

Mobile Application Development

Higher Diploma in Science in Computer Science

Produced
by

Eamonn de Leastar (edelestar@wit.ie)

Department of Computing, Maths & Physics
Waterford Institute of Technology

<http://www.wit.ie>

<http://elearning.wit.ie>



Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRCE



Navigation Principles

(from developer.android.com)

Agenda

- Navigation Within an App
- Navigating Into an App
- Navigation Between Apps

Navigation Within an App

- Back and Up
- Changing Screens
- Sibling Screens

Back & Up

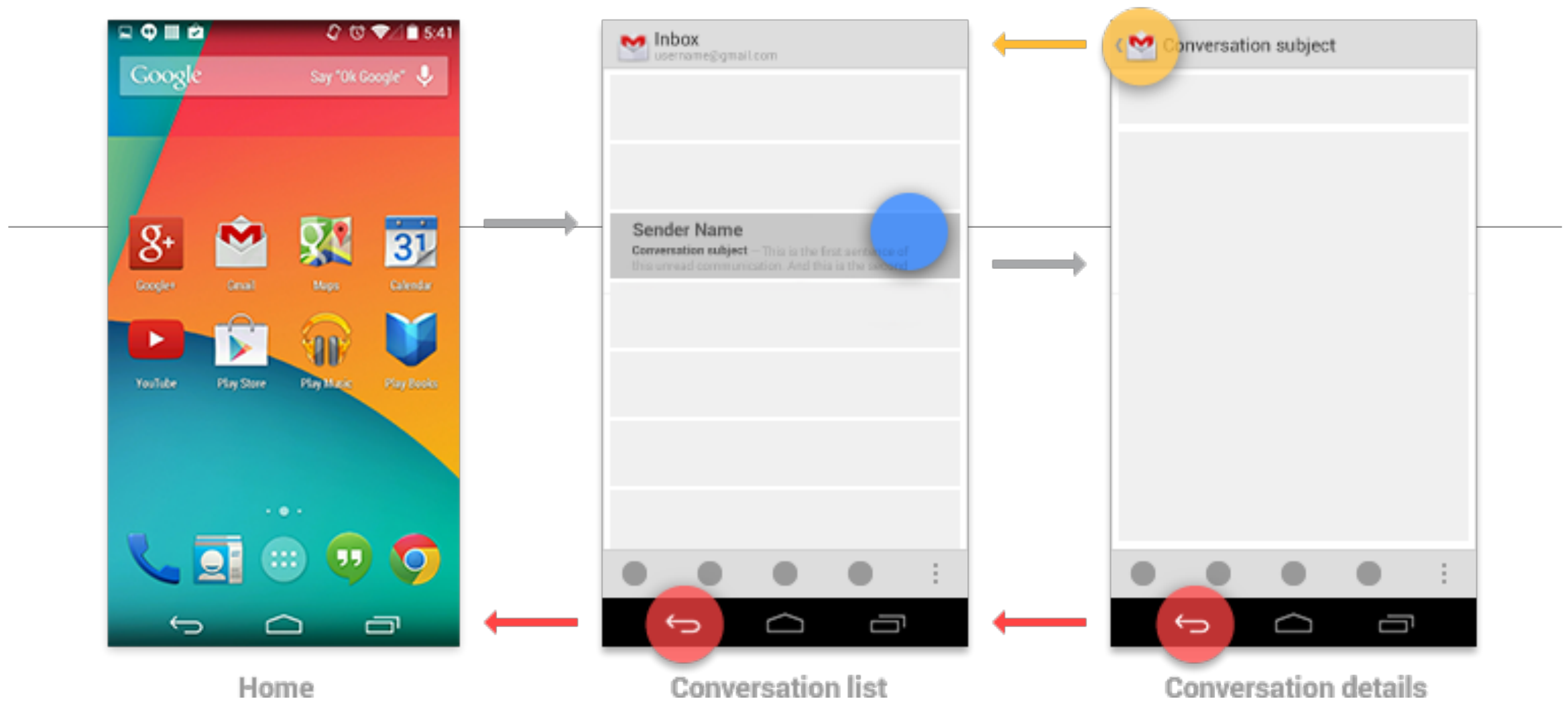
- The system Back button is used to navigate, in reverse chronological order, through the history of screens the user has recently worked with.



- It is generally based on the temporal relationships between screens, rather than the app's hierarchy.
- The Up button is used to navigate within an app based on the hierarchical relationships between screens.



- For instance, if screen A displays a list of items, and selecting an item leads to screen B (which presents that item in more detail), then screen B should offer an Up button that returns to screen A.
- If a screen is the topmost one in an app (that is, the app's home), it should not present an Up button.



- When the previously viewed screen is also the hierarchical parent of the current screen, pressing the Back button has the same result as pressing an Up button—this is a common occurrence.
- However, unlike the Up button, which ensures the user remains within your app, the Back button can return the user to the Home screen, or even to a different app.

Back Button - Behaviours



- The Back button also supports a few behaviors not directly tied to screen-to-screen navigation:
 - Dismisses floating windows (dialogs, popups)
 - Dismisses contextual action bars, and removes the highlight from the selected items
 - Hides the onscreen keyboard (IME)

Navigation Within Your App

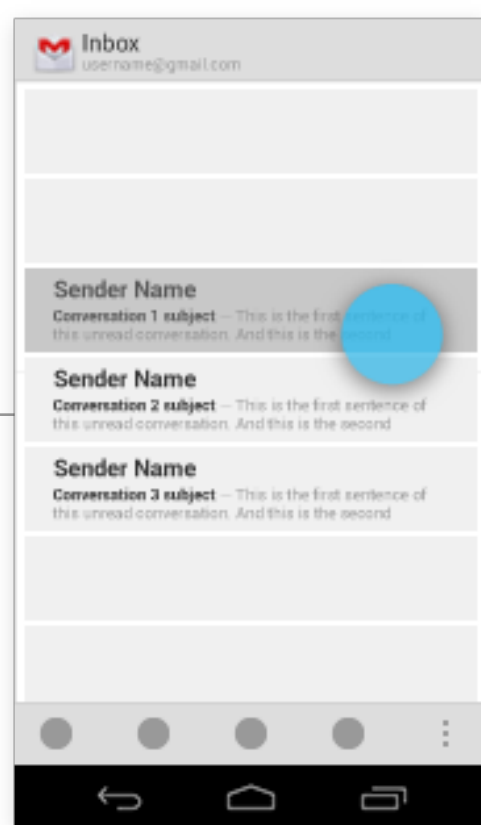
- Navigating to screens with multiple entry points
 - Sometimes a screen doesn't have a strict position within the app's hierarchy, and can be reached from multiple entry points — such as a settings screen that can be reached from any other screen in your app.
 - In this case, the Up button should choose to return to the referring screen, behaving identically to Back.

Hierarchy Exceptions

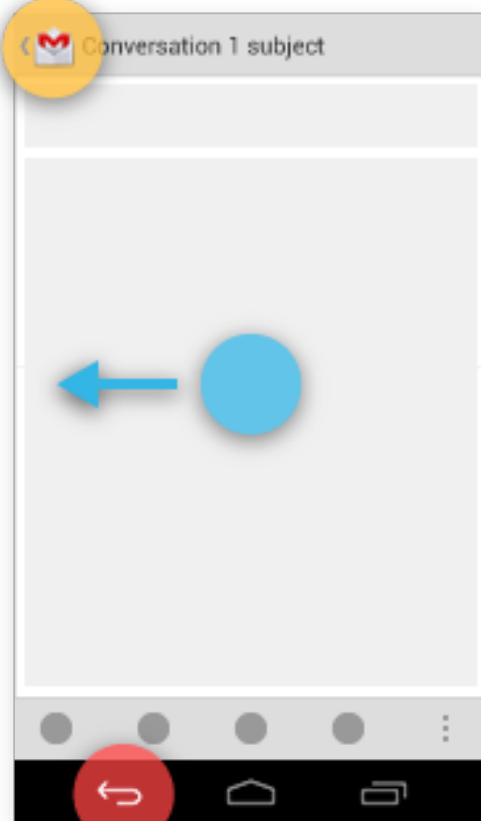
- Switching to some views may not necessarily change the behavior of Up or Back: with the screen remaining in the same place within the app's hierarchy, and no new navigation history is created.
- Examples of such view changes are:
 - Switching views using tabs and/or left-and-right swipes
 - Switching views using a dropdown (aka collapsed tabs)
 - Filtering a list
 - Sorting a list
 - Changing display characteristics (such as zooming)

Navigating between sibling screens

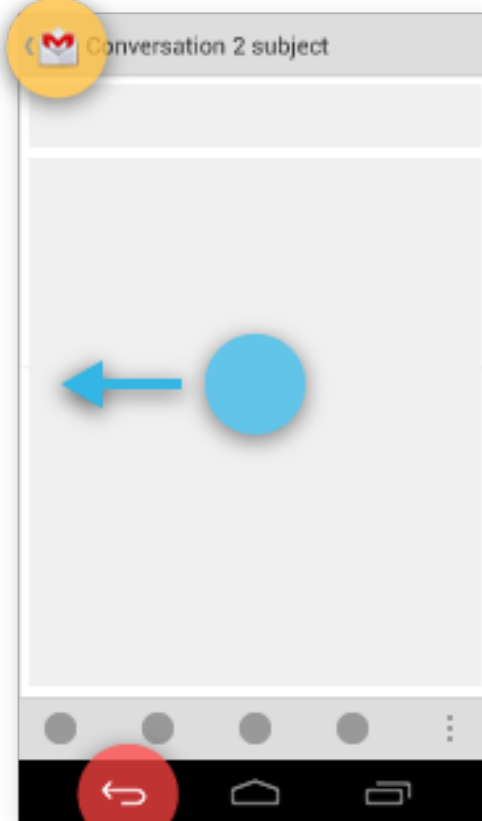
- When your app supports navigation from a list of items to a detail view of one of those items, it's often desirable to support direction navigation from that item to another one which precedes or follows it in the list.



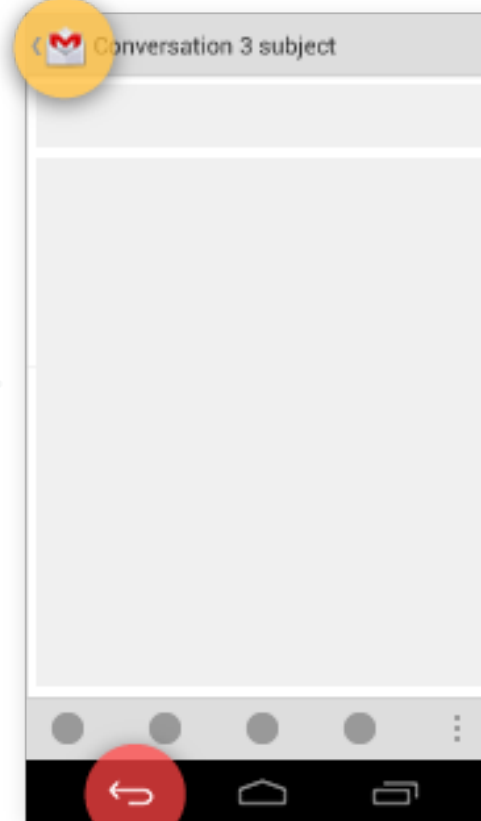
Conversation list



Conversation 1 details



Conversation 2 details



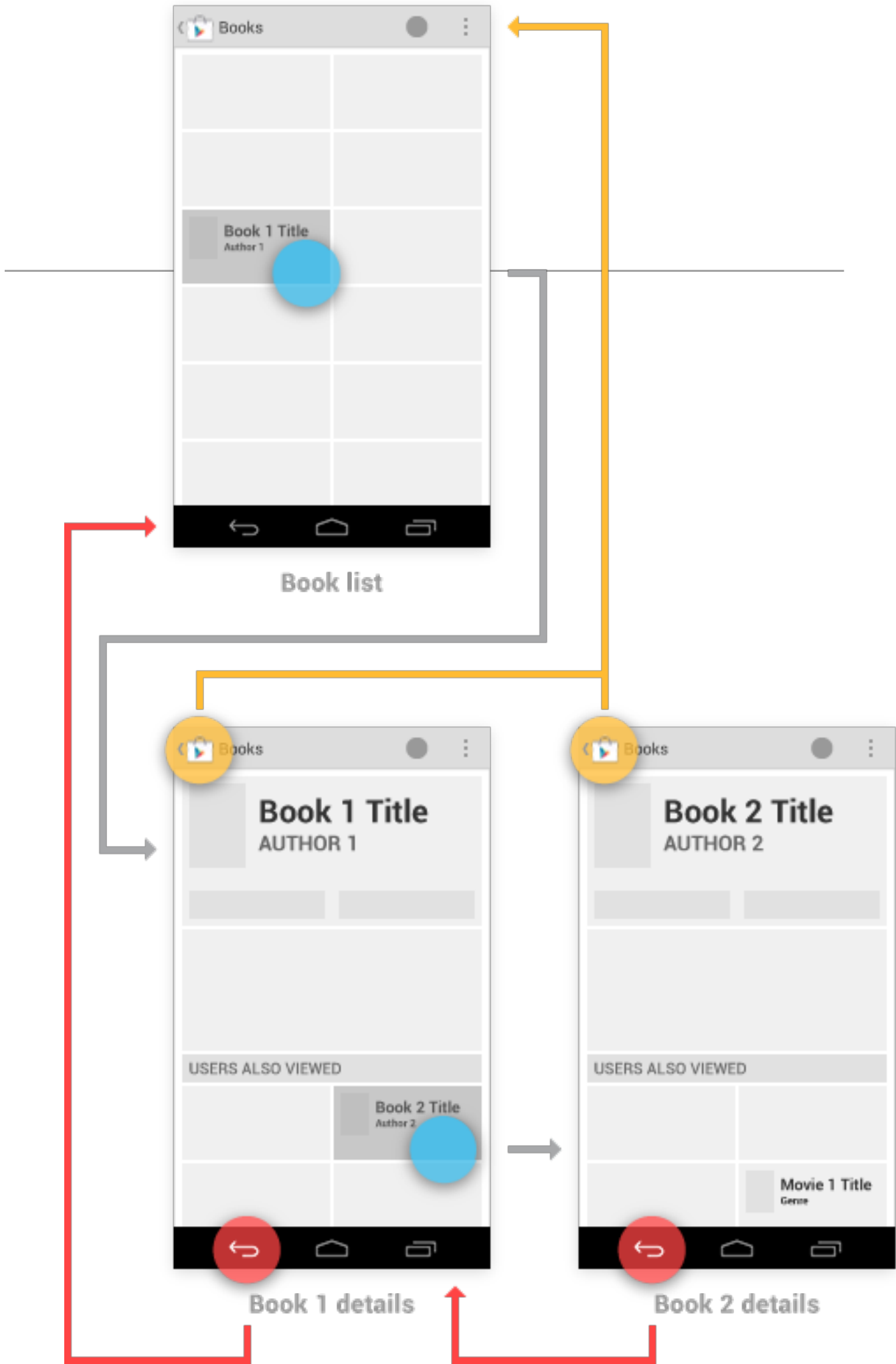
Conversation 3 details

- For example, in Gmail, it's easy to swipe left or right from a conversation to view a newer or older one in the same Inbox.
- Just as when changing view within a screen, such navigation does not change the behavior of Up or Back.

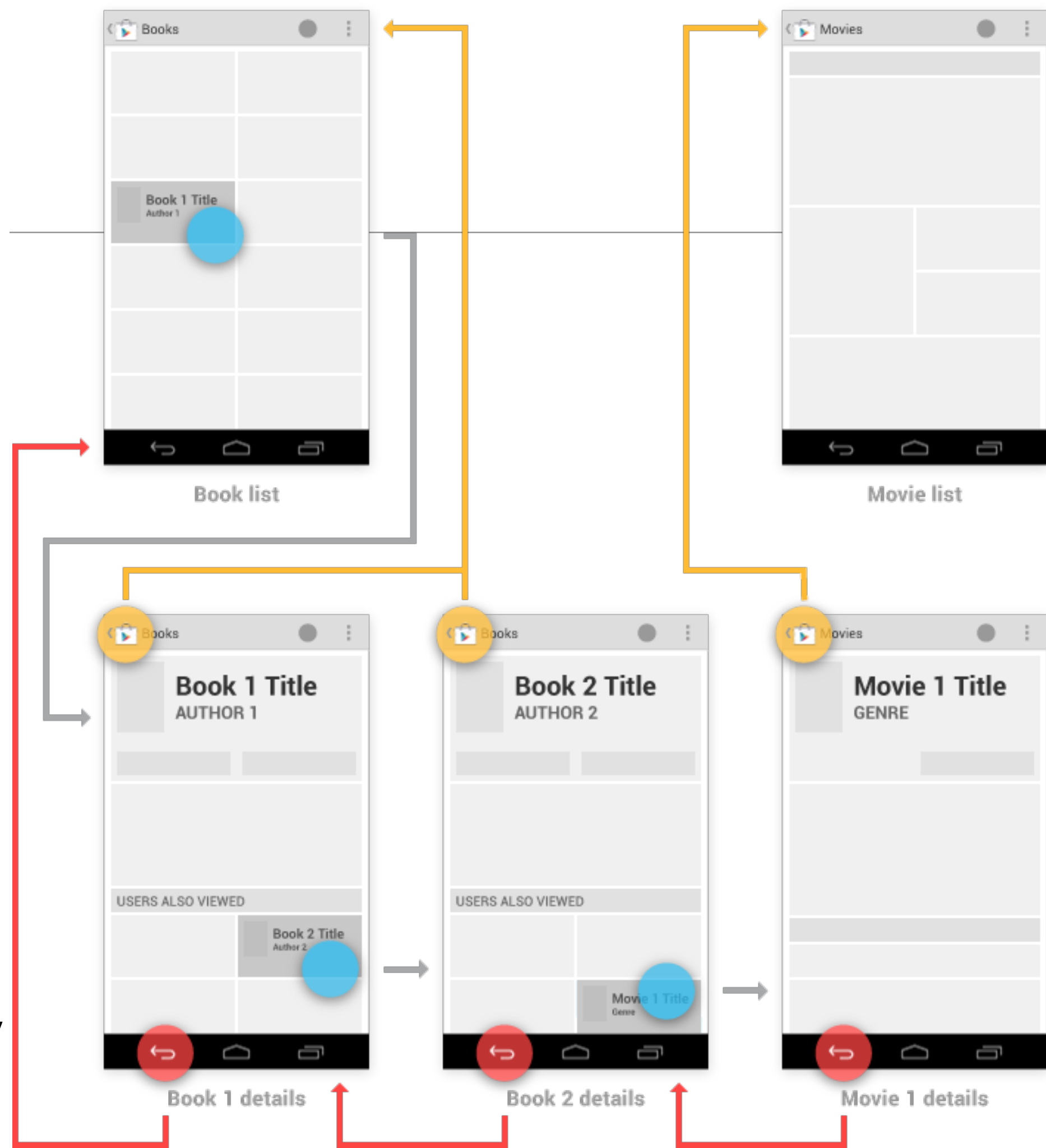
- However, a notable exception to this occurs when browsing between related detail views not tied together by the referring list — for example, when browsing in the Play Store between apps from the same developer, or albums by the same artist.

- In these cases, following each link does create history, causing the Back button to step through each previously viewed screen.

- Up should continue to bypass these related screens and navigate to the most recently viewed container screen.



- You have the ability to make the Up behavior even smarter based on your knowledge of detail view.
- Extending the Play Store example from above, imagine the user has navigated from the last Book viewed to the details for the Movie adaptation.
- In that case, Up can return to a container (Movies) which the user hasn't previously navigated through.



(Material drawn from developer.android.com)