

Android & Java Frameworks: Introduction



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

- Understand how software frameworks are used in Android & Java



Overview of Frameworks in Android

Overview of Frameworks in Android & Java

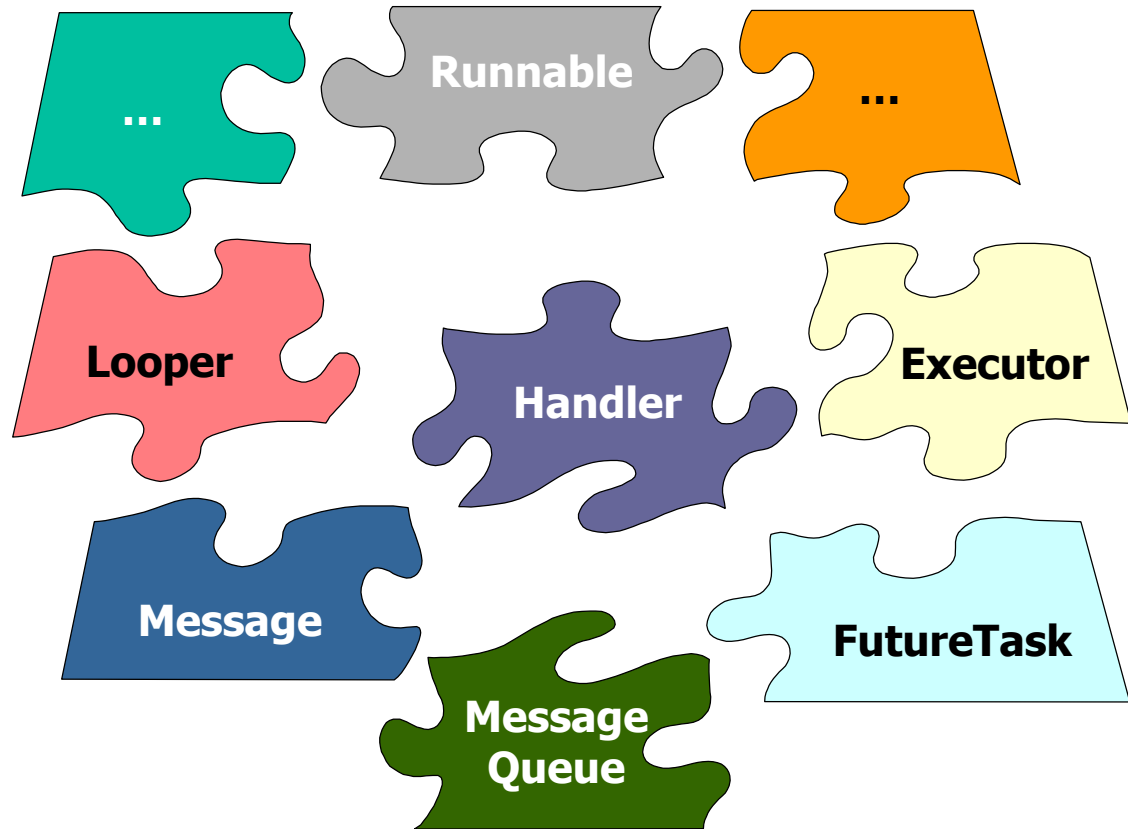
- A framework is an integrated set of components that provide a reusable architecture for a family of related apps



See www.dre.vanderbilt.edu/~schmidt/frameworks.html

Overview of Frameworks in Android & Java

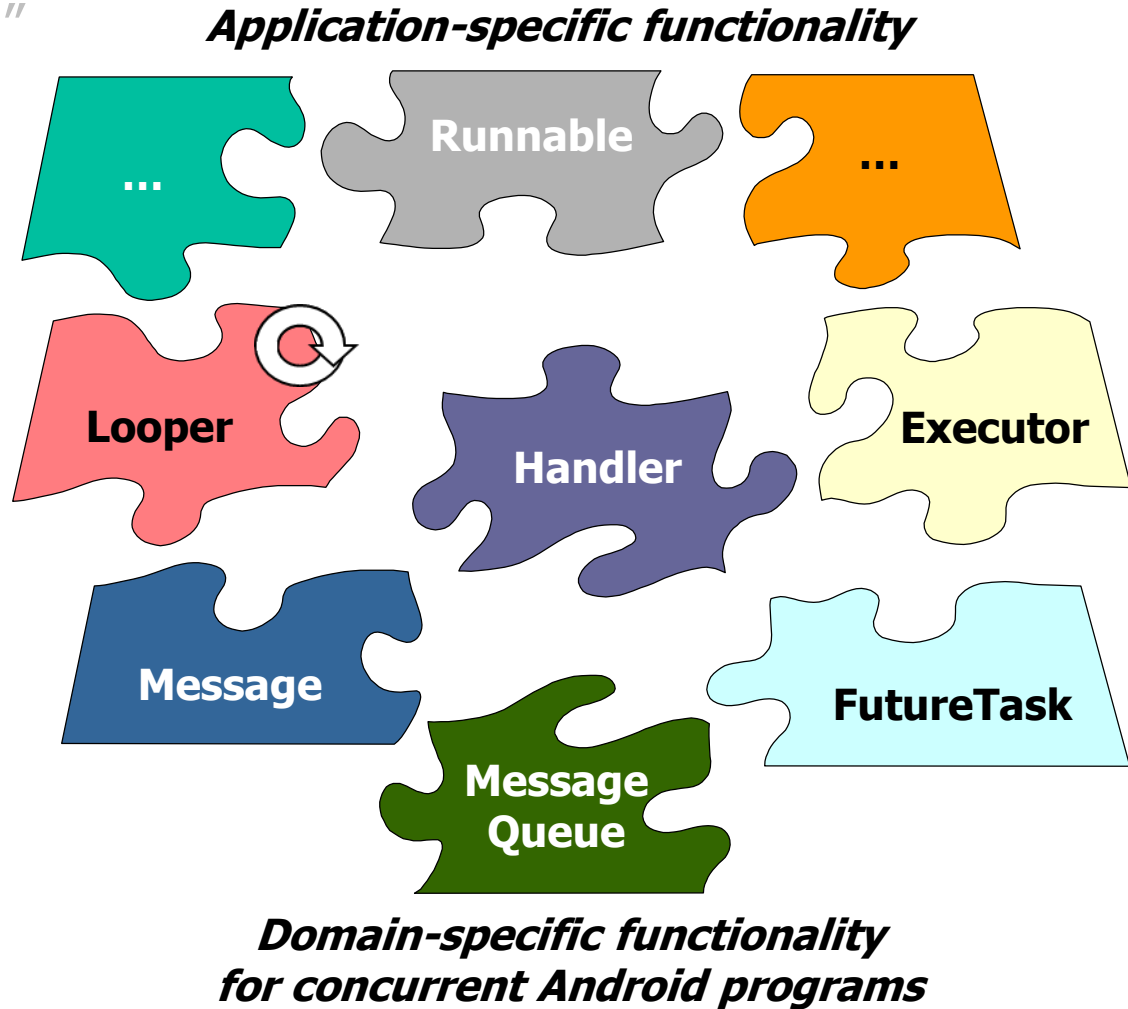
- A framework is an integrated set of components that provide a reusable architecture for a family of related apps & exhibits three key characteristics
 - Exhibit “inversion of control” (IoC) via callbacks



See en.wikipedia.org/wiki/Inversion_of_control & [en.wikipedia.org/wiki/Callback_\(computer_programming\)](https://en.wikipedia.org/wiki/Callback_(computer_programming))

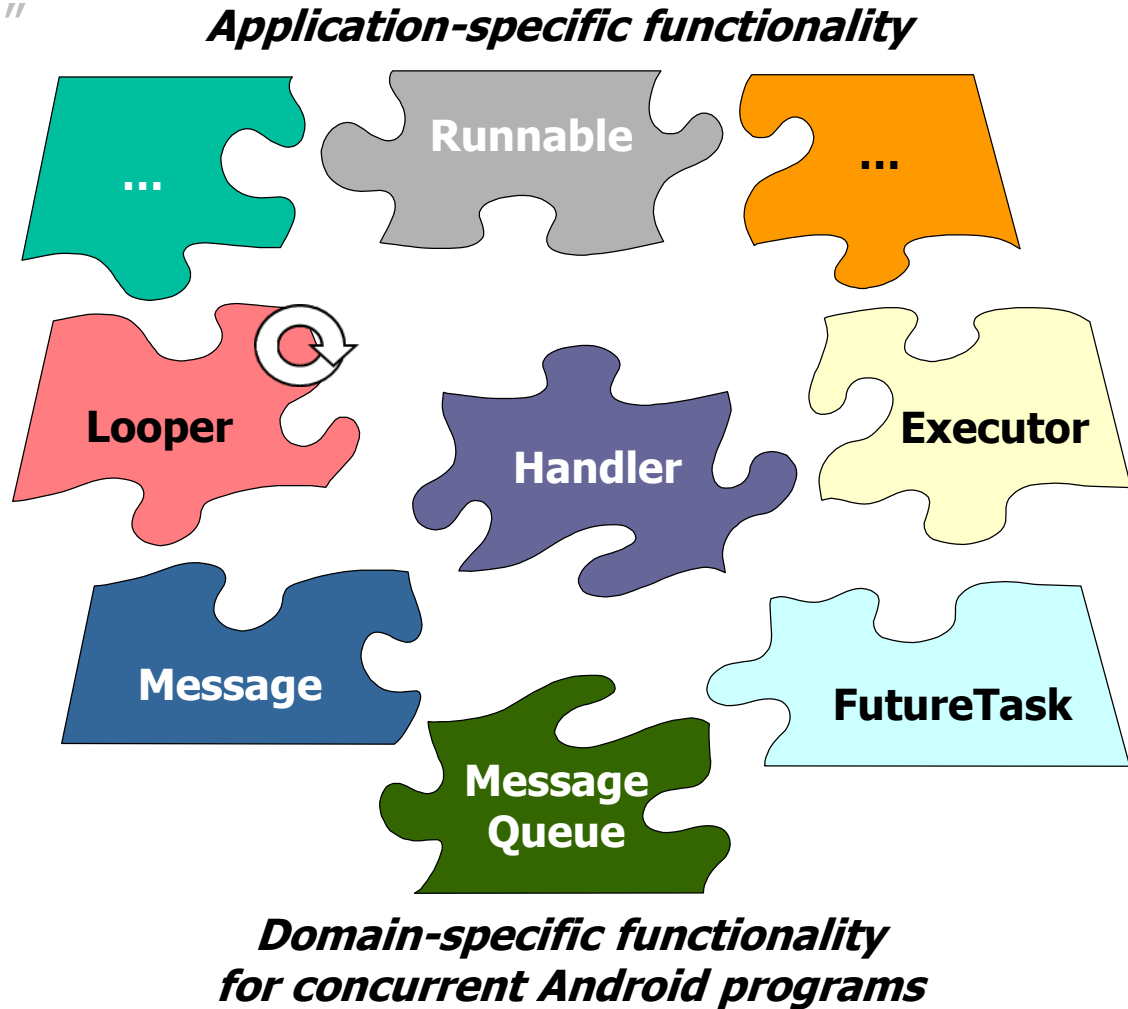
Overview of Frameworks in Android & Java

- A framework is an integrated set of components that provide a reusable architecture for a family of related apps & exhibits three key characteristics
 - Exhibit “inversion of control” (IoC) via callbacks
 - Integrated domain-specific structure & functionality



Overview of Frameworks in Android & Java

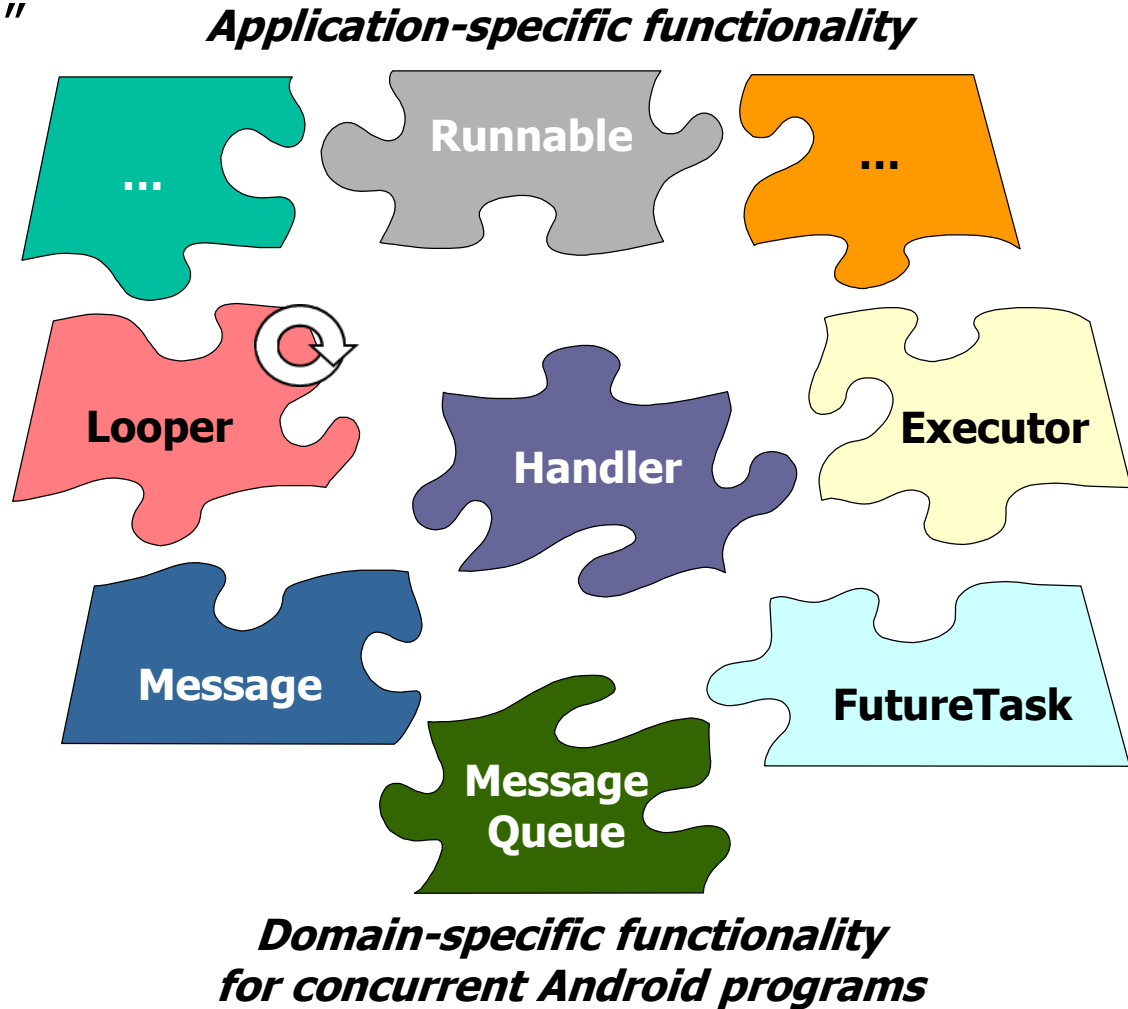
- A framework is an integrated set of components that provide a reusable architecture for a family of related apps & exhibits three key characteristics
 - Exhibit “inversion of control” (IoC) via callbacks
 - Integrated domain-specific structure & functionality
 - Provide semi-complete (portions of) apps



See www.laputan.org/drc/drc.html

Overview of Frameworks in Android & Java

- A framework is an integrated set of components that provide a reusable architecture for a family of related apps & exhibits three key characteristics
 - Exhibit “inversion of control” (IoC) via callbacks
 - Integrated domain-specific structure & functionality
 - Provide semi-complete (portions of) apps



See next part of lesson on “*Android & Java Frameworks: Key Characteristics*”

Overview of Frameworks in Android & Java

- Android & Java provide many frameworks

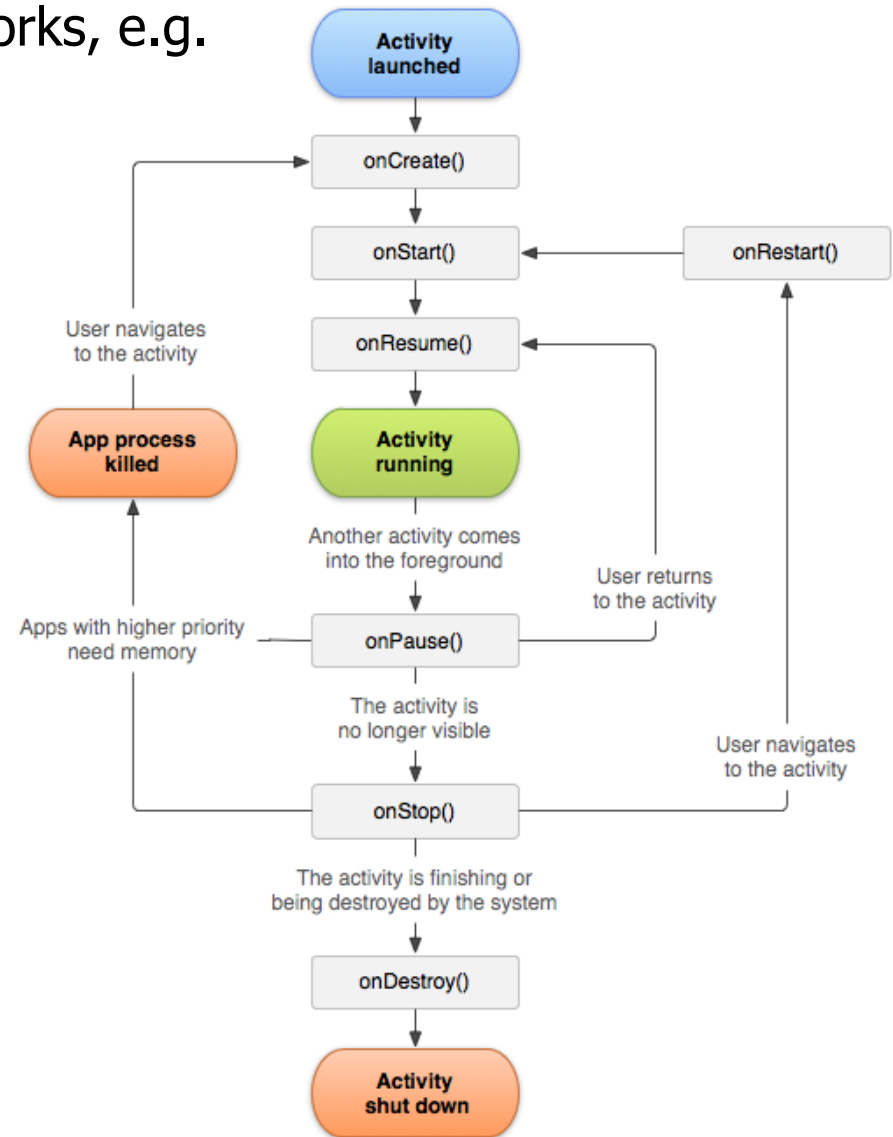


Overview of Frameworks in Android & Java

- Android & Java provide many frameworks, e.g.

- Android**

- Android's Activity framework manages lifecycle hook methods dispatched in the UI thread



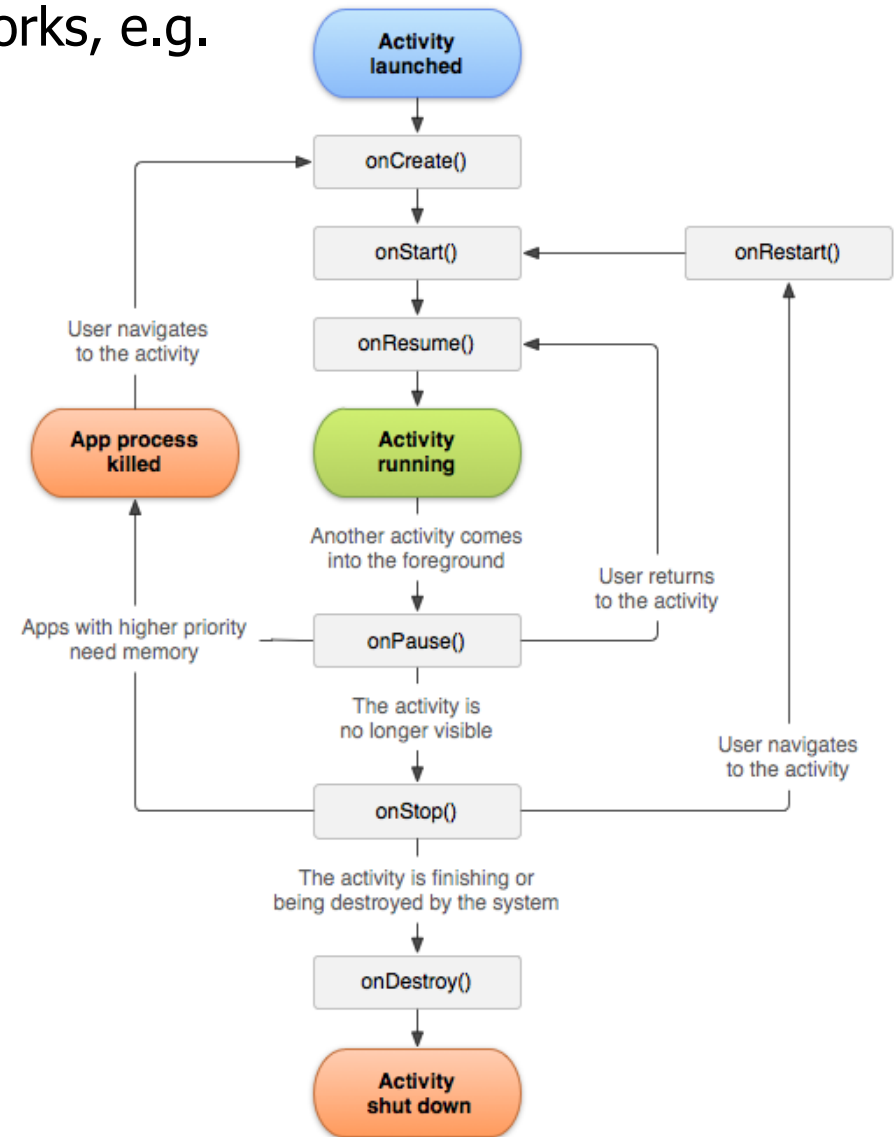
See developer.android.com/training/multiple-threads/communicate-ui.html

Overview of Frameworks in Android & Java

- Android & Java provide many frameworks, e.g.

- Android**

- Android's Activity framework manages lifecycle hook methods dispatched in the UI thread
- e.g., onCreate(), onStart(), onStop(), onDestroy(), etc.



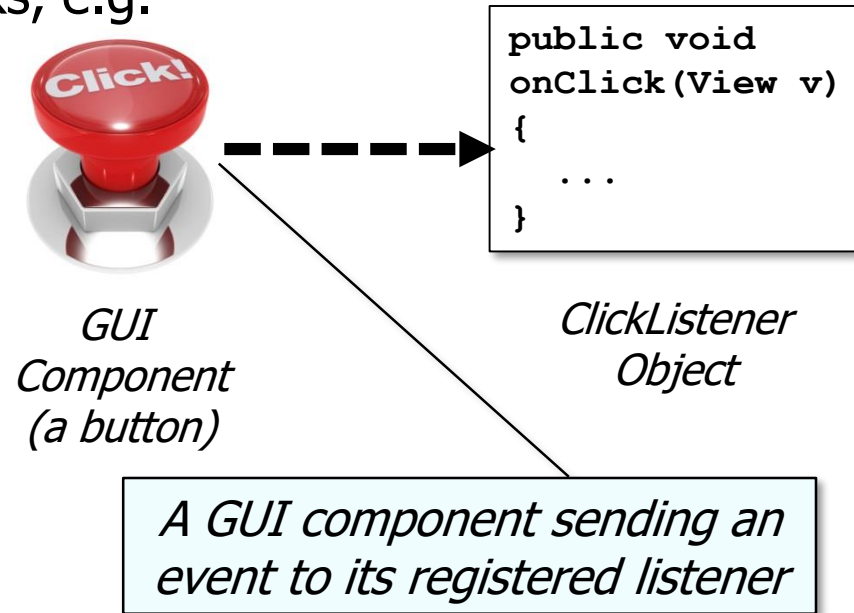
See developer.android.com/training/basics/activity-lifecycle

Overview of Frameworks in Android & Java

- Android & Java provide many frameworks, e.g.

- Android**

- Android's Activity framework manages lifecycle hook methods dispatched in the UI thread
- A listener for button clicks is called back by Android's GUI framework



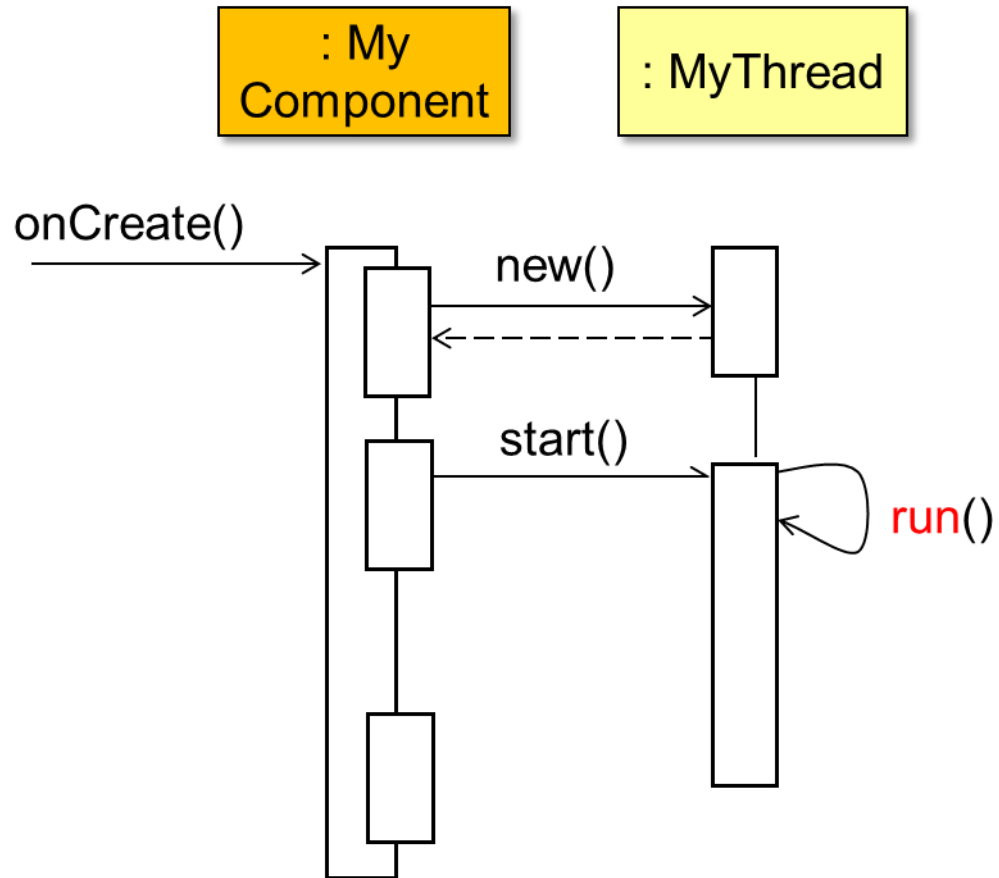
Overview of Frameworks in Android & Java

- Android & Java provide many frameworks, e.g.

- Android**

- Java**

- A thread invokes the run() hook method of a runnable



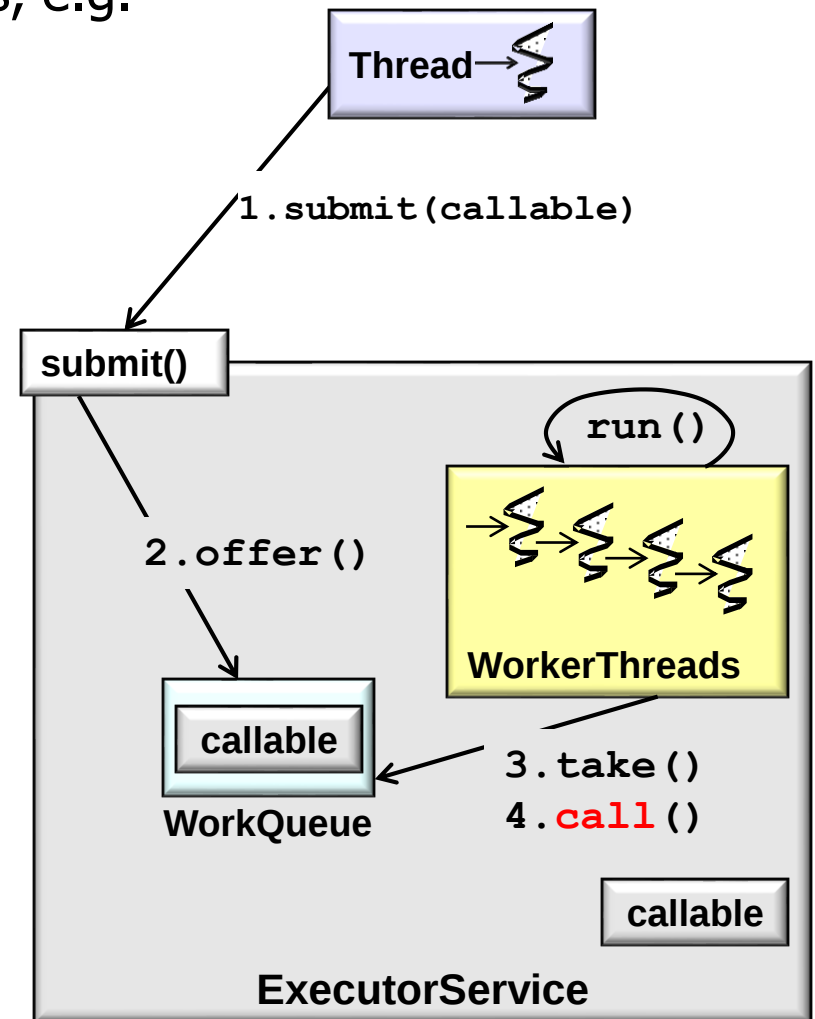
Overview of Frameworks in Android & Java

- Android & Java provide many frameworks, e.g.

- Android**

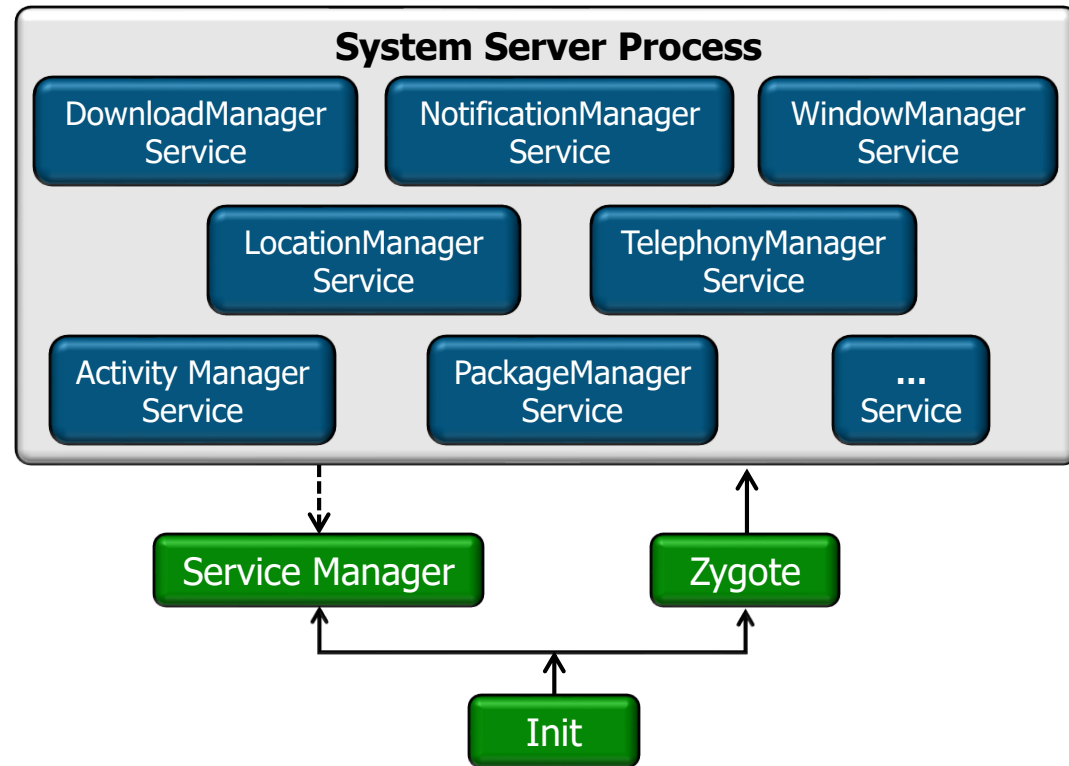
- Java**

- A thread invokes the `run()` hook method of a runnable
- The `ExecutorService` invokes the `call()` hook method of a callable



Overview of Frameworks in Android & Java

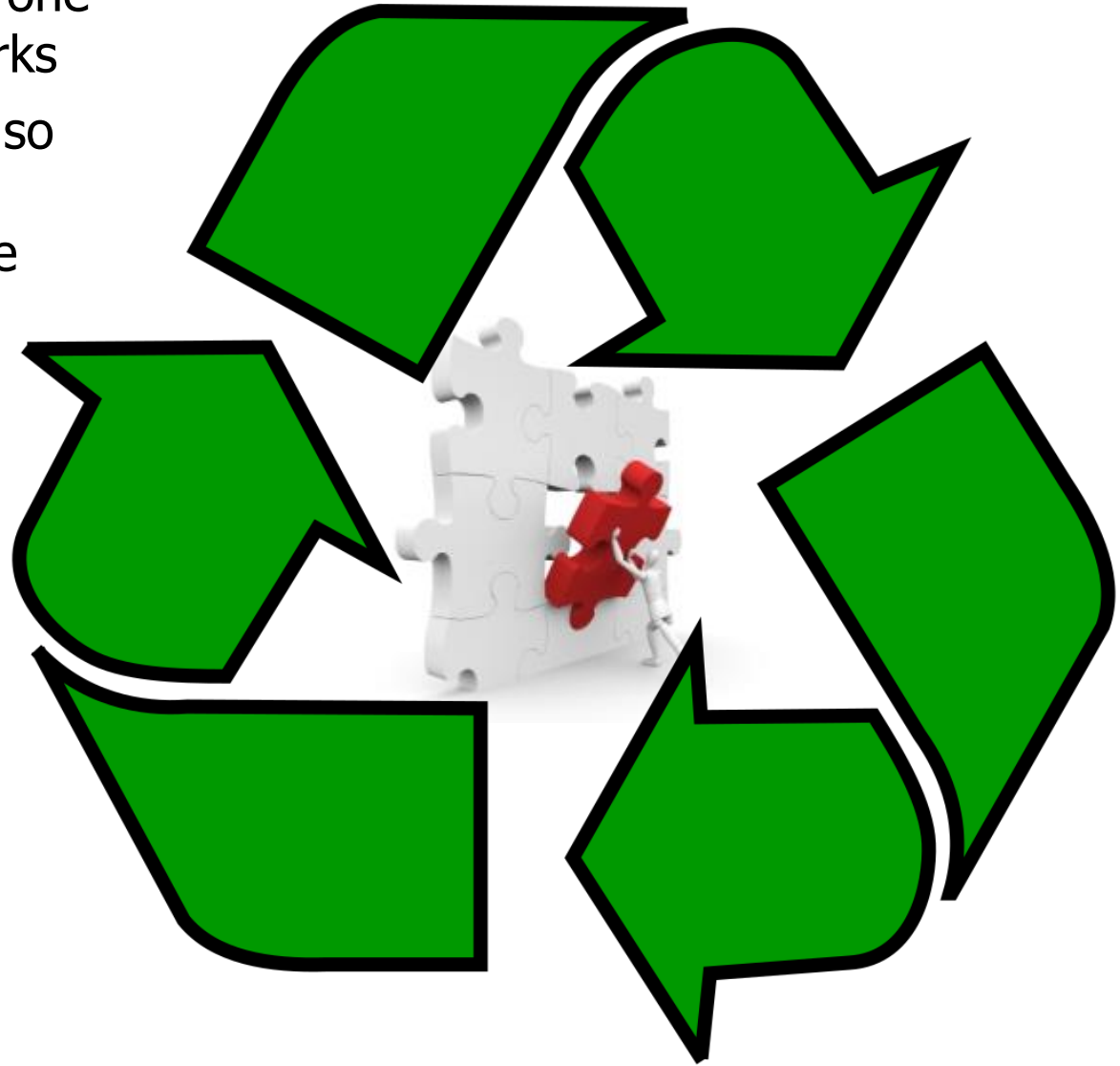
- *All* Android apps run inside one or more software frameworks



Your mobile apps must use multiple Android frameworks

Overview of Frameworks in Android & Java

- All Android apps run inside one or more software frameworks
- The motivation for using so many frameworks is to enhance systematic reuse



See en.wikipedia.org/wiki/Code_reuse#Systematic_software_reuse

Overview of Frameworks in Android & Java

- *All* Android apps run inside one or more software frameworks
 - The motivation for using so many frameworks is to enhance systematic reuse
 - Apps (& app developers) thus don't need to "reinvent the wheel"



See en.wikipedia.org/wiki/Reinventing_the_wheel

Overview of Frameworks in Android & Java

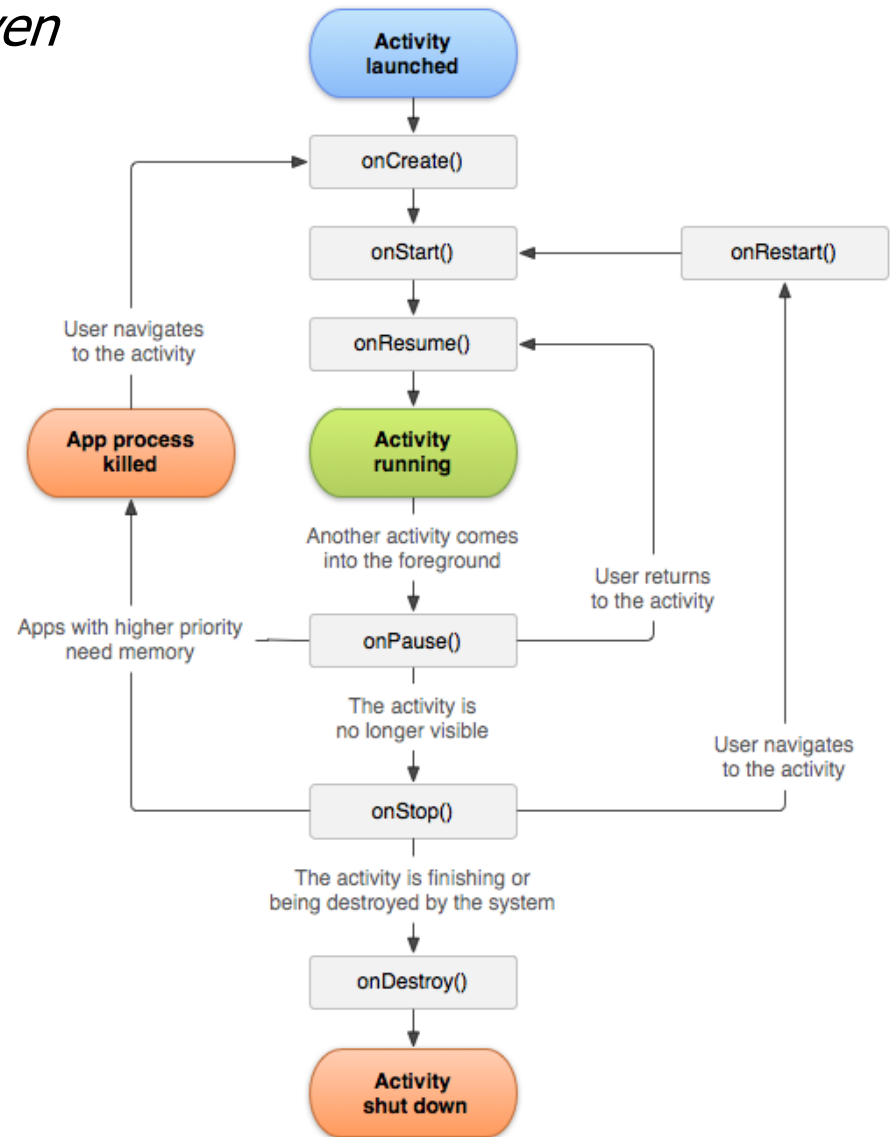
- Android frameworks use an *event-driven programming model* to integrate app classes into them



See en.wikipedia.org/wiki/Event-driven_programming

Overview of Frameworks in Android & Java

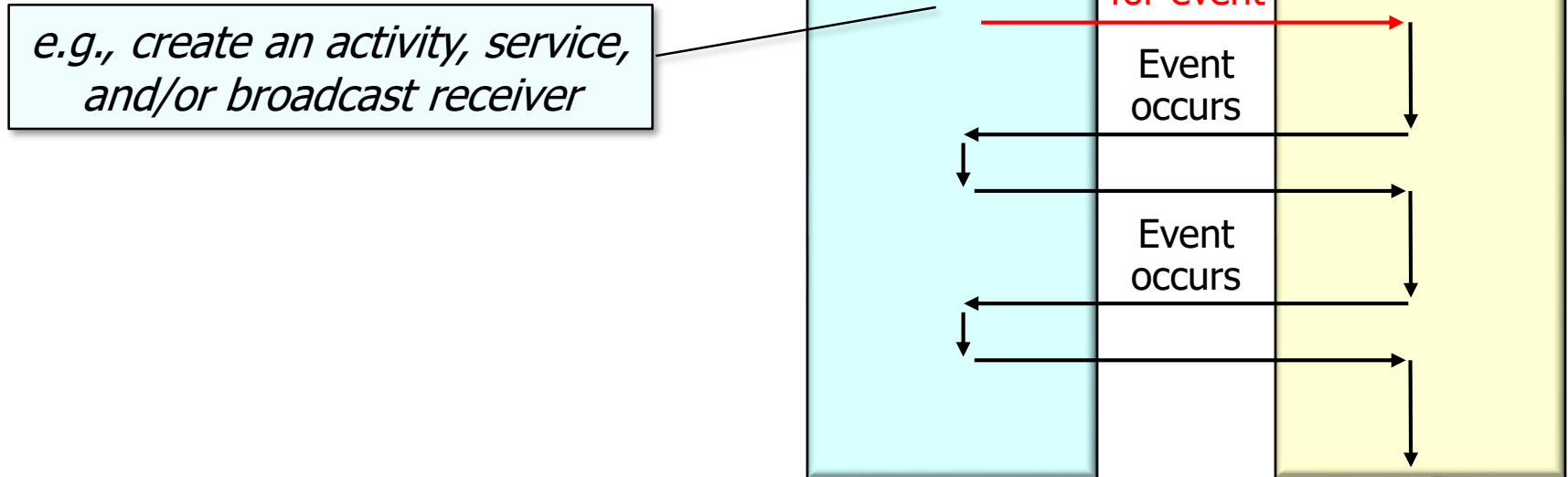
- Android frameworks use an *event-driven programming model* to integrate app classes into them
- In this programming paradigm, the flow of the program is determined by events, e.g.
 - User actions (button presses, gestures, etc.)
 - Sensor inputs/outputs
 - Messages from other threads



See en.wikipedia.org/wiki/Event-driven_programming

Overview of Frameworks in Android & Java

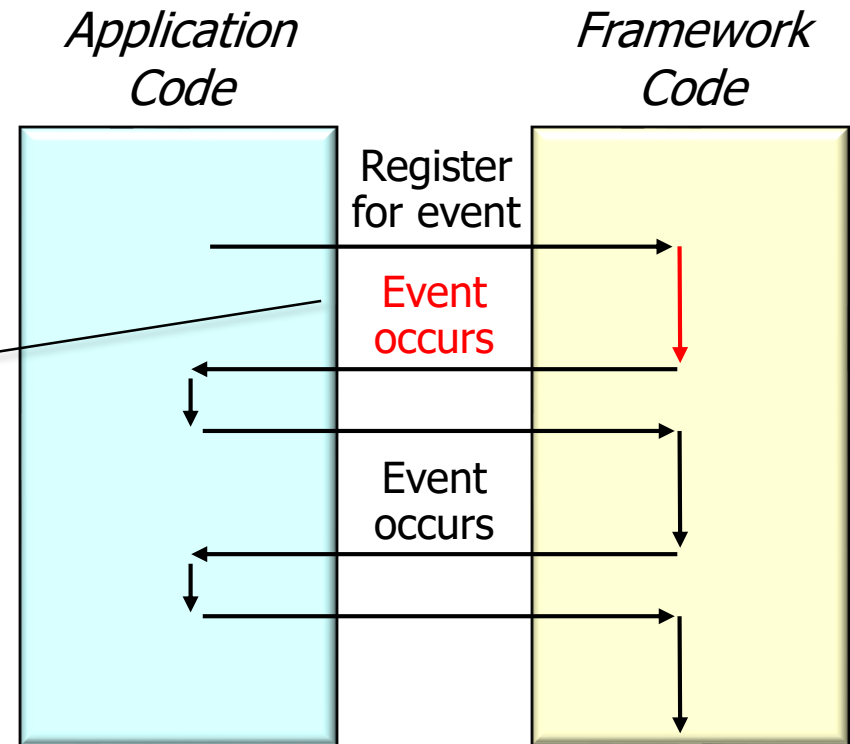
- The control flow in framework-driven Android apps traverses between the framework(s) & the app classes



Overview of Frameworks in Android & Java

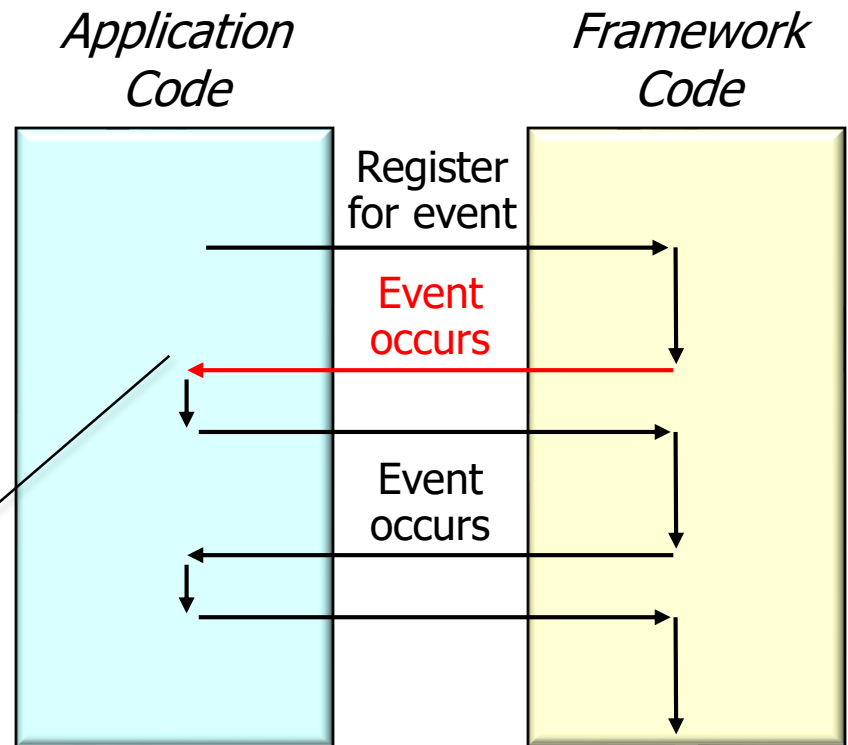
- The control flow in framework-driven Android apps traverses between the framework(s) & the app classes

e.g., a message occurs that triggers a lifecycle event



Overview of Frameworks in Android & Java

- The control flow in framework-driven Android apps traverses between the framework(s) & the app classes
- A framework calls to app code when an event of interest occurs

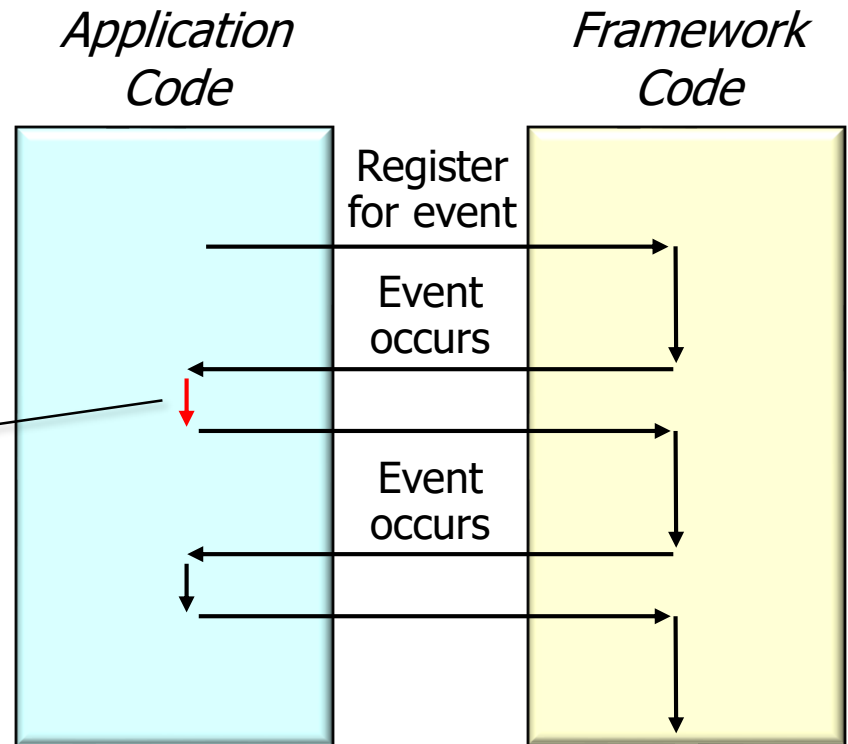


e.g., dispatch lifecycle hook methods like `onCreate()`, `onStart()`, `onReceive()`, etc.

Overview of Frameworks in Android & Java

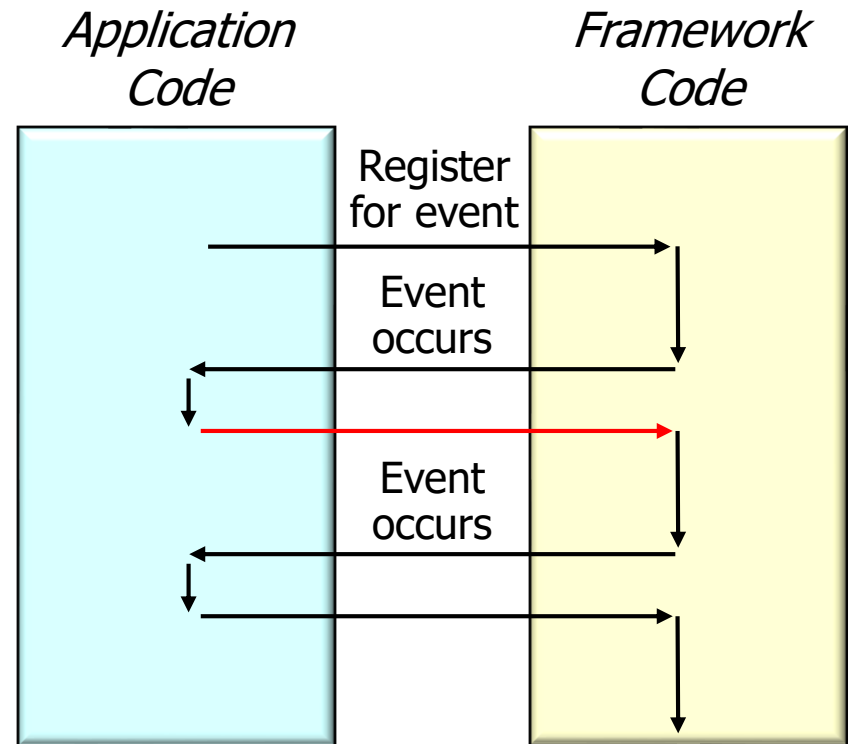
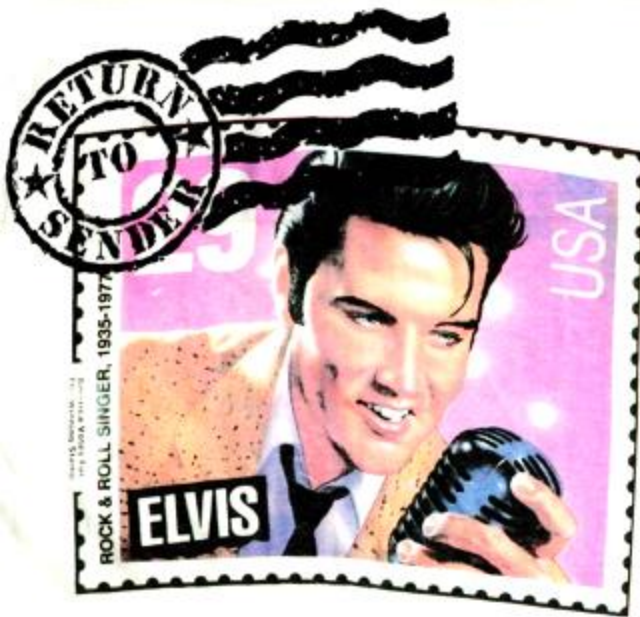
- The control flow in framework-driven Android apps traverses between the framework(s) & the app classes
- A framework calls to app code when an event of interest occurs

e.g., the app performs its processing in the context of framework thread(s)



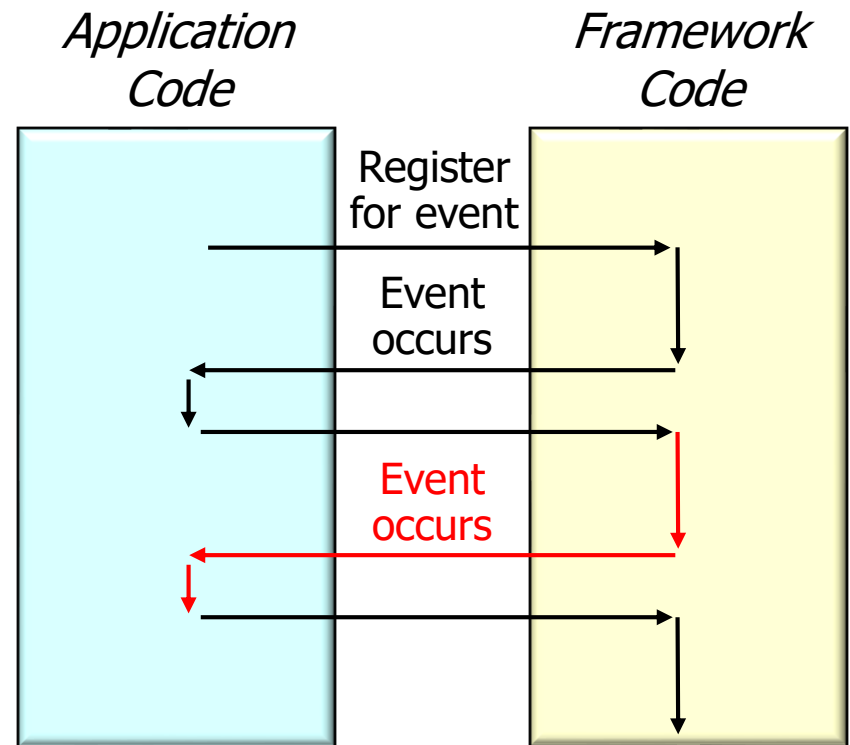
Overview of Frameworks in Android & Java

- The control flow in framework-driven Android apps traverses between the framework(s) & the app classes
 - A framework calls to app code when an event of interest occurs
 - Control returns to the framework after an app callback is done



Overview of Frameworks in Android & Java

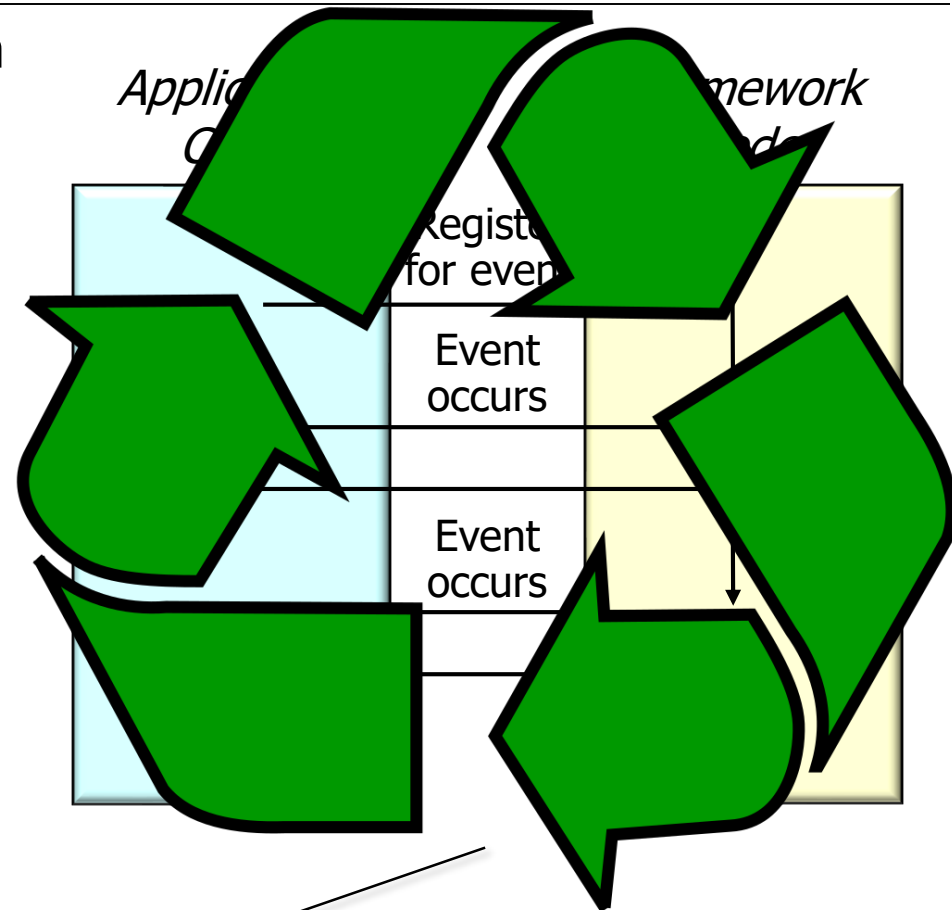
- The control flow in framework-driven Android apps traverses between the framework(s) & the app classes
 - A framework calls to app code when an event of interest occurs
 - Control returns to the framework after an app callback is done
- Lather, rinse, repeat until app is done ...



See en.wikipedia.org/wiki/Lather,_rinse,_repeat

Overview of Frameworks in Android & Java

- The control flow in framework-driven Android apps traverses between the framework(s) & the app classes
 - A framework calls to app code when an event of interest occurs
 - Control returns to the framework after an app callback is done
 - Lather, rinse, repeat until app is done ...



Separating concerns this way helps enhance systematic software reuse & improve various quality attributes

See www.sei.cmu.edu/reports/95tr021.pdf

End of Android & Java Frameworks: Introduction