



The Android Emulator

The emulator is the perfect tool for testing and debugging your applications, particularly if you don't have a real device (or don't want to risk it) for experimentation.

The emulator is an implementation of the Dalvik virtual machine, making it as valid a platform for running Android applications as any Android phone. Because it's decoupled from any particular hardware, it's an excellent baseline to use for testing your applications.

A number of alternative user interfaces are available to represent different hardware configurations, each with different screen sizes, resolutions, orientations, and hardware features to simulate a variety of mobile device types.

Full network connectivity is provided along with the ability to tweak the Internet connection speed and latency while debugging your applications. You can also simulate placing and receiving voice calls and SMS messages.

The ADT plug-in integrates the emulator into Eclipse so that it's launched automatically when you run or debug your projects. If you aren't using the plug-in or want to use the emulator outside of Eclipse,

you can telnet into the emulator and control it from its console. For more details on controlling the emulator, check the documentation at <http://code.google.com/android/reference/emulator.html>.

At this stage, the emulator doesn't implement all the mobile hardware features supported by Android, including the camera, vibration, LEDs, actual phone calls, the accelerometer, USB connections, Bluetooth, audio capture, battery charge level, and SD card insertion/ejection.

Dalvik Debug Monitor Service (DDMS)

The emulator lets you see how your application will look, behave, and interact, but to really see what's happening under the surface, you need the DDMS. The Dalvik Debug Monitoring Service is a powerful debugging tool that lets you interrogate active processes, view the stack and heap, watch and pause active threads, and explore the filesystem of any active emulator.

The DDMS perspective in Eclipse also provides simplified access to screen captures of the emulator and the logs generated by LogCat.

If you're using the ADT plug-in, the DDMS is fully integrated into Eclipse and is available from the DDMS perspective. If you aren't using the plug-in or Eclipse, you can run DDMS from the command line, and it will automatically connect to any emulator that's running.

The Android Debug Bridge (ADB)

The *Android debug bridge (ADB)* is a client-service application that lets you connect with an Android Emulator or device. It's made up of three components: a daemon running on the emulator, a service that runs on your development hardware, and client applications (like the DDMS) that communicate with the daemon through the service.

As a communications conduit between your development hardware and the Android device/emulator, the ADB lets you install applications, push and pull files, and run shell commands on the target device.

Using the device shell, you can change logging settings, and query or modify SQLite databases available on the device.

The ADT tool automates and simplifies a lot of the usual interaction with the ADB, including application installation and update, log files, and file transfer (through the DDMS perspective).

To learn more about what you can do with the ADB, check out the documentation at

<http://code.google.com/android/reference/adb.html>.