1. 可能存在的问题：
2. 关于注释，还是需要进行预处理。
3. 关于typedef声明，类的指针形式引起的调用。
4. 关于多层调用，如果是只有一个函数，其余为变量，就可以都找出来；如果包含有多层函数调用则只能找出离变量最近的那层调用关系。
5. 对模版类的处理。如某个类成员函数的定义如下：

void classA<T>::method();

1. 关于include问题。可能出现相互包含或是循环包含的情况，只要加上ifdef等语句就不会报错，但是对解析会有影响。
2. 已解析的产生关系的情况：
3. 类A和类B存在函数调用关系
4. 调用者的变量可出现在 类成员变量、函数参数以及函数局部变量。【全局变量需加入考虑】

三． 对应OO那篇论文所提到的所有耦合，有以下列表：

1. 数据抽象耦合(DAC)
2. 类B是类A的一个属性，即类B作为类A的成员函数出现。【Done】
3. 类B是类A中方法的一个元素。即类B的对象声明出现在类A的某个成员函数的局部变量声明中。【Done】
4. 类B是类A某个成员函数的返回类型。【待考虑】
5. 消息传递耦合(MPC)

1） 类B的成员函数被类A的成员函数调用。【Done】

2） 类A和类B共享数据（如全局变量，或是其类C的成员变量）【待考虑】

1. 类与方法的交互(CM)

类A的某个成员函数以类B类型作为参数或是返回类型。【参数的已Done返回类型待考虑】

1. 继承耦合。

类A作为类B的父类出现。【，已找出父类，关系待考虑】

2013/8/7

讨论调用关系如何处理以及权重等问题列表：

1. 再一次确定某个类被其他类调用时，该类所在顶点与其他类所表示的顶点组成一条超边。
2. 在解析类间关系时，保存了每一个类调用其他类的次数。比如有A,B,C,D四个类，类A分别被B,C,D三个类分别调用了2,3,1次，那么这条超边的权重如何确定？是取最大值还是平均值。还有一个想法是，如果B,C,D调用了A的次数差别比较大，那么选出调用次数大于某个阈值的点重新作为一条超边，原始超边还是不变。
3. 对于顶点的权重，我们考虑了某个类的成员变量。如果类A中出现的成员变量类型为其他类类型，则类A所代表的顶点的权重即为这些类的个数之和。如类A中出现类B,C,D三个类类型的成员变量，则顶点A的权重为3.