Programming Handheld Systems

Adam Porter
The Fragment Class
Tablet UIs

Tablets have larger displays than phones do.

They can support multiple UI panes / user behaviors at the same time.

The 1 activity – 1 thing the user can do heuristic may not work for larger devices.
QuoteViewer

Application uses two Activities

One shows titles of Shakespeare plays & allows user to select one title

The other shows a quote from selected play
The QuoteViewer UI

This layout is reasonable on a phone
But inefficient on a larger device
Better Layout

Use two cooperating layouts units on one screen
Fragment

Represents a behavior / portion of UI within an Activity

Multiple Fragments can be embedded in an Activity to create a multi-pane UI

A single Fragment can be reused across multiple Activities
Fragment Lifecycle

Fragment lifecycle is coordinated with the lifecycle of its containing Activity.

Fragments have their own lifecycles and receive their own callbacks.
Fragment Lifecycle States

Resumed
  Fragment is visible in the running activity

Paused
  Another activity is in the foreground and has focus, containing activity is visible

Stopped
  The fragment is not visible
Activity Created: onAttach()

Fragment is first attached to its activity
onCreate()

Initialize the fragment
**onCreateView()**

**Fragment sets up & returns its user interface**

- onAttach()
- onCreate()
- onCreateView()
onActivityCreated() 

Containing activity has completed its onCreate() & fragment has been installed
Activity Started: OnStart()

Hosting Activity about to become visible onStart ()
Activity Resumed: onResume()

Hosting Activity is about to become visible and ready for user interaction
Activity Paused: onPause()

Hosting Activity is visible, but does not have focus
Activity Stopped: onStop()

Hosting Activity is no longer visible
Activity Destroyed: onDestroyView()

View previously created by onCreateView() has been detached from the Fragment

Typical Actions

Clean up view resources
onDestroy()

Fragment is no longer in use

Typical Actions

Clean up Fragment resources

onDestroyView()

onDestroy()
**onDetach()**

**Fragment no longer attached to its activity**

**Typical Actions**
- Null out references to Hosting Activity

Diagram:
- onDestroyView()
- onDestroy()
- on_detach()
Adding Fragments to Activities

Two general ways to add Fragments to an Activity’s layout

Declare it statically in the Activity’s layout file

Add it programmatically using the fragmentManager
Fragment Layout

Layout can be inflated/implemented in onCreateView()
onCreateView() must return the View at the root of the Fragment's layout
This View is added to the containing Activity
Hamlet
King Lear
Julius Ceasar

Now cracks a noble heart. Good-night, sweet prince; And flights of angels sing thee to thy rest.
Adding Fragments Dynamically

While an Activity’s running you can add a Fragment to it’s layout.

Get reference to the FragmentManager

Begin a FragmentTransaction

Add the Fragment

Commit the FragmentTransaction
Dynamic Layout

Fragment transactions allow you to dynamically change your app’s user interface.

Can make the interface more fluid & take better advantage of available screen space.
Configuration Changes

If you call `setRetainInstance(true)`, Android won’t destroy the Fragment on configuration changes.
Configuration Changes

Results in some changes to lifecycle callback sequence
onDestroy() will not be called
onCreate() will not be called
FragmentStaticConfigLayout

Functionality is similar to previous examples
In landscape mode
Both Fragments use a large font
TitleFragment takes more horizontal space & allows long titles to span multiple lines
FragmentStaticConfigLayout

In portrait mode
Both Fragments use a smaller font
TitleFragment will use less space and will ellipsize long titles, limiting them to a single line
Next Time

User Interface classes