Using CentOS to build the XMLFoundation HTTP server for Android

This was written on December 4, 2013 using current versions of all the necessary components on 64 bit CentOS.

Updated August 2015

Download and decompress the following:

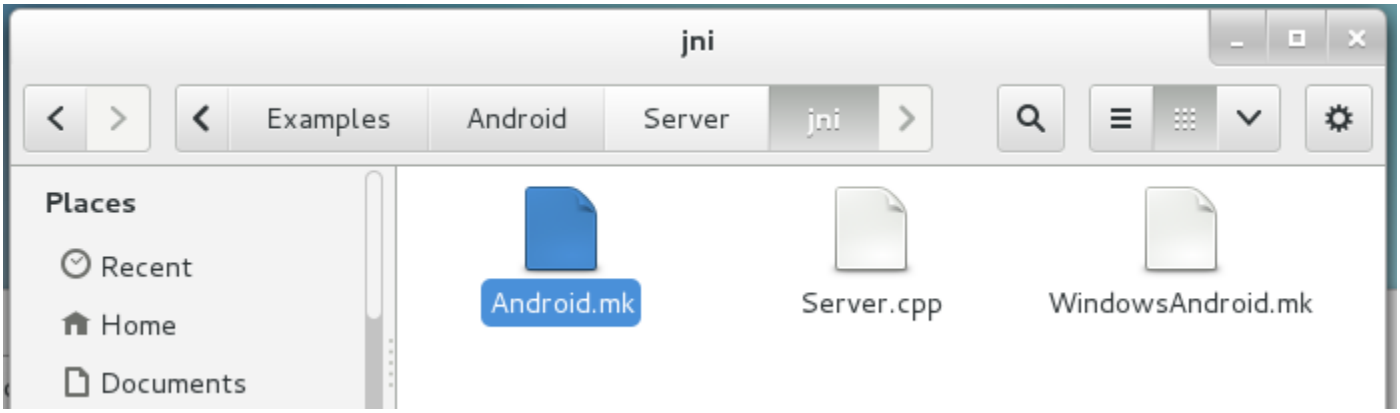
Java Development Kit (JDK): http://www.oracle.com/technetwork/java/javase/downloads/index.html

Android NDK : http://developer.android.com/tools/sdk/ndk/index.html

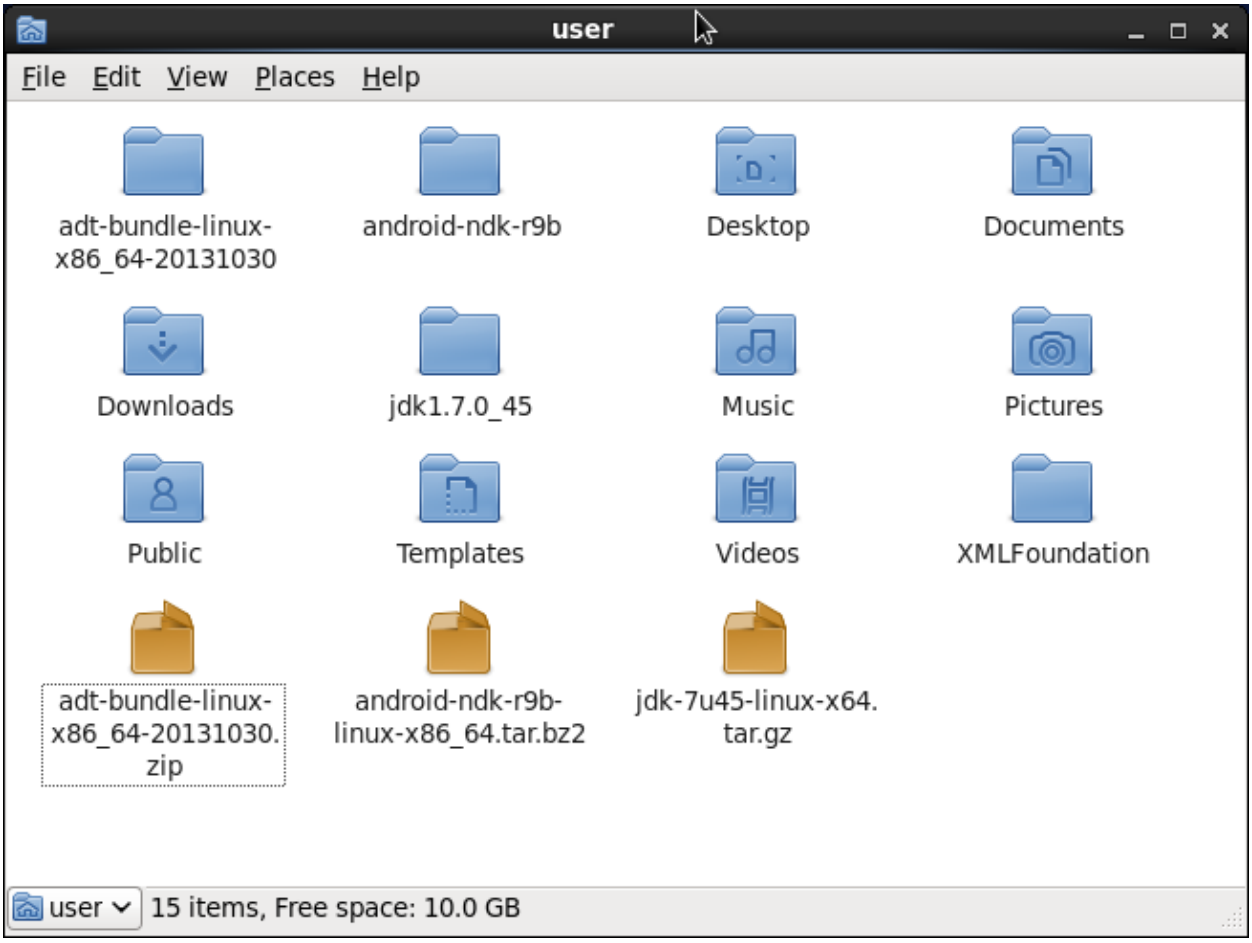
Android SDK : http://developer.android.com/sdk/index.html

Rename LinuxAndroid.mk to Android.mk by doing this:

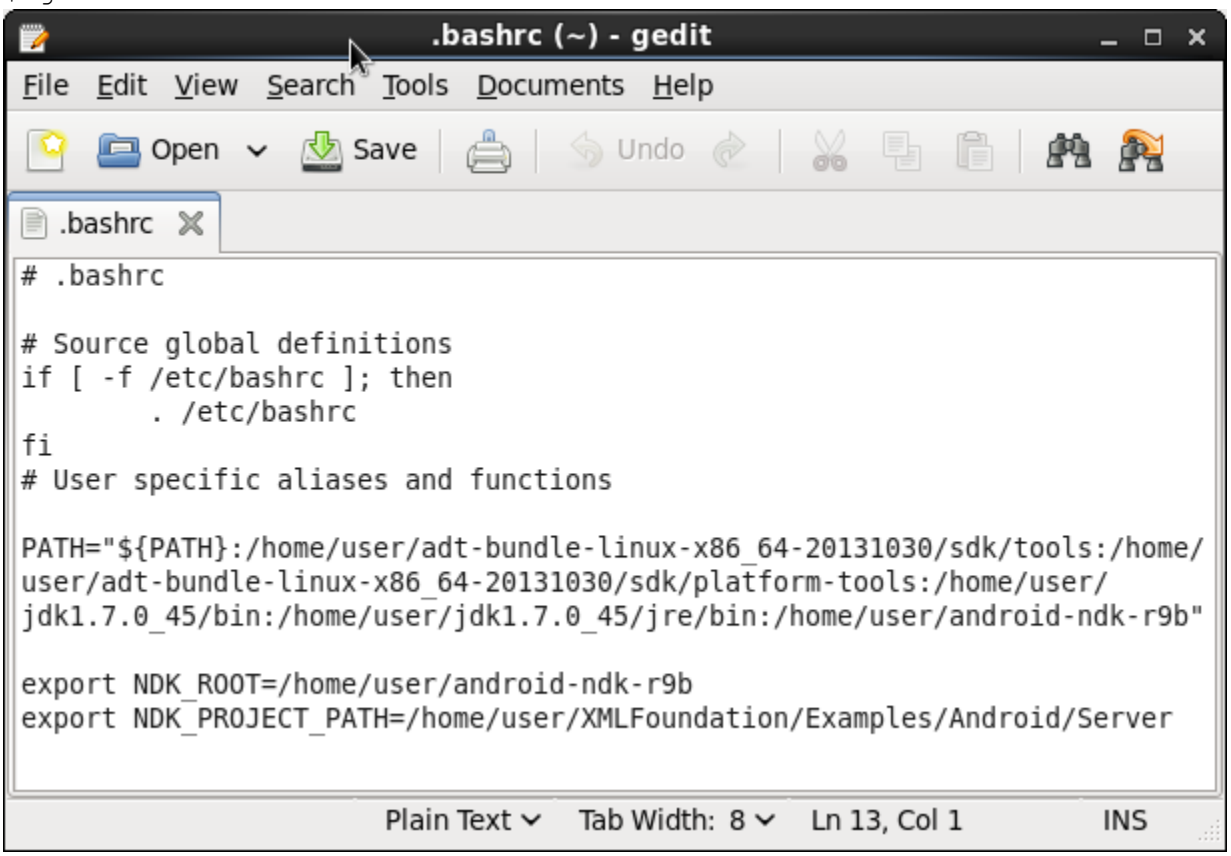
```
cd XMLFoundation/Examples/Android/Server/jni
cp Android.mk WindowsAndroid.mk
rm Android.mk
cp LinuxAndroid.mk Android.mk
```



Here you can see the downloads and the equivalent directories that they have been decompressed to. In gedit (next page) - you see how the environment must be setup for this directory configuration.




```
$ gedit .bashrc
```



This lists the individual PATH entries that MUST appear in the PATH=

```
/home/user/adt-bundle-linux-x86_64-20131030/sdk/tools
/home/user/adt-bundle-linux-x86_64-20131030/sdk/platform-tools
/home/user/jdk1.7.0_45/bin
/home/user/jdk1.7.0_45/jre/bin
/home/user/android-ndk-r9b
```

And make sure you export the two NDK_ variables

You must also install the 32 bit runtime (for ADT not for XMLFoundation)

or Eclipse will fail and give a false error that it cannot find /sdk/build-tools/android-4.3/aapt or /sdk/build-tools/android-4.3/adb

```
# yum install glibc.i686
# yum install zlib.i686
# yum install libstdc++
# yum install libstdc++.so.6
```

Additioanlly you need OpenGL for the Android emulator to work.

```
# yum install mesa-libGLU-devel
```

Mesa-libGLU-devel is likely the only thing you NEED to get the emulator working – but I added Mesa-libGLU-devel at the same time as these for some GUI work I am doing.

```
# yum install freeglut freeglut-devel libX11-devel
```

Reboot the machine, or restart the graphical shell to pick up the new environment settings.

Start a terminal window and type

```
# ndk-build
```

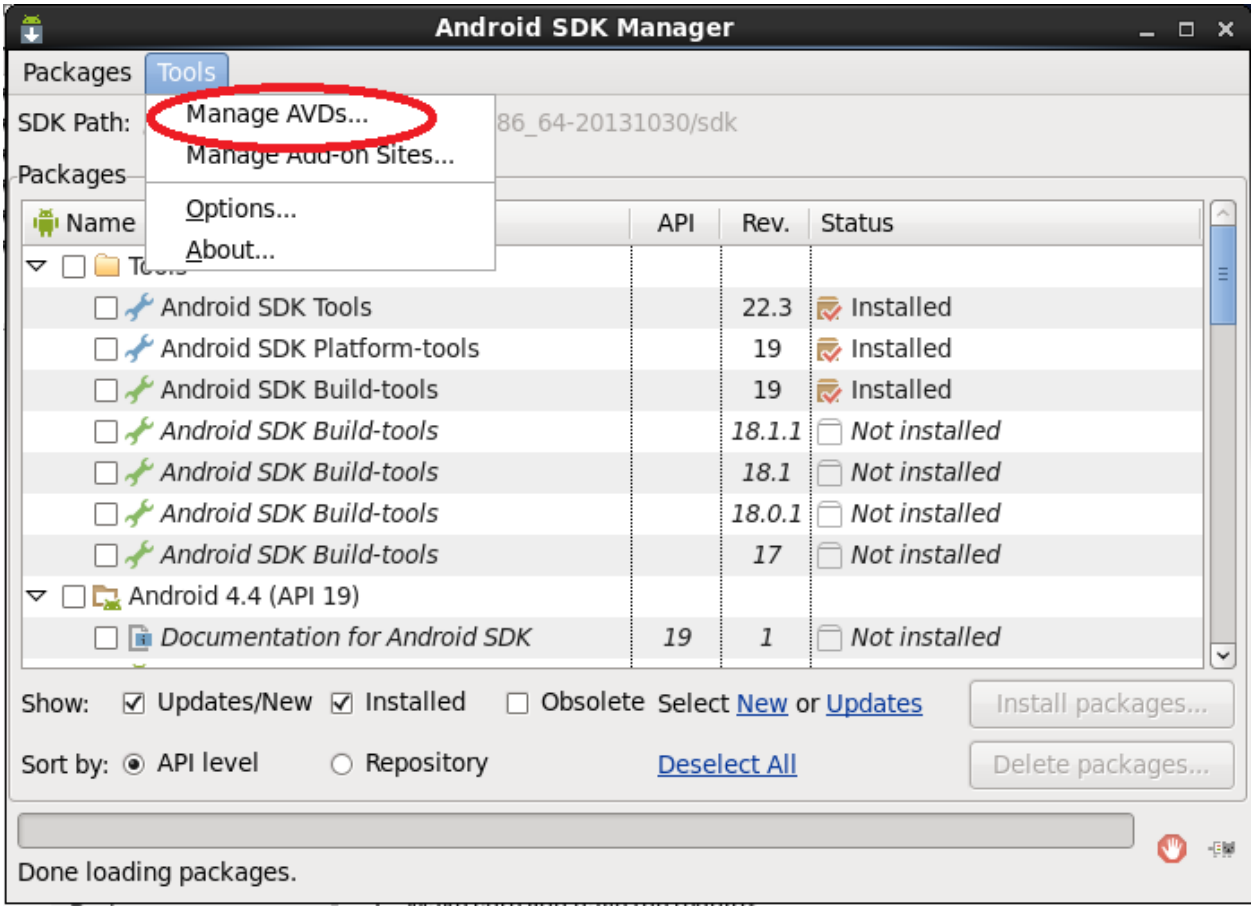
The following page contains the output you will see when the XMLFoundation library is built.

```
user@localhost:~
File  Edit  View  Search  Terminal  Help
[user@localhost ~]$ pwd
/home/user
[user@localhost ~]$ ndk-build
[armeabi] Compile++ thumb: Server <= Server.cpp
[armeabi] Compile++ thumb: Server <= ServerCore.cpp
[armeabi] Compile++ thumb: Server <= GSocketHelpers.cpp
[armeabi] Compile++ thumb: Server <= GArray.cpp
[armeabi] Compile++ thumb: Server <= GBTree.cpp
[armeabi] Compile++ thumb: Server <= GException.cpp
[armeabi] Compile++ thumb: Server <= GHash.cpp
[armeabi] Compile++ thumb: Server <= GList.cpp
[armeabi] Compile++ thumb: Server <= GProfile.cpp
[armeabi] Compile++ thumb: Server <= GStack.cpp
[armeabi] Compile++ thumb: Server <= GString.cpp
[armeabi] Compile++ thumb: Server <= GStringList.cpp
[armeabi] Compile++ thumb: Server <= GDirectory.cpp
[armeabi] Compile++ thumb: Server <= GPerformanceTimer.cpp
[armeabi] Compile++ thumb: Server <= TwoFish.cpp
[armeabi] Compile++ thumb: Server <= SHA256.cpp
[armeabi] Compile++ thumb: Server <= BZip.cpp
[armeabi] Compile++ thumb: Server <= GZip.cpp
[armeabi] Compile++ thumb: Server <= Base64.cpp
[armeabi] Compile++ thumb: Server <= PluginBuilderLowLevelStatic.cpp
[armeabi] Compile++ thumb: Server <= GHTTPMultiPartPOST.cpp
[armeabi] Compile++ thumb: Server <= AttributeList.cpp
[armeabi] Compile++ thumb: Server <= CacheManager.cpp
```

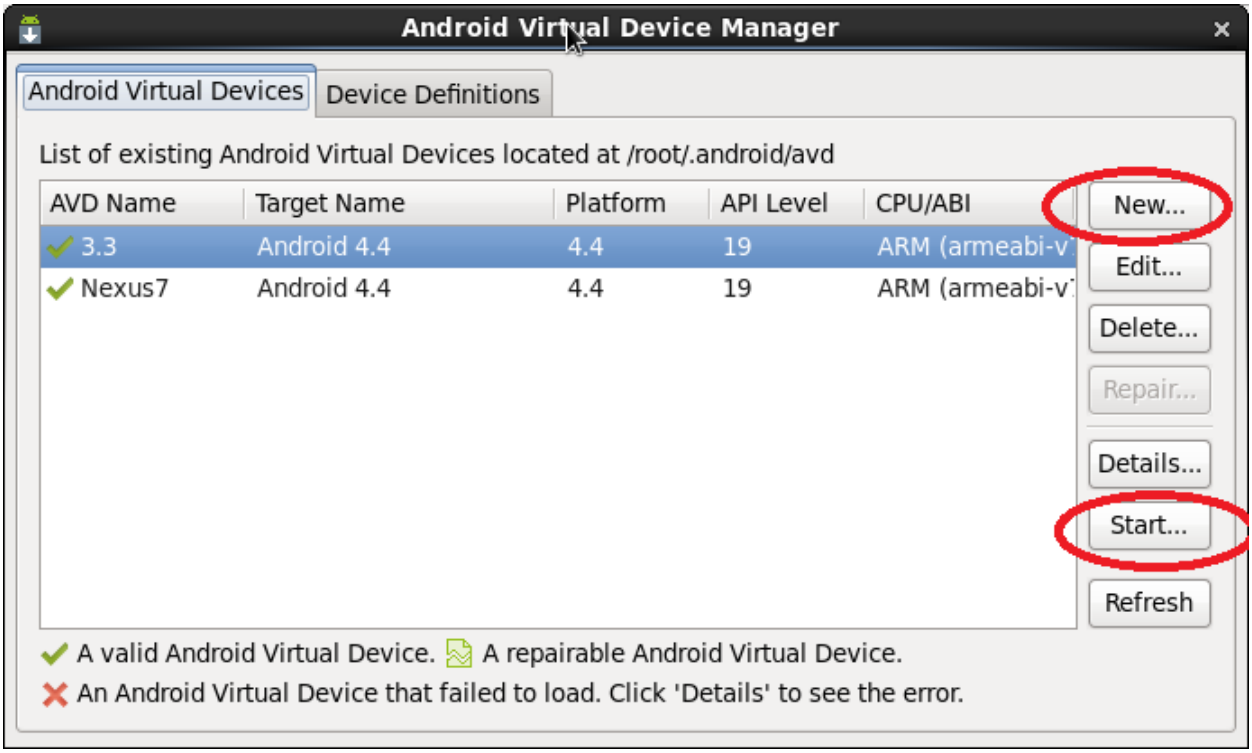
Setup an emulator definition. At the prompt type “android”. To start the SDK manager.

[root@localhost user]# android

Then “Manage AVDs…” as shown in the next image



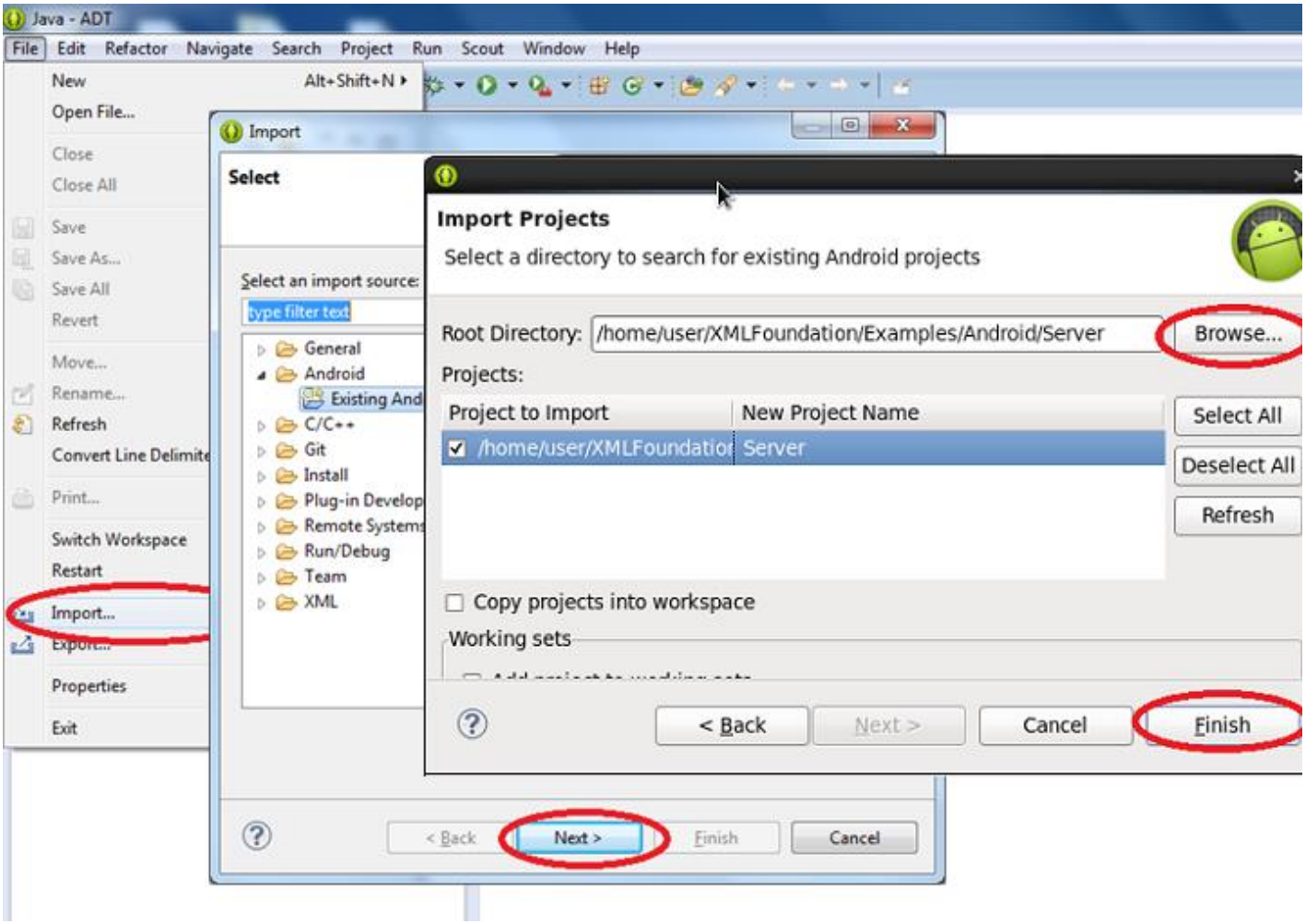
Create a new definition - then start it.



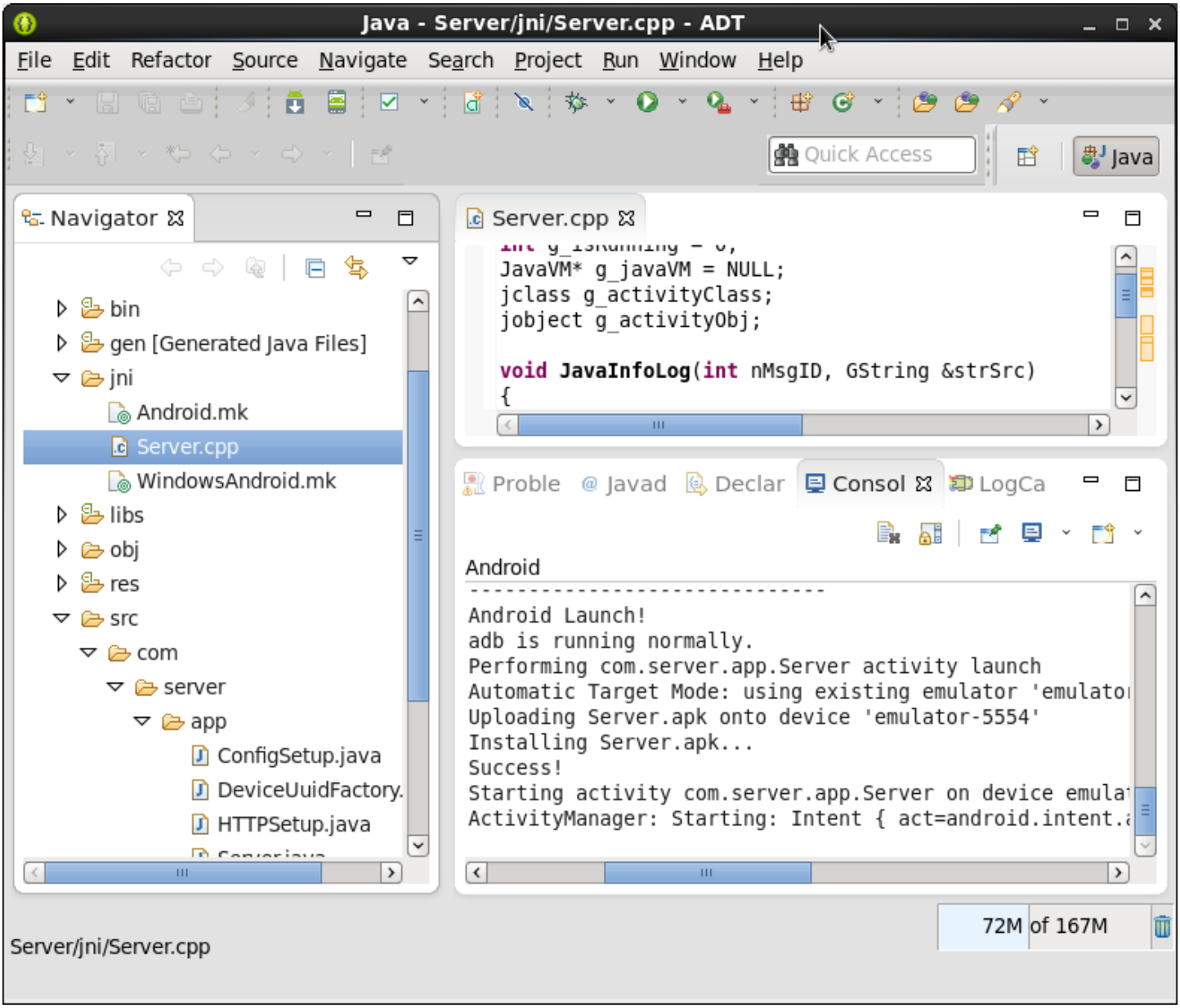
Start ADT (You must start it as root)

```
user@localhost:/home/user/adt-bundle-linux-x86_64-20131
File  Edit  View  Search  Terminal  Help
[user@localhost ~]$ su
Password:
[root@localhost user]# cd adt*
[root@localhost adt-bundle-linux-x86_64-20131030]# cd ec*
[root@localhost eclipse]# ./eclipse
```

and import the Android example from the XMLFoundation:



Open “Window..Navigator”, “Run” the project


Java - Server/jni/Server.cpp - ADT
File Edit Refactor Source Navigate Search Project Run Window Help
Quick Access
Java
Navigator
bin
gen [Generated Java Files]
jni
Android.mk
Server.cpp
WindowsAndroid.mk
libs
obj
res
src
com
server
app
ConfigSetup.java
DeviceUuidFactory.
HTTPSetup.java
Server.cpp
JavaVM* g_javaVM = NULL;
jclass g_activityClass;
jobject g_activityObj;
void JavaInfoLog(int nMsgID, GString &strSrc)
{
Proble
Javad
Declar
Consol
LogCa
Android
Android Launch!
adb is running normally.
Performing com.server.app.Server activity launch
Automatic Target Mode: using existing emulator 'emulato
Uploading Server.apk onto device 'emulator-5554'
Installing Server.apk...
Success!
Starting activity com.server.app.Server on device emula
ActivityManager: Starting: Intent { act=android.intent.
Server/jni/Server.cpp
72M of 167M

When the app starts on the phone, press Start.