　　在windows安装Android的开发环境不简单也说不上算复杂，本文写给第一次想在自己Windows上建立Android开发环境投入Android浪潮的朋友们，为了确保大家能顺利完成开发环境的搭建，文章写的尽量详细，希望对准备进入Android开发的朋友有帮助。

　　本教程将分为五个步骤来完成Android开发环境的部署。

[第一步：安装JDK。](http://www.admin10000.com/document/5334.html#1)

[第二步：配置Windows上JDK的变量环境 。](http://www.admin10000.com/document/5334.html#2)

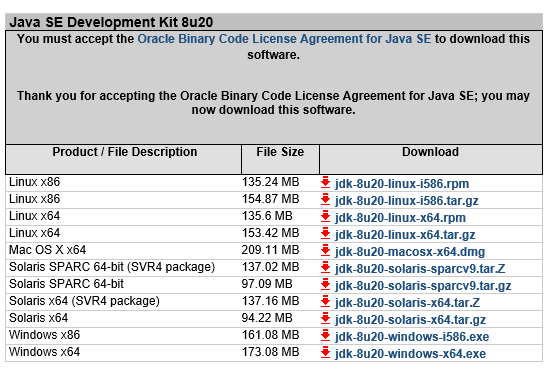
[第三步： 下载安装Eclipse 。](http://www.admin10000.com/document/5334.html#3)

[第四步：下载安装Android SDK 。](http://www.admin10000.com/document/5334.html#4)

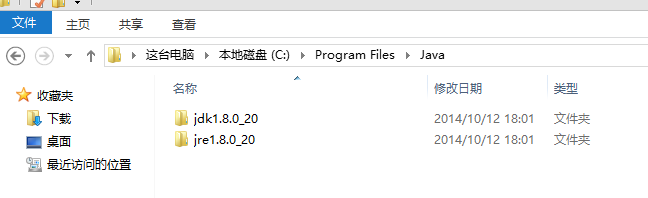
[第五步：为Eclipse安装ADT插件。](http://www.admin10000.com/document/5334.html#5)

　　**第一步：安装JDK**

　　要下载Oracle公司的JDK可以百度“JDK”进入Oracle公司的JDK下载页面（当前下载页面地址为<http://www.oracle.com/technetwork/java/javase/downloads/index.html>），选择自己电脑系统的对应版本即可。

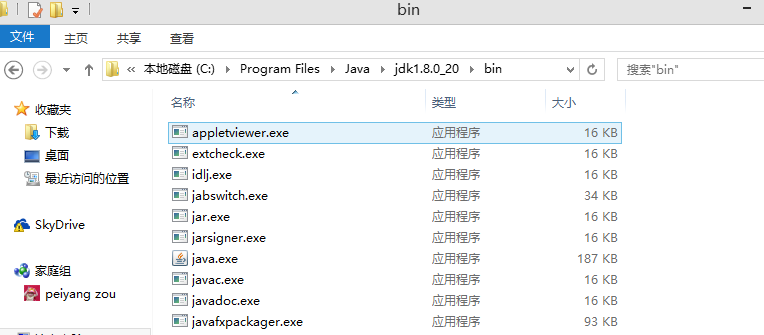


　　下载到本地电脑后双击进行安装。JDK默认安装成功后，会在系统目录下出现两个文件夹，一个代表jdk，一个代表jre。



　　JDK的全称是Java SE Development Kit，也就是Java 开发工具箱。SE表示标准版。JDK是Java的核心，包含了Java的运行环境（Java Runtime Environment），一堆Java工具和给开发者开发应用程序时调用的Java类库。

　　我们可以打开jdk的安装目录下的Bin目录，里面有许多后缀名为exe的可执行程序，这些都是JDK包含的工具。通过第二步讲到的配置JDK的变量环境，我们可以方便地调用这些工具及它们的命令。



　　JDK包含的基本工具主要有：

　　javac：Java编译器，将源代码转成字节码。

　　jar：打包工具，将相关的类文件打包成一个文件。

　　javadoc：文档生成器，从源码注释中提取文档。

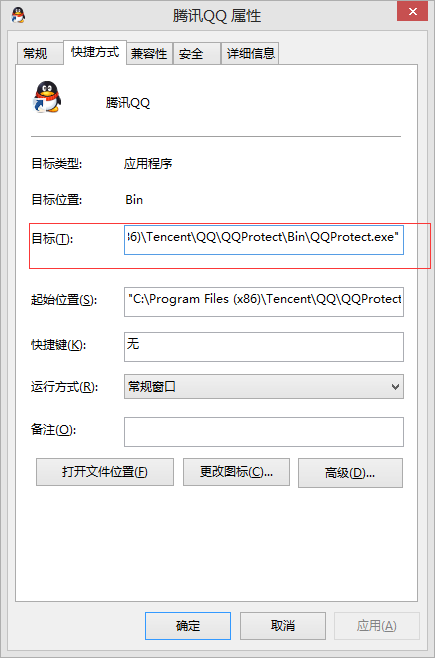
　　jdb：debugger，调试查错工具。

　　java：运行编译后的java程序。

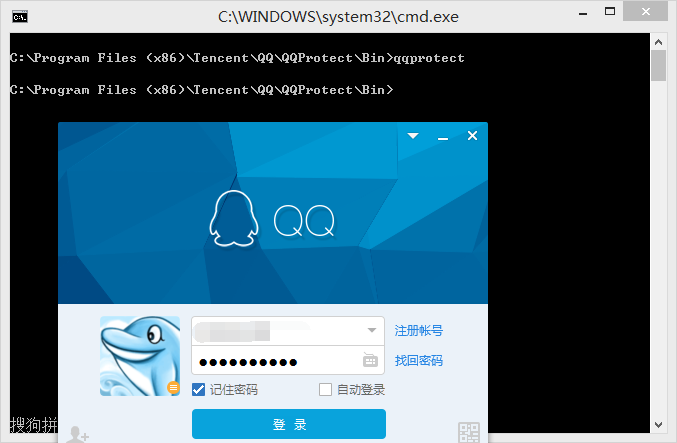
　　**第二步：配置Windows上JDK的变量环境**

　　很多刚学java开发的人按照网上的教程可以很轻松配置好Windows上JDK的变量环境，但是为什么要这么配置并没有多想。

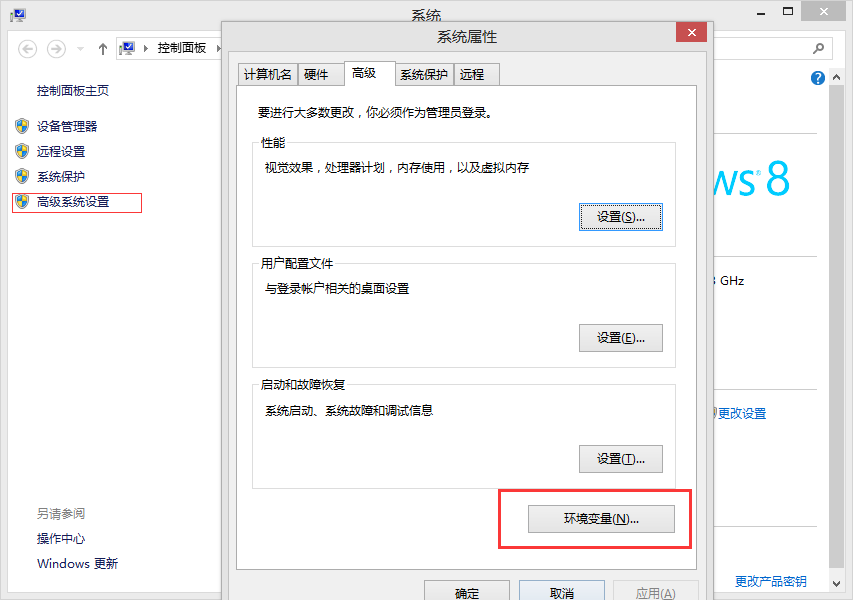
　　我们平时打开一个应用程序，一般是通过桌面的应用程序图标双击或单击系统开始菜单中应用程序的菜单链接，无论是桌面的快捷图标还是菜单链接都包含了应用程序的安装位置信息，打开它们的时候系统会按照这些位置信息找到安装目录然后启动程序。



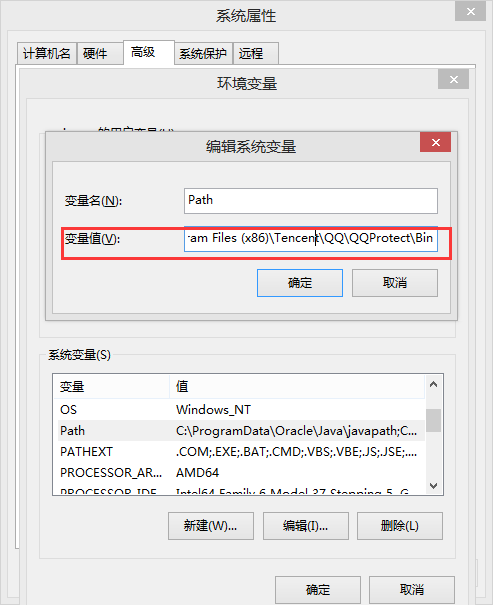
　　知道了一个应用程序的安装目录位置，我们也可以通过命令行工具打开，如QQ的位置为：C:\Program Files (x86)\Tencent\QQ\QQProtect\Bin，QQ的应用程序名为为QQProtect.exe，那么我们打开命令行工具，然后进入到“C:\Program Files (x86)\Tencent\QQ\QQProtect\Bin”目录，再输入“QQProtect”，即可运行qq。



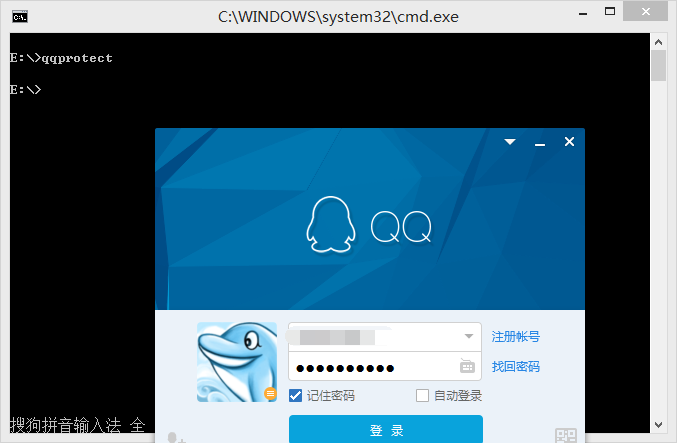
　　如果我们希望打开命令行工具后，直接输入“QQProtect”就能启动qq程序，而不是每次都进入qq的安装目录再启动，这个时候可以通过配置系统环境变量Path来实现。右击“我的电脑”，选择“属性”，在打开窗口中点击左边的“高级系统设置”，出现“系统属性”窗口，在“高级”选项卡下面点击“环境变量”。



　　编辑系统变量名“Path”，在“Path”变量（字符串内容）的后面追加qq的安装目录：;C:\Program Files (x86)\Tencent\QQ\QQProtect\Bin 注意追加的时候要在目录字符串的前面加个英文的分号;，英文分号是用来区分Path里面不同的路径。



　　确定保存后，再回到命令窗口，不管在任何目录下，你只要输入qqprotect的命令，qq就会启动。



　　通过启动qq的例子，我们总结下：当要求系统启动一个应用程序时，系统会先在当前目录下查找，如果没有则在系统变量Path指定的路径去查找。前面我们说了JDK包含了一堆开发工具，这些开发工具都在JDK的安装目录下，为了方便使用这些开发工具，我们有必要把JDK的安装目录设置了系统变量。这就是为什么在Windows安装了JDK后需要设置JDK的bin目录为系统环境变量的原因。

　　为了配置JDK的系统变量环境，我们需要设置三个系统变量，分别是JAVA\_HOME，Path和CLASSPATH。下面是这三个变量的设置防范。

　　JAVA\_HOME

　　先设置这个系统变量名称，变量值为JDK在你电脑上的安装路径：C:\Program Files\Java\jdk1.8.0\_20。创建好后则可以利用%JAVA\_HOME%作为JDK安装目录的统一引用路径。

　　Path

　　PATH属性已存在，可直接编辑，在原来变量后追加：;%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin 。

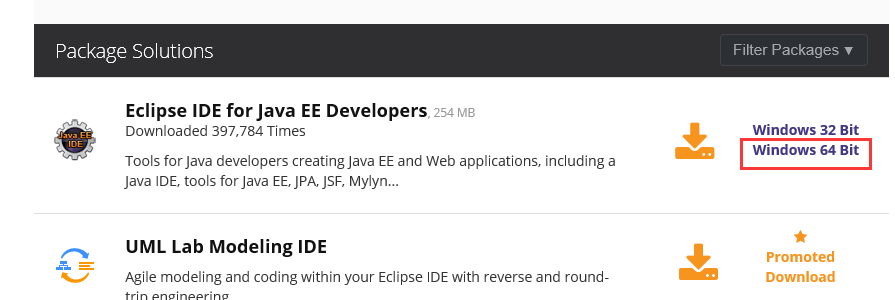
　　CLASSPATH

　　设置系统变量名为：CLASSPATH  变量值为：.;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar 。  
注意变量值字符串前面有一个”.”表示当前目录，设置CLASSPATH 的目的，在于告诉Java执行环境，在哪些目录下可以找到您所要执行的Java程序所需要的类或者包。

　　**第三步： 下载安装Eclipse**

　　Eclipse为Java应用程序及Android开发的IDE（集成开发环境）。Eclipse不需要安装，下载后把解压包解压后，剪切eclipse文件夹到你想安装的地方，打开时设置你的工作目录即可。

　　Eclipse的版本有多个，这里选择下载Eclipse IDE for Java EE Developers这个版本。



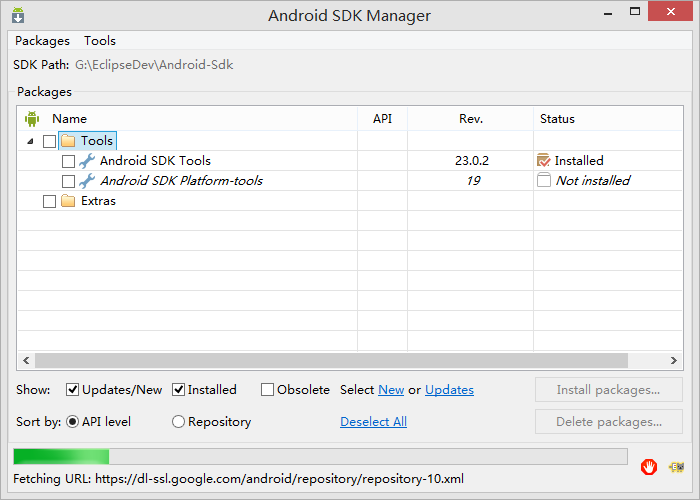
　　**第四步：下载安装Android SDK**

　　配置了JDK变量环境，安装好了Eclipse，这个时候如果只是开发普通的JAVA应用程序，那么Java的开发环境已经准备好了。我们要通过Eclipse来开发Android应用程序，那么我们需要下载Android SDK（Software Development Kit）和在Eclipse安装ADT插件，这个插件能让Eclipse和Android SDK关联起来。

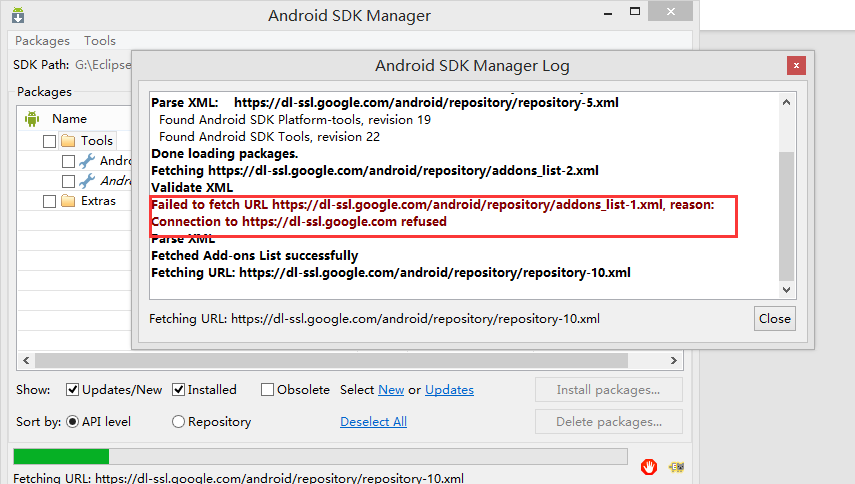
　　Android SDK提供了开发Android应用程序所需的API库和构建、测试和调试Android应用程序所需的开发工具。  
打开<http://developer.android.com/sdk/index.html>，我们发现google提供了集成了Eclipse的Android Developer Tools，因为我们这次是已经下载了Eclipse，所以我们选择单独下载Android SDK。

　　下载后双击安装，指定Android SDK的安装目录，为了方便使用Android SDK包含的开发工具，我们在系统环境变量中的Path设置Android SDK的安装目录下的tools目录。

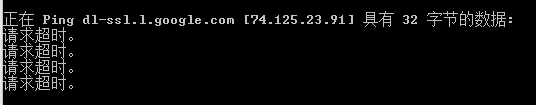
　　在Android SDK的安装目录下，双击“SDK Manager.exe”，打开Android SDK Manager，Android SDK Manage负责下载或更新不同版本的SDK包，我们看到默认安装的Android SDK Manager只安装了一个版本的sdk tools。



　　打开Android SDK Manager，它会获取可安装的sdk版本，但是国内有墙，有时候会出现获取失败的情况。



　　从弹出的log窗口中，我们可以看到连接 “[https://dl-ssl.google.com](https://dl-ssl.google.com/)”失败了。我们通过ping命令，发现果然网络不通。



　　从万能的互联网上，我们找到了解决这个问题的方案，而且行之有效。

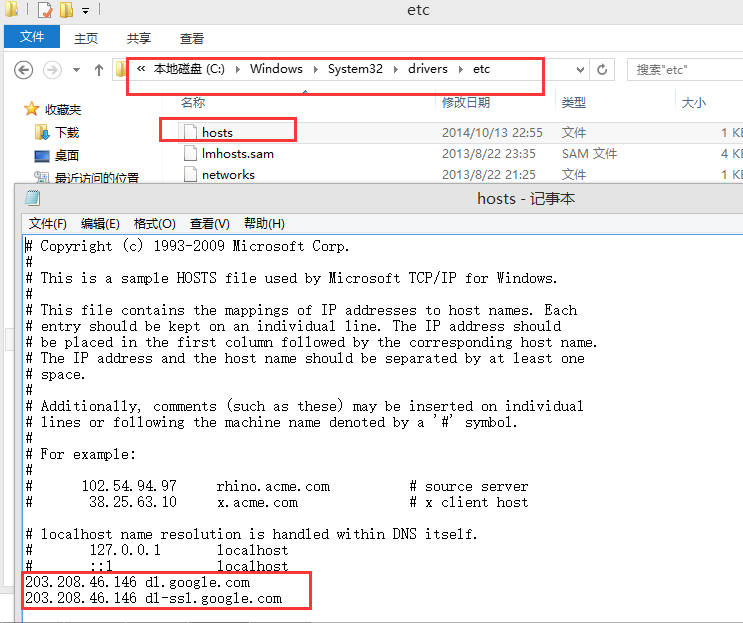
　　更改host文件

　　首先更改host文件，host文件在C:\Windows\System32\drivers\etc目录下，用记事本打开“hosts”文件，将下面两行信息追加到hosts文件末尾，保存即可。如果你的是windows8系统可能没有权限修改host文件，可以右击hosts文件，将Users组设置为可对hosts文件完全控制的权限即可。

　　203.208.46.146 dl.google.com

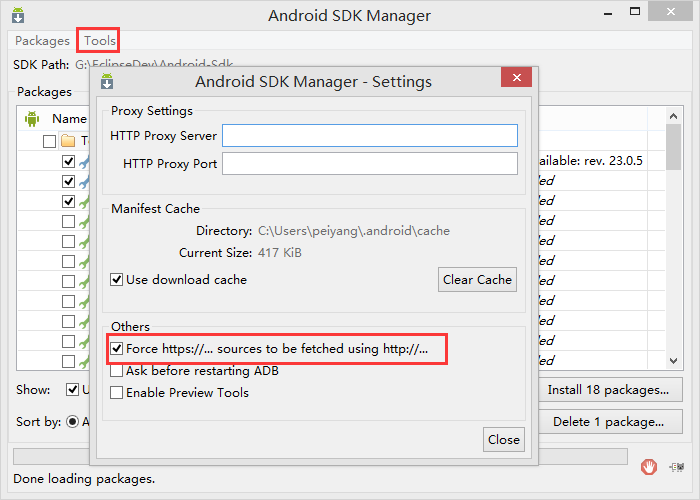
　　203.208.46.146 dl-ssl.google.com

　　上面两行放在host文件的意思是将本地访问dl.google.com和dl-ssl.google.com定向到ip地址为203.208.46.146的服务器上。

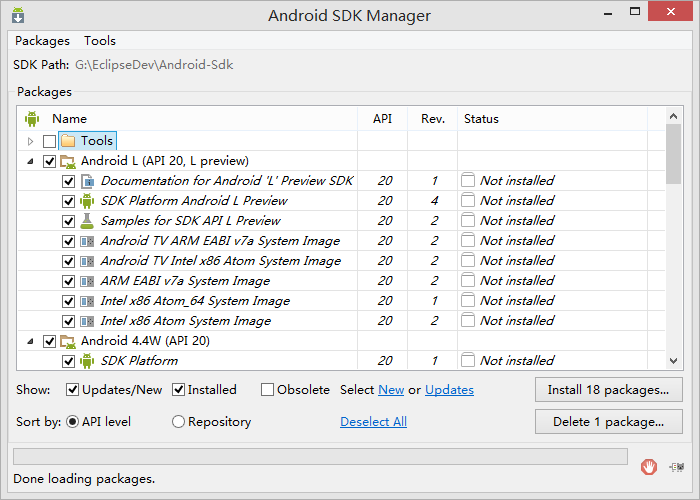


　　将Android SDK Manage上的https请求改成http请求

　　打开Android SDK Manager，在Tools下的 Options 里面，有一项 Force [https://..sources](https://..sources/) to be fetched using [http://…](http://.../) 将这一项勾选上，就可以了。



　　再打开Android SDK Manager.exe，正常情况下就可以下载Android的各个版本的sdk了。你只需要选择想要安装或更新的安装包安装即可。这里是比较耗时的过程，还会出现下载失败的情况，失败的安装包只需要重新选择后再安装就可以了。

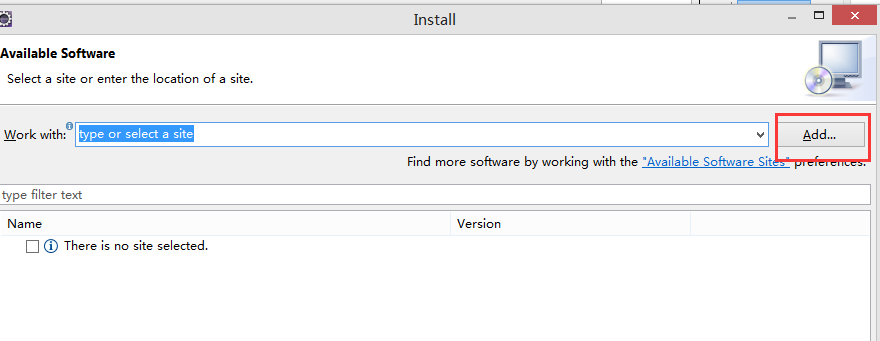


　　**第五步：为Eclipse安装ADT插件**

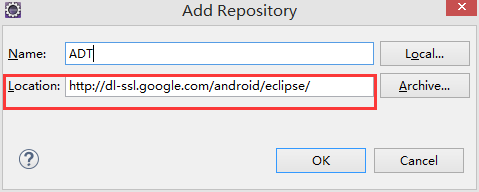
　　前面我们已经配置好了java的开发环境，安装了开发Android的IDE，下载安装了Android SDK，但是Eclipse还没有和Android SDK进行关联，也就是它们现在是互相独立的，就好比枪和子弹分开了。为了使得Android应用的创建，运行和调试更加方便快捷，Android的开发团队专门针对Eclipse IDE定制了一个插件：Android Development Tools（ADT）。

　　下面是在线安装ADT的方法：

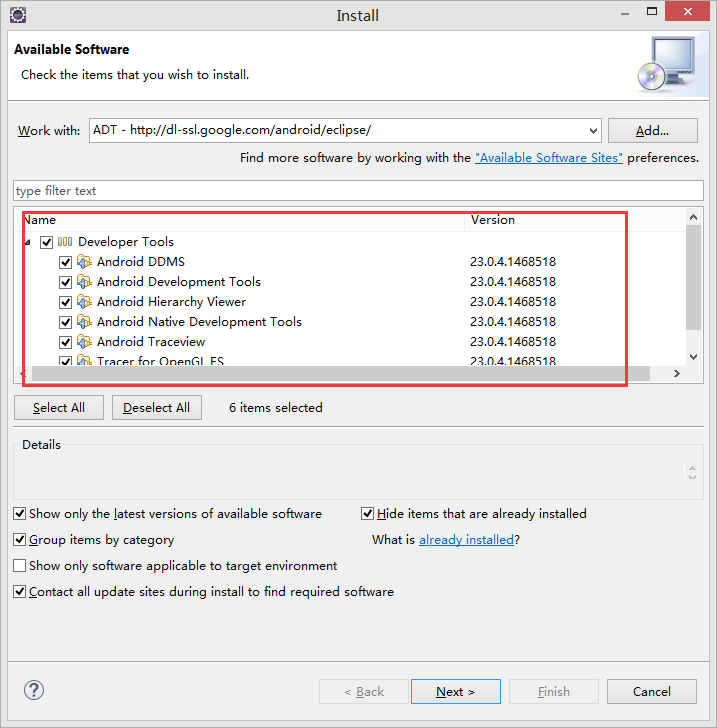
　　启动Eclipse，点击 Help菜单 -> Install New Software… ?，点击弹出对话框中的Add… 按钮。



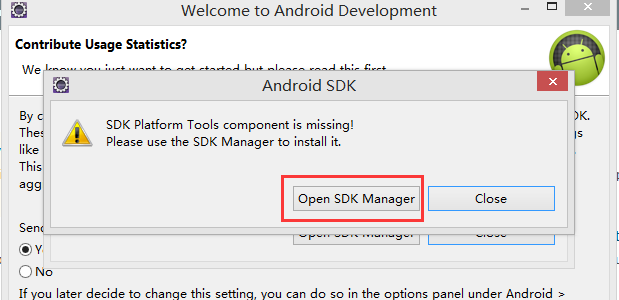
　　然后在弹出的对话框中的Location中输入：<http://dl-ssl.google.com/android/eclipse/>，Name可以输入ADT，点击“OK”按钮。



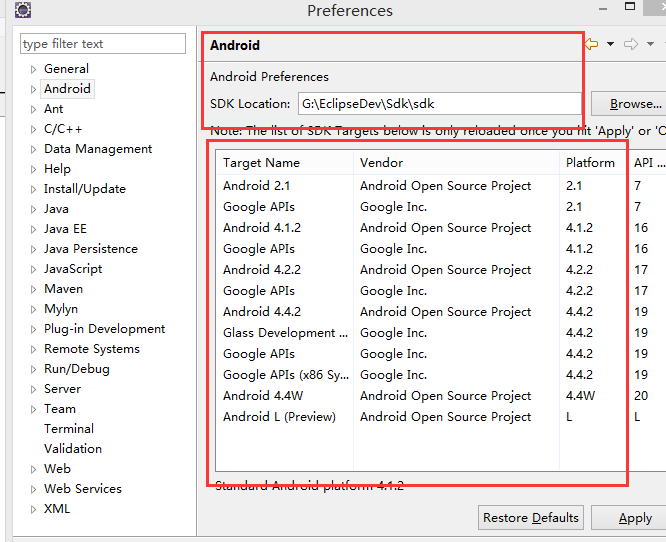
　　在弹出的对话框选择要安装的工具，然后下一步就可以了。



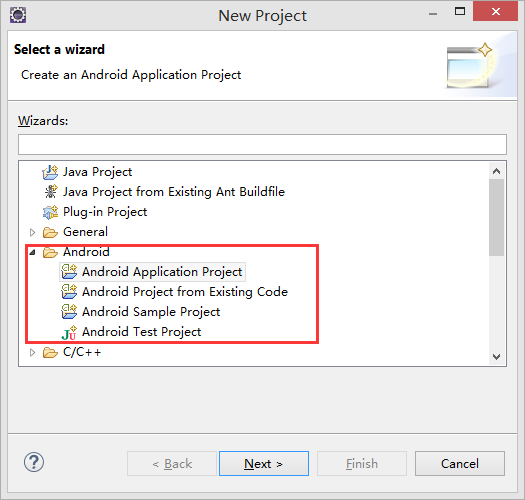
　　安装好后会要求你重启Eclipse，Eclipse会根据目录的位置智能地和它相同目录下Android sdk进行关联，如果你还没有通过sdk manager工具安装Android任何版本的的sdk，它会提醒立刻安装它们。



　　如果Eclipse没有自动关联Android sdk的安装目录，那么你可以在打开的Eclipse选择 Window -> Preferences ，在弹出面板中就会看到Android设置项，填上安装的SDK路径，则会出现刚才在SDK中安装的各平台包，按OK完成配置。



　　到这里，我们在windows上的Android上的开发环境搭建就完成了，这时候，你用Eclipse的File——》New——》Project…新建一个项目的时候，就会看到建立Android项目的选项了。



　　原文来自：[边写边唱](http://www.cnblogs.com/zoupeiyang/p/4034517.html)

**eclipse android adt安装后没有菜单栏没有相应的插件?**  
  
如果安装过程没错,直接在Eclipse ->window->customize Perpective->Command Groups Availability 然后把以andorid 开头全部选项就可以了