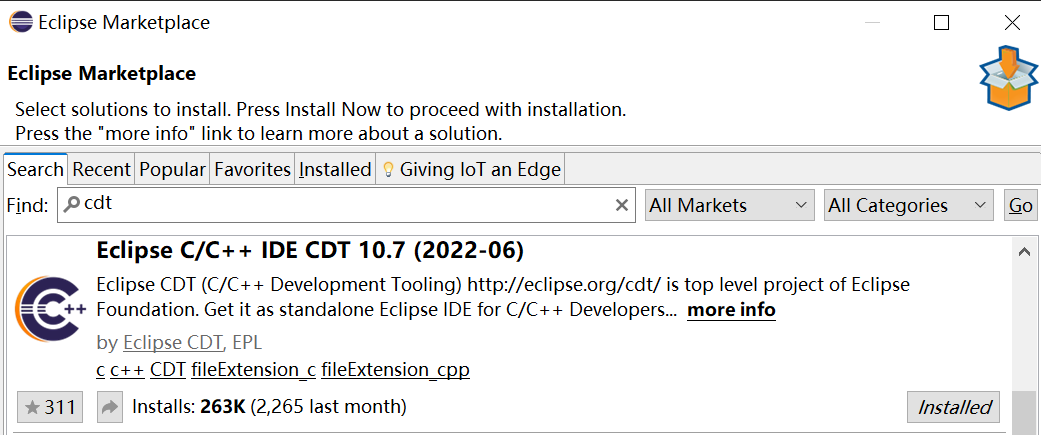
# 在eclipse上使用cdt-clangd插件

cdt-clangd依赖于lsp4e和lsp4j，lsp4e和lsp4j跟随cdt打包发布，因此直接安装cdt即可。值得注意的是lsp4j是与eclipse平台无关的。

由于cdt-clangd插件并未打包发布，因此此处以源码方式运行cdt-clangd插件。

1. 在eclipse上安装cdt



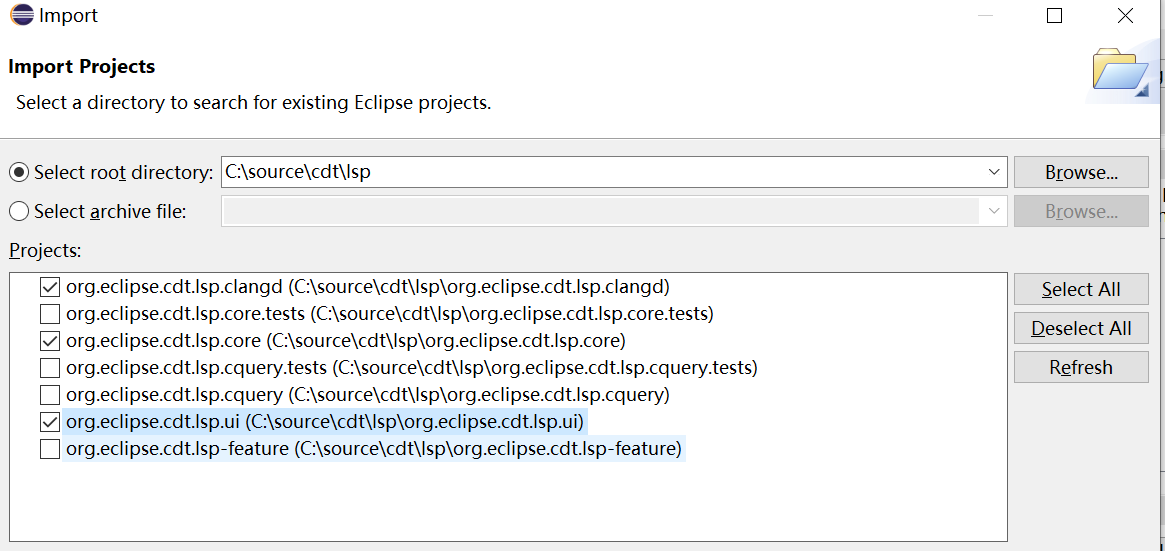
1. 下载clangd源码

>git clone <https://github.com/eclipse-cdt/cdt.git>

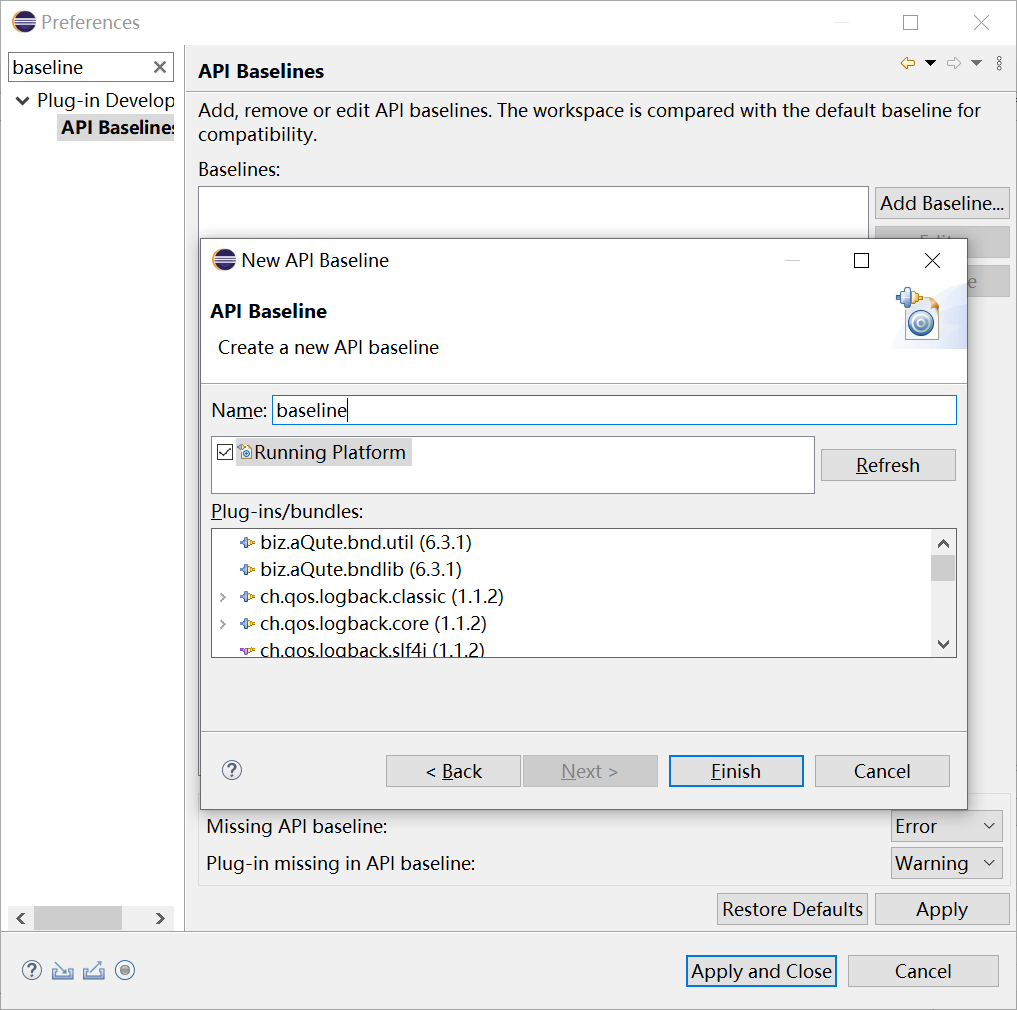
>cd cdt/lsp

cdt/lsp文件夹下的org.eclipse.cdt.lsp.clangd org.eclipse.cdt.lsp.core org.eclipse.cdt.lsp.ui三个插件工程是我们所需要的。

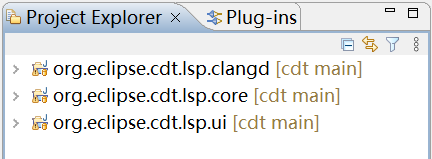
1. 导入上述三个工程



1. 配置api baseline，解决编译错误

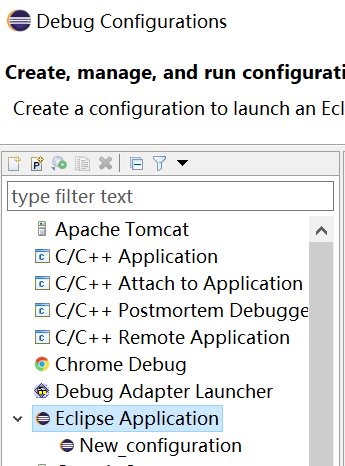


1. 导入工程，解决编译错误之后的工程结构如下

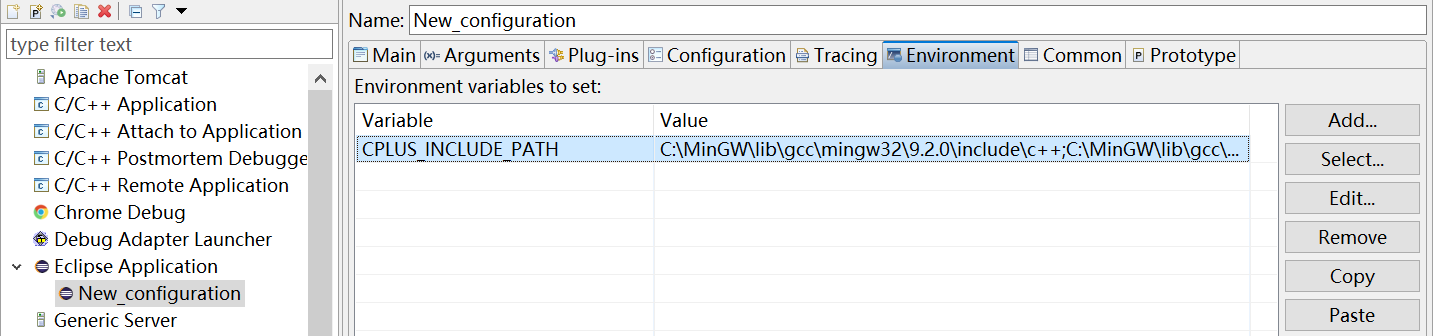


1. 以源码方式运行以上三个插件工程

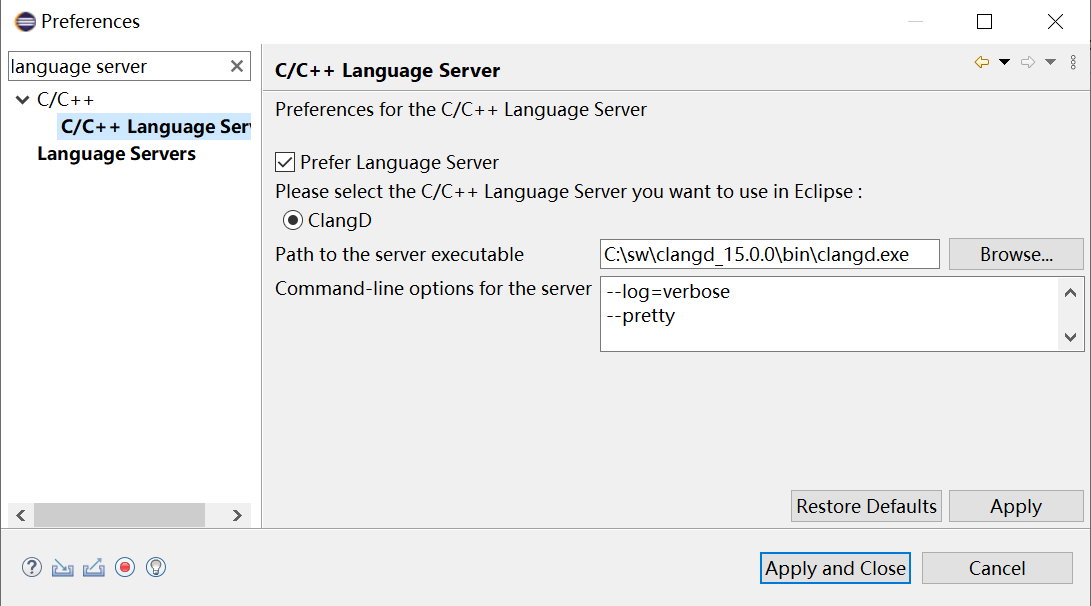
创建一个Eclipse Application的Launch Configuration



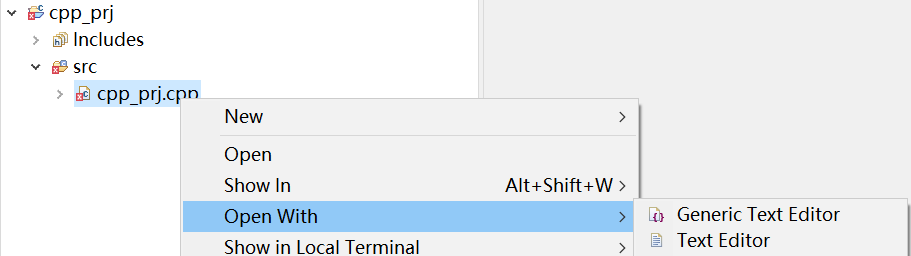
配置该Launch Configuration的环境变量，增加CPLUS\_INCLUDE\_PATH变量，变量值为c++的头文件搜索路径，多个路径以分号分隔，clangd需要从环境变量中搜索头文件。



1. 运行此Launch Configuration
2. 在弹出的界面配置clangd，指定clangd可执行程序的路径



1. 创建c++工程，以generic text editor方式打开一个cpp文件



1. 在编辑器使用clangd进行代码提示

打开之后可以看到iostream有报错，这可能是因为mingw的版本问题，这会导致std相关的代码提示出现异常，但不影响其他类库。因此此处我们引入vector头文件和string头文件。如下图所示，vector类的代码提示是正常可用的，其它代码也没有出现解析错误，代码高亮也是正常的。

