Recording in Progress

This class is being recorded
Please turn off your video and/or video if you do not wish to be recorded
CMSC436: Programming Handheld Systems
The Android Platform
The Android Platform

A software stack for mobile devices:
  - OS kernel and interfaces, system libraries, frameworks & key apps

Android SDK for creating apps
  - Libraries & development tools
  - Lots of documentation. Start browsing today!

See: http://developer.android.com/training
The Android Architecture

- System Apps
- Java API Framework
- Native C/C++ Libraries
- Hardware Abstraction Layer (HAL)
- Android Runtime
- Linux Kernel
The Android Architecture

- **System Apps**
- **Java API Framework**
- **Native C/C++ Libraries**
- **Android Runtime**
- **Hardware Abstraction Layer (HAL)**

- **Linux Kernel**
  - Drivers
    - Audio
    - Binder (IPC)
    - Display
Linux Kernel – Standard Services

Security
Memory & process management
File & network I/O
Device drivers
Linux Kernel - Android-Specific

Power management
Low memory killer
Interprocess communication (IPC)
The Android Architecture

- System Apps
- Java API Framework
- Native C/C++ Libraries
- Android Runtime
- Hardware Abstraction Layer (HAL)
- Linux Kernel
Hardware Abstraction Layer (HAL)

Provides standard interfaces between API framework and device hardware

Defines and interface for various hardware classes, such as Camera, Audio, Graphics, etc.

Android loads library modules for hardware components on demand
The Android Architecture

- System Apps
- Java API Framework
- Native C/C++ Libraries
- Android Runtime
- Hardware Abstraction Layer (HAL)
- Linux Kernel
## Libraries

<table>
<thead>
<tr>
<th>Bionic libc</th>
<th>Webkit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surface Manager</td>
<td>OpenGL</td>
</tr>
<tr>
<td>Media Framework</td>
<td>SQLite</td>
</tr>
<tr>
<td>FreeType</td>
<td>SSL</td>
</tr>
</tbody>
</table>
The Android Architecture

System Apps

Native C/C++ Libraries
- Webkit
- OpenMAX AL
- Libc
- Media Framework
- OpenGL ES
- ...

Android Runtime
- Android Runtime (ART)
- Core Libraries

Hardware Abstraction Layer (HAL)

Linux Kernel
Android Runtime

Two main components

Core Java libraries with some Java 8+ feature support
Android Runtime (ART)
Core Java Libraries

Basic java classes -- java.*, javax.*
App lifecycle, basic services -- android.*, androidx
Internet/Web services -- org. *
Unit testing -- junit.*
Java 8+ support

Android does not support all Java 8+ language features

Some supported features (in API level 24 or higher)
  - Lambda expressions
  - Method references
  - `java.util.function` and `java.util.stream`

See:
  - https://developer.android.com/studio/write/java8-support
Android Runtime (ART)

Since Android 5.0, apps are executed in a managed runtime environment (ART)

On older platforms, apps run in the Dalvik Virtual Machine
ART Design Goals

Designed for resource-constrained environments

- Slower CPU
- Less RAM
- Limited battery life
Major ART Features

Ahead-of-time (AOT) and just-in-time (JIT) compilation

Optimized garbage collection (GC)

API level 28+ conversion of an app package's Dalvik Executable format (DEX) files to more compact machine code

Better debugging support, including a dedicated sampling profiler, detailed diagnostic exceptions and crash reporting, and the ability to set watchpoints to monitor specific fields
Typical Workflow

App written in Java, Kotlin or C++
Compiled to Java bytecode files
Tool chain converts java bytecode files to a single dex-formatted bytecode file
Virtual machine executes bytecode file
ART

Compiles dex bytecode to native binary for improved runtime performance

Applies system-dependent optimizations at installation time, runtime, and in background

Results in faster execution at cost of larger executable
The Android Architecture

- **System Apps**
  - Java API Framework
  - Content Providers
  - View System

- **Native C/C++ Libraries**

- **Android Runtime**
  - Managers: Activity, Location, Package, Notification
  - Resource, Telephony, Window

- **Hardware Abstraction Layer (HAL)**

- **Linux Kernel**
Package Manager

Keeps track of app packages on device
Window Manager

Manages the windows comprising an app
Notification Bar

Main Window

Subwindow
View System

Provides common user interface elements
  e.g., icons, text entry boxes, buttons and more
Tabs
Resource Manager

Manages non-compiled resources

e.g., strings, graphics, & layout files
Activity Manager

Manages app lifecycle and navigation stack
ContentProvider

Inter-application data sharing
Location Manager

Provides location & movement information
User’s Location
Notification Manager

Place notification icons in the status bar area when important events occur
The Android Architecture

System Apps

Java API Framework

Native C/C++ Libraries

Hardware Abstraction Layer (HAL)

Android Runtime

Linux Kernel
Applications

Standard apps include:

- Home – main screen
- Contacts – contacts database
- Phone – dial phone numbers
- Browser – view web pages
- Email reader – compose & read email messages
Applications

Nothing special about these apps
You can substitute your own or 3rd party app for any of them
Next

Android Development Environment