

第 2 章 Android 开发环境

2.1 如何配置 Android 开发环境

Google 公司推荐的 Android 开发环境是 Elipse+ADT。除了这两个工具外，还必须安装一些其他的 SDK，并在 Eclipse 中进行设置。建议读者尽可能使用最新版本的 SDK 和开发工具。搭建 Android 开发环境必须的工具和 SDK 如下：

- JDK (java development kit)
- Eclipse
- Android SDK
- ADT
- Android NDK

除了上述的工具外，我们还需要掌握什么呢？是 Linux？尽管 Android 采用 Linux 作为操作系统的内核，但是基于 Android SDK 上的应用全部采用 Java 语言来开发的，并运行在 Dalvik 虚拟机中。所以熟练地掌握 Java 语言是开发 Android 应用的基础。

2.2 支持 Android 开发的操作系统

支持 Android 开发的操作系统有：

- Windows XP 或 Vista
- Mac OS X 10.4.8 or later (x86 only)
- Linux (tested on Linux Ubuntu Dapper Drake)

2.3 安装 Java 开发包 (JDK)

JDK 原来是 SUN 公司开发的 Java 运行和开发环境，现在属于 Oracle 公司。读者可以在 <http://www.oracle.com/technetwork/java/javase/downloads/index.html> 下载 JDK 的最新版本，如图 2-1、图 2-2 所示。版本包括：JDK 5 和 JDK6。与此同时我们还必须下载最新版本的 JDK Standard Edition。接下来按照安装向导进行操作即可。

对于图 2-2，选择左上角的“Accept License Agreement”单选按钮即可进入下载选择界面，选择下载相应的平台上的 JDK。对于 Java SE 6 来说，并未直接提供 Mac OS X 下载的安装包，因此使用 Mac OS X 进行 Android 开发的读者需要使用 Mac OS X 本身的更新来安装 JDK。但在 Java SE 7 的下载页面中直接提供了 Mac OS X 平台的 JDK 安装包。读者也可直接下载该文件安装即可。



图 2-1 JDK 下载界面

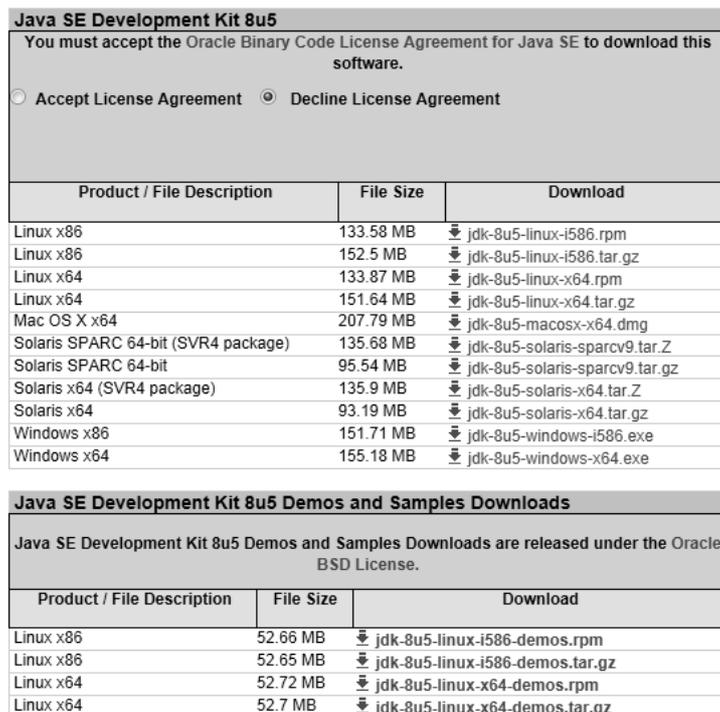


图 2-2 选择界面

2.4 安装配置 Eclipse 开发环境

多数的开发人员使用当下流行的 Eclipse 集成开发环境进行 Android 的开发，Eclipse 是一款开源的集成开发环境，它能够极大的提高开发应用的效率。最重要的是它提供了丰富的插件来帮助我们开发 Android 应用。我们可以到 <http://www.eclipse.org/downloads> 下载 Eclipse 的最

新版本。完成 JDK 的安装和配置后，只要直接将 Eclipse 压缩包解压，并执行 eclipse.exe 文件就可以运行 Eclipse 了。二者可以在 Windows、Mac 和 Linux 操作下使用。

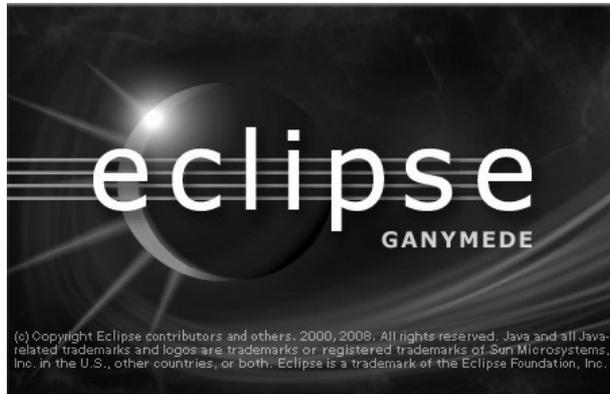


图 2-3 Eclipse 主界面

在不同的操作系统下，Eclipse 的安装有不同的需要，例如在 Windows 操作系统下完整安装 Eclipse 环境需要大约 400MB 磁盘空间，而其压缩需要 175MB。

为了使 Eclipse 更符合自己的要求，还可以对其进行一些配置：

- 改变默认的 Java 编辑器字体
- 显示行号
- 修改 Java 的默认代码格式
- 使 Java 编辑器更智能

2.5 其他开发环境

JetBrains IntelliJ Idea9.0.4 是一款综合的 Java 编程环境，被许多开发人员和行业专家誉为市场上最好的 IDE。它提供了一系列最实用的工具组合：智能编码辅助和自动控制，支持 J2EE，Ant，JUnit 和 CVS 集成，非平行的编码检查和创新的 GUI 设计器。IDEA 把 Java 开发人员从一些耗时的常规工作中解放出来，显著地提高了开发效率。具有运行更快速，生成更好的代码；持续的重新设计和日常编码变得更加简易，与其它工具的完美集成；很高的性价比等特点。在 4.0 版本中支持 Generics，BEA WebLogic 集成，改良的 CVS 集成以及 GUI 设计器。但是仅有 Eclipse 与 Android SDK 练习最为紧密，我们所选择的开发环境必须完全的支持 JDK5 或 JDK6。

2.6 安装 Android 软件开发包（SDK）

Android SDK 是在线安装的（下载地址<http://www.android.com>），进入网站后会显示如图 2-4 的界面，我们可以看到 Android 的最新版本并可单击了解。单击 Android SDK 选项，弹出图 2-5 的界面，直接单击下载 SDK（默认的是 Windows 32 下的开发包），再单击按钮的同时我们也将 ADT 下载完成，ADT（Android develop tools）它是主要针对 Android 开发的插件，即打包好的工具集。



图 2-4 网站界面

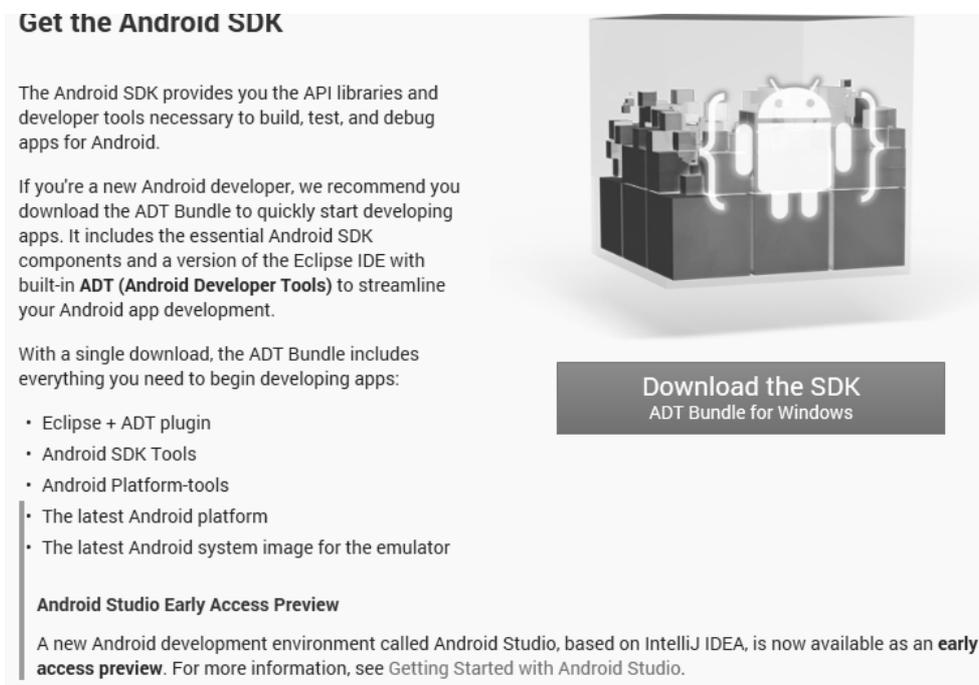


图 2-5 下载界面

在图 2-4 下载界面中，我们需要注意最下面的一段话，介绍最新版本的开发环境叫做 Android Studio，它不是基于 Linux 的，而是基于 IntelliJ IDEA。但现在由于代码的格式化、预览的支持还不是特别的稳定，所以还没有普及，可以试用一下。安装 SDK 时我们可以看到所有从网上直接下载的安装包实际上是一个空壳，下载后在 Android SDK 安装目录有一个 SDK Manager.exe 文件，它用来帮助我们下载不同版本的 SDK。我们需要哪个版本直接选择单击下载即可，界面如图 2-4 所示。

界面主要包括三部分：一部分是从 Google 公司的官网上获取 Android SDK 目前支持的 Android 版本安装包列表，另一部分 Tools 是 SDK 所需要开发的工具，还有 Extras 是可用扩展工具因此在启动该程序之前需要有快速和稳定的 Internet 链接。读者可以从这个列表中选择相

应的 Android 版本。然后单击界面右下方的 Install packages 按钮进行安装，安装过程仍然需要链接 Internet，如图 2-5 所示。

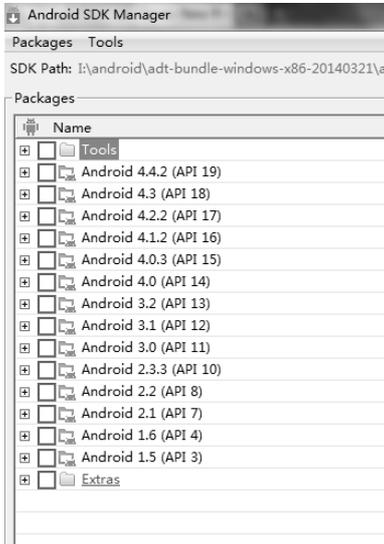


图 2-6 SDK 主界面

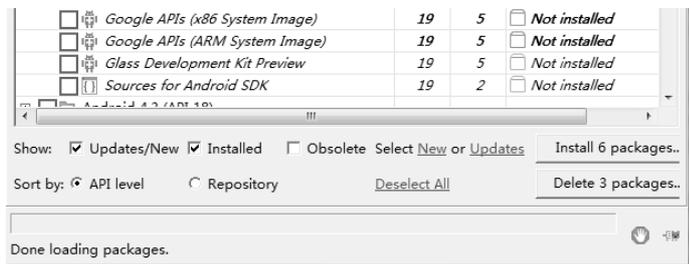


图 2-7 选择安装界面

然后选中 Accept，单击 Install 安装。最后选择安装一个 Android 版本。这里需要读者注意的是，在下载过程中，由于我国的国情，Google 的网站链接是间断性的，所以会导致我们的下载失败，为了避免这样的事情发生，我们需要打开 SDKManager，单击 Tools→options，会出现如图 2-8 所示的界面，我们只需选中 Force http://...sources to be fetched using http://...这样我们选中的资源就可以下载了。

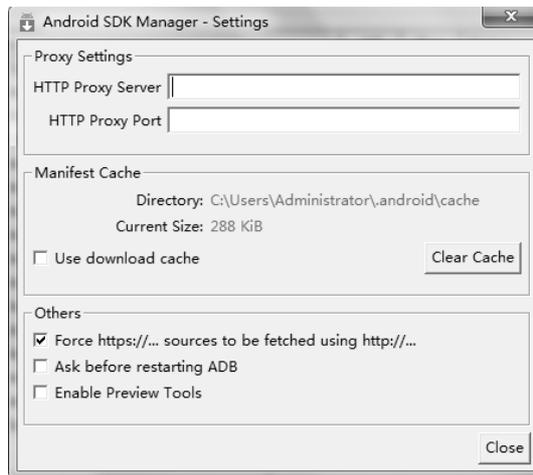


图 2-8 option 界面

若需安装的 Package 较多，则在线安装时间会比较长。为避免浪费更多的时间，读者可以从其他的机器上复制已经安装好的 Android SDK 到自己的机器上。建议在安装完本机器后可将其备份，以备不时之需。

2.7 安装与配置 Android Eclipse 插件 (ADT)

ADT 是 Google 为 Android 开发者提供的 Eclipse 插件。我们可以从下面的网站地址获取在线安装 ADT 的 URL 或离线安装 ADT 的安装包下载地址。

<http://androidappdocs.appspot.com/sdk/eclipse-adt.html>

安装步骤如下：

如果在线安装 ADT，需要单击 help 按钮，如图 2-9 所示。然后单击 Install New Software 菜单项，打开 Install newsoftware 对话框，如图 2-10 所示。

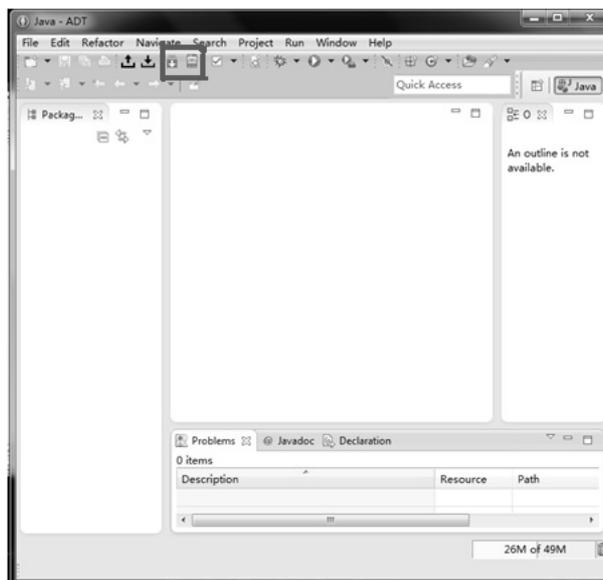


图 2-9 Java-ADT 界面

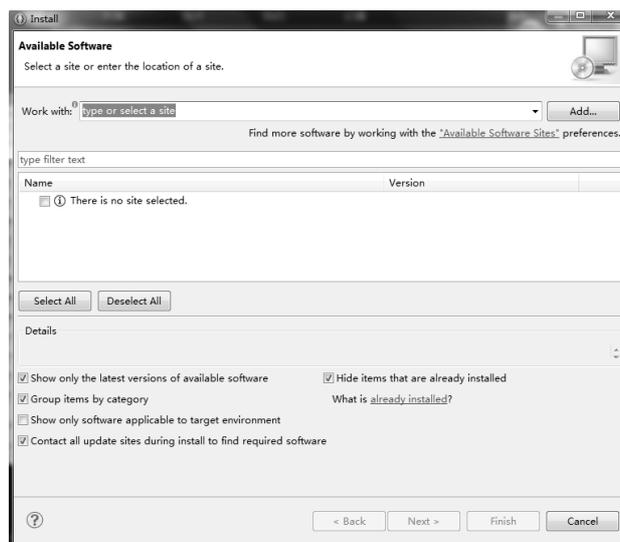


图 2-10 Install 界面

单击右侧的 Add 按钮弹出 Add Repository 对话框，如图 2-11 所示。

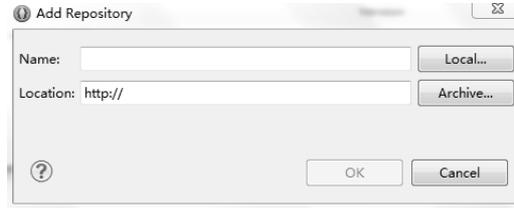


图 2-11 Add Repository

在 Name 文本框中输入 ADT（或其他与系统已有名字不重复的），在 Location 文本框中输入如下网站地址：[Http://dl-ssl.google.com/android/eclipse/](http://dl-ssl.google.com/android/eclipse/)。然后单击 OK 按钮关闭 Add Repository 对话框，就会在 Install 对话框（如图 2-12 所示）显示 ADT 包含的安装列表，展开后如图 2-13 所示。

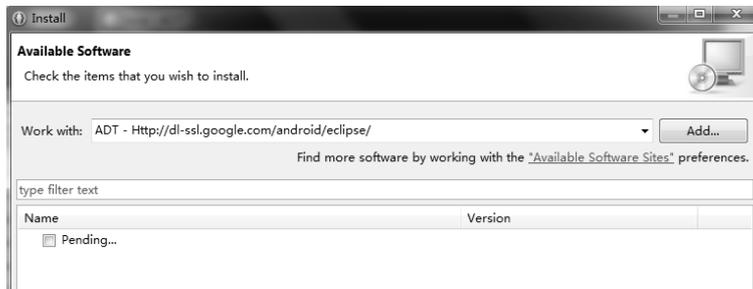


图 2-12 ADT 安装列表

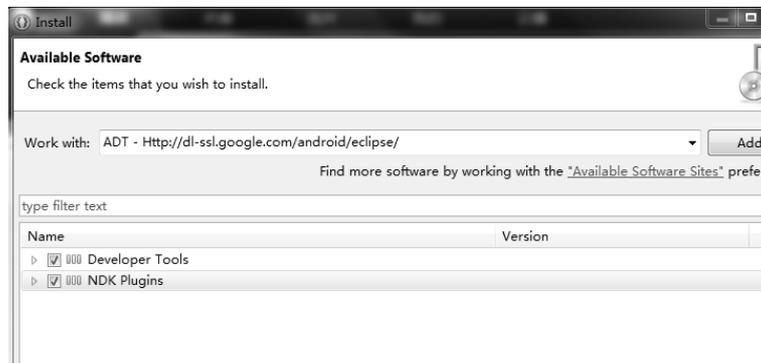


图 2-13 ADT 安装

选中后然后单击 Next 按钮，接下来按提示进行安装即可。

2.8 创建模拟器 AVD

这里我们需要注意的是，我们安装的 Java-ADT 与标准的 Eclipse 没有太大的区别，唯一不同的就是多了两个按钮（在图 2-9 中用边框圈出）。左面的是 SDK Manager，其实与我们所下载的文件夹中所带的是一样的。右面的按钮是 Android 虚拟设备管理器。虽然 Android 设备多种多样，但是它的分辨率是有要求的，我们可以单击设备虚拟管理器进行查看，如图 2-14 所示。

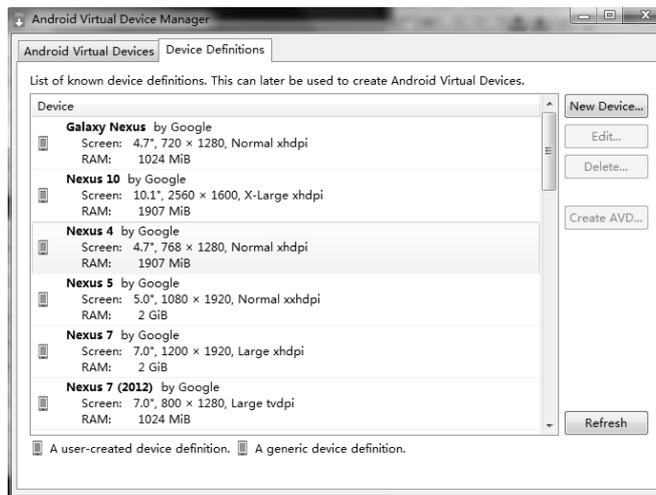


图 2-14 虚拟设备管理器

我们需要了解以下一些市面上主流的分辨率，例如：

- VGA: 480*640
- QVGA: 240*320
- HVGA: 320*480
- WVGA: 480*800
- FWVGA: 480*854

Android 的发展趋势是屏幕越做越大，但是当我们创建一个设备的时候，需要考虑以下电脑的配置问题，配置比较低时建议大家选择分辨率较低的模拟器，否则运行速度会非常的慢。这里我们以 HVGA 为例创建，选择分辨率后，按 Create AVD 会弹出图 2-15 的空白界面，填写的信息和选择的属性如图所示，然后单击 OK 键，模拟设备创建成功，回到 Device Manager 界面，我们发现在 Android virtual devices 中多了一个 iphon 设备如图 2-16 所示。

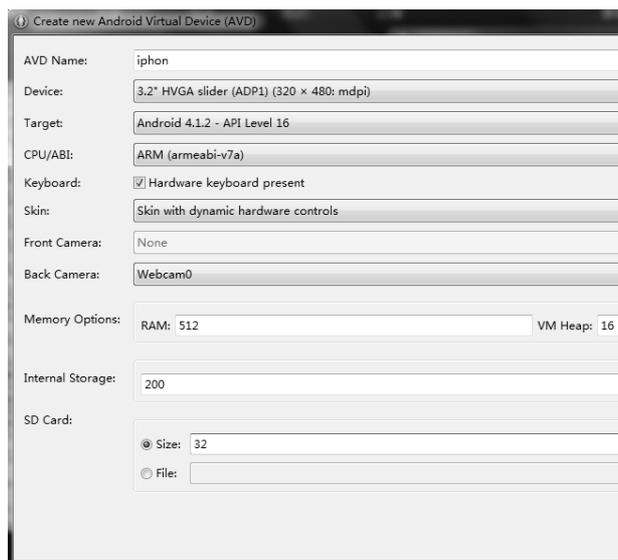


图 2-15 创建模拟设备

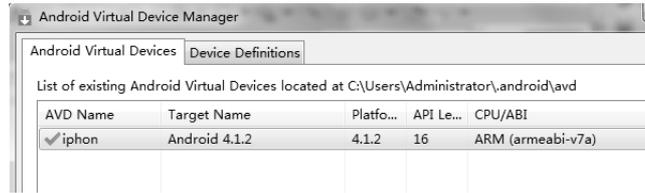


图 2-16 iphon

开启设备后，弹出如图 2-17 的界面，这里有一个 wipe user data 的选项，建议读者不要选择，因为它会将手机里的数据清空。单击 Launch 得到如图 2-18 的效果图。

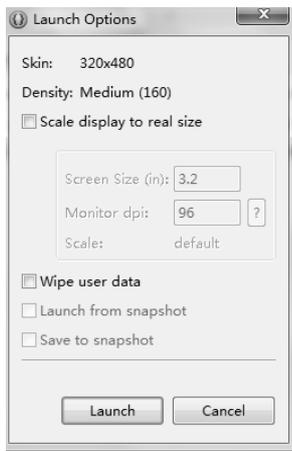


图 2-17 Launch Options



图 2-18 模拟器界面

由于我们选用的是 ARM。所以开机时间比较长，大约 1~2 分钟，开机后，我们可以看到这个模拟器与我们的手机不仅界面是一样的，所实现的功能也是相同的，如图 2-19 所示。到此我们的模拟设备创建完成。

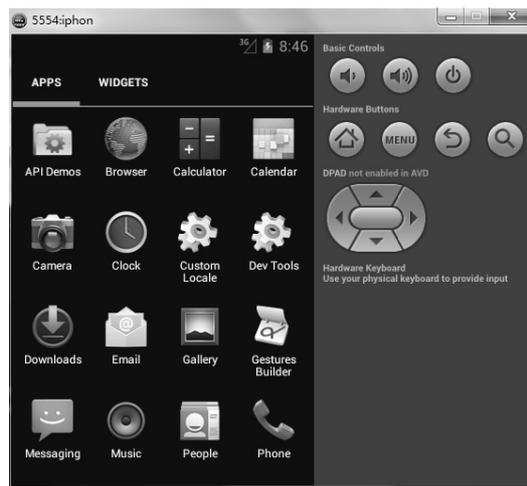


图 2-19 Menu 界面

在图 2-18 所示的左上角出现的 5554:iphon 代表的是当前模拟器的端口号。介绍完模拟器的创建。现在我们回到 Java-ADT 界面，在右侧一般会出现 DDMOS 的按钮，它的界面如图

2-20 所示。如果你安装的 ADT 中默认没有 DDMS，可以单击图中右侧绿色标记的按钮，选择 DDMS 添加即可。DDMS 由 2 个界面组成，每个界面又包括一下的一些属性：

- Name: 当前 Eclipse 上所开启的模拟器。
- Threads: 用来监视进程里面的线程。
- Heap: 观察应用程序的堆栈信息。
- Emulator Control: 对模拟器进行一些设置，例如打电话、发短信和位置发送等。
- Network Statistics: 网络连接状况。
- File Explorer: 文件管理器，主要显示 Android。
- System information: 显示系统信息。

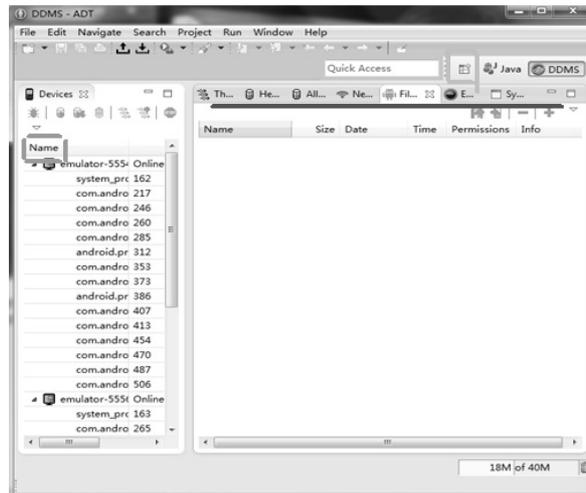


图 2-20 DDMS 界面

下面我们对 Emulator Control 进行一下演示它的界面如图 2-21 所示，speed 是改变模拟器的速度，当选择不同时，模拟器信号处的字母也会不同。

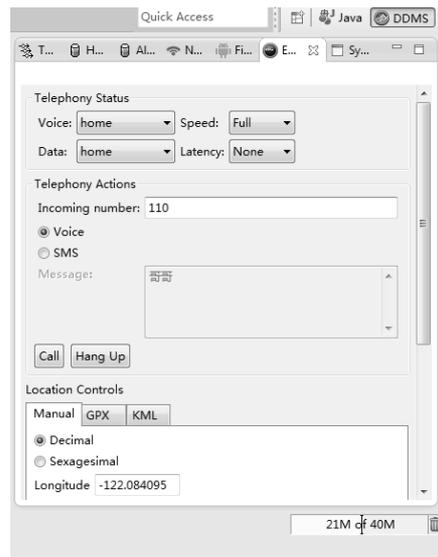


图 2-21 Emulator Control 界面

当我们选择 `iphon` 时，在号码处输入 110，按 `call` 键。`iphon` 的模拟器显示如图 2-22 所示，`Hang Up` 结束通话。我们也可建立多个模拟器，实现它们之间的通讯功能，每个模拟器的电话号码就是该模拟器的端口号。在实现短信功能时，需要注意的是文本输入不支持中文。

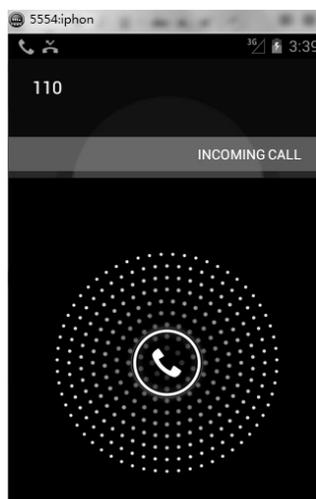


图 2-22 电话功能效果图

小结

1. 搭建 Android 开发环境必须的工具和 SDK 如下：

- JDK (java development kit)
- Eclipse
- Android SDK
- ADT
- Android NDK

2. 支持 android 开发的操作系统有：

- Windows XP 或 Vista
- Mac OS X 10.4.8 or later (x86 only)
- Linux (tested on Linux Ubuntu Dapper Drake)

3. 对于 Android 初学者来说，搭建 Android 开发和测试环境是必须要经历的一关，因为只有真实的环境中才能更好地理解和使用环境中的各项工具，并且积累各种开发技巧。若想更好地理解本章知识，需要掌握一些 Java 知识，当你理解本章的内容后，就说明已经踏进了 Android 的大门，接下来将会介绍大量的示例和知识。