

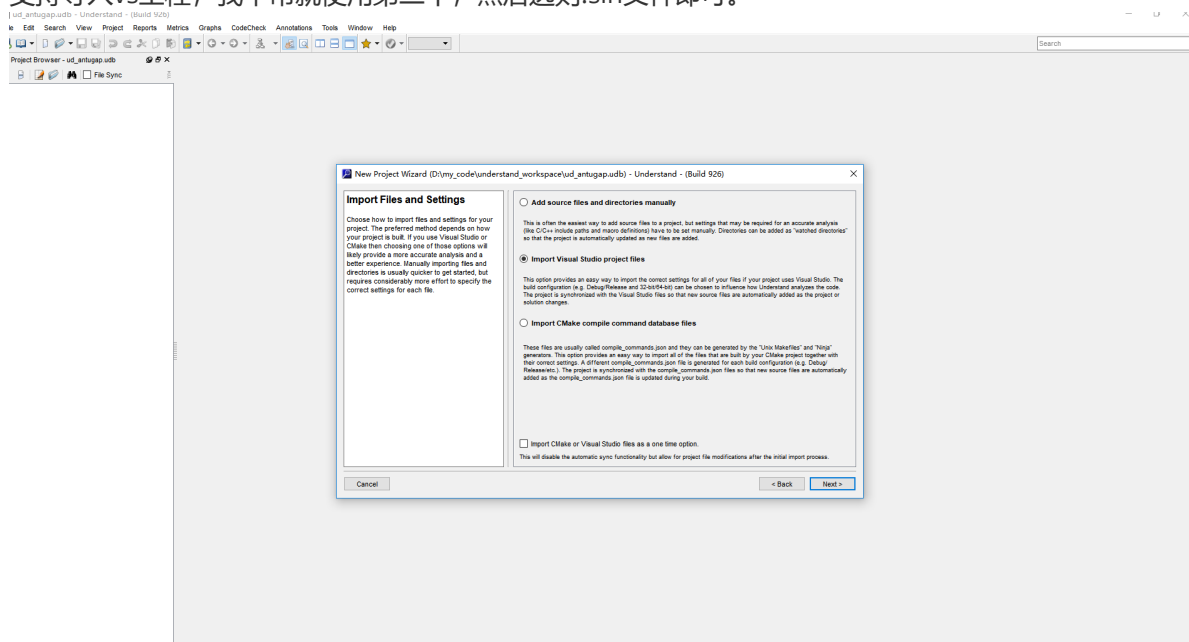
1 背景

相对于大家熟知的source insight 源码阅读工具，understand工具似乎差了不少。但是，两者的优缺点很显而易见， source insight 优缺点(捡重点说)：

- 很好的“即点即现”功能，鼠标所到就可以看到相关内容的定义/声明；
 - 不能够很好的维护/支持 函数的调用层级关系；
- 相对的 understand源码阅读工具，不支持“即点即现”功能，但却很漂亮的维护/支持了函数的层级调用关系。如果你有这方面的需求，建议尝试用用understand工具，它能够更好地为你理清清楚函数的调用层级关系。以下，understand的功能解释/使用。

2 导入项目

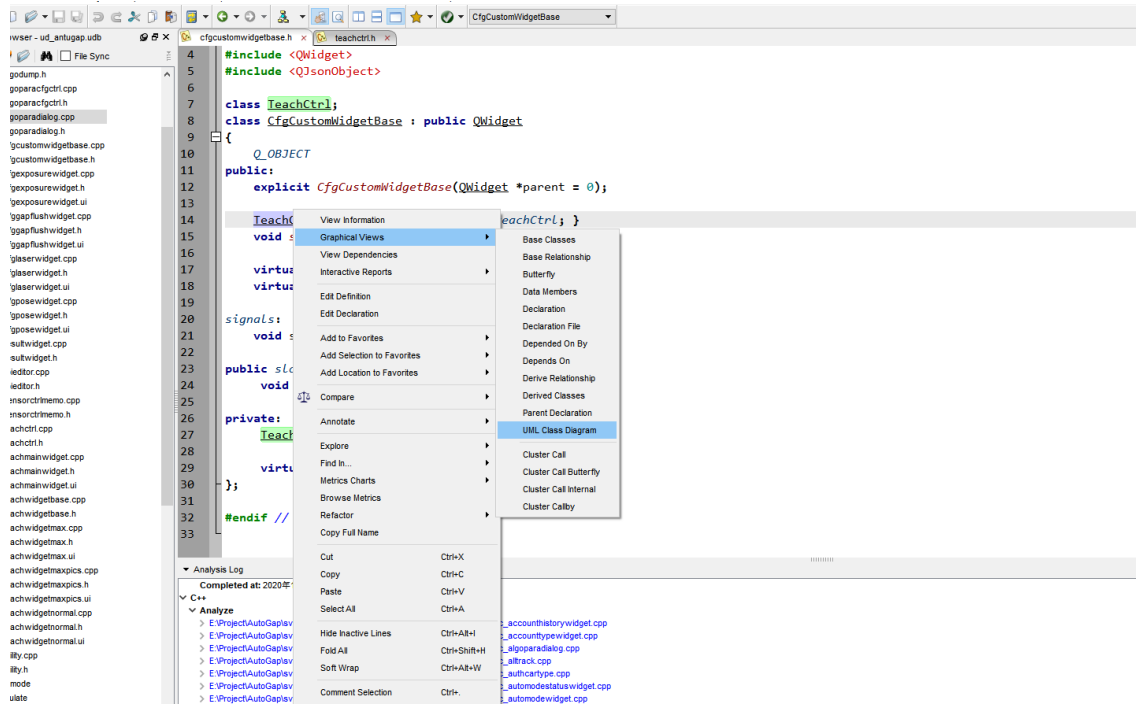
支持导入vs工程，我平常就使用第二个，然后选则.sln文件即可。



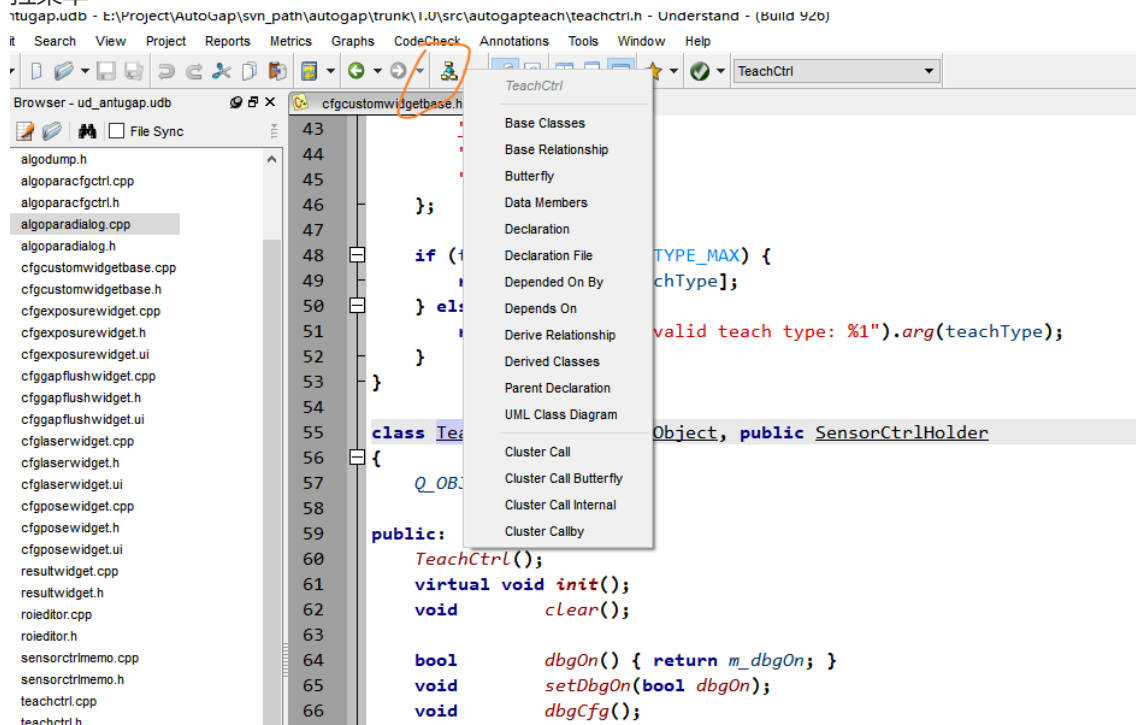
3 项目视图

项目视图包含很多的功能，能够自动生成各种流程图结构图，帮助你快速理清代码逻辑、结构等，以便快速理解项目流程

- 鼠标放到类或函数上，然后右键弹出菜单，鼠标移动到Graphical Views，然后弹出二级菜单，如下图所示：



- 或者也可以这样，点击要查看的类或者函数，然后找到代码上面菜单栏中的如下图标，点击显示下拉菜单

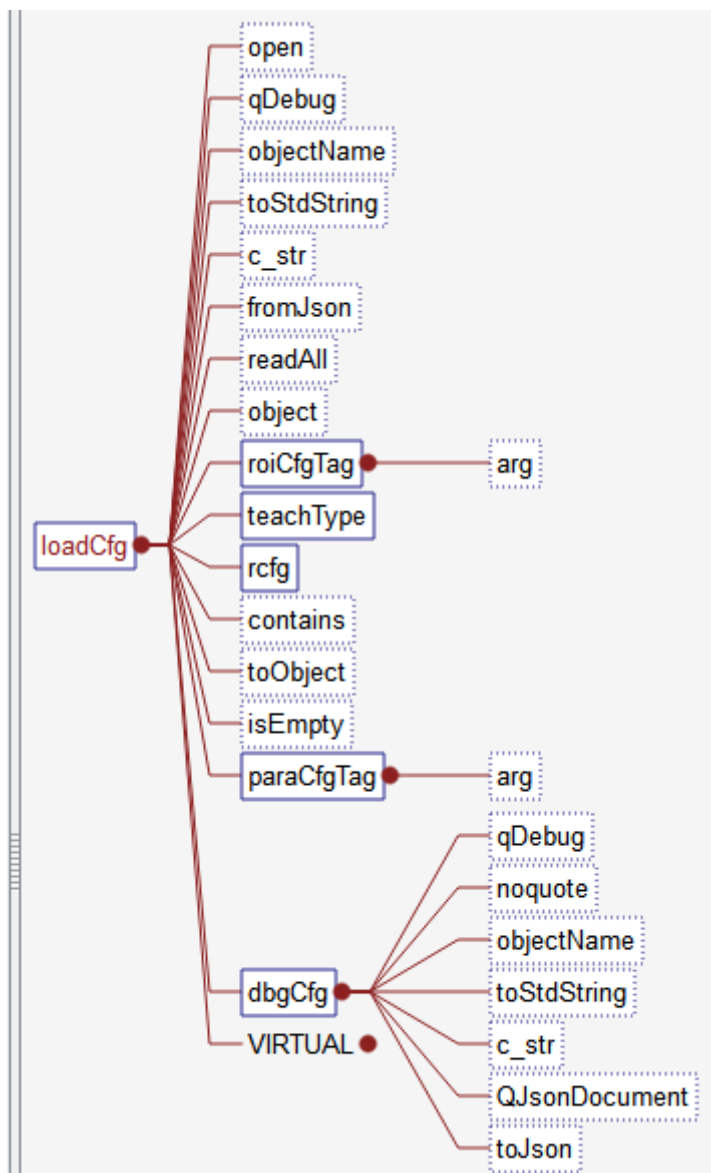


4 层级关系视图分类

1. Butterfly

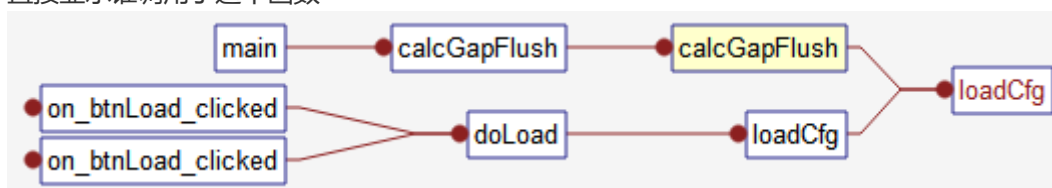
如果两个实体间存在关系，就显示这两个实体间的调用和被调用关系;

显示从此函数发出的整个呼叫链。实体之间的每条线被读作“x调用y”。



3. Called By

直接显示谁调用了这个函数



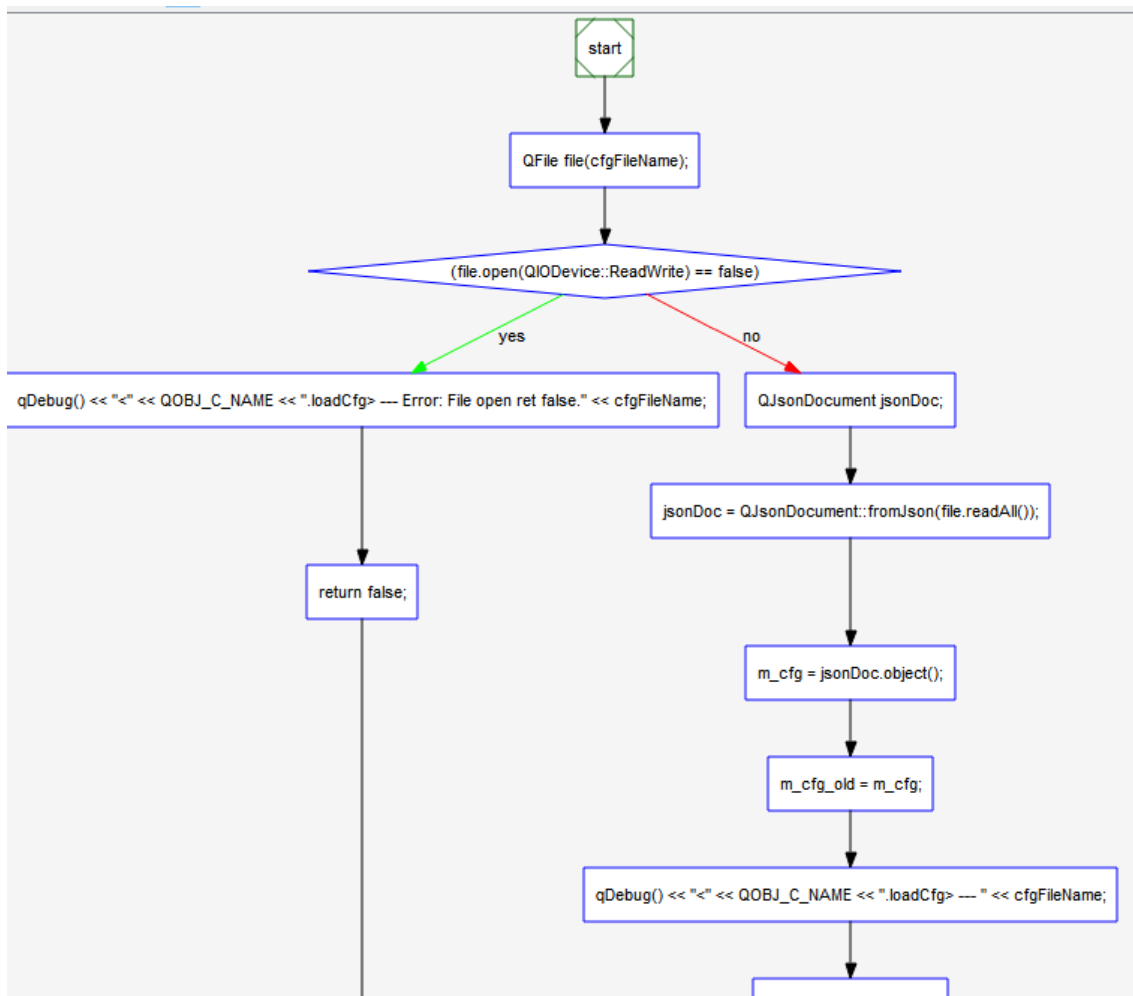
4. Calls Relationship / Calledby Relationship

展示了两个实体之间的调用和被调用关系，操作方法：首先右键你要选择的第一个实体，然后点击另一个你要选择的实体，如果选择错误，可以再次点击其他正确即可，然后点击ok；

5 结构关系视图

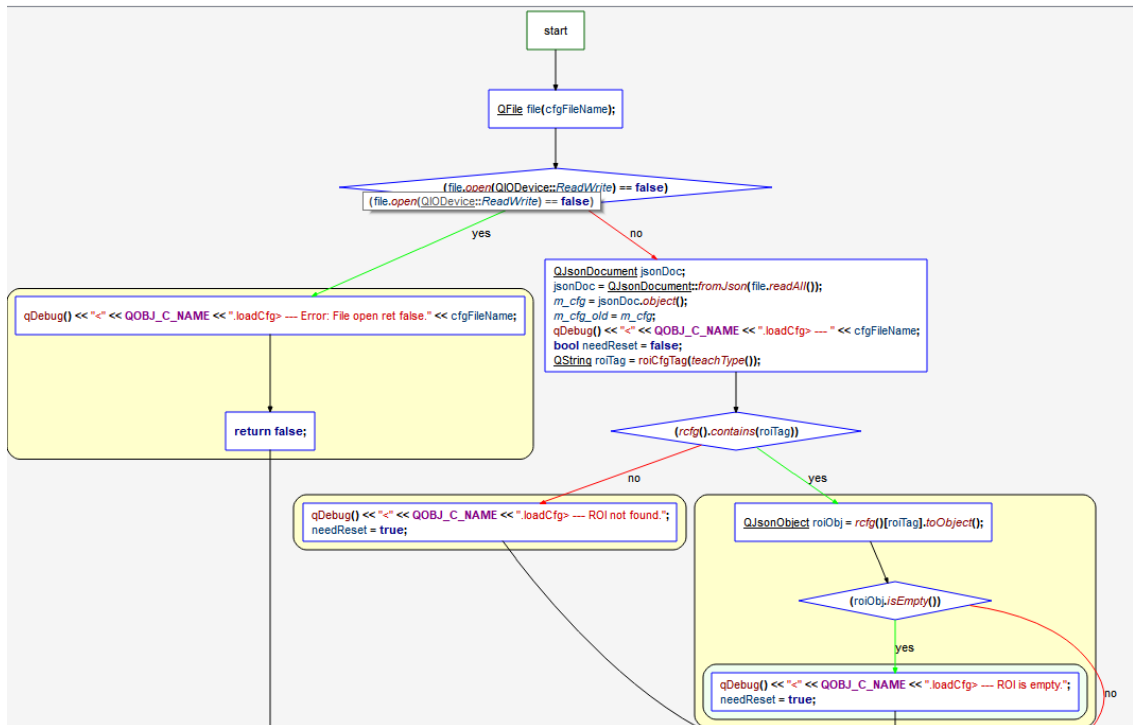
1. Control Flow

函数实体流程图，不完美但清晰。



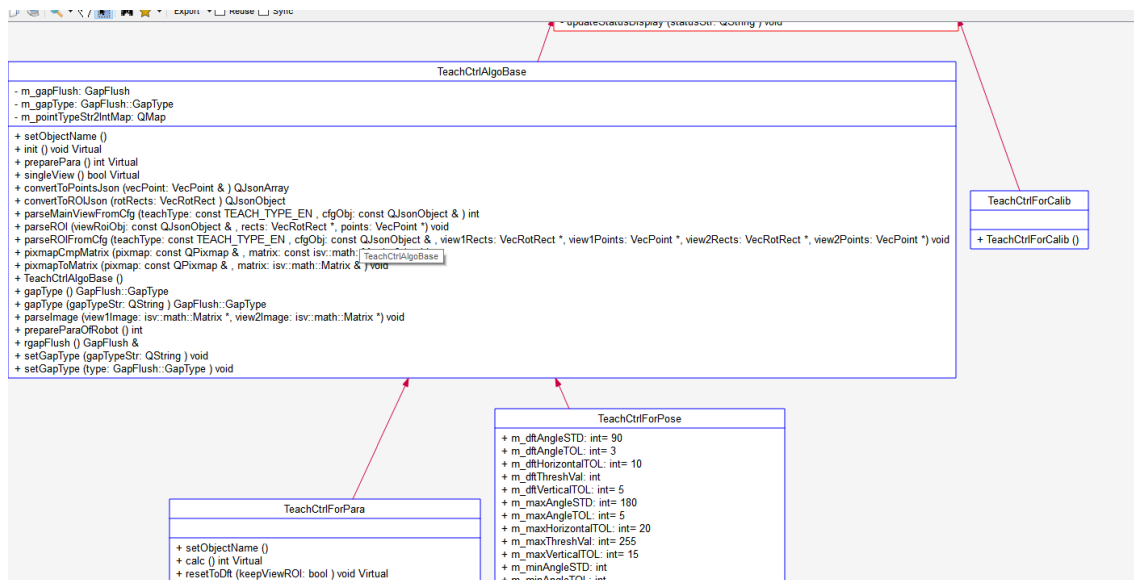
2. Cluster Control Flow

展示一个实体的流程图或者类似实体类型，这个比上一个更具有交互性；



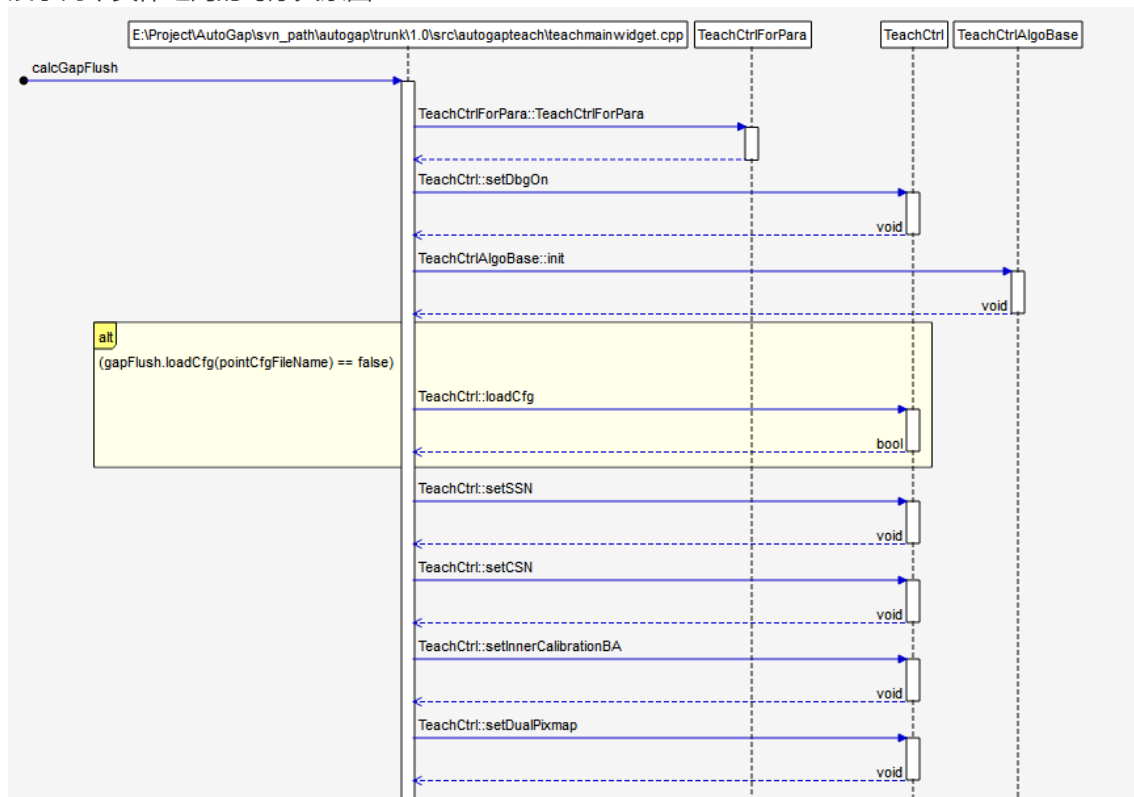
3. UML Class Diagram

展示这个项目中或者一个文件中定义的类以及与此类关联的类



4. UML Sequence Diagram

展示两个实体之间的时序关系图



更多功能等待大家探索