次的事件处理自动生成所需要的附加代码。)

解决方案：原来是C/C++标头