2、Creating an Android Development Environment

SmartPhone Application

Development Environment

- Android applications, like most mobile phone applications, are developed in a host-target development environment.
  - To develop your application on a host computer (where resources are abundant) and download it to a target mobile phone for testing and ultimate use.
  - Applications can be tested and debugged either on a real Android device or on an emulator. For most developers, using an emulator is easier for initial development and debugging, followed by final testing on real devices.
- To write your own Android mobile phone applications, you'll first need to collect the required tools and set up an appropriate development environment on your PC or Mac.
  - To download them and install on your computer.

Development Environment

- Android applications are developed in a host-target development environment.
  - To develop an application on a host computer (where resources are abundant) and download it to a target mobile phone for testing and ultimate use.
- Applications can be tested and debugged either on a real Android device or on an emulator.
  - For most developers, using an emulator is easier for initial development and debugging, followed by final testing on real devices.

- To write your own Android mobile phone applications, you'll first need to collect the required tools and set up an appropriate development environment on your PC or Mac.
  - Linux, Windows, and OS X are all supported development environments, and we'll show you how to install the latest set of tools on each.

- The Android SDK supports several different integrated development environments (IDEs). Here we will focus on Eclipse because it is the IDE that is best integrated with the SDK and free.
- No matter which operating system you are using, you will need essentially the same set of tools:
  - The Eclipse IDE
  - Sun's Java Development Kit (JDK)
  - The Android Software Developer's Kit (SDK)
  - The Android Developer Tool (ADT), a special Eclipse plug-in
  - Phone driver

Setting Up Android Development Environment

- Software：
  - ①JDK 5 or JDK 6 (JDK，not JRE)
  - ②Eclipse 3.3 or later
  - ③The Android Software Developer's Kit (SDK)
  - ④ The Android Developer Tool (ADT), a special Eclipse plug-in
  - ⑤smart phone driver

Android Environment configuration

- JDK1.6
  - download：http://java.sun.com/javase/downloads/
- Eclipse
  - download：http://www.eclipse.org/downloads/
- ADT
  - download： http://developer.android.com
- Android SDK 2.x
  - download： http://developer.android.com
- Phone driver

Quick Start

- 1. start
  - Install JDK
    - Set environment parameters：JAVA_HOME、CLASSPATH
  - Eclipse
- 2. install Android SDK
- 3. setup ADT on Eclipse
- 4. install phone-driver
- SDK Setup Done!

Install Eclipse

- Eclipse 3.3 or later
- Eclipse JDT plugin (included in most Eclipse IDE packages)
  - If you need to install or update Eclipse, you can download it from http://www.eclipse.org/downloads/.
- Several types of Eclipse packages are available for each platform. For developing Android applications, we recommend that you install one of these packages:
  - Eclipse IDE for Java EE Developers
  - Eclipse IDE for Java Developers
  - Eclipse for RCP/Plug-in Developers
  - Eclipse Classic (versions 3.5.1 and higher)

Android SDK

- Android 2.1 Platform
- Android 1.6 Platform
- Android 1.5 Platform
- Older Platforms
  - Android 2.0.1 Platform
  - Android 2.0 Platform
  - Android 1.1 Platform

Operating System

- Windows XP (32-bit) or Vista (32- or 64-bit)
- Mac OS X 10.5.8 or later (x86 only)
- Linux (tested on Linux Ubuntu Hardy Heron)
  - 64-bit distributions must be capable of running 32-bit applications. For information about how to add support for 32-bit applications.

Hardware environment

Component type	Approximate size	Comments
SDK Tools	50 MB	need
Android platform (each)	150 MB	At least one
SDK Add-on (each)	100 MB	Optional
USB Driver for Windows	10 MB	Optional, but Windows need
Samples (per platform)	10 MB	Optional
Offline documentation	250 MB	Optional

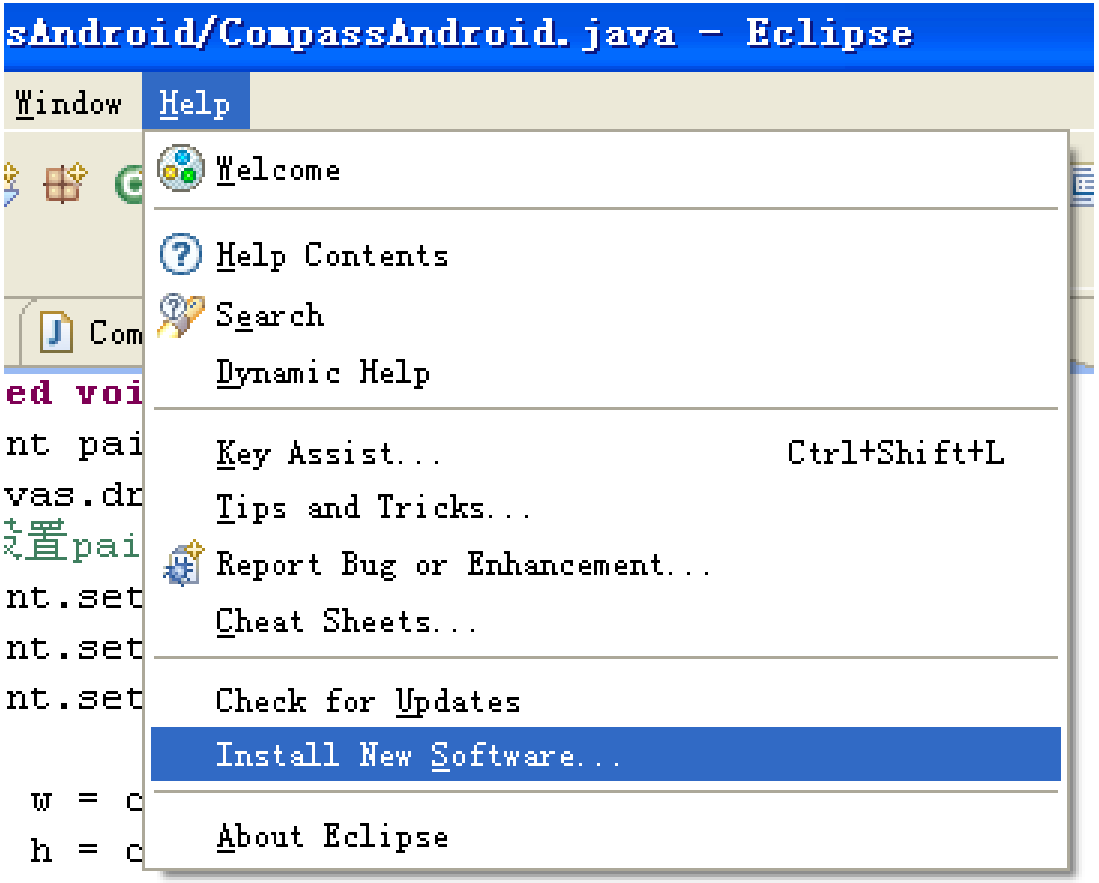
- Does not include JAVA、Eclipse。

Constants

int	BASE	Oct. 2008	The original, first, version of Android.
int	BASE_1_1	Feb. 2009	First Android update, officially called 1.1.
int	CUPCAKE	May 2009	Android 1.5.
int	CUR_DEVELOPMENT		Magic version number for a current development build, which has not yet turned into an official release.
int	DONUT	Sep. 2009	Android 1.6.
int	ECLAIR	Nov. 2009	Android 2.0: Applications targeting this or a later release will get these new changes in behavior: ▫The Service.onStartCommand function will return the new START_STICKY behavior instead of the old compatibility START_STICKY_COMPATIBILITY.
int	ECLAIR_0_1	Dec. 2009	Android 2.0.1
int	ECLAIR_MR1	Jan. 2010	Android 2.1

Install Android

- Extract，Install New Software on Eclipse



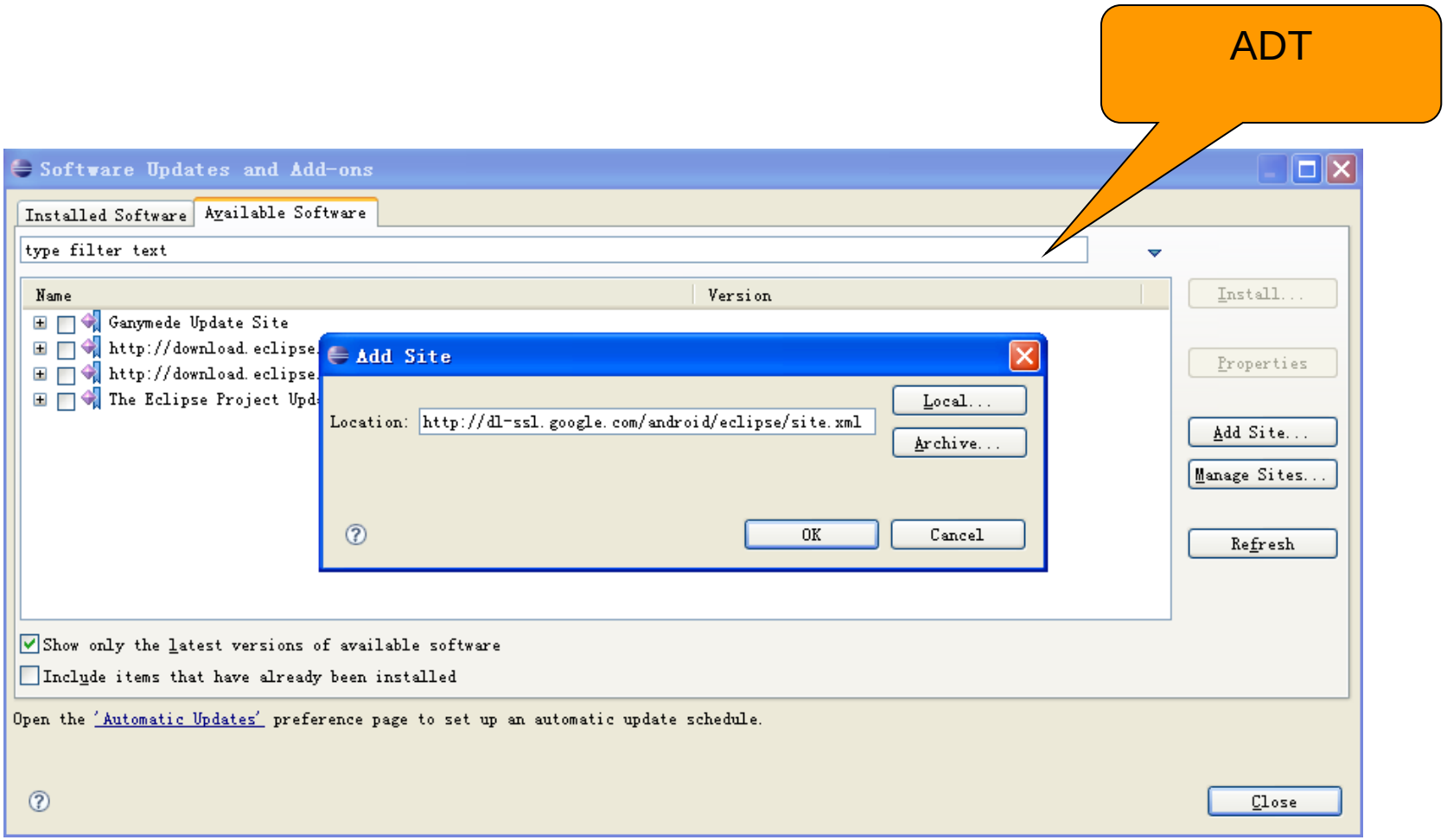
Install the Android plug-in (ADT):

- The plug-in is installed in much the same way as any other Eclipse plug-in:
  - Start Eclipse.
  - From the menu bar, select "Help Software Updates Find and Install...".
  - In the Install/Update dialog, select "Search for new features to install" and click on "Next."
  - In the Install dialog, click on "New Remote Site." A "New Update Site" dialog pops up. Enter a name for the plug-in ("Android Plugin" will do), and the URL for updates: . Click "OK."
  - The new site should now appear in the list of sites on the Install dialog. Click "Finish."

Install the Android plug-in (ADT):

- In the Search Results dialog, select the checkbox for "Android Plugin Developer Tools" and click "Next."
- The license agreement for the plug-in appears. Read it, and if you agree, select "Accept terms of the license agreement" and click "Next." Click "Finish."
- You will get a warning that the plug-in is not signed. Choose to install it anyway by clicking "Install All."
- Restart Eclipse.
- After Eclipse restarts, you need to tell it where the SDK is located. From the menu bar, select "Window Preferences." In the Preferences dialog, select "Android" in the left column.
- Use the "Browse" button to navigate to the place you installed the Android SDK, and click on "Apply," then on "OK."

Android Development Environment



DDMS

- Dalvik Debug Monitor Service, is unique to Android. It provides Android-specific debug information, including:
  - Devices
    - This shows you what devices (emulated or hardware) are available to run your applications.
  - Emulator Control
    - To adjust parameters that define how the telephony and location emulators work. When running on the emulator, we'll use this to manually send location updates to the location provider.
  - LogCat
    - This is a view of the very powerful logging facility available under Android, which allows you to see everything going on in the target system, and to filter out the information you really care about.
  - Threads, Heap, and File Explorer
    - This is a tabbed set of views where you can follow the running threads in the application, see how the heap is being used, and select files from the folder hierarchy.

website

- http://androidappdocs.appspot.com/index.html
- http://www.android.com/
- http://www.android123.com.cn/
- http://www.androidcn.org/index.php
- https://sites.google.com/site/androidappcourse/home（翻墙）
- 。 。 。 。 。 。

Thank you!