

Android Common Services & Apps (Part 2): Service Frameworks & Packaged Apps



Douglas C. Schmidt

d.schmidt@vanderbilt.edu

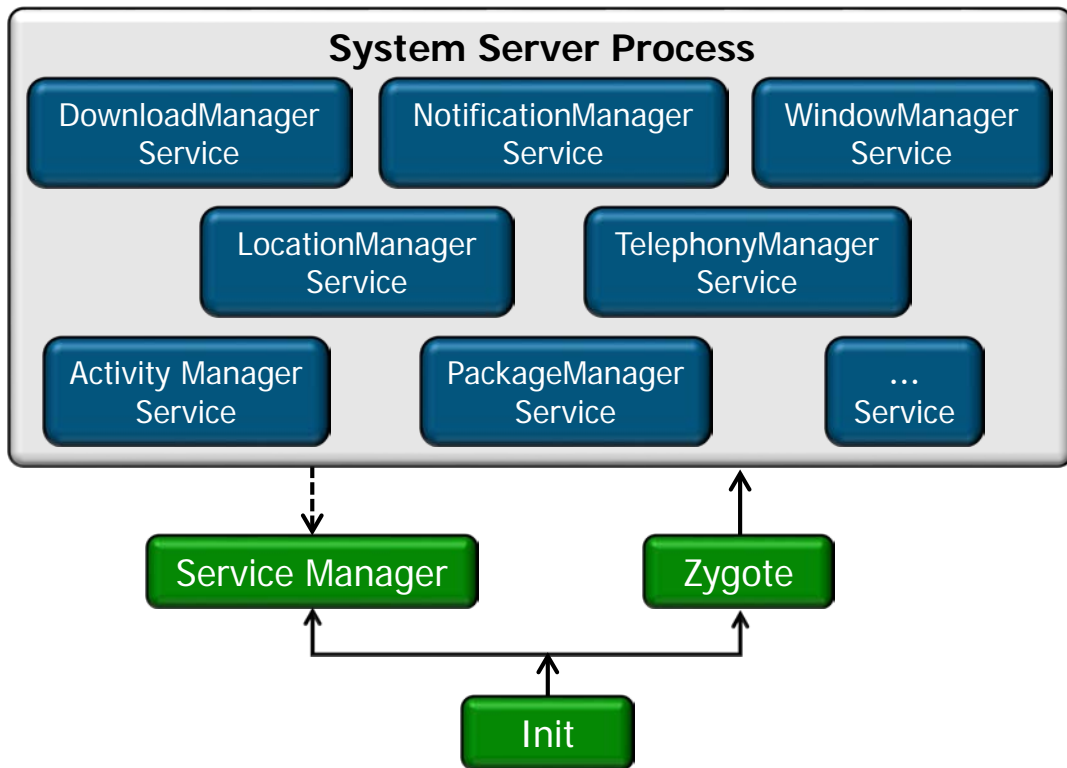
www.dre.vanderbilt.edu/~schmidt

Institute for Software
Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA



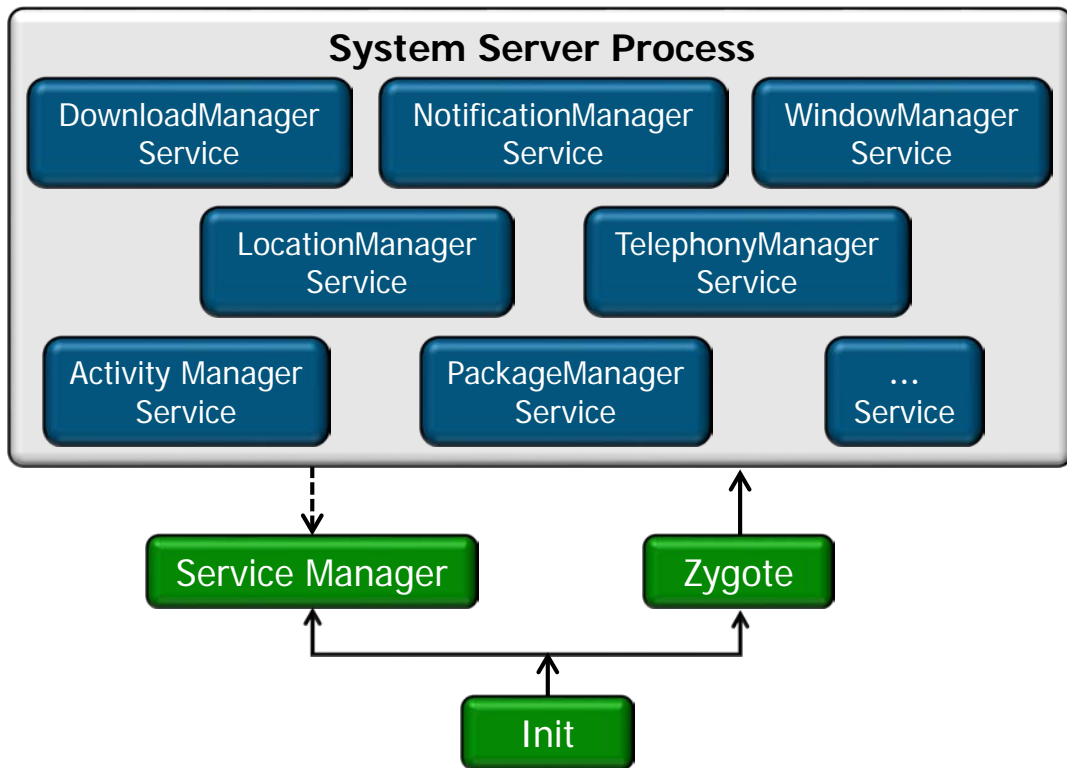
Learning Objectives in this Part of the Lesson

1. Name & recognize common services in the Android Application Framework layer



Learning Objectives in this Part of the Lesson

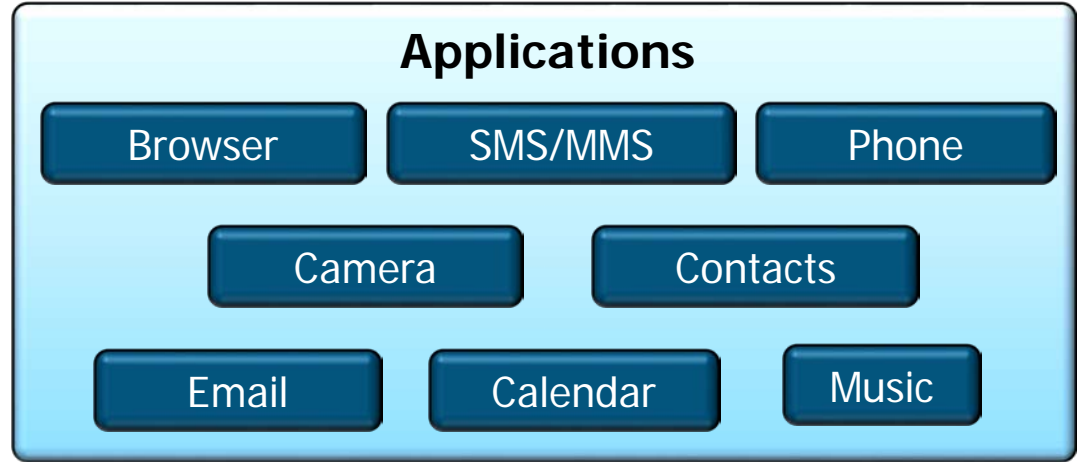
1. Name & recognize common services in the Android Application Framework layer



You'll use these services extensively when developing & running Android apps!

Learning Objectives in this Part of the Lesson

1. Name & recognize common services in the Android Application Framework layer
2. Name & recognize common apps that are available on an Android device

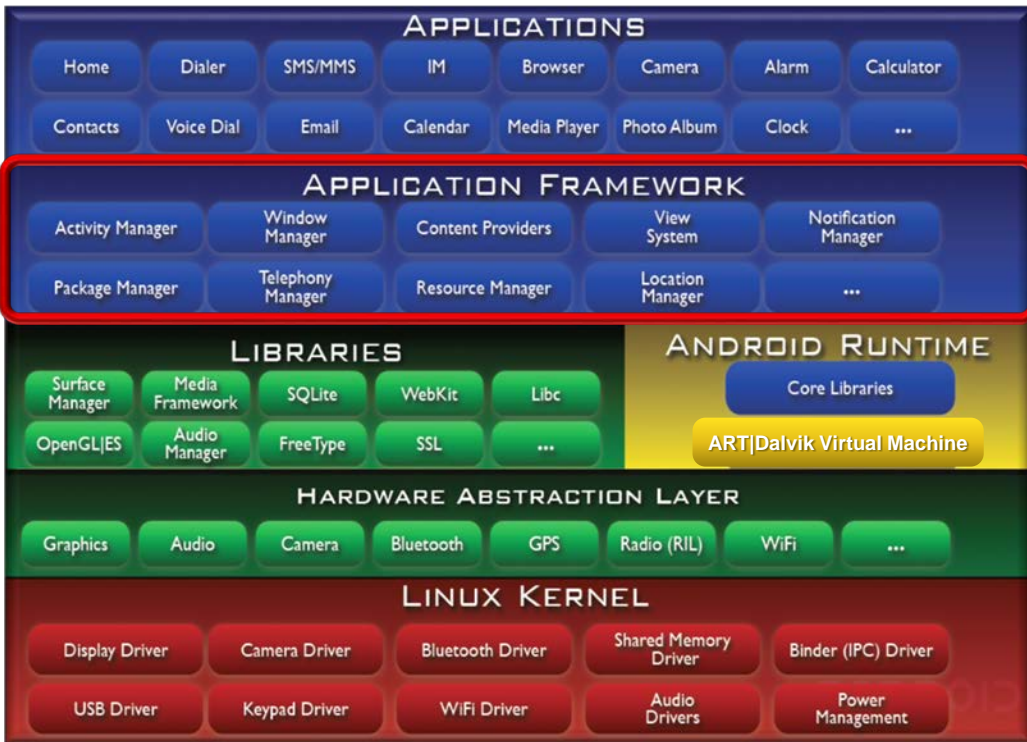


You probably use these (& other) Android apps every day!

Overview of the Android Application Framework

Overview of Android: Application Framework Layer

- Android's *Application framework* layer contains system services that provide apps w/capabilities & info they need to do their work

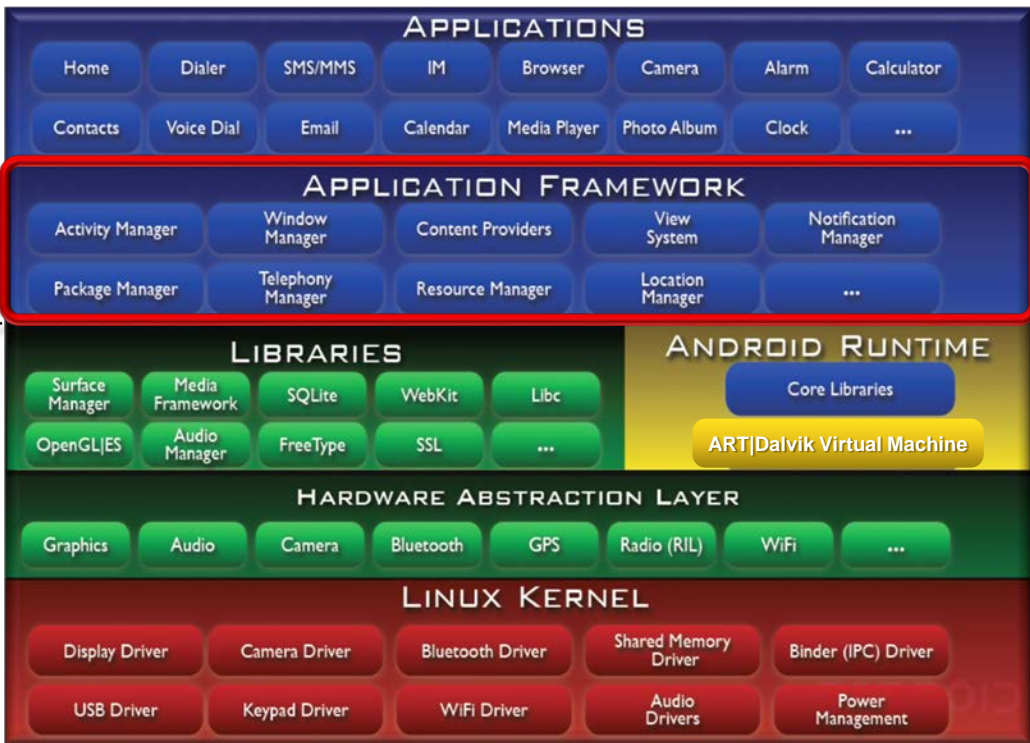


See anatomyofandroid.com/2013/10/03/system-services

Overview of Android: Application Framework Layer

- Android's *Application framework* layer contains system services that provide apps w/capabilities & info they need to do their work

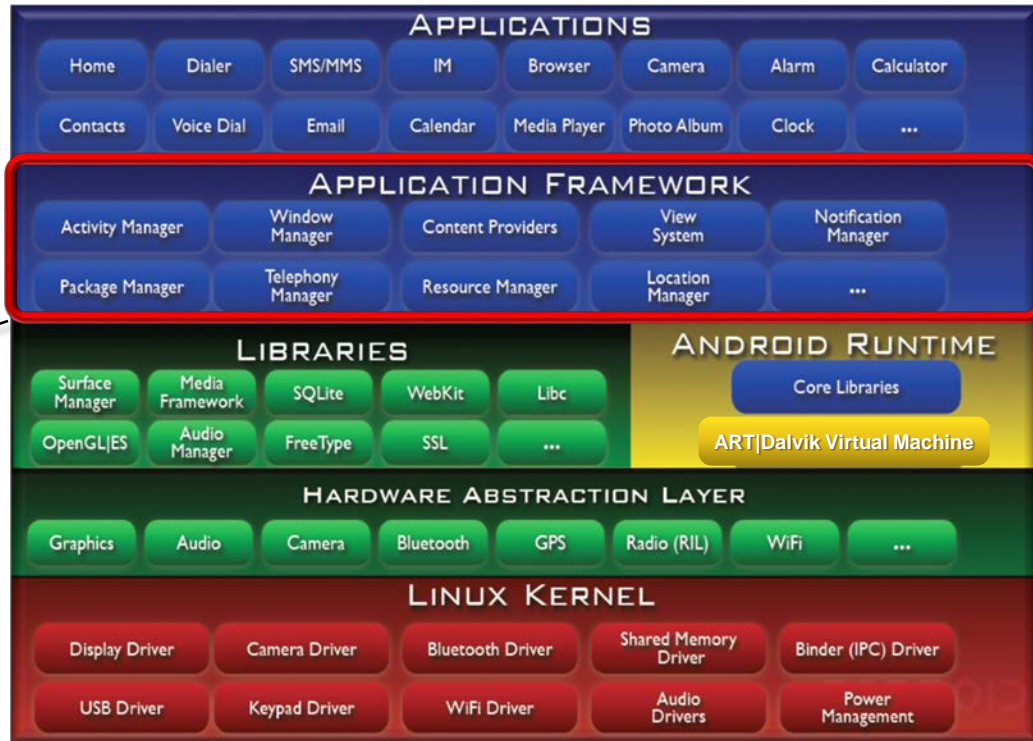
- Expose hardware & Linux OS kernel capabilities to apps*
- Run continuously during system operation*
- Control flow is driven by various events & callbacks*



Overview of Android: Application Framework Layer

- Android's *Application framework* layer contains system services that provide apps w/capabilities & info they need to do their work

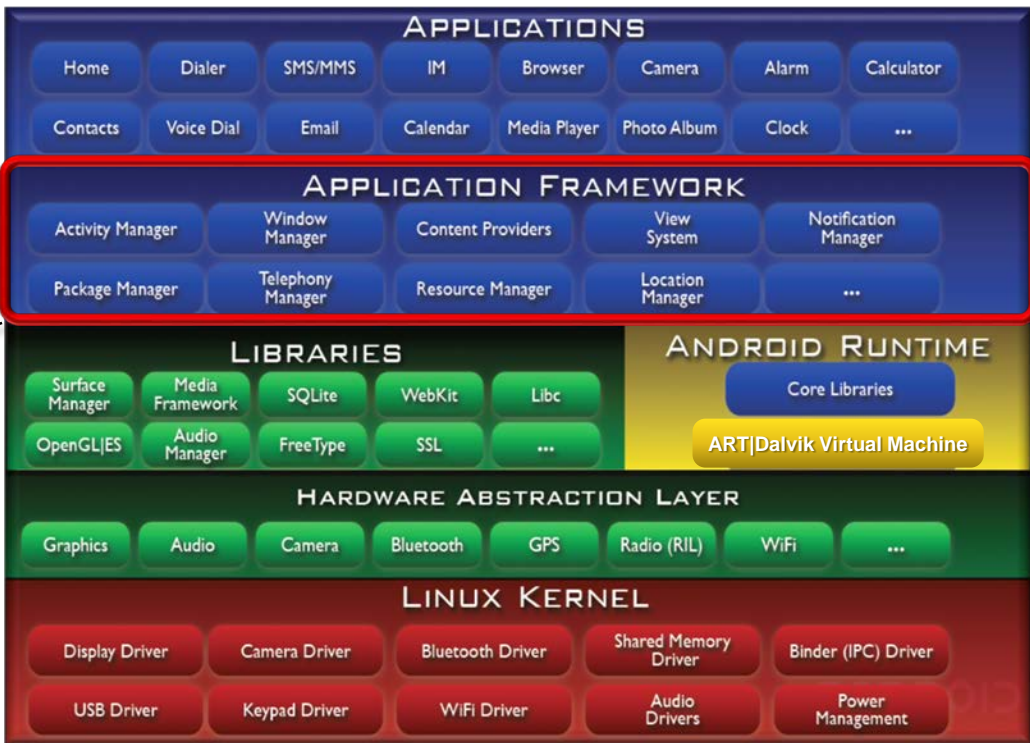
- *Expose hardware & Linux OS kernel capabilities to apps*
- *Run continuously during system operation*
- *Control flow is driven by various events & callbacks*



Overview of Android: Application Framework Layer

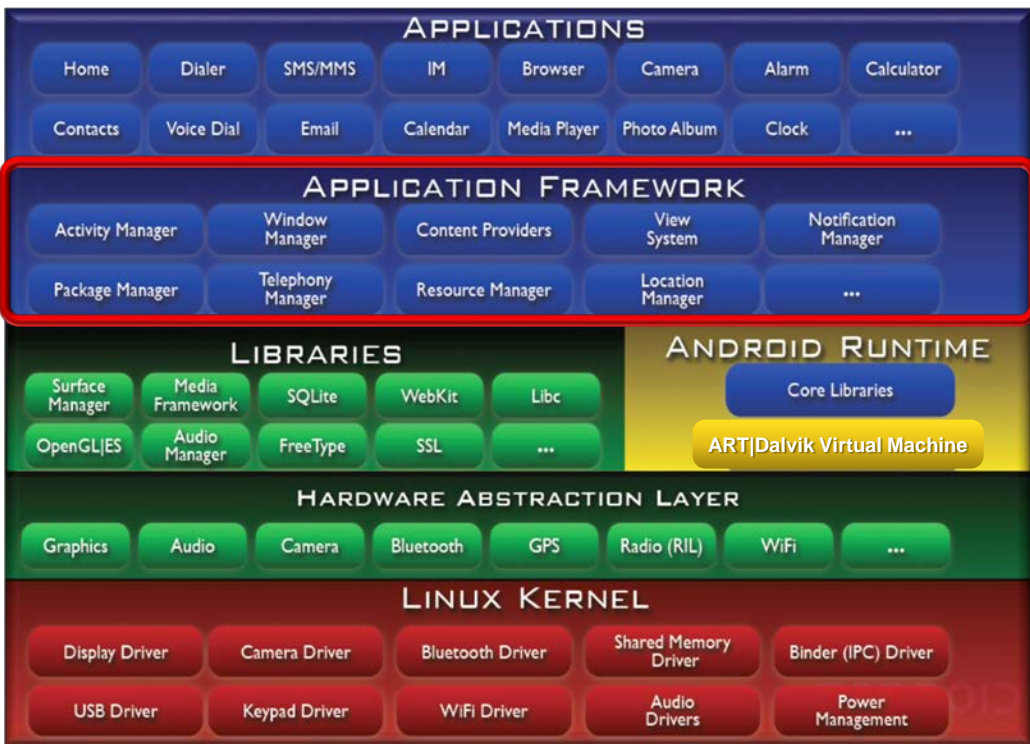
- Android's *Application framework* layer contains system services that provide apps w/capabilities & info they need to do their work

- *Expose hardware & Linux OS kernel capabilities to apps*
- *Run continuously during system operation*
- *Control flow is driven by various events & callbacks*



Overview of Android: Application Framework Layer

- Android's *Application framework* layer contains system services that provide apps w/capabilities & info they need to do their work



These system services are largely written in Java, with some C/C++ native code

Overview of Android: Application Framework Layer

- Developers can access Android's system services via the method `getSystemService()` available in an activity or any other context

```
public abstract Object getSystemService (String name)  
Added in API level 1
```

Return the handle to a system-level service by name. The class of the returned object varies by the requested name. Currently available names are:

`WINDOW_SERVICE` ("window")

The top-level window manager in which you can place custom windows. The returned object is a `WindowManager`.

`ACTIVITY_SERVICE` ("activity")

A `ActivityManager` for interacting with the global activity state of the system.

`NOTIFICATION_SERVICE` ("notification")

A `NotificationManager` for informing the user of background events.

`LOCATION_SERVICE` ("location")

A `LocationManager` for controlling location (e.g., GPS) updates.

`DOWNLOAD_SERVICE` ("download")

A `DownloadManager` for requesting HTTP downloads

Overview of Android: Application Framework Layer

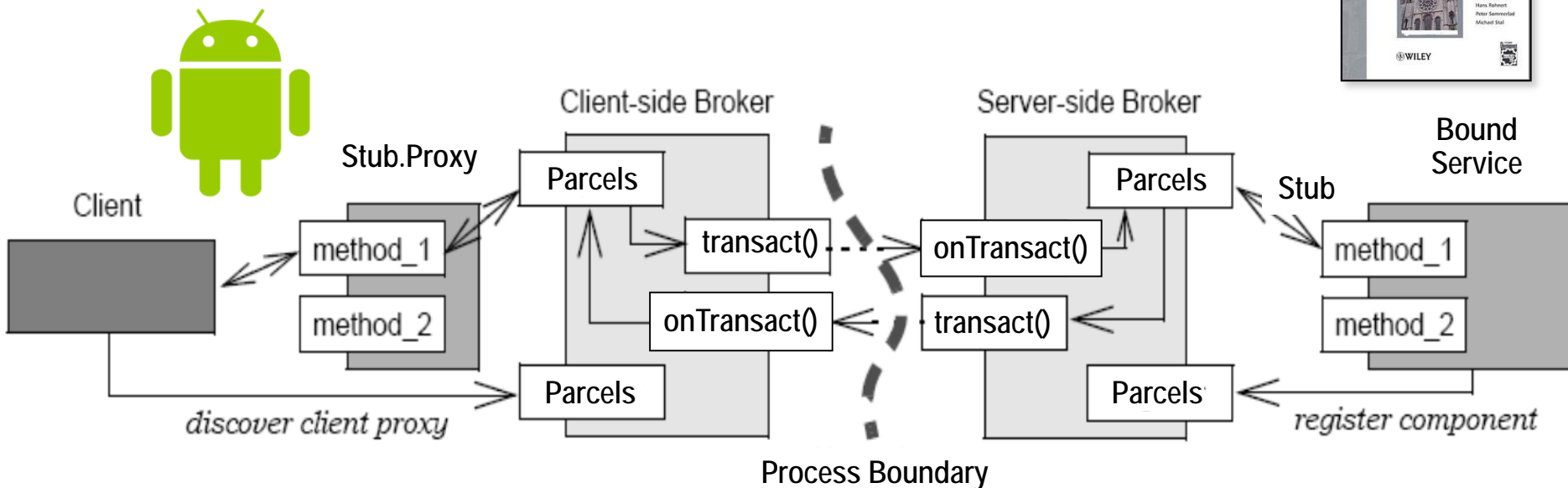
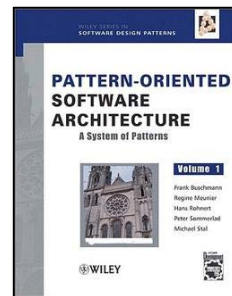
- Developers can reuse the system services in Android's application framework layer to program powerful apps effectively & productively



See sites.google.com/site/io/inside-the-android-application-framework

Overview of Android: Application Framework Layer

- Developers can reuse the system services in Android's application framework layer to program powerful apps effectively & productively, e.g.
- Leverage time-proven patterns (e.g., Broker & Proxy) & frameworks (e.g., Binder, HaMeR, & AsyncTask)



Overview of Android: Application Framework Layer

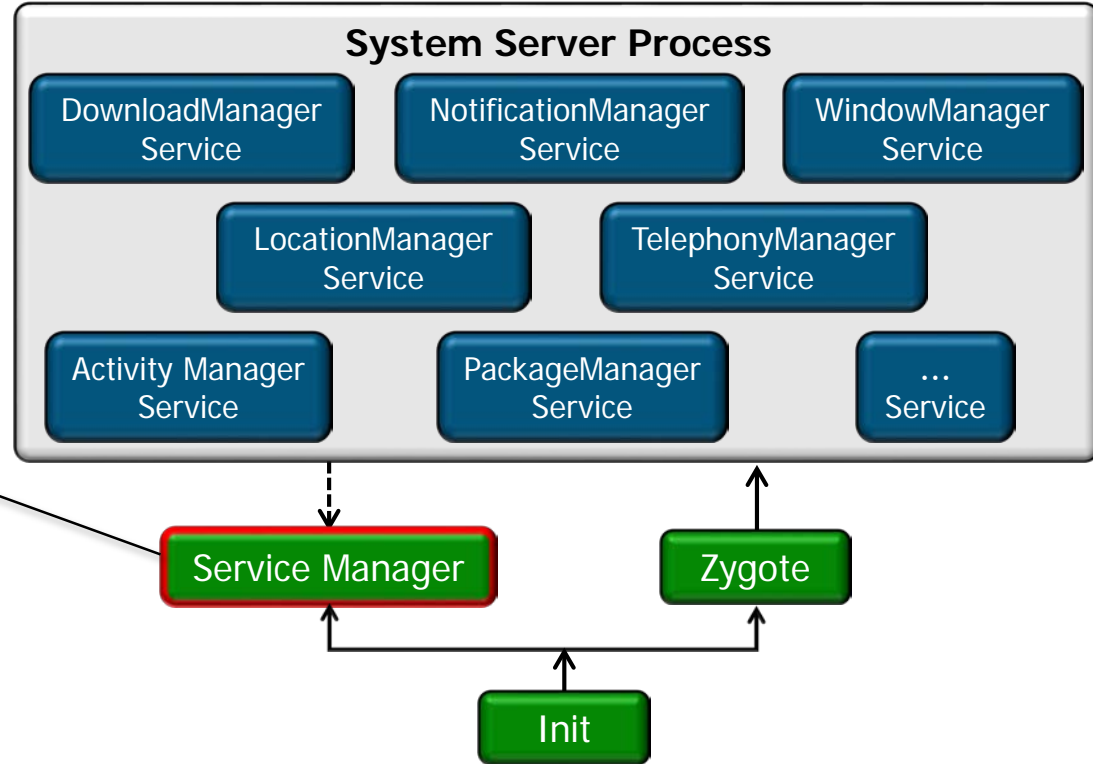
- Developers can reuse the system services in Android's application framework layer to program powerful apps effectively & productively, e.g.
 - Leverage time-proven patterns (e.g., Broker & Proxy) & frameworks (e.g., Binder, HaMeR, & AsyncTask)
 - Apps (& app developers) don't need to "reinvent the wheel"



Example System Services in Application Framework Layer

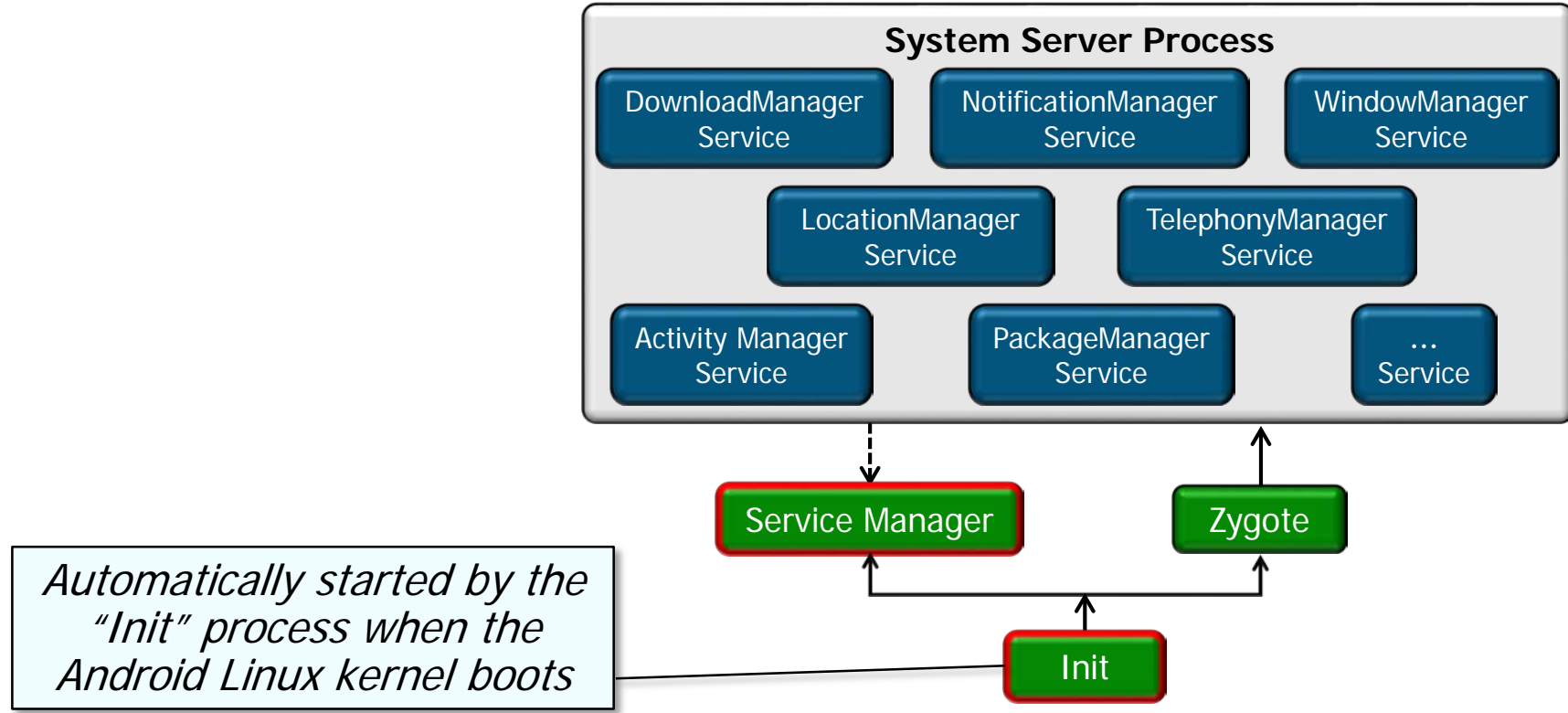
- Android system services are coordinated by the ServiceManager

A "super server" that ensures system services are started & restarted throughout the lifetime of an Android device



Example System Services in Application Framework Layer

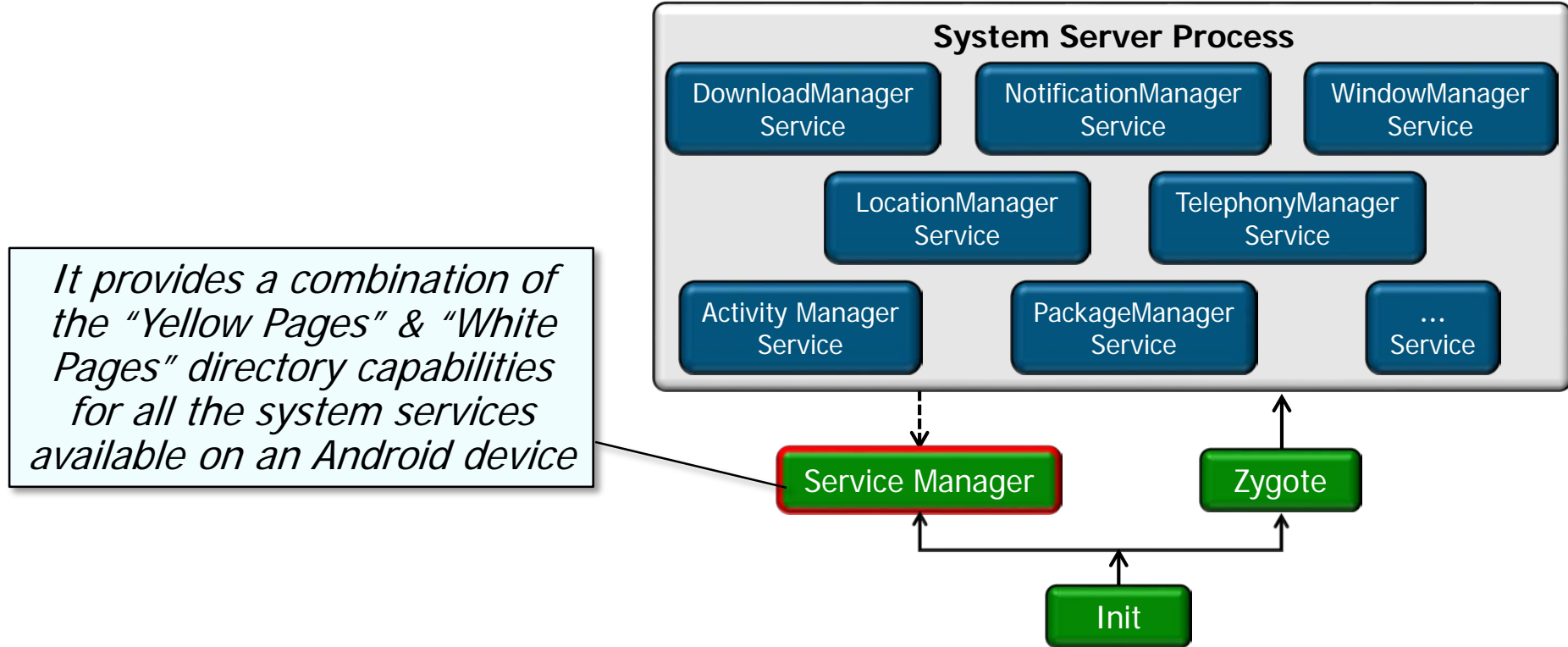
- Android system services are coordinated by the ServiceManager



See anatomyofandroid.com/2013/10/11/service-manager

Example System Services in Application Framework Layer

- Android system services are coordinated by the ServiceManager

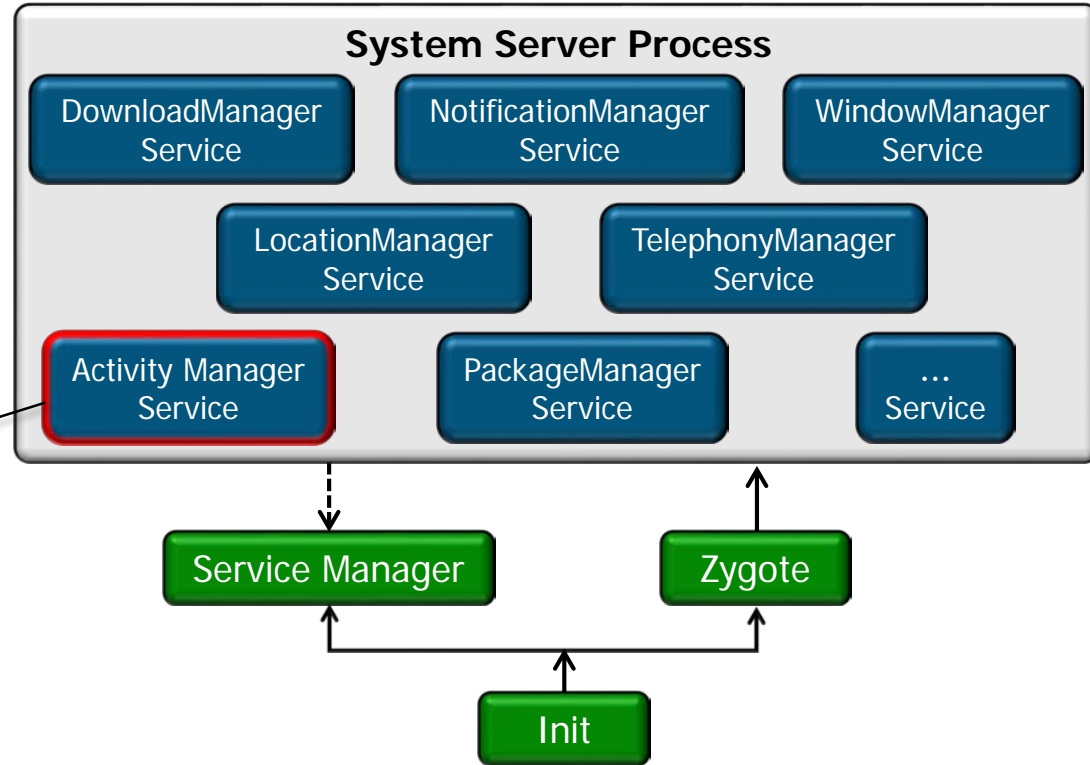


It provides a combination of the "Yellow Pages" & "White Pages" directory capabilities for all the system services available on an Android device

See en.wikipedia.org/wiki/Yellow_pages & en.wikipedia.org/wiki/Telephone_directory

Example System Services in Application Framework Layer

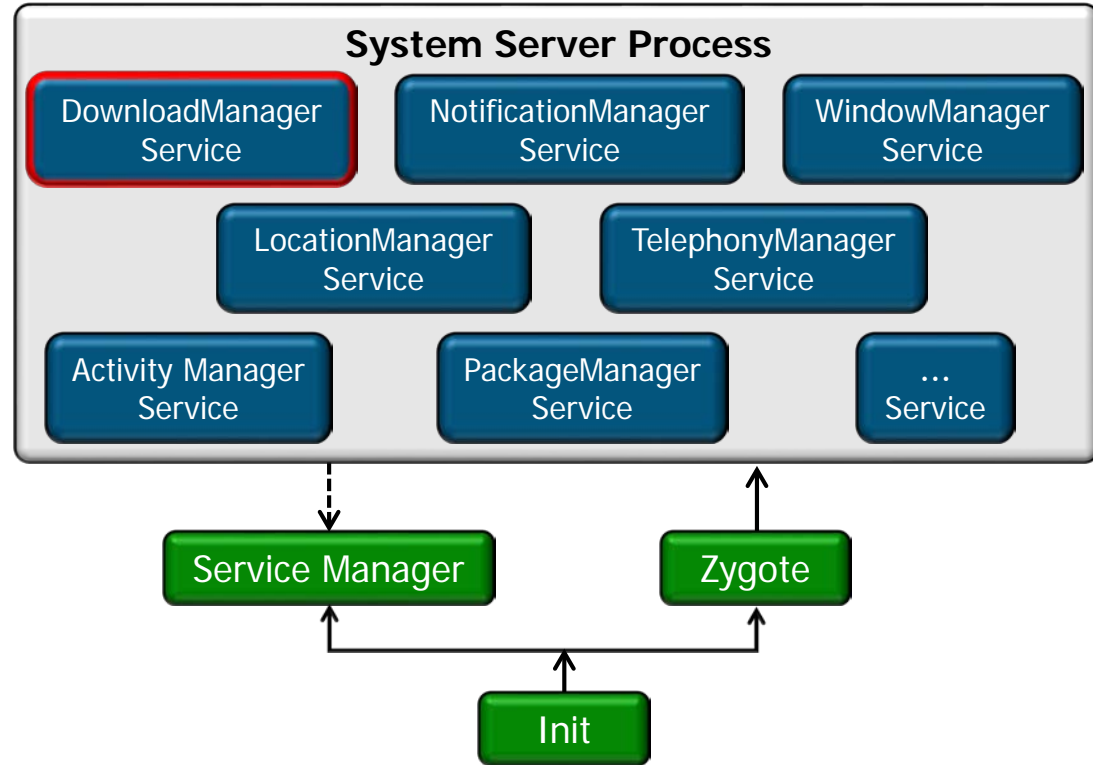
- Android system services are coordinated by the ServiceManager
- Activity Manager
 - Controls all components in the system



We focus on the Activity Manager Service throughout this course

Example System Services in Application Framework Layer

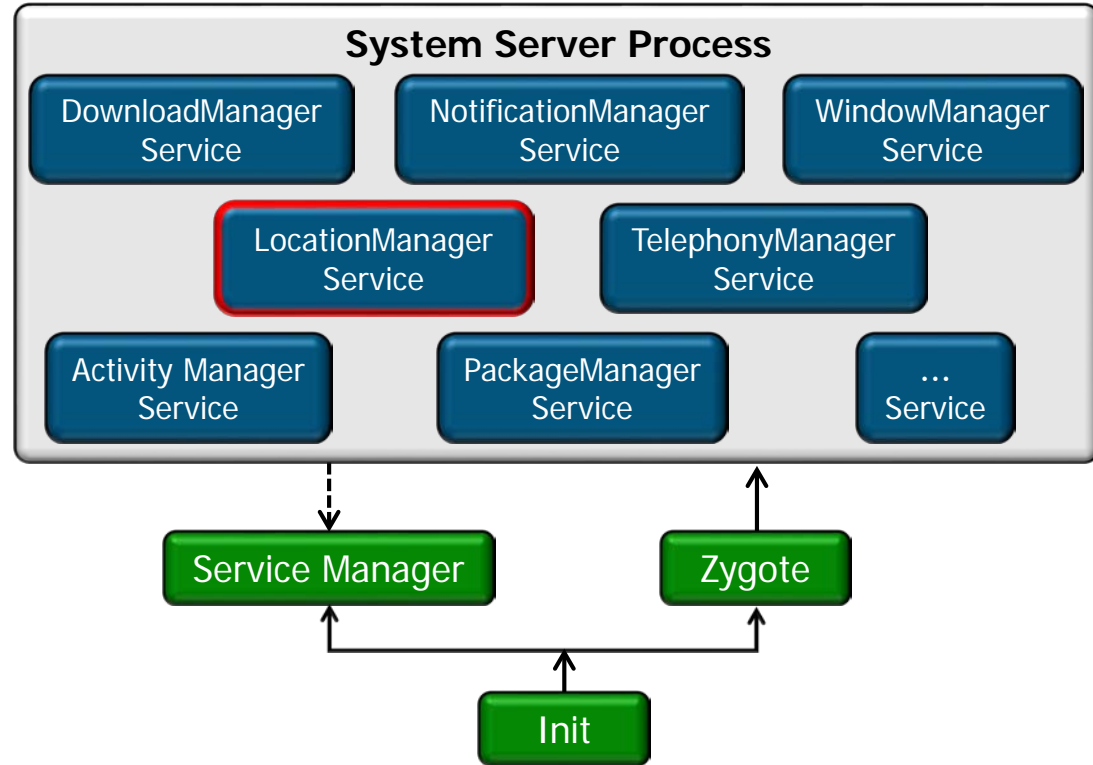
- Android system services are coordinated by the ServiceManager
 - Activity Manager
 - Download Manager
 - Handles long-running HTTP downloads stored to a particular file



See developer.android.com/reference/android/app/DownloadManager.html

Example System Services in Application Framework Layer

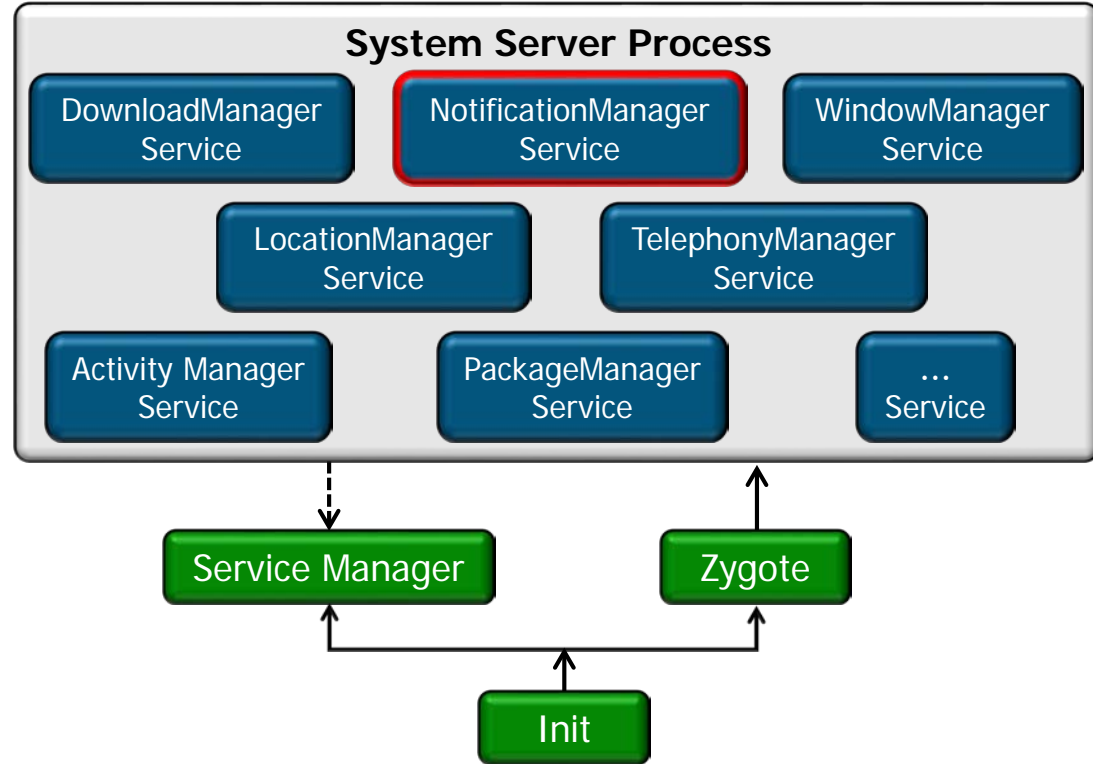
- Android system services are coordinated by the ServiceManager
 - Activity Manager
 - Download Manager
 - Location Manager
 - Provides access to the system location services



See developer.android.com/reference/android/location/LocationManager.html

Example System Services in Application Framework Layer

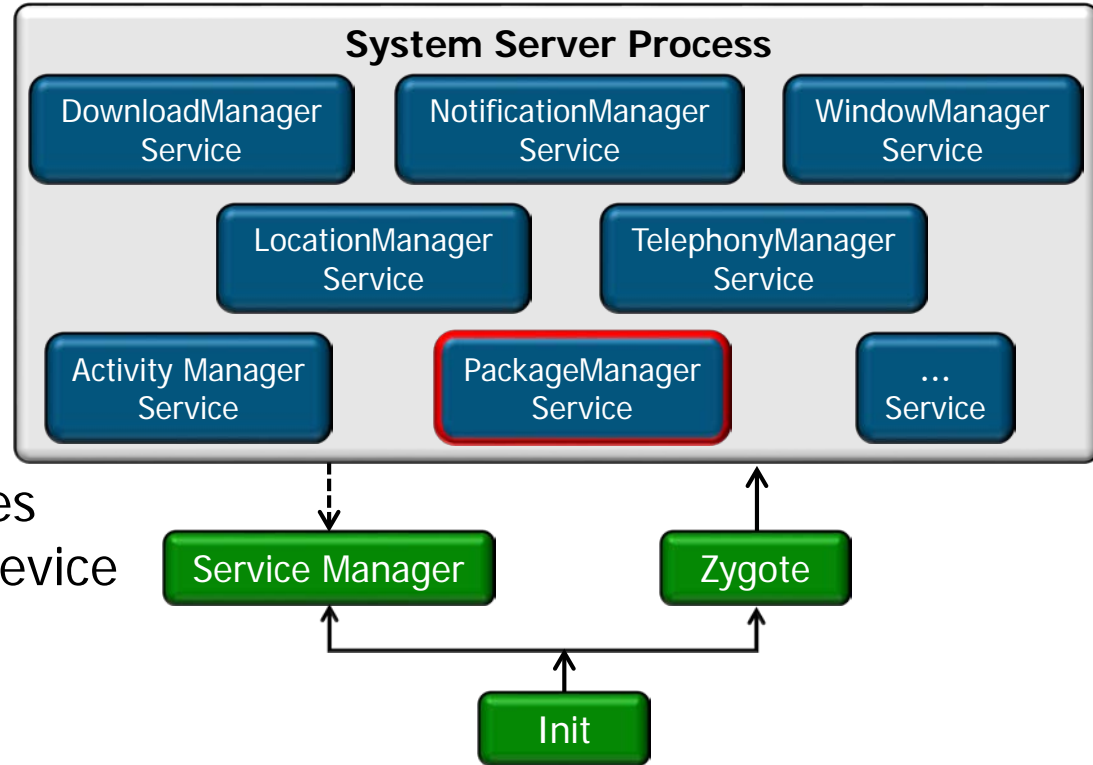
- Android system services are coordinated by the ServiceManager
 - Activity Manager
 - Download Manager
 - Location Manager
 - Notification Manager
 - Inform user of events that happen in the background



See developer.android.com/reference/android/app/NotificationManager.html

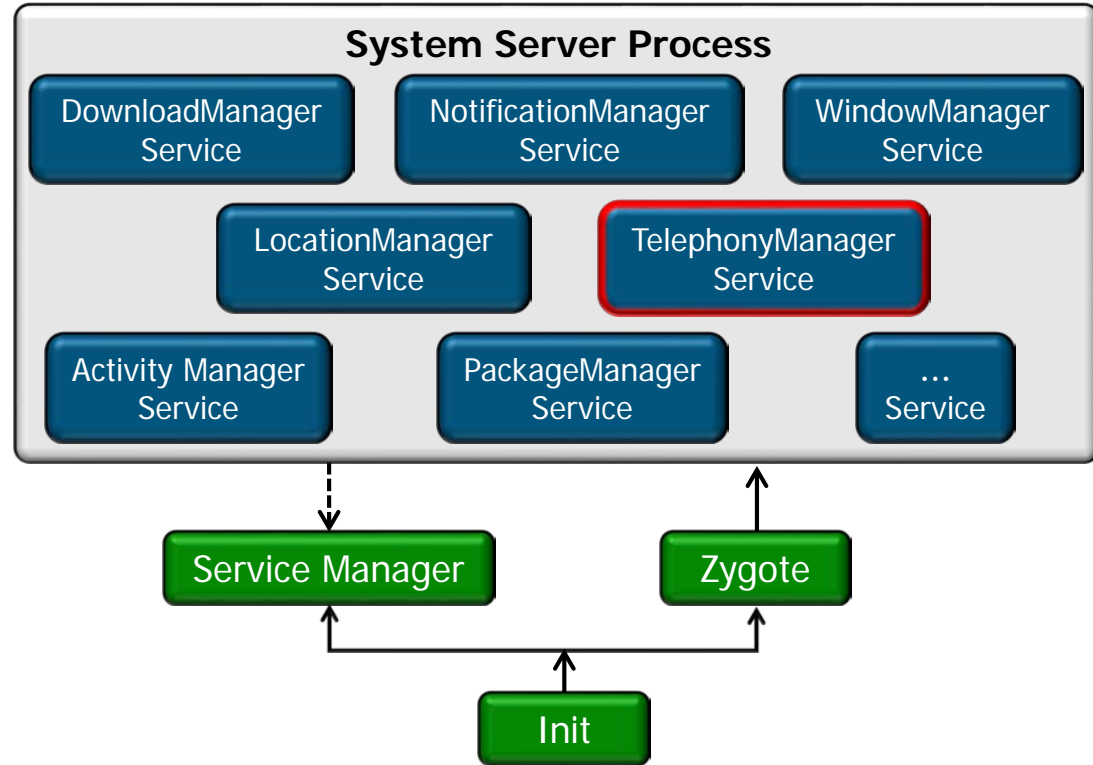
Example System Services in Application Framework Layer

- Android system services are coordinated by the ServiceManager
 - Activity Manager
 - Download Manager
 - Location Manager
 - Notification Manager
 - Package Manager
 - Stores various kinds of info related to app packages currently installed on the device



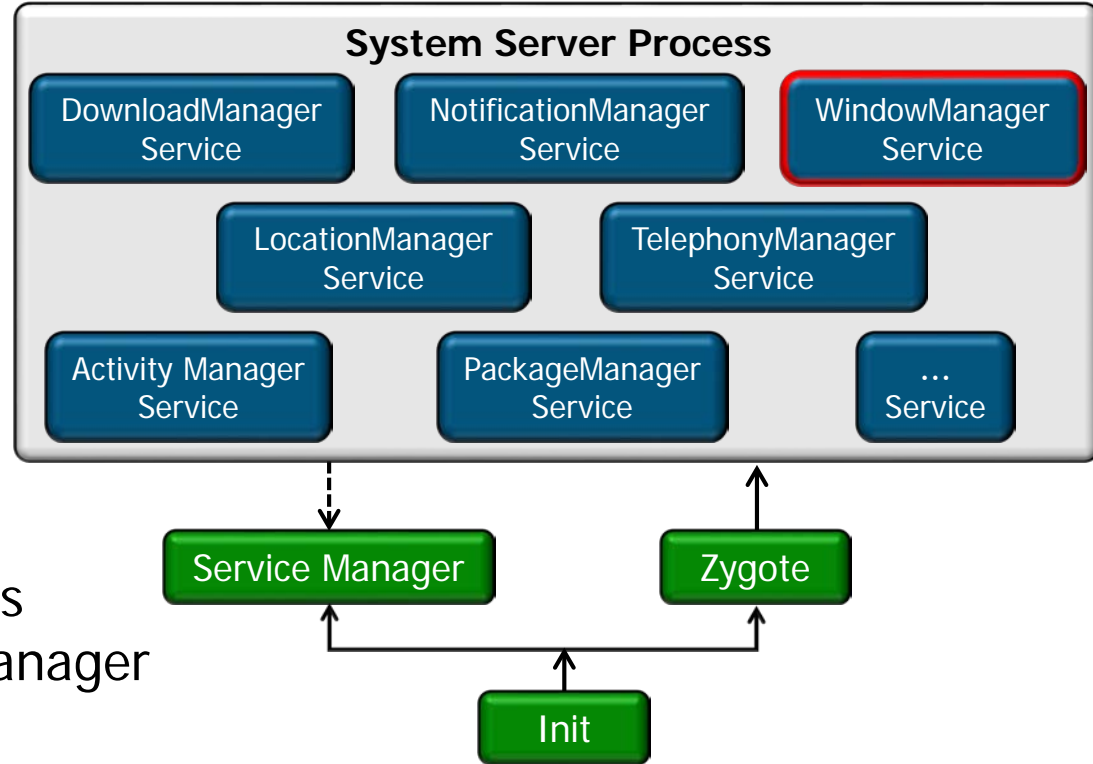
Example System Services in Application Framework Layer

- Android system services are coordinated by the ServiceManager
 - Activity Manager
 - Download Manager
 - Location Manager
 - Notification Manager
 - Package Manager
 - Telephony Manager
 - Provides access to info about telephony services on the device



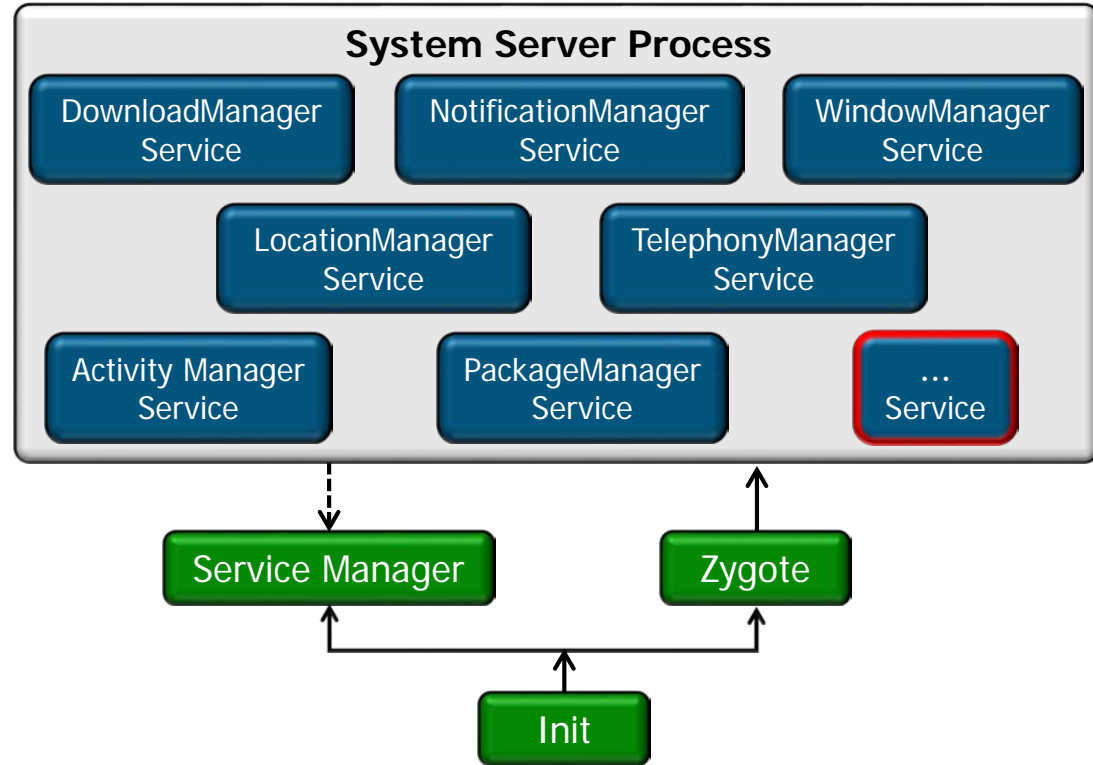
Example System Services in Application Framework Layer

- Android system services are coordinated by the ServiceManager
 - Activity Manager
 - Download Manager
 - Location Manager
 - Notification Manager
 - Package Manager
 - Telephony Manager
 - Window Manager
 - Provides interface that apps use to talk with window manager



Example System Services in Application Framework Layer

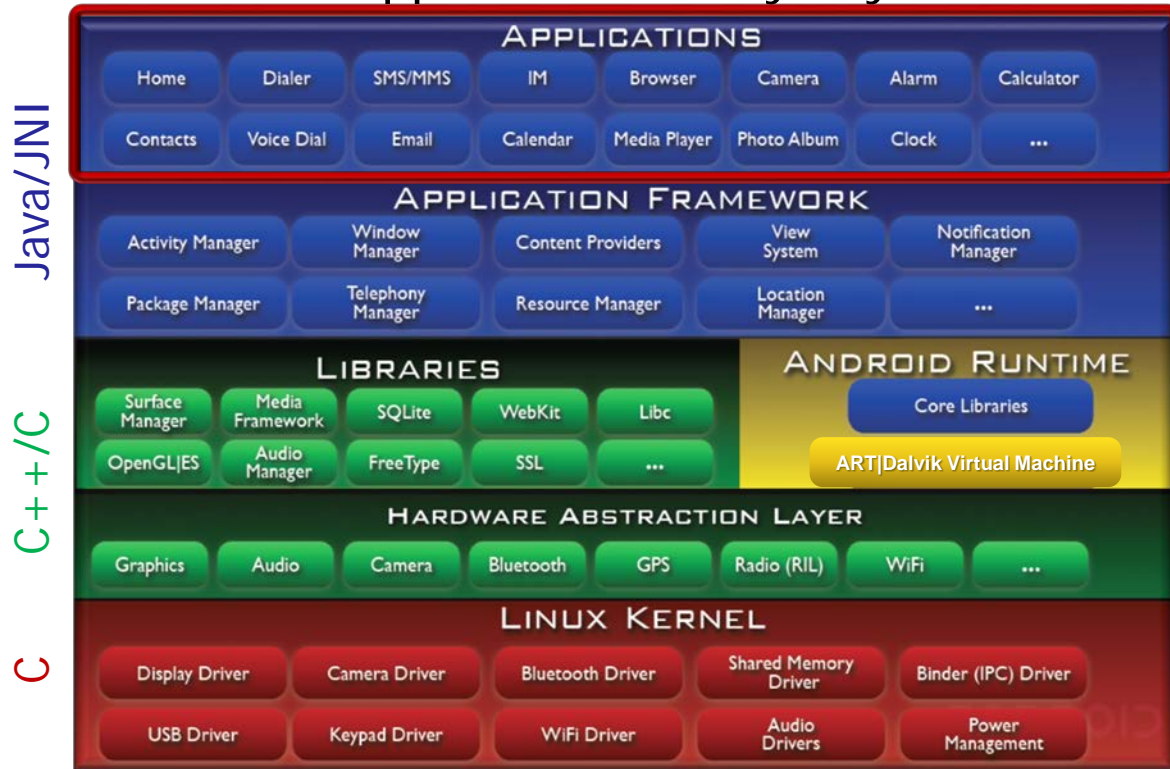
- Android system services are coordinated by the ServiceManager
 - Activity Manager
 - Download Manager
 - Location Manager
 - Notification Manager
 - Package Manager
 - Telephony Manager
 - Window Manager
 - etc.



Overview of Common Android Apps

Overview of Common Android Apps

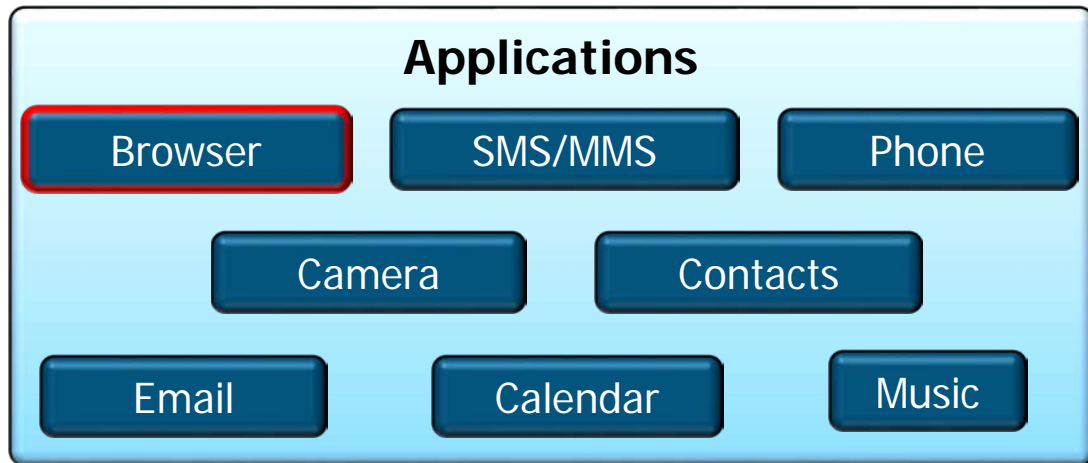
- On the top of Android's software stack are apps we use every day



See android.googlesource.com/platform/packages/apps

Overview of Common Android Apps

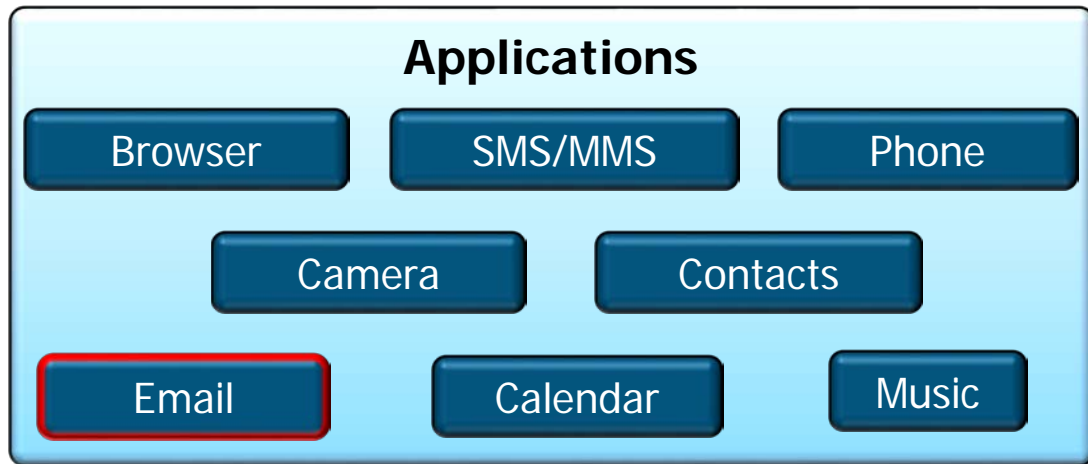
- On the top of Android's software stack are apps we use every day, e.g.
 - Browser
 - Retrieves, presents, & traverses info resources on the World Wide Web



See [packages/apps/Browser](https://developer.android.com/packages/apps/Browser)

Overview of Common Android Apps

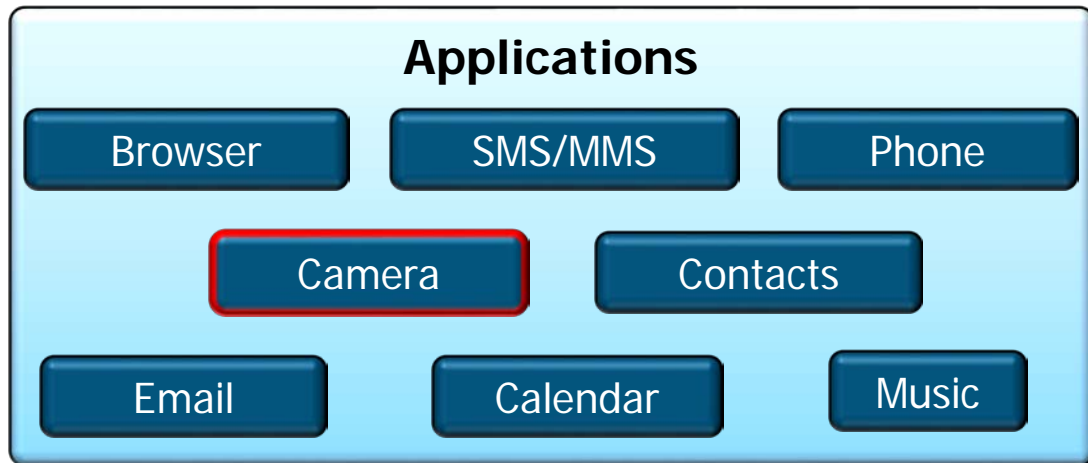
- On the top of Android's software stack are apps we use every day, e.g.
 - Browser
 - Email
 - Provides email message composition, reception, & management functions



See [packages/apps/Email](https://developer.android.com/packages/apps/Email)

Overview of Common Android Apps

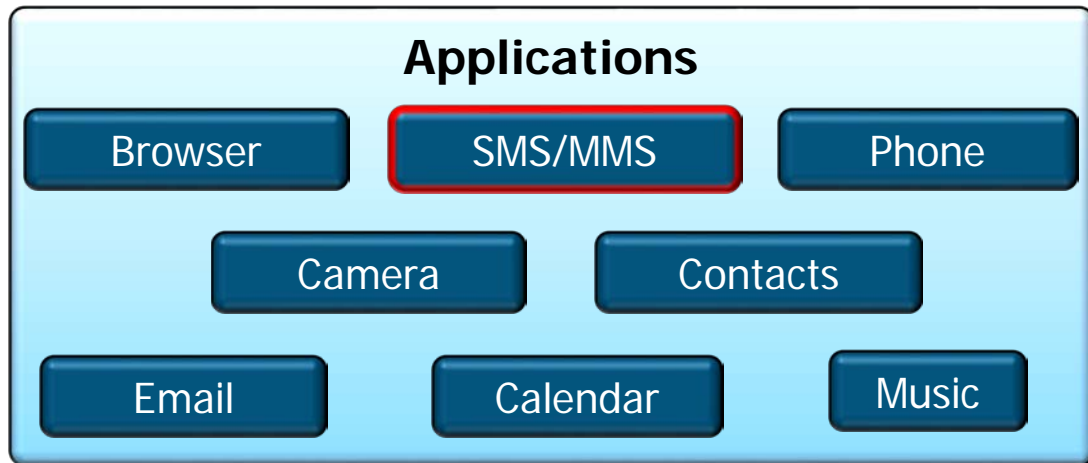
- On the top of Android's software stack are apps we use every day, e.g.
 - Browser
 - Email
 - Camera
 - Encodes digital images & videos digitally & stores them for later viewing



See [packages/apps/Camera](https://developer.android.com/packages/apps/Camera)

Overview of Common Android Apps

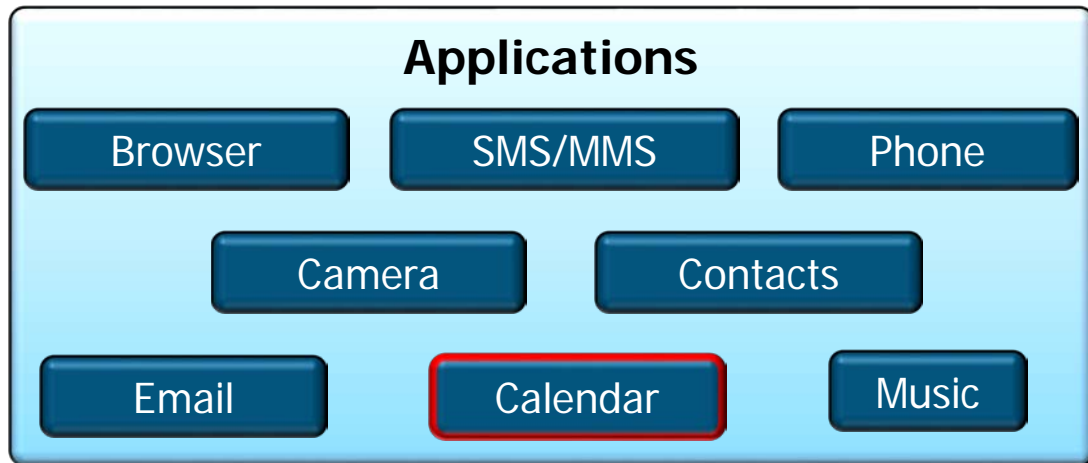
- On the top of Android's software stack are apps we use every day, e.g.
 - Browser
 - Email
 - Camera
 - SMS/MMS
 - Send/receive messages with multimedia content to/from mobile phones over a cellular network



See [packages/apps/Mms](https://developer.android.com/packages/apps/Mms)

Overview of Common Android Apps

- On the top of Android's software stack are apps we use every day, e.g.
 - Browser
 - Email
 - Camera
 - SMS/MMS
 - Calendar
 - Tracks time-based events & appointments

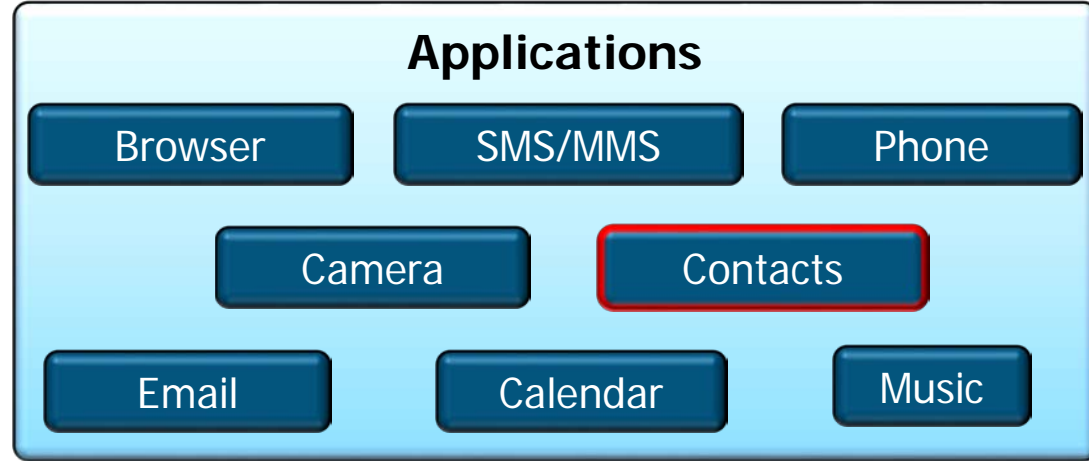


See [packages/apps/Calendar](https://developer.android.com/packages/apps/Calendar)

Overview of Common Android Apps

- On the top of Android's software stack are apps we use every day, e.g.

- Browser
- Email
- Camera
- SMS/MMS
- Calendar
- Contacts

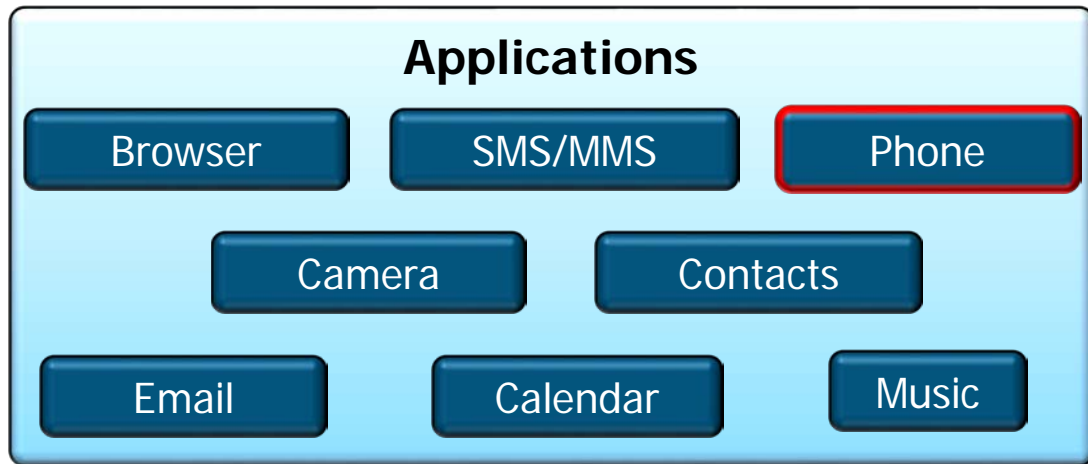


- Facilitate the recording, tracking, & management of certain types of personal information about friends, family, & colleagues

See [packages/apps/Contacts](https://developer.android.com/packages/apps/Contacts)

Overview of Common Android Apps

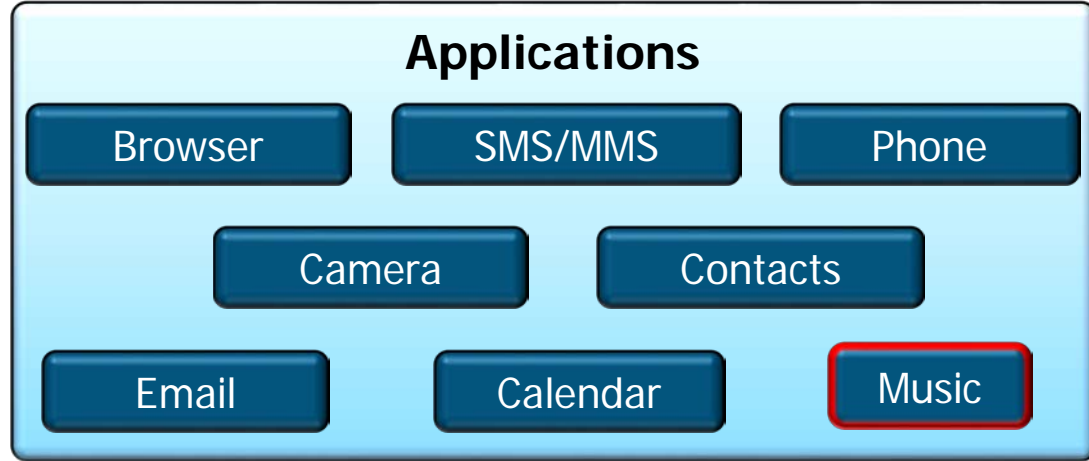
- On the top of Android's software stack are apps we use every day, e.g.
 - Browser
 - Email
 - Camera
 - SMS/MMS
 - Calendar
 - Contacts
 - Phone
 - Allows user to place & receive phone calls



See [packages/apps/Phone](https://developer.android.com/packages/apps/Phone)

Overview of Common Android Apps

- On the top of Android's software stack are apps we use every day, e.g.
 - Browser
 - Email
 - Camera
 - SMS/MMS
 - Calendar
 - Contacts
 - Phone
 - Music
 - Select & play audio/music files on a device



See [packages/apps/Music](https://developer.android.com/packages/apps/Music)

Overview of Common Android Apps

- On the top of Android's software stack are apps we use every day, e.g.

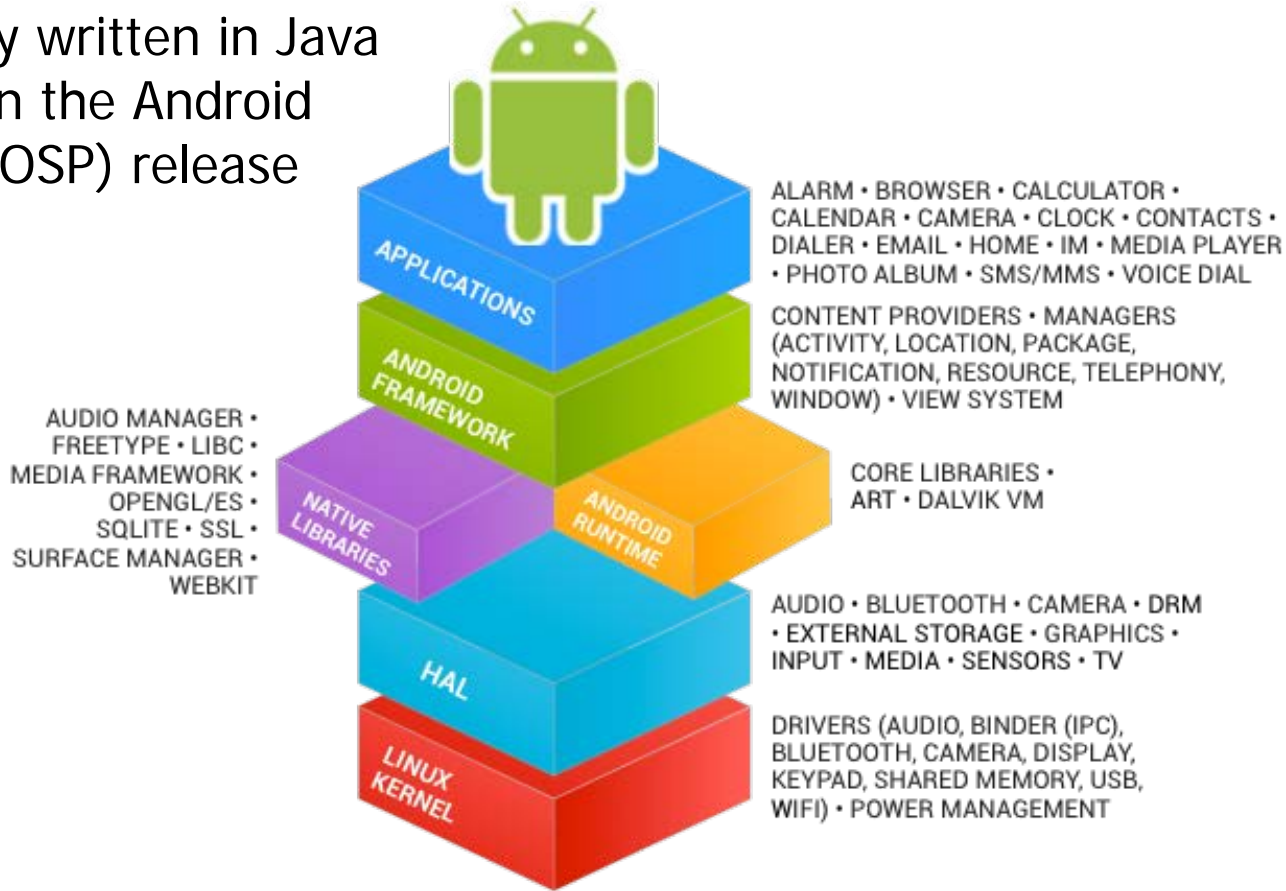
- Browser
- Email
- Camera
- SMS/MMS
- Calendar
- Contacts
- Phone
- Music
- etc.

[47 Directory Files]		
BasicSmsReceiver	Gallery	PhoneCommon
Bluetooth	Gallery2	Protips
Browser	HTMLViewer	Provision
Calculator	InCallUI	QuickSearchBox
Calendar	KeyChain	Settings
Camera	Launcher2	SmartCardService
Camera2	Launcher3	SoundRecorder
CellBroadcastReceiver	LegacyCamera	SpareParts
CertInstaller	ManagedProvisioning	SpeechRecorder
Contacts	Mms	Stk
ContactsCommon	Music	Tag
DeskClock	MusicFX	Terminal
Dialer	Nfc	TvSettings
Email	OneTimeInitializer	UnifiedEmail
Exchange	PackageInstaller	VoiceDialer
FMRadio	Phone	

See [packages/apps](#)

Overview of Common Android Apps

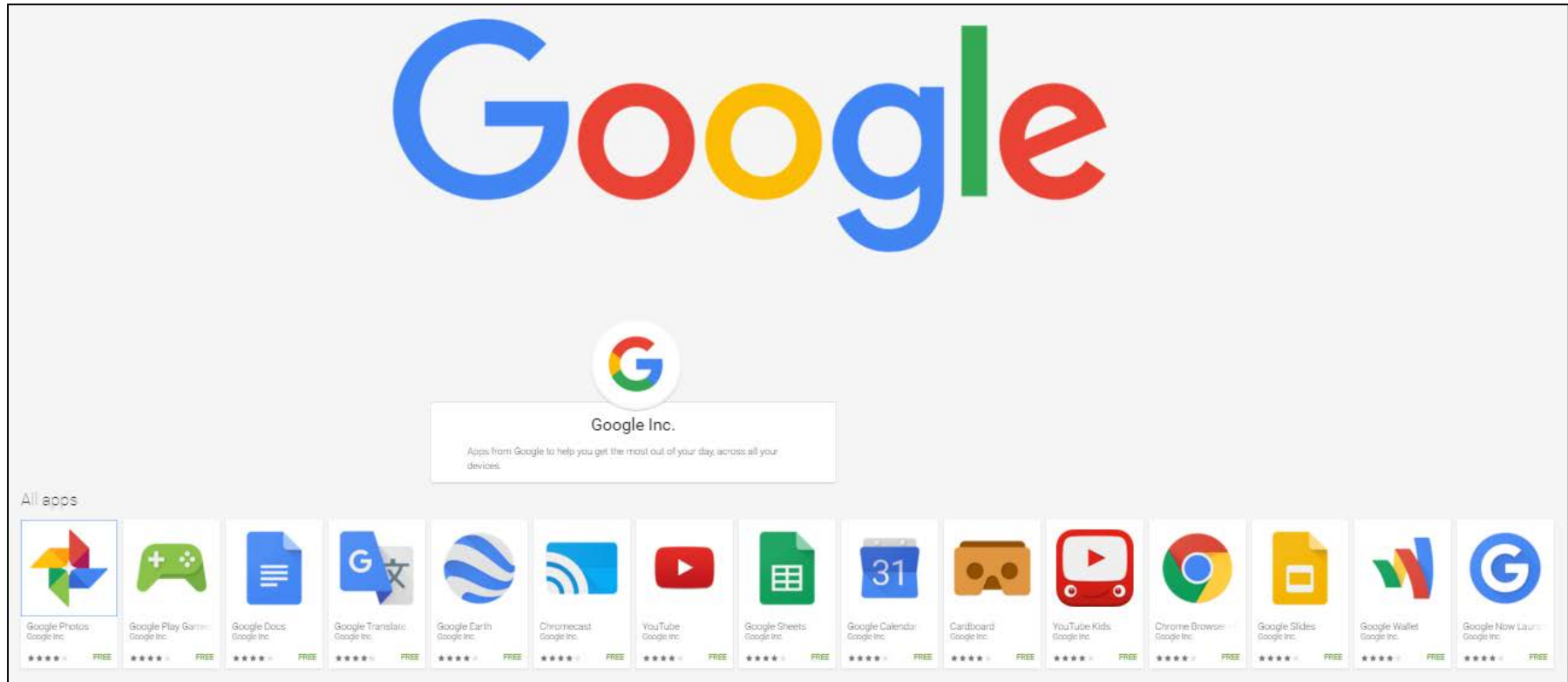
- Android apps are largely written in Java & many pre-packaged in the Android Open Source Project (AOSP) release



See source.android.com/source

Overview of Common Android Apps

- Other popular apps are not available in open-source form



See play.google.com/store/apps/dev?id=5700313618786177705

Packaging & Distributing Android Apps

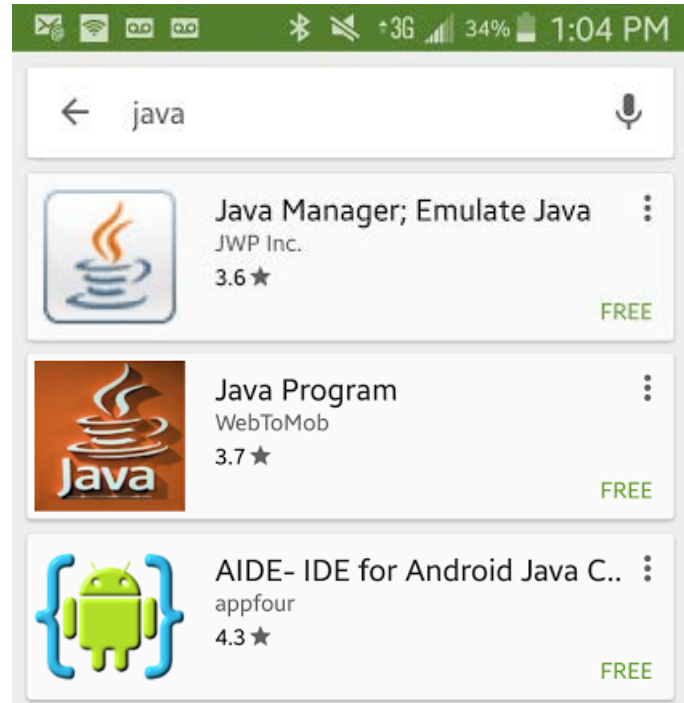
- Apps are packaged into “APK” archives that contain certain files & directories
 - All the app’s code
 - e.g., .dex files
 - Resources
 - e.g., binary versions of the XML code
 - Assets
 - e.g., logos
 - Certificates
 - e.g., for authentication
 - An AndroidManifest
 - e.g., defines all the app components



See en.wikipedia.org/wiki/Android_application_package

Packaging & Distributing Android Apps

- You can distribute your apps via the Google Play Store



See play.google.com/store/apps?hl=en

Packaging & Distributing Android Apps

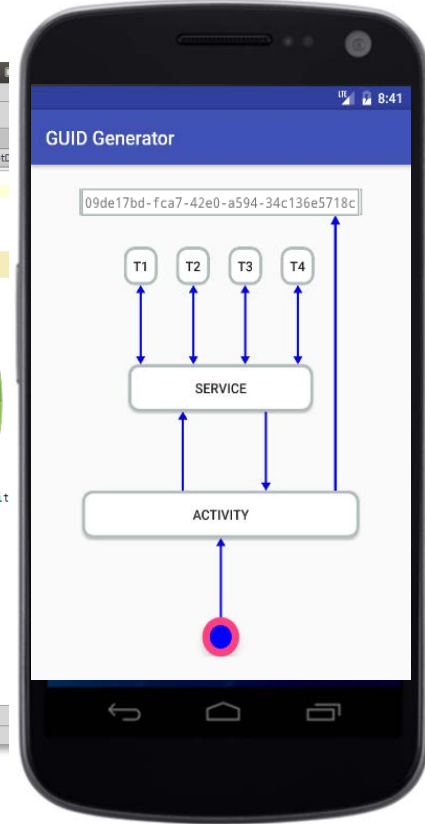
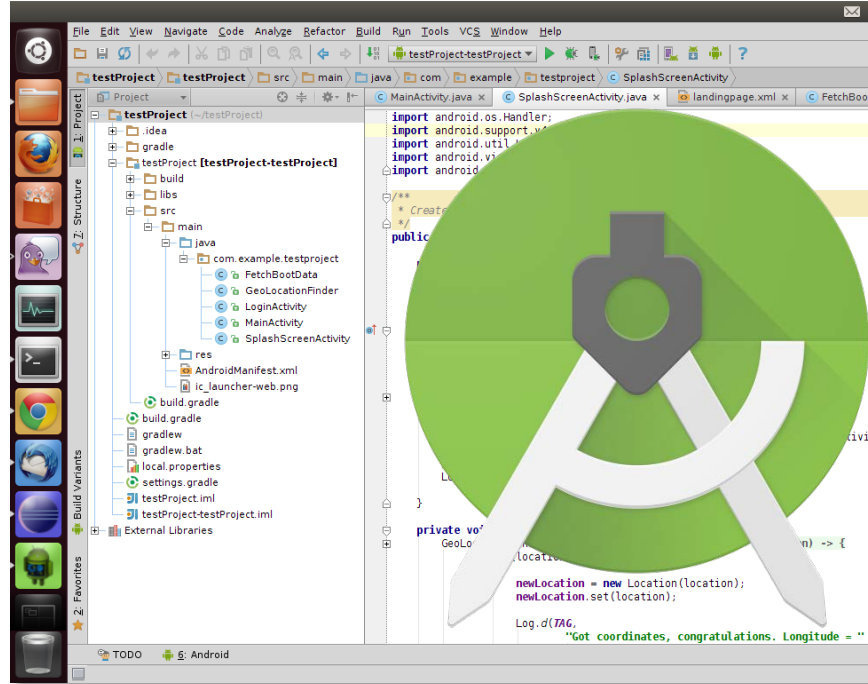
- You can distribute your apps via the Google Play Store
- Google documents best practices & useful tips to help distribute successful apps on the Google Play Store



See developer.android.com/distribute/googleplay/guide.html

Packaging & Distributing Android Apps

- You can built/distribute apps once you've mastered the material we cover!



You'll use this knowledge extensively when developing & deploying your own apps!

End of Common Services & Apps (Part 2): Service Frameworks & Packaged Apps