Mobile Application Development

Introduction

What is Android?

Android

- Android is an operating system **based on the Linux kernel**, and designed primarily for **touchscreen** mobile devices such as smartphones and tablet computers.
- Initially developed by **Android, Inc**., which Google backed financially and later bought in 2005.
- The project responsible for developing the Android system is called the **Android Open Source Project (AOSP)** and is primarily lead by Google with **Open Handheld Alliance (OHA)**.
- **Android was unveiled in 2007**. The first publicly available smartphone running Android, the HTC Dream, was released on **October 22, 2008**.
- As of July 2013, **Google Play** store had over 1 million apps published, and **over 50 billion downloads**.
- Android surpassed a billion shipments of devices by reaching close to **1.16 billion end users in 2014**.

Building 44

Android Versions

- The relative number of devices accessing the Play Store recently and running a given version of the Android platform, as of February 2, 2015.

Version	Code Name	Release Date	API Level	Market Share
5.0	Lollipop	November 3, 2014	21	1.6%
4.4	Kit Kat	October 31, 2013	19	39.7%
4.3.x	Jelly Bean	July 24, 2013	18	6.3%
4.2.x		November 13, 2012	17	19.8%
4.1.x		July 9, 2012	16	18.4%
4.0.3 – 4.0.4	Ice Cream Sandwich	December 16, 2011	15	6.4%
2.3.3-2.3.7	Gingerbread	February 9, 2011	10	7.4%
2.2	Frovo	May 20, 2010	8	0.4%

Variations in Device Capabilities

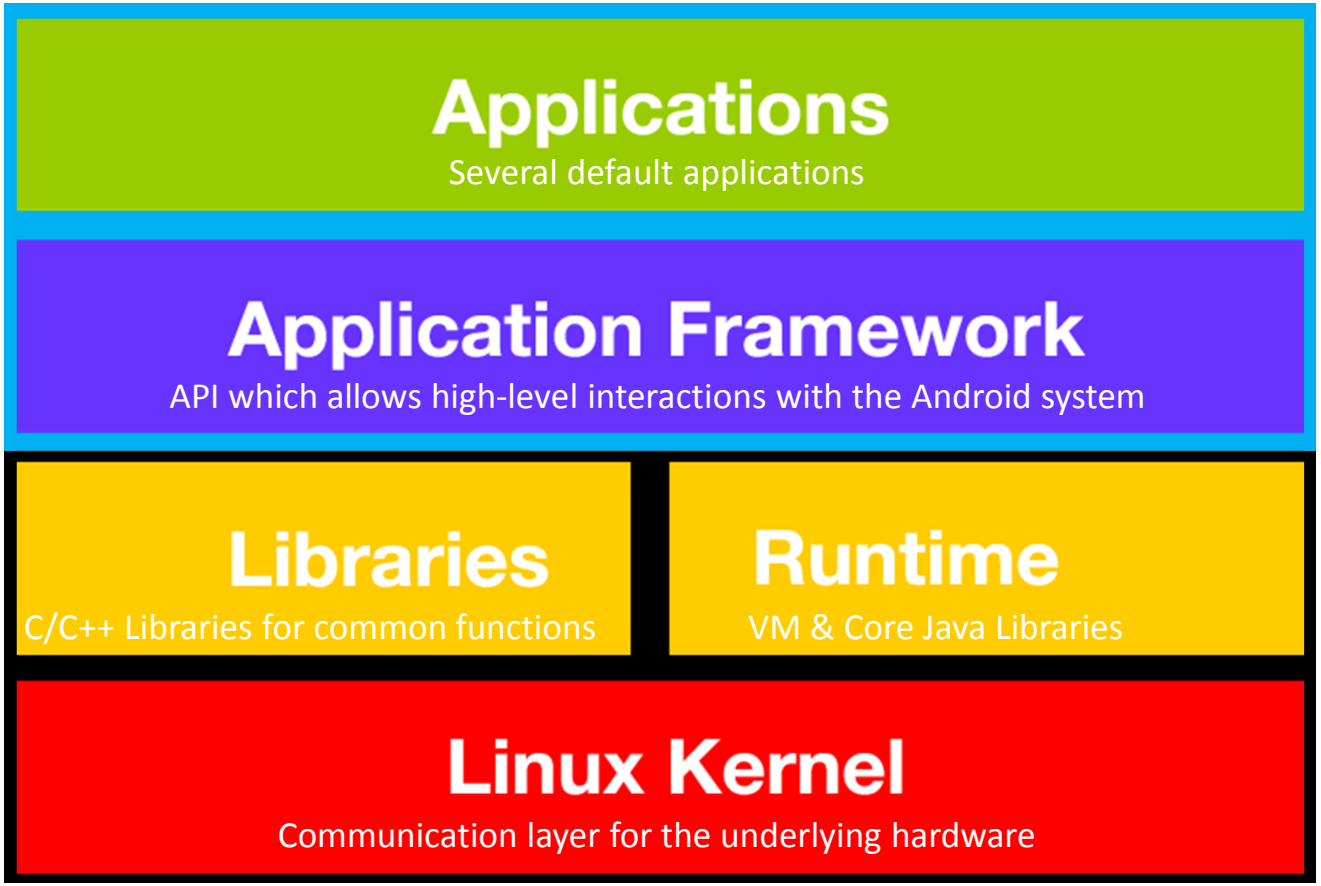
- Screen Size
- Processing Power
- RAM
- Storage
- Battery Life
- GSM/EDGE/3G/4G
- Wi-Fi
- Bluetooth
- Stylus Support
- Video Camera
- Touchscreen
- GPS
- Accelerometer
- Gyroscope
- Barometer
- Magnetometer
- Dedicated gaming control
- Proximity and pressure sensors
- Thermometer
- Accelerated 2D and accelerated 3D graphics

Development Challenges

- Various Types of Devices
- Different Android Versions
- Device Capabilities

ANDROID
PLATFORM COMPONENTS

Android Platform Components



Android Developer typically works with these two layers to create Android Apps.

The Linux Kernel, Libraries and the Runtime are encapsulated by the application framework

Android Platform Components

- Application - The Android Open Source Project contains several default application, like
  - Email Client
  - SMS Program
  - Calendar
  - Maps
  - Browser
  - Contacts
  - etc.
- All applications are written using the Java language.

Android Platform Components

- Application Framework – Consist of API which allows high-level interactions with the Android system from Android applications.
  - Managing the lifecycle of applications
  - Providing access to non-code resources (localized string, graphics, and layout files)
  - Enabling applications to access data from other applications or to share their own data
  - Providing user interface components
  - etc.

Android Platform Components

- Libraries - Includes a set of C/C++ libraries used by components of the Android system. Exposed to developers through the Android application framework
  - Recording and playback of audio and video formats
  - Access to the display system and supports 2D and 3D
  - WebKit library is responsible for browser support
  - FreeType library is responsible for font support
  - SQLite a relational database that is available on the device
  - etc.

Android Platform Components

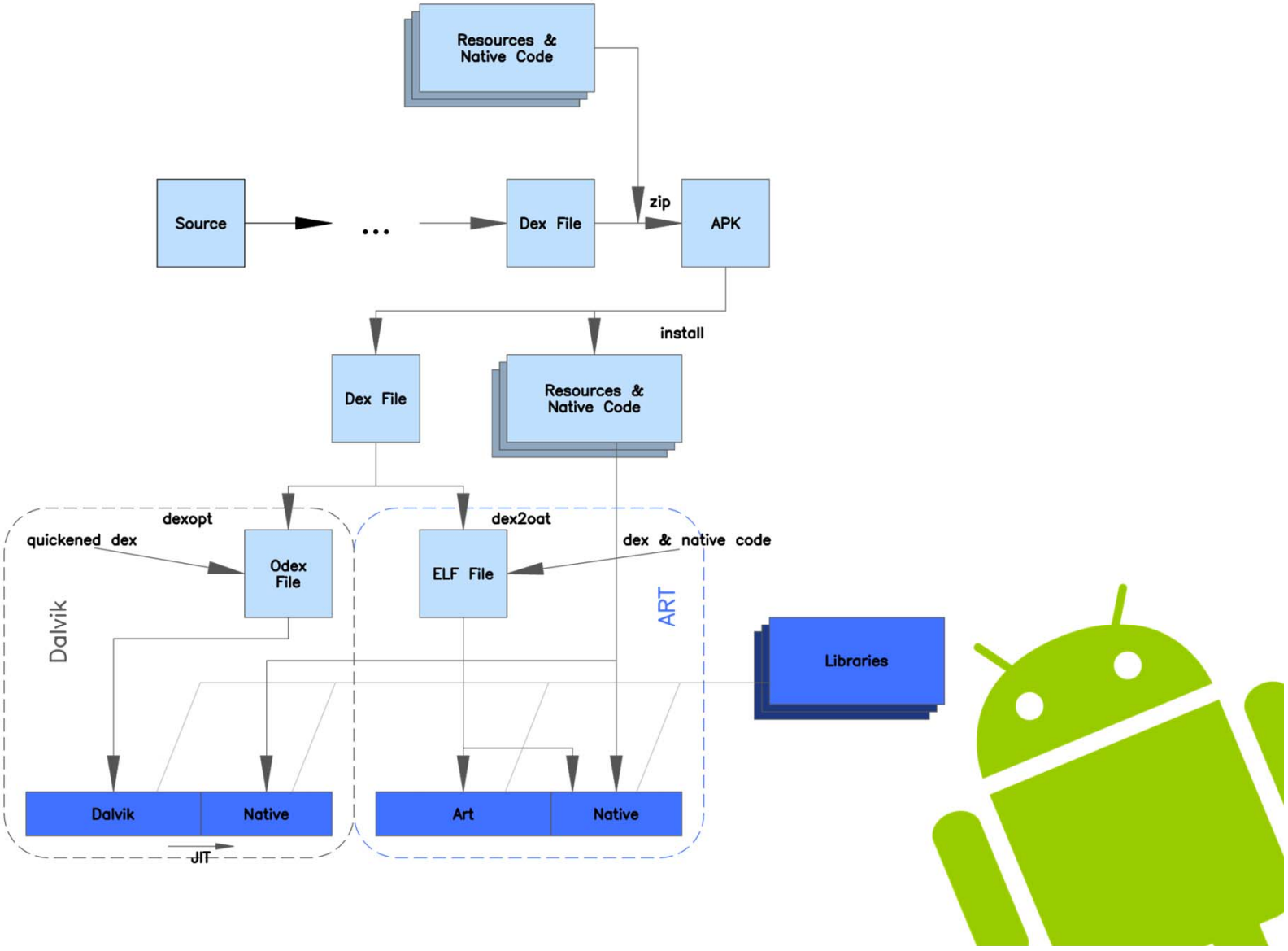
- Runtime - Core Java Libraries - Providing most of the functionality available in the core libraries of the Java language
  - Data Structures
  - Utilities
  - File Access
  - Network Access
  - Graphics
  - etc.

Android Platform Components

- Android Runtime (ART) - Providing environment on which every Android application runs
  - Replaces **Dalvik** (Used from 2.2 to 4.4)
  - **Dalvik** used JIT (Just-in-Time) Compilation
  - **ART** (5.0 onwards) uses AOT (Ahead-of-Time) Compilation
    (Creates "Executable and Linkable Format" ELF)

http://en.wikipedia.org/wiki/Android_Runtime

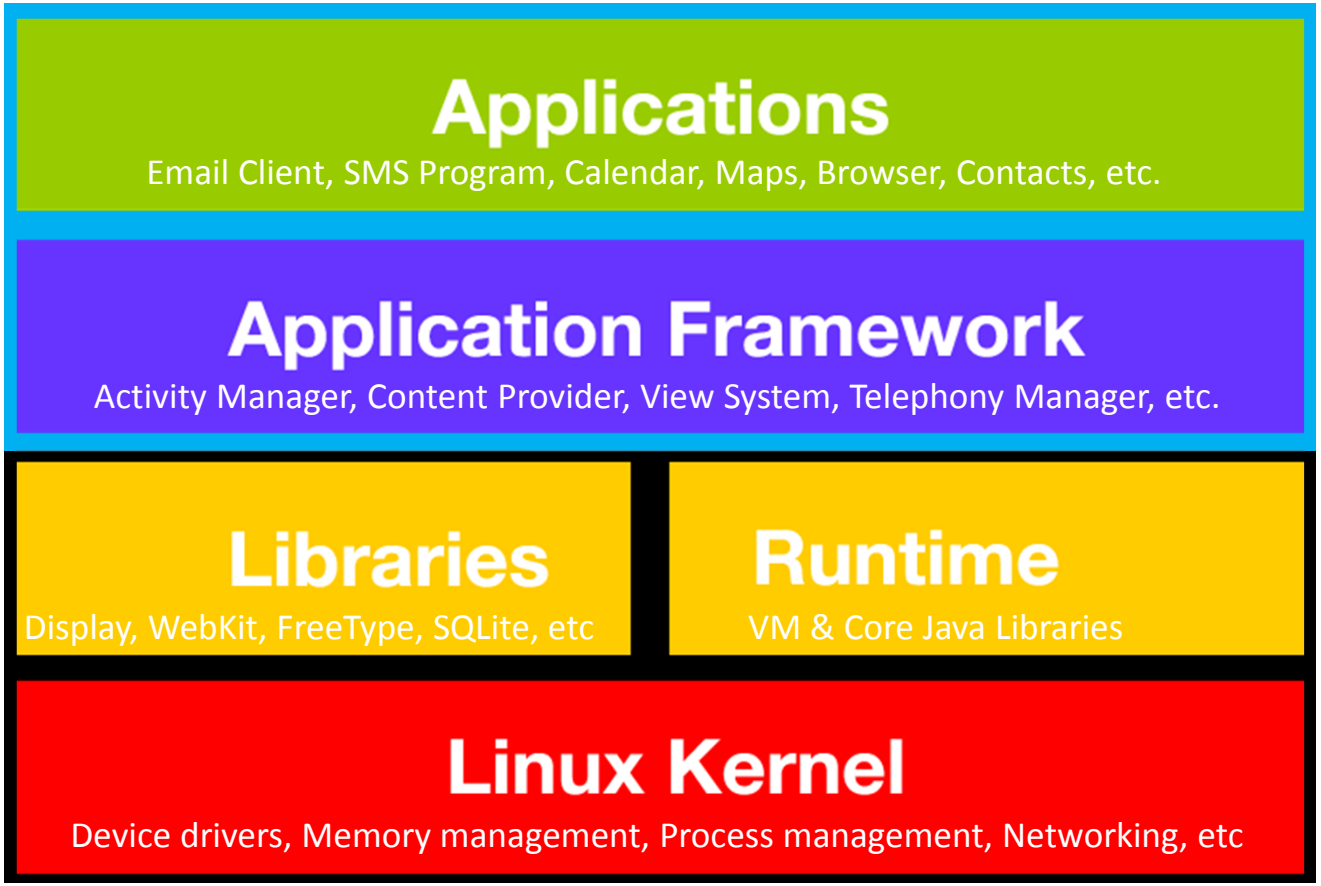
ART vs Dalvik



Android Platform Components

- Linux Kernel- Communication layer for the underlying hardware.
  - Device drivers
  - Memory management
  - Process management
  - Networking
  - etc.

Android Platform Components



Android Developer typically works with these two layers to create Android Apps.

The Linux Kernel, Libraries and the Runtime are encapsulated by the application framework

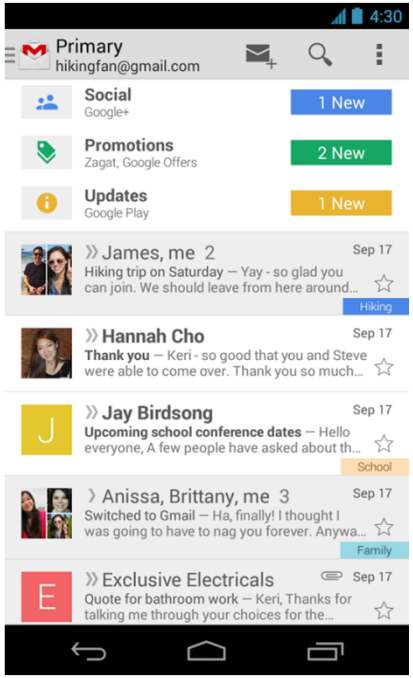
ANDROID
APPLICATION BUILDING BLOCKS

Application Building Blocks

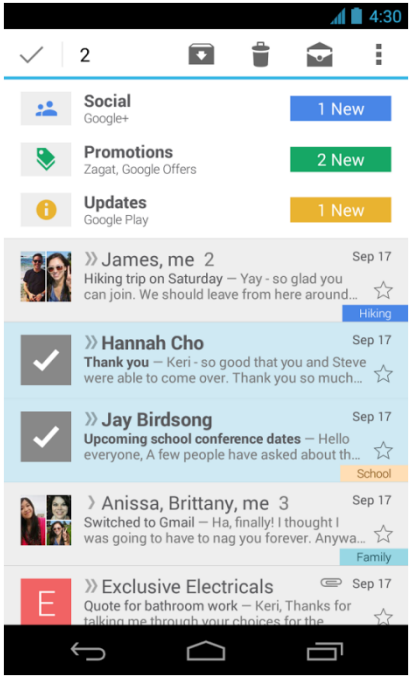
- App components are the essential building blocks of an Android application. Each component is a different point through which the system can enter your app.
- There are four different types of app components:
  - Activities
  - Services
  - Content Providers
  - Broadcast Receivers

Activities

- An *activity* represents a single screen with a user interface.

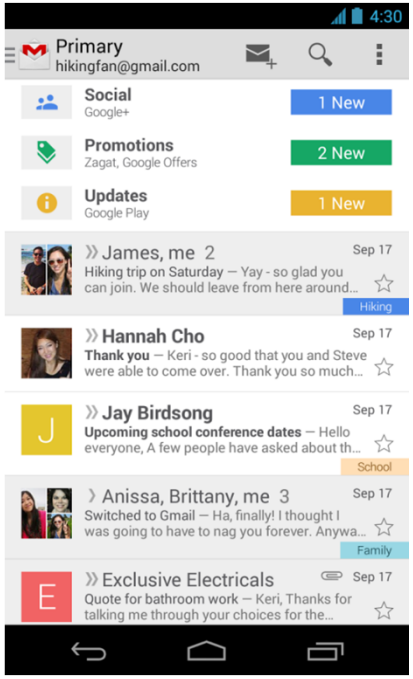


Same Activity

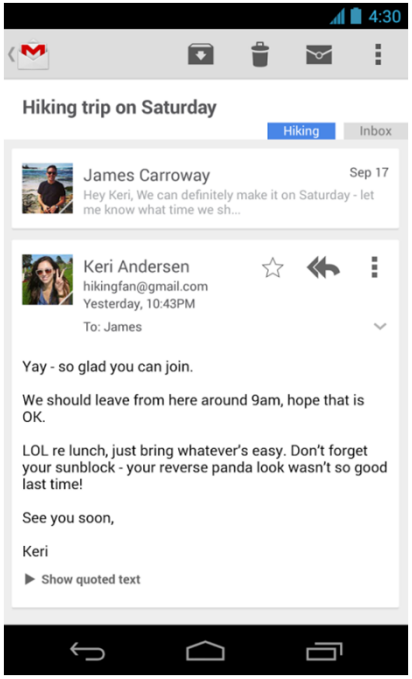


Activities

- An app (for example: email app) might have one activity that shows a list of new emails, another for reading emails.

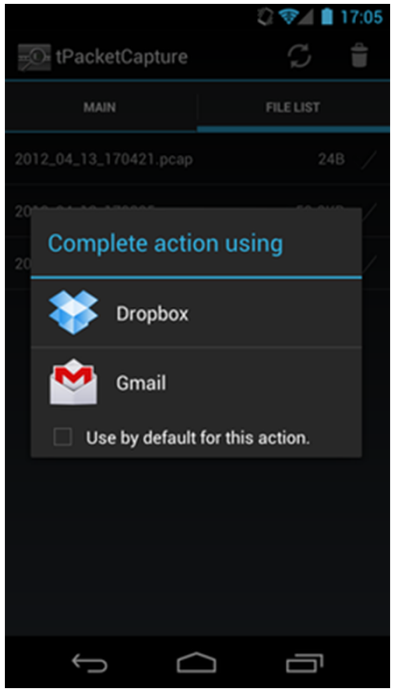


Two Different Activities



Activities

- A different app can start any one of these activities. For example, another app can start the activity in the email app that composes new mail.



Services

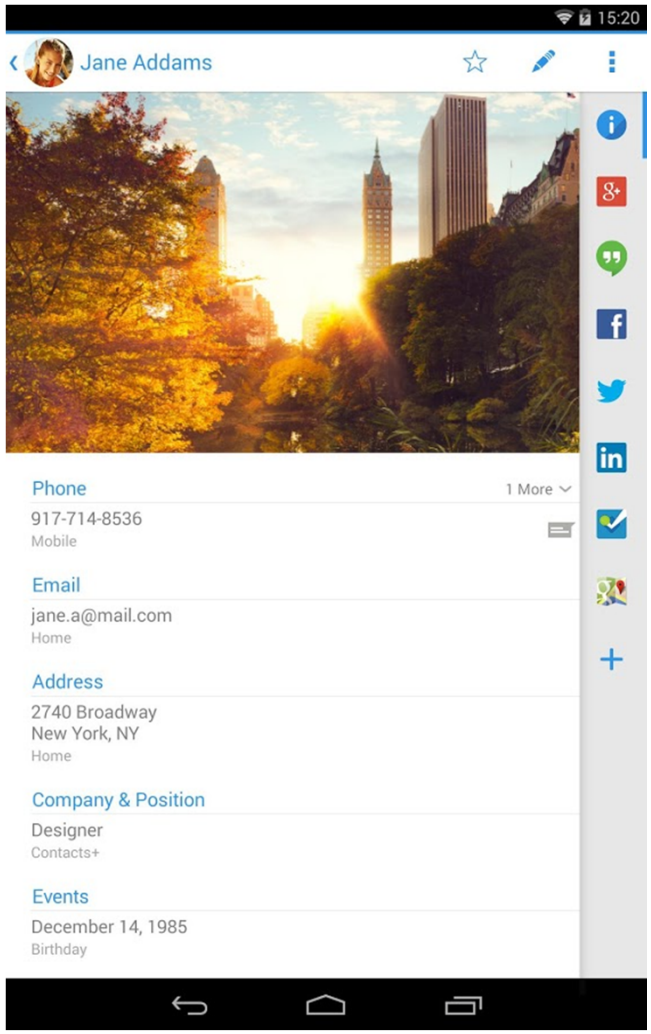
- A service is a component that runs in the background to perform long-running operations.
- A service does not provide a user interface.
- For example, play music in the background while the user is in a different app, or it might fetch data over the network.

Content Providers

- A content provider manages a shared set of app data.
- You can store the data in the file system, an SQLite database, on the web, or any other persistent storage location your app can access.
- Through the content provider, other apps can query or even modify the data (if the content provider allows it).

Content Providers

- For example, the Android system provides a content provider that manages the user's contact information.
- As such, any app with the proper permissions can query part of the content provider to read and write information about a particular person.



Broadcast Receivers

- A broadcast receiver is a component that responds to system-wide broadcast announcements.
- Broadcast receivers don't display a user interface, but in some cases a status bar notification is created to alert the user when a broadcast event occurs.
- For example, a broadcast announcing that the screen has turned off, the battery is low, or a picture was captured.
- Although Apps can also initiate broadcasts, many broadcasts also originate from the system

Application Components

- There are four different types of app components:
  - Activities
  - Services
  - Content Providers
  - Broadcast Receivers

Activating Components

- A unique aspect of the Android system design is that any app can start another app's component.

- Because the system runs each app in a separate process with file permissions that restrict access to other apps, your app cannot directly activate a component from another app. The Android system, however, can.

- So, to activate a component in another app, you must deliver a message to the system that specifies your intent to start a particular component. The system then activates the component for you.

Activating Components

- Three of the four component types—Activities, Services, and Broadcast Receivers—are activated by an asynchronous message called an intent.
- Intents bind individual components to each other at runtime (you can think of them as the messengers that request an action from other components), whether the component belongs to your app or another.
- Content Provider is activated when targeted by a request from a ContentResolver - handles all direct transactions between the content provider and the component requesting information (for security).

WRAP-UP

Summary

- Android Platform
- Android Platform Components
  - Application
  - Application Framework
  - Libraries
  - Runtime
    - Core Java Library
    - Virtual Machine
  - Linux Kernel
- Application Building Blocks
  - Application Components
    - Activities
    - Services
    - Content Providers
    - Broadcast Receivers
  - Activating Components
    - Intent
    - ContentResolver

Q & A