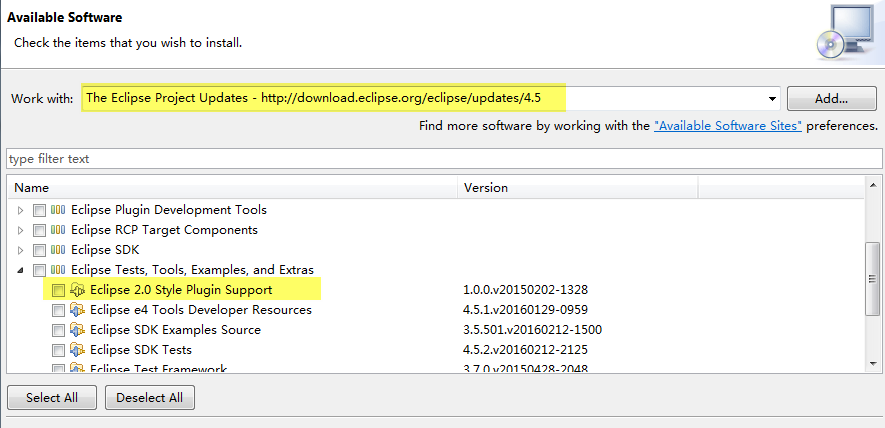
**eclipse笔记**

# luna和mars中安装fatjar失败的解决办法

可能由于fatjar年久失修,在高版本的eclipse中安装fatjar插件时会出现错误,解决方法是,在eclipse中安装对eclipse2.0版本插件支持的插件.方法如下:

Help-->Install New Software-->Work with :

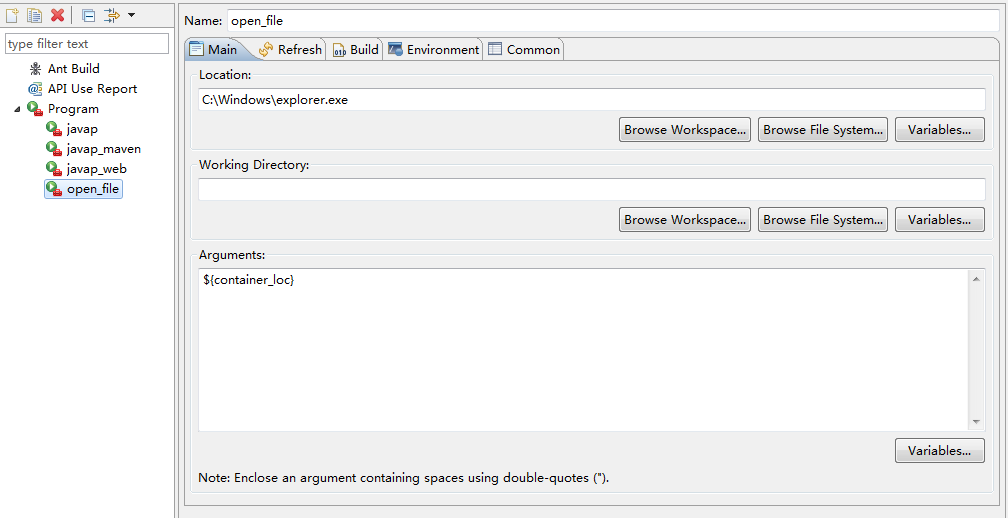
The Eclipse Project Updates - http://download.eclipse.org/eclipse/updates/4.5



安装成功后再安装fatjar就不会报错了

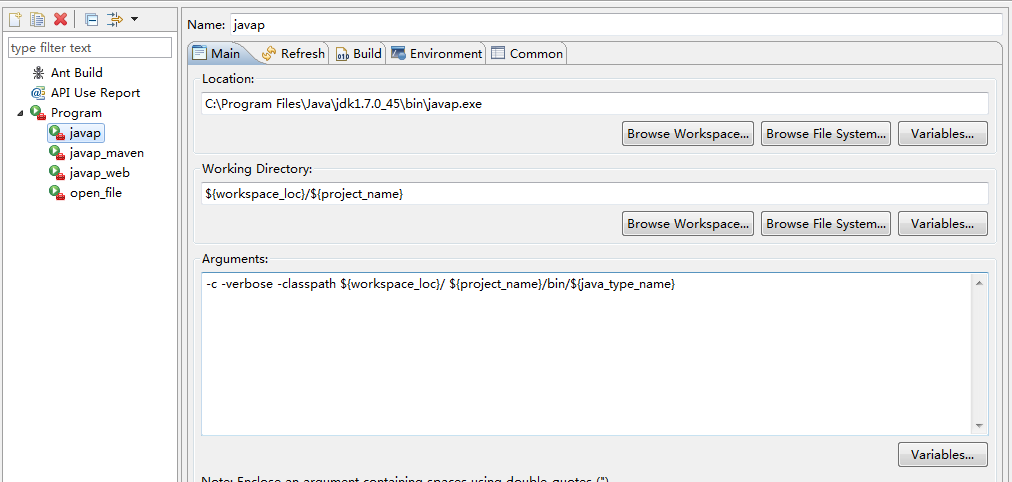
# 快速定位文件位置

在run->External Tools中设置



# 在eclipse中使用javap

在run->External Tools中设置



# eclipse.ini

eclipse的启动由%ECLIPSE\_HOME%/eclipse.ini控制，如果$ECLIPSE\_HOME 没有被定义，则eclipse安装目录下的默认eclipse.ini会生效。eclipse.ini是一个文本文件，其内容相当于在eclipse运行时添加到eclipse.exe之后的命令行参数。

格式：

1：所有的选项及其相关的参数必须在单独的一行之内，若参数形如“项 值”形式，中间的空格需要换行书写,如果值中有空格则需要用双引号括起来;

2：所有在-vmargs之后的参数将会被传输给JVM，所有如果所有对Eclipse 设置的参数必须写在-vmargs之前.

-nosplash

描述：指定启动时不显示闪屏

举例：-nosplash

-vm [jre path]

描述：指定启动时所使用的Java虚拟机

举 例：例如要使用自己的Java虚拟机，则-vm D:/j2sdk1.4.2\_04/jre/bin/java.exe，这样还有一个好处，就是可以开启一个Console，能够显示控制台信息，当然若 使用-vm D:/j2sdk1.4.2\_04/jre/bin/javaw.exe则不会再显示控制台

-vmargs [Java VM arguments]

描述：指定启动时要使用的Java虚拟机参数

注：此参数一定要放在所有参数变量的最后面

如 -vmargs

-Xms40m

-Xmx256m

-XX:PermSize=64M

-XX:MaxPermSize=128M

# Memory Analyzer

## concept(概念)

### Heap Dump

A heap dump is a snapshot of the memory of a Java process at a certain point of time. There are different formats for persisting this data, and depending on the format it may contain different pieces of information, but in general the snapshot contains information about the java objects and classes in the heap at the moment the snapshot was triggered. Usually a full GC is triggered before the heap dump is written so it contains information about the remaining objects.

当做dump操作时会先进行一次Full GC

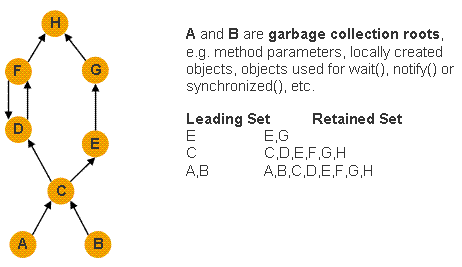
### Shallow vs. Retained Heap

Shallow Heap是只包含这个对象自身,而不包含所引用的对象

Retained Heap包含对象自身和该对象所引用的对象所占用的Heap大小

**Retained set** of X is the set of objects which would be removed by GC when X is garbage collected.

**Retained heap** of X is the sum of shallow sizes of all objects in the retained set of X, i.e. memory kept alive by X.



上图中当E不被引用后G将会被回收,但H不会被回收,所以E的Retained Set包含E和G而不包含H

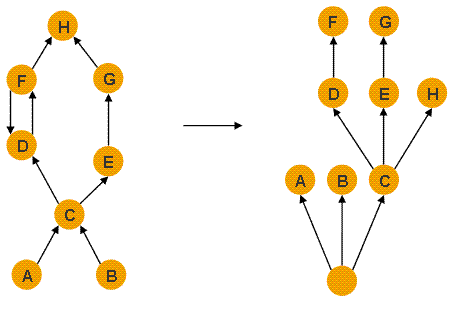
### Dominator Tree(支配树)

An object x **dominates** an object y if every path in the object graph from the start (or the root) node to y must go through x.

如果所有直接或间接引用Y的对象都通过X来引用,那么X是Y的支配者

The **immediate dominator** x of some object y is the dominator closest to the object y.

引用关系图-->Dominator Tree的转换如下图:



### Garbage Collection(GC) Roots

A garbage collection root is an object that is accessible from outside the heap.

The root set is all the data that is directly accessible to the program.This is the contents of the stack(s) , global/static variables.

当GC Root消亡时,只被该GC Root直接或间接引用的堆对象将会被垃圾回收掉

## task(功能)

### Leak Suspect Report(泄露怀疑报告)

打开该功能的方式:

