The Dalvik Virtual Machine

One of the key elements of Android is the Dalvik virtual machine. Rather than use a traditional Java virtual machine (VM) such as Java ME (Java Mobile Edition), Android uses its own custom VM designed to ensure that multiple instances run effi ciently on a single device.

The Dalvik VM uses the device's underlying Linux kernel to handle low-level functionality including security, threading, and process and memory management. It's also possible to write C/C++ applications that run directly on the underlying Linux OS. While you *can* do this, in most cases there's no reason you should need to.

This book focuses exclusively on writing applications that run within Dalvik. If your inclinations run toward exploring the Linux kernel and C/C++ underbelly of Android, modifying Dalvik, or otherwise tinkering with things under the hood, check out the Android Internals Google Group at http://groups.google.com/group/android-internals

All Android hardware and system service access is managed using Dalvik as a middle tier. By using a VM to host application execution, developers have an abstraction layer that ensures they never have to worry about a particular hardware implementation.

The Dalvik VM executes Dalvik executable fi les, a format optimized to ensure minimal memory footprint. The .dex executables are created by transforming Java language compiled classes using the tools supplied within the SDK. You'll learn more about how to create Dalvik executables in the next chapter.