

Android程序分析环境搭建-Windows篇

上一篇讲了如何在Ubuntu上搭建Android程序分析环境，现在，我们继续讲解如何在Windows上完成这些操作。

Windows分析环境搭建

如果读者不打算在Windows环境上编译Android源码，则最低可以使用Windows XP作为系统的配置环境；反之，如果需要使用 `Docker` 来编译Android系统源码，则需要满足 `Docker` 运行的最低系统要求，即最低需要Windows 7旗舰版以上，笔者推荐使用Windows 10。

安装JDK

尽管现在Android软件可以使用Kotlin语言来开发，但在未来很长的一段时间内，使用Java语言开发Android软件仍然会占很大的比例，JDK也就成了Android软件开发必装的软件之一。Windows平台的JDK安装只有一个可选项，即Oracle公司提供的JDK。可以到它的官网上进行下载安装。下载地址为：<http://www.oracle.com/technetwork/java/javase/downloads/index.html>，打开下载页面，可以看到JDK的多个版本，目前Android软件开发支持到最新的版本是JDK 8。下载最新的JDK 8安装包exe文件，然后双击运行，按照默认的安装选项，不停的下一步即可完成安装。

安装完成后手动添加 `JAVA_HOME` 环境变量，它的值为JDK安装后的完整路径。例如JDK版本8u131，按照默认的安装路径，可将 `JAVA_HOME` 环境变量设置为“`C:\Program Files\Java\jdk1.8.0_131`”。然后将“`%JAVA_HOME%\bin`”添加到系统的 `PATH` 变量中，方便 `Android Studio` 或其他第三方软件找到 `Java` 编译器的位置。

安装完成后检查一下 `Java` 是否安装成功。单击“开始”按钮，选择“运行”，在出现的对话框中输入CMD命令打开命令提示符窗口，在窗口中输入 `java -version`，如果屏幕上出现如下所示的输出信息，说明安装成功。

```
> java -version
java version "1.8.0_131"
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)
```

安装Android SDK

新版本的 `Android Studio` 包含了Android SDK，如果读者不打算使用 `Android Studio` 来开发Android软件，而选择命令行方式使用Android SDK，则可以下载Android SDK的单独版本；反之，则可以跳过本小节，直接参考 `Android Studio` 的安裝配置方法。


```

ndk-bundle                | 14.1.3816874 | NDK
patcher;v4                | 1             | SDK Patch Applier v4
platform-tools            | 25.0.6        | Android SDK Platform-
Tools
.....
platforms;android-24      | 2             | Android SDK Platform
24
platforms;android-25      | 3             | Android SDK Platform
25
platforms;android-7       | 3
.....
sources;android-23        | 1             | Sources for Android 23
sources;android-24        | 1             | Sources for Android 24
sources;android-25        | 1             | Sources for Android 25
system-images;a...ult;armeabi-v7a | 4             | ARM EABI v7a System
Image
system-images;a...-10;default;x86 | 4             | Intel x86 Atom System
Image
.....
system-images;a...google_apis;x86 | 4             | Google APIs Intel x86
Atom Sys...
system-images;a...gle_apis;x86_64 | 4             | Google APIs Intel x86
Atom_64 ...
tools                     | 26.0.2        | Android SDK Tools

Available Updates:
  ID      | Installed | Available
  ----- | -
tools    | 26.0.1    | 26.0.2
done

```

Installed packages 部分是已经安装的包。Available Packages 是可以下载的包，其中 Path 是包的下载安装目录路径，以分号";"分隔；Version 是对应的版本号；Description 是包的描述信息。例如下载安装Android SDK提供的最新版本的模拟器emulator，可以执行如下命令：

```
> sdkmanager emulator
```

命令执行后会弹出提示接受LICENSE，输入"y"后回车，此时会开始下载模拟器。执行以下命令可以下载平台相关的工具，里面包含了adb、fastboot等命令：

```
> sdkmanager platform-tools
```

执行以下命令可以更新所有可更新的包：

```
> sdkmanager --update
```

Android SDK默认只提供了最新版本的SDK Tools，并不包含开发用的平台SDK与构建工具，如果想要开发Android程序，还需要下载其他额外的包。以Android 7.1开发为例，还需要执行如下命令：

```
> sdkmanager build-tools;25.0.3
> sdkmanager docs
> sdkmanager platforms;android-25
> sdkmanager sources;android-25
```

下载完所有需要的包后，将常用的工具所在路径加到系统的 `PATH` 环境变量中。其中包含 `adb`、`fastboot` 的 `platform-tools` 目录；包含 `aapt`、`apksigner`、`dexdump`、`zipalign` 的 `build-tools/25.0.3` 目录；包含 `android`、`monitor`、`emulator` 的 `tools` 目录等。

完成以上操作后，在命令提示符下输入“`emulator -version`”与“`adb version`”命令查看是否能成功运行。执行结果如下所示：

```
> emulator -version
Android emulator version 26.0.3.0 (build_id 3965150)
Copyright (C) 2006-2015 The Android Open Source Project and many others.
This program is a derivative of the QEMU CPU emulator (www.qemu.org).

This software is licensed under the terms of the GNU General Public
License version 2, as published by the Free Software Foundation, and
may be copied, distributed, and modified under those terms.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
> adb version
Android Debug Bridge version 1.0.39
Revision 3db08f2c6889-android
Installed as E:\Android\sdk\platform-tools\adb.exe
```

至此，Android SDK的安装配置就算完成了。

安装Android NDK

Android NDK可以单独用来开发Android原生的动态库与可执行程序，也可以配合Android SDK开发包含原生代码的Android软件。如果之前安装了Android SDK，可以在命令行下执行如下命令安装Android NDK。

```
> sdkmanager ndk-bundle
```

新版本的Android工程使用Cmake来构建，并使用lldb来调试，如果同时需要编写Java与Native代码，还需要在这里安装它们。执行如下命令：

```
> sdkmanager cmake;3.6.3155560
> sdkmanager lldb;2.3
```

除了Android SDK方式安装Android NDK外，也可以从Android官网下载最新的Android NDK压缩包进行安装，地址是：<https://developer.android.com/ndk/downloads/index.html>。如图所示，点击链接中的Windows版本，下载最新的Android NDK压缩包即可。

NDK 下载

平台	软件包	大小（字节）	SHA1 校验和
Windows 32 位	android-ndk-r14b-windows-x86.zip	707533928	070443eaa7fa37ed337f91c655e02ca708d37c92
Windows 64 位	android-ndk-r14b-windows-x86_64.zip	769151176	a625e8c599bccdb9061b61dcf3d1f1a01071613f
Mac OS X	android-ndk-r14b-darwin-x86_64.zip	824705073	2bf582c43f6da16416e66203d158a6dfaba4277c
Linux 64 位 (x86)	android-ndk-r14b-linux-x86_64.zip	840626594	becd161da6ed9a823e25be5c02955d9cbca1dbeb

解压下载的压缩包后，将NDK的根目录添加到系统的 `NDK_HOME` 环境变量，并将其添加到系统的 `PATH` 环境变量。在命令提示符下执行 `ndk-build -v`，可以正常显示如下输出信息，说明配置完成了。

```
> ndk-build -v
GNU Make 3.81
Copyright (C) 2006 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE.

This program built for i586-pc-mingw32
```

Android Studio集成开发环境

`Android Studio` 是全新的Android软件开发IDE，它包含了整套Android开发所需的SDK以及配置工具，如果读者决定使用 `Android Studio` 来开发Android程序，可以跳过前面安装配置Android SDK与Android NDK的步骤，直接使用 `Android Studio` 来安装它们。

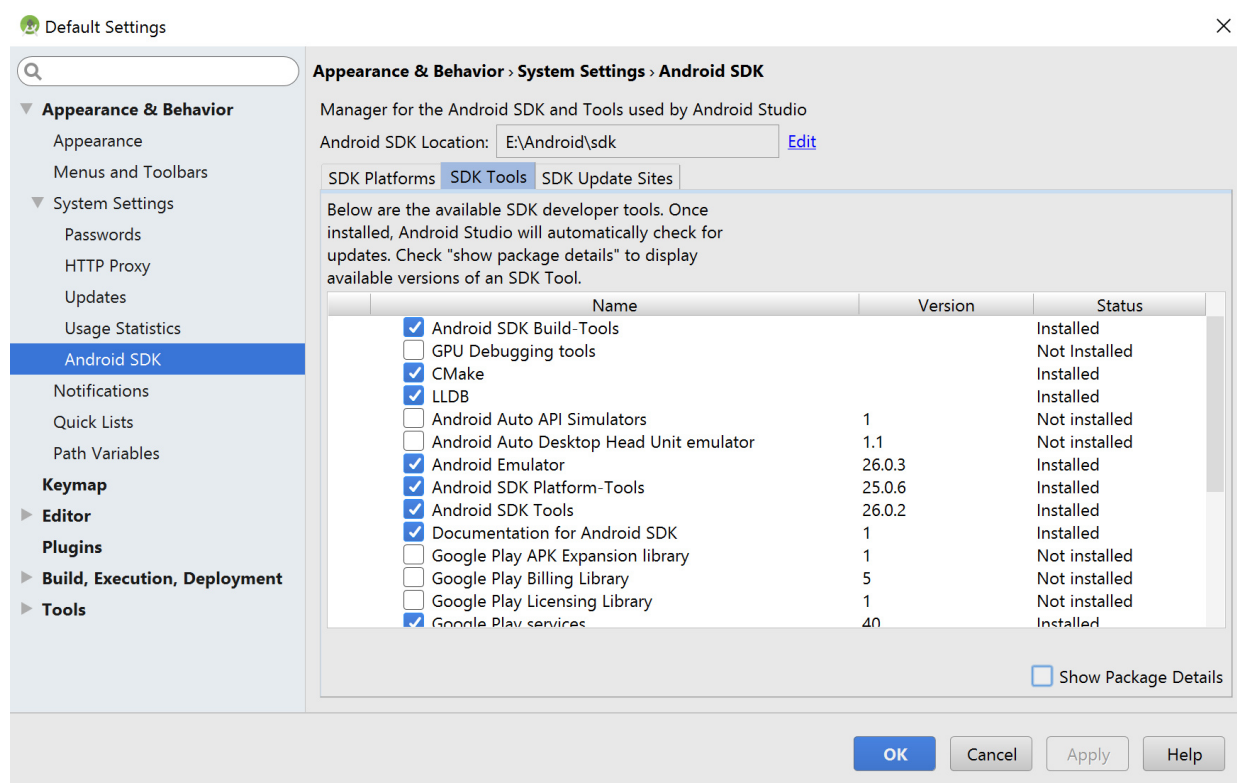
首先到<https://developer.android.com/studio/index.html> 下载最新的 `Android Studio`。

Windows平台分为包含了Android SDK的exe安装版本，与不包含Android SDK的exe安装版本及zip压缩包版本。如图所示：

平台	Android Studio 软件包	大小	SHA-1 校验和
Windows (64 位)	android-studio-bundle-162.3934792-windows.exe 包含 Android SDK (推荐)	1,893 MB (1,985,351,576 bytes)	9d787c0cf453e40ad1b0621f0e5a9653270dcc22a58fa7c9fab2223531c83a41
	android-studio-ide-162.3934792-windows.exe 无 Android SDK	422 MB (442,578,936 bytes)	939cf6a1556c9078f4cbc05d1d2b8175f365ea5485661b04579788c423c38c95
	android-studio-ide-162.3934792-windows.zip 无 Android SDK, 无安装程序	438 MB (460,075,724 bytes)	87cdb1295137ae75e5dbc7f9b6c499079d05d1141efa769c085e08c18fcec437
Windows (32 位)	android-studio-ide-162.3934792-windows32.zip 无 Android SDK, 无安装程序	438 MB (459,516,639 bytes)	67098fdd64d257c2aa98db29af8a6341fa787d9d0b37c8ea06eb5ec00dc51986
Mac	android-studio-ide-162.3934792-mac.dmg	434 MB (455,228,488 bytes)	e20fb9ae57c9ca35285dfc40594a0e23bf2c47648a43ca6b7da0e7e58d092b86
Linux	android-studio-ide-162.3934792-linux.zip	438 MB (459,976,690 bytes)	6e33a232466820a15c884af9faefab772b8267ede056aaedb63f291ceb5e95a7

不包含Android SDK的版本, 会在第一次运行时, 联网下载最新版本的SDK及构建工具。包含Android SDK的版本, 则自带了最新版本的SDK, 按照向导不停的下一步即可安装完成。

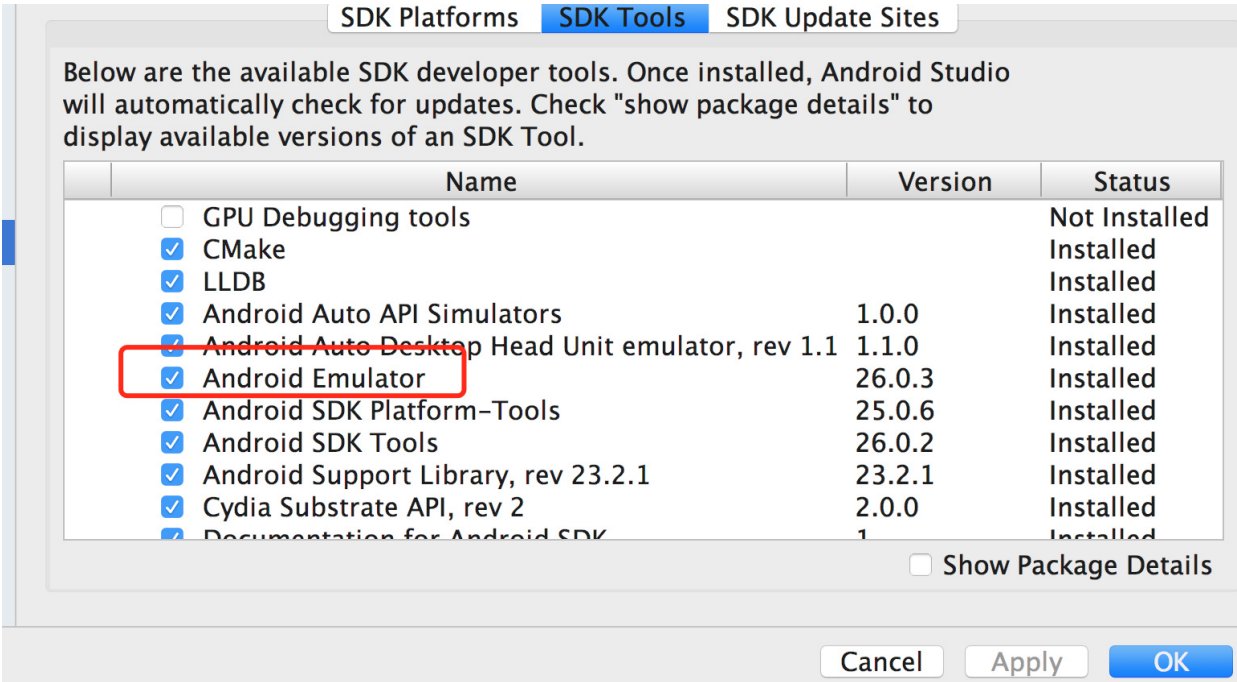
安装完成后启动 `Android Studio`, 点击主界面上的Configure->SDK Manager, 会弹出SDK管理界面。在这里可以下载安装需要的软件包与构建工具。如图所示, Cmake与LLDB可以直接在这里勾选安装, 非常方便。



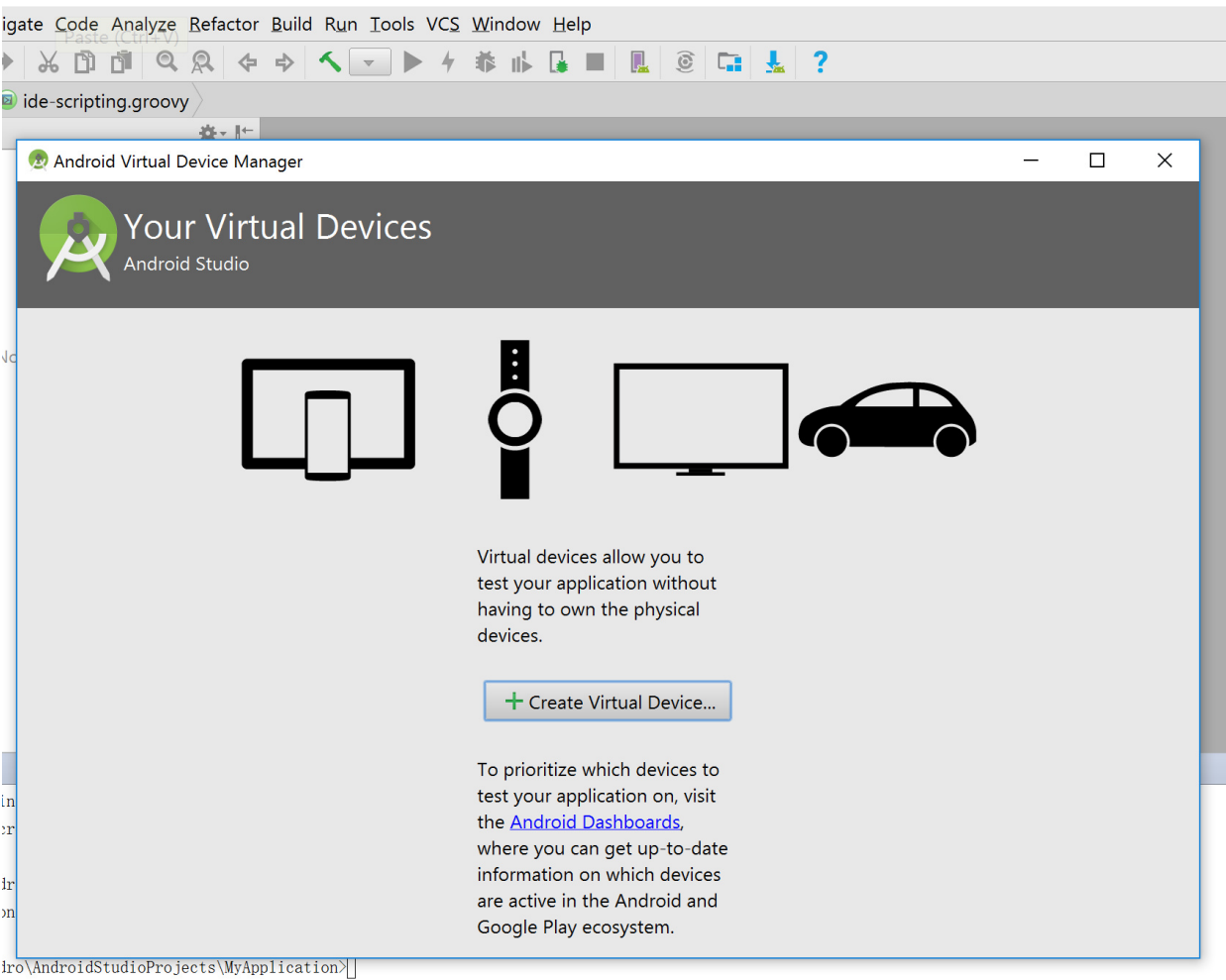
使用 `Android Studio` 的过程中, 不需要一次性下载所有版本的SDK与构建工具, 在编译第三方项目或构建新项目时, `Android Studio` 会根据实际情况, 提示用户下载相应版本的SDK。用户体验方面比之前基于Eclipse的ADT开发工具要优秀得多。

创建Android模拟器

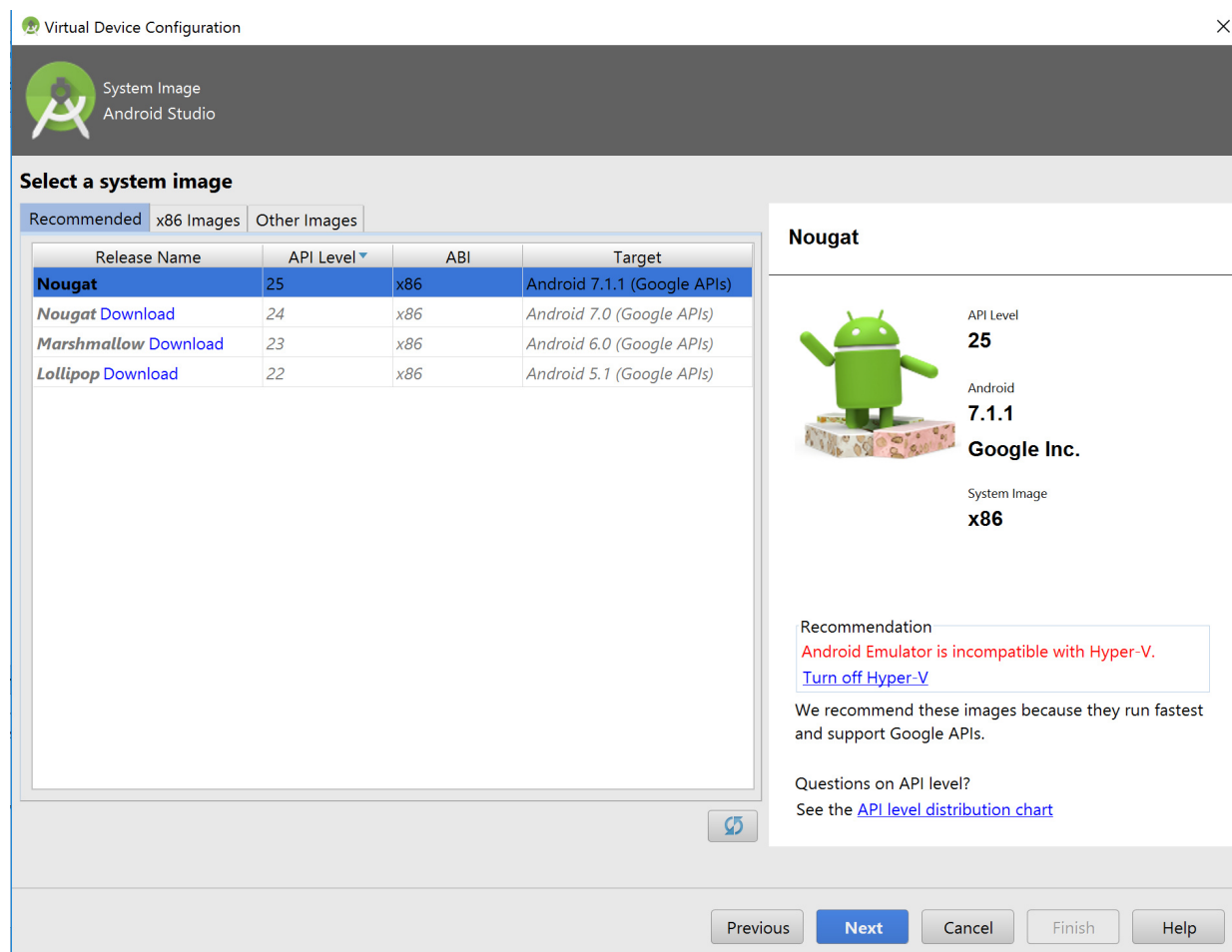
在创建Android模拟器前，请在 Android Studio 中确认本机已经安装模拟器必须的镜像文件。如图所示：



使用 Android Studio 创建Android模拟器，需要先打开一个Android工程。在打开的工程界面上，点击菜单Tools->Android->AVD Manager，会弹出Android模拟器创建对话框，如图所示：



点击Create Virtual Device...按钮，弹出虚拟设备创建窗口，选择设备的类型，例如Nexus 6p，点击Next，选择创建的系统版本。支持的系统版本会在这里全部列出来，默认情况下会选择本机安装的最新系统版本镜像，对于本机没有安装的系统版本，会在系统版本名称的旁边显示一个Download按钮，点击该按钮，可以下载相应的虚拟设备镜像。如图所示，选择Nougat后点击Next，确认设备类型与版本号无误后，点击Finish即可完成创建。



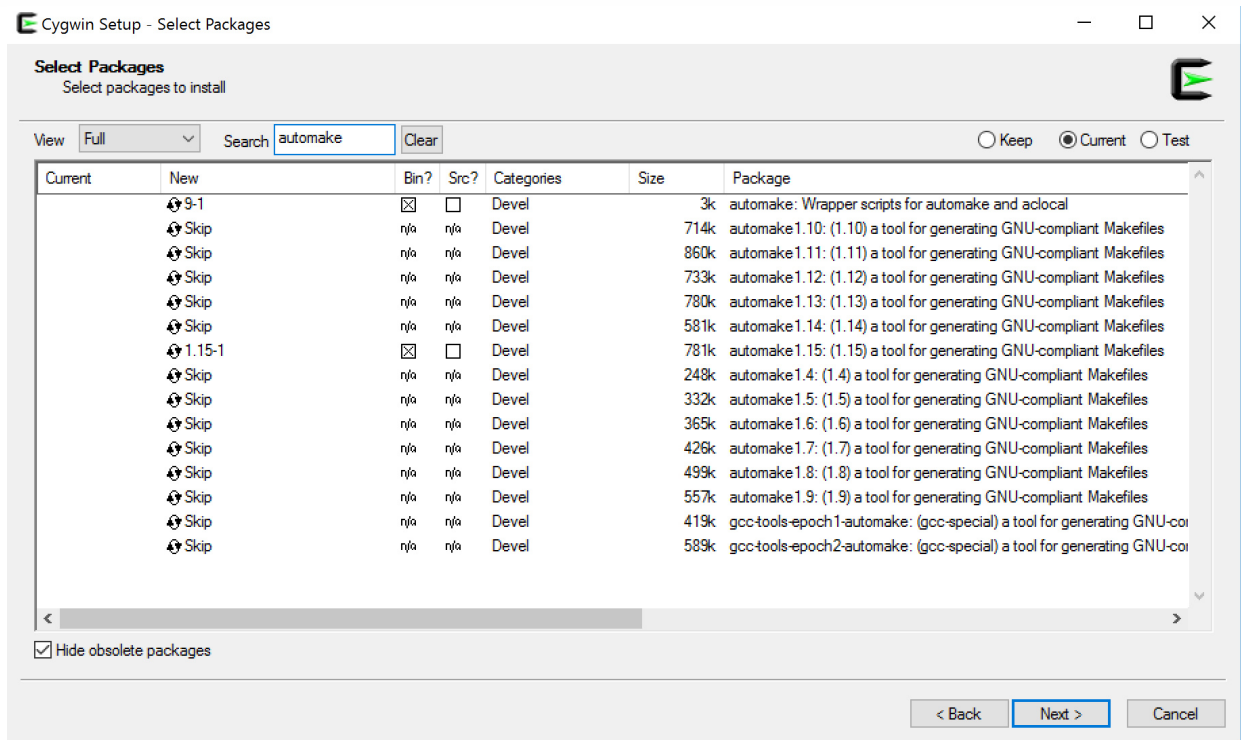
在返回的AVD Manager管理界面，选择创建好的模拟器，点击Action栏的绿色启动按钮，或者在选择模拟器上双击，就会启动模拟器。

使用时间较长的模拟器，可以在AVD Manager管理界面中选中它，然后右键选择Wipe Data清空数据，将模拟器内容重置，或者选择Delete删除后重新创建。

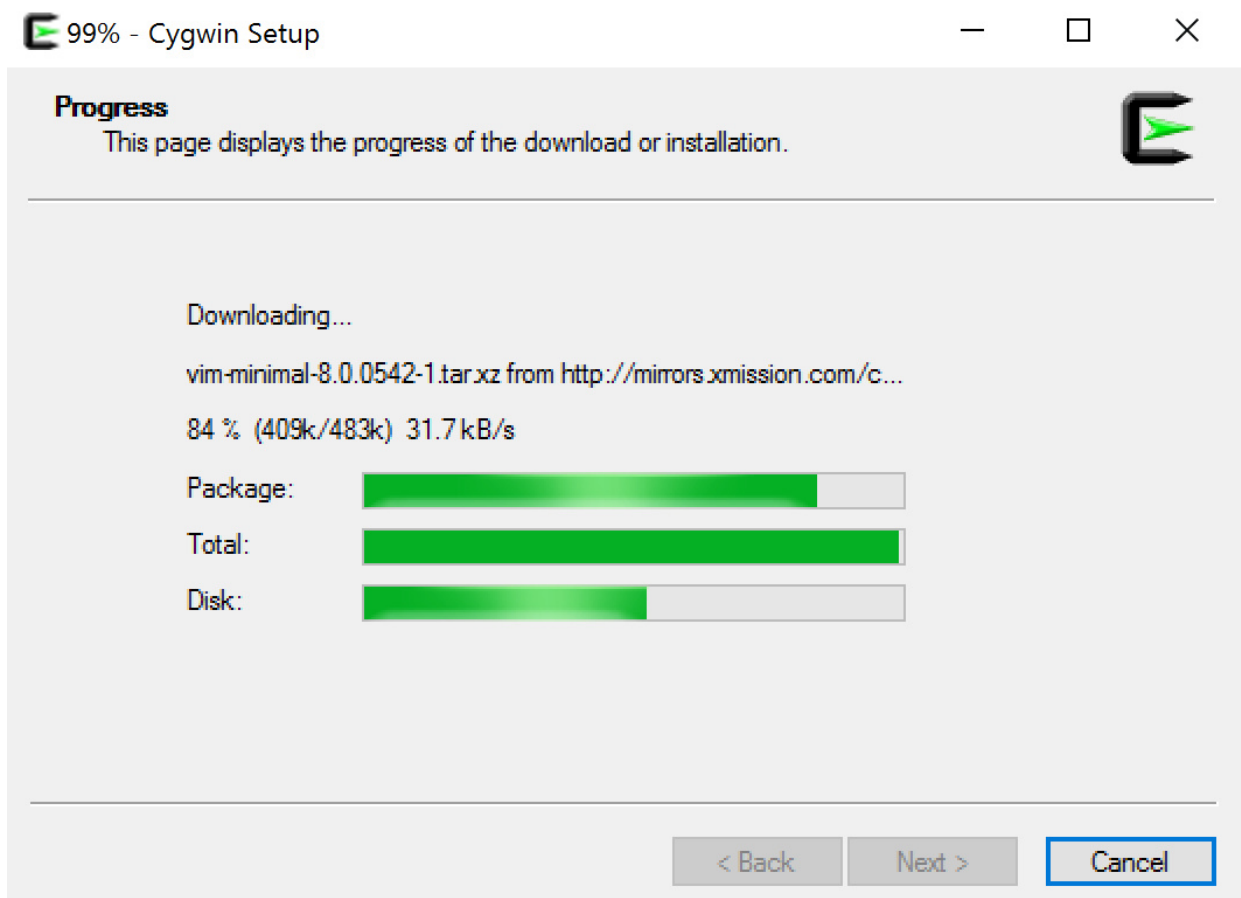
CygWin

为了保证在内容讲解的过程中，工具的操作与输出信息达到一致，因此以后会使用了大量的UNIX命令来完成演示操作。在Windows平台上，有多种UNIX命令的模拟环境。对于Windows 10之前的系统，可以使用的模拟环境有 Cygwin、MinGW、MSys2，Windows 10系统则加入了 Bash On Ubuntu On Windows。它们在操作上差别不大，这里演示时使用的是 Cygwin 与 Bash On Ubuntu On Windows，读者可以根据个人的喜欢选择它们四者之一。

首先到 Cygwin 的官方下载最新版本的安装程序，地址是：<https://www.cygwin.com>。对于64位的Windows系统，下载地址目前为https://www.cygwin.com/setup-x86_64.exe。下载安装程序后双击运行安装，在确定 Cygwin 的安装目录后，不停的下一步，直到命令安装界面，如图所示：



输入需要安装的命令，例如 `automake`，Cygwin 会列出所有与这个名字相关的软件包，在结果的 Bin 栏上打勾，选择要安装的 `automake` 版本，如果需要安装其源码，可以在 Src 栏上打勾。确定好后，可以输入其他的软件名，搜索下一个需要安装的软件，例如本人经常使用到的 `grep`、`openssl`、`tree` 等命令。所有命令都选择勾选好后，点击 Next 进行安装，确认要安装的软件的版本正确无误后，再一次点击 Next，Cygwin 会联网下载所有选中的软件包，如图所示：



完装完成后，会在桌面上生成Cygwin64 Terminal的图标，双击运行就会进行Shell执行环境，在该环境下，可以执行与macOS、Linux上一样的Shell命令，例如，执行 `uname -a` 查看机器信息；执行 `id` 查看用户的ID信息；执行 `ls` 列举目录；执行 `wc -l` 统计输出行数等。效果如下所示：

```
android@DESKTOP-0E964KM ~
$ uname -a
CYGWIN_NT-10.0 DESKTOP-0E964KM 2.8.0(0.309/5/3) 2017-04-01 20:47 x86_64
Cygwin

android@DESKTOP-0E964KM ~
$ id
uid=197609(android) gid=197609(android)
groups=197609(android),401408(Medium Mandatory Level),197610(Ssh
Users),559(Performance Log Users),545(Users),4(INTERACTIVE),66049(CONSOLE
LOGON),11(Authenticated Users),15(This
Organization),68564(MicrosoftAccount+fei_cong@hotmail.com),113(Local
account),66048(LOCAL),262180(Cloud Account Authentication)

android@DESKTOP-0E964KM ~
$ ls /cygdrive/c | wc -l
45
```

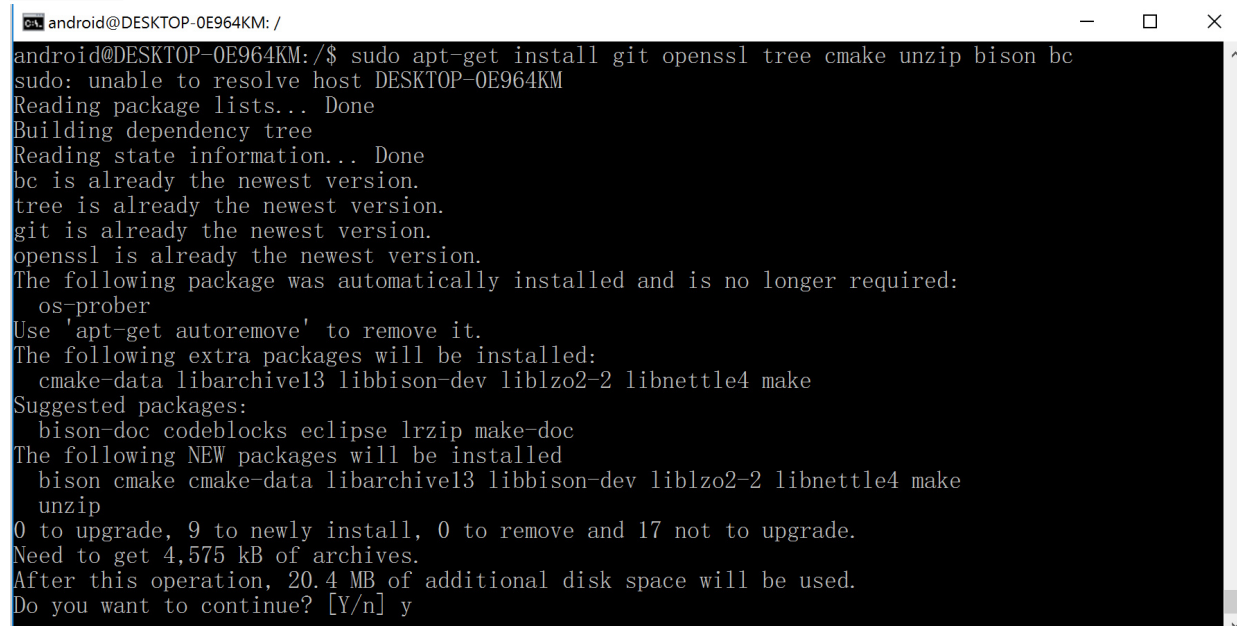
Bash on Ubuntu On Windows

对于Windows 10系统，可以使用它全新的名为 `Bash On Ubuntu On Windows` 的Shell模拟环境。该环境随Windows 10最新的Linux子系统一起提供。具体的安装步骤如下：

1. 升级到最新的Windows 10系统，以便系统能够支持开启"开发者模式"。
2. 打开你的Windows 10的设置，在"更新和安全"选项中选择"开发者"，然后选择"开发者模式"。
3. 在Windows 更新和安全选项中，选择"Windows 更新"，在"更新设置"中选择高级选项，勾选"内部预览版本"选项。
4. 打开"控制面板"-"程序"，选择"打开或者关闭Windows特性"，找到Windows Subsystem for Linux(beta)勾选并确定，系统会下载更新并提示重启机器。
5. 重启完成后，打开命令提示符，输入 `bash` 命令并回车，会提示继续安装 `Bash On Windows`，选择yes，系统会安装有关的系统文件，等待安装完成后，系统会切换到Shell模式。

安装完成后，可以在Windows菜单中找到一个名为"Bash on Ubuntu on Windows"的Ubuntu系统图标，它就是Shell的启动入口。另外，在命令提示符下输入 `bash`，也可以快速切换到Shell环境。

在“Bash on Ubuntu on Windows”的Shell环境中，可以执行与 Ubuntu系统上一样的 `sudo apt-get install` 命令来安装软件包，效果如图所示。



```
android@DESKTOP-0E964KM: /
android@DESKTOP-0E964KM:/$ sudo apt-get install git openssl tree cmake unzip bison bc
sudo: unable to resolve host DESKTOP-0E964KM
Reading package lists... Done
Building dependency tree
Reading state information... Done
bc is already the newest version.
tree is already the newest version.
git is already the newest version.
openssl is already the newest version.
The following package was automatically installed and is no longer required:
  os-prober
Use 'apt-get autoremove' to remove it.
The following extra packages will be installed:
  cmake-data libarchive13 libbison-dev liblzo2-2 libnettle4 make
Suggested packages:
  bison-doc codeblocks eclipse lrzip make-doc
The following NEW packages will be installed:
  bison cmake cmake-data libarchive13 libbison-dev liblzo2-2 libnettle4 make
  unzip
0 to upgrade, 9 to newly install, 0 to remove and 17 not to upgrade.
Need to get 4,575 kB of archives.
After this operation, 20.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

编译Android源码

`Docker` 在Windows上运行的最低版本是Windows 7，需要借助 `Docker Toolbox` 在 `VirtualBox` 虚拟机中的容器进行通信，效率上相对较低。Windows 10版本是原生支持 `Docker`，因此，推荐读者使用Windows 10来编译Android系统源码。

首先到 `Docker` 官网<https://www.docker.com> 下载最新的Windows版本的 `Docker Toolbox`，目前稳定版的地址为<https://download.docker.com/win/stable/DockerToolbox.exe>。下载完成后，双击运行安装。

安装完成后，桌面上同样会有 `Docker` 与 `Kitematic` 等软件的图标。在Windows系统上使用它们来编译Android系统源码，与macOS上的操作是一样的，此处不再赘述。

小结

本篇主要介绍了在Windows平台上，如何搭建安卓的开发与分析环境，以及如何在Windows系统上编译安卓系统的源码。

更多精彩内容，欢迎关注微信公众号【feicong_sec】

