App Navigation

Dr. Sarah Abraham

University of Texas at Austin
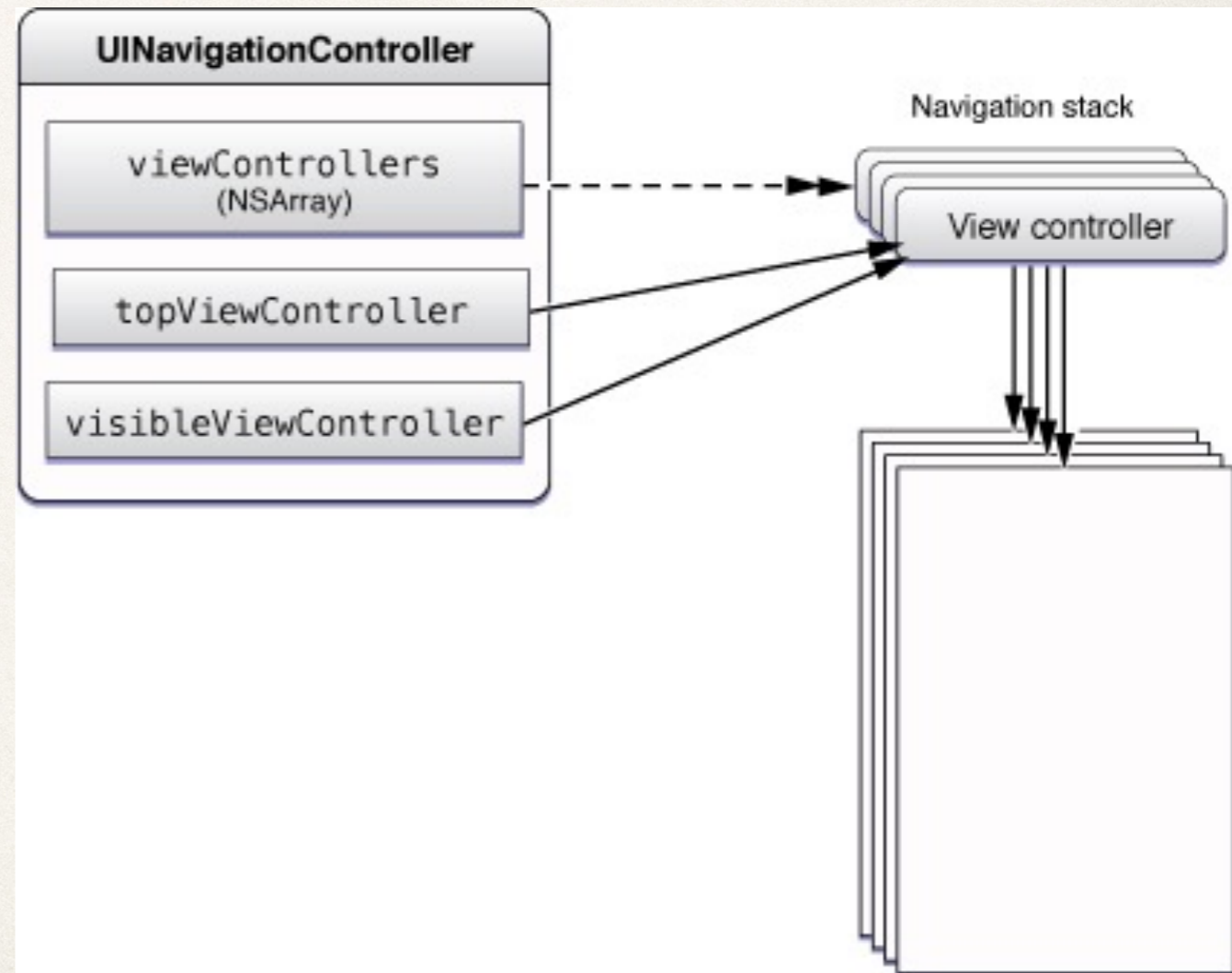*CS329e*
Spring 2020

Multiple Screens in Apps

✤ A single screen is rarely enough to encapsulate all app functionality

✤ Program must know:

  ✤ What screen displays what information

  ✤ When to change screens

  ✤ How to return to previous screens

Navigation Controllers

* **Container controller** holding some number of view controllers

* Child view controllers are referenced in an internal array

  * Pushed onto view controller stack to display

  * Popped from view controller stack to remove

* Navigation bar provided to move between view controllers

Navigation Stack

✤ First item in stack is root view controller

   ✤ Never popped

✤ Back button can automatically pop top view controller

Navigation Controller Example

Album list controller
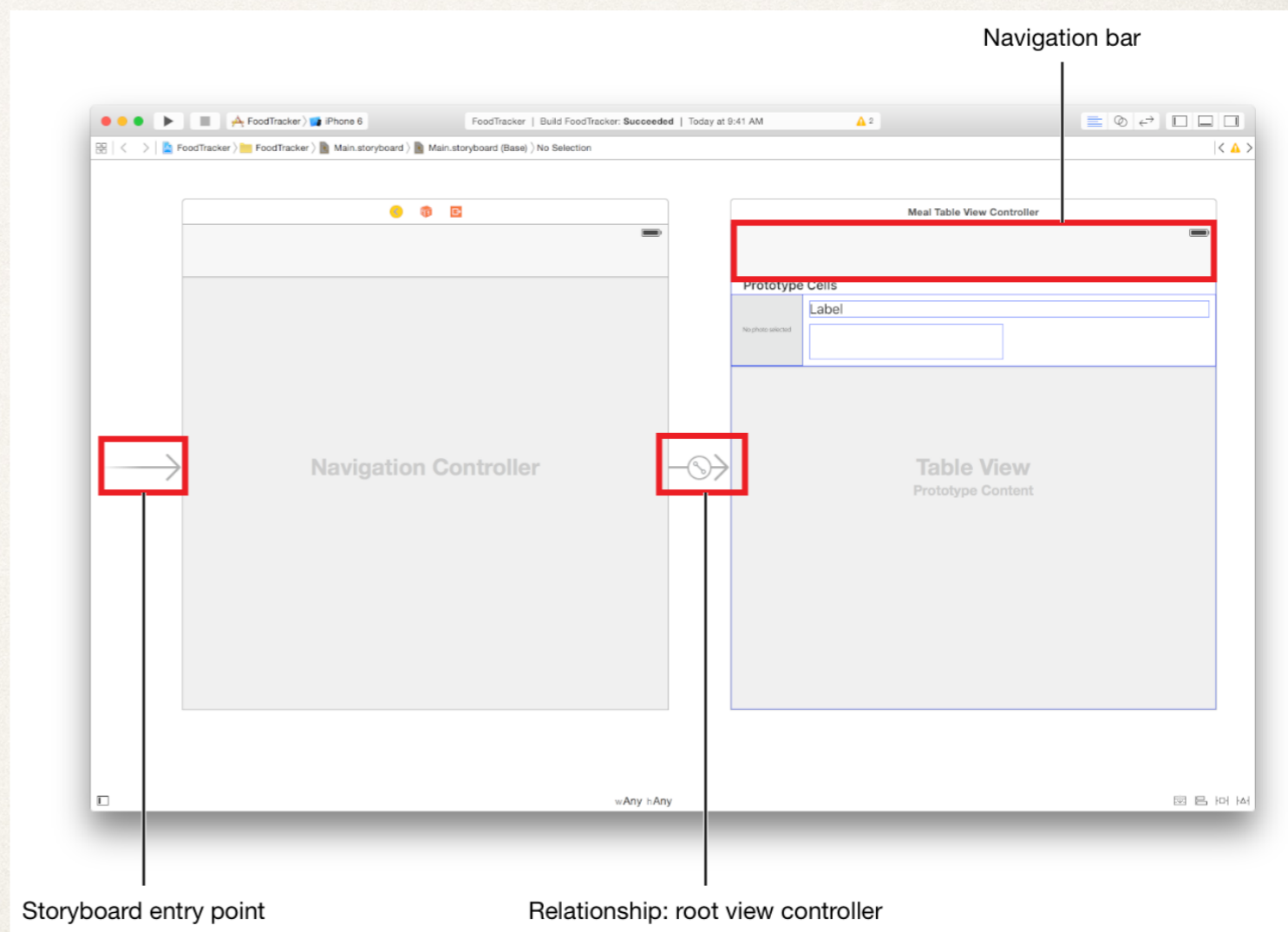
Photo album controller

Photo controller

Segues

* Named connection between one part of user interface and another

* Transitions between two view controllers in the storyboard

* Created in Interface Builder or programmatically

* Automatically triggered at runtime

    * Can notify app about transition for additional control

Segue Actions

✤ Add segue action using control drag (connect to next view controller)

✤ Types of segues available:

  ✤ Show

  ✤ Show detail

  ✤ Present modally

  ✤ Popover presentation

  ✤ Custom

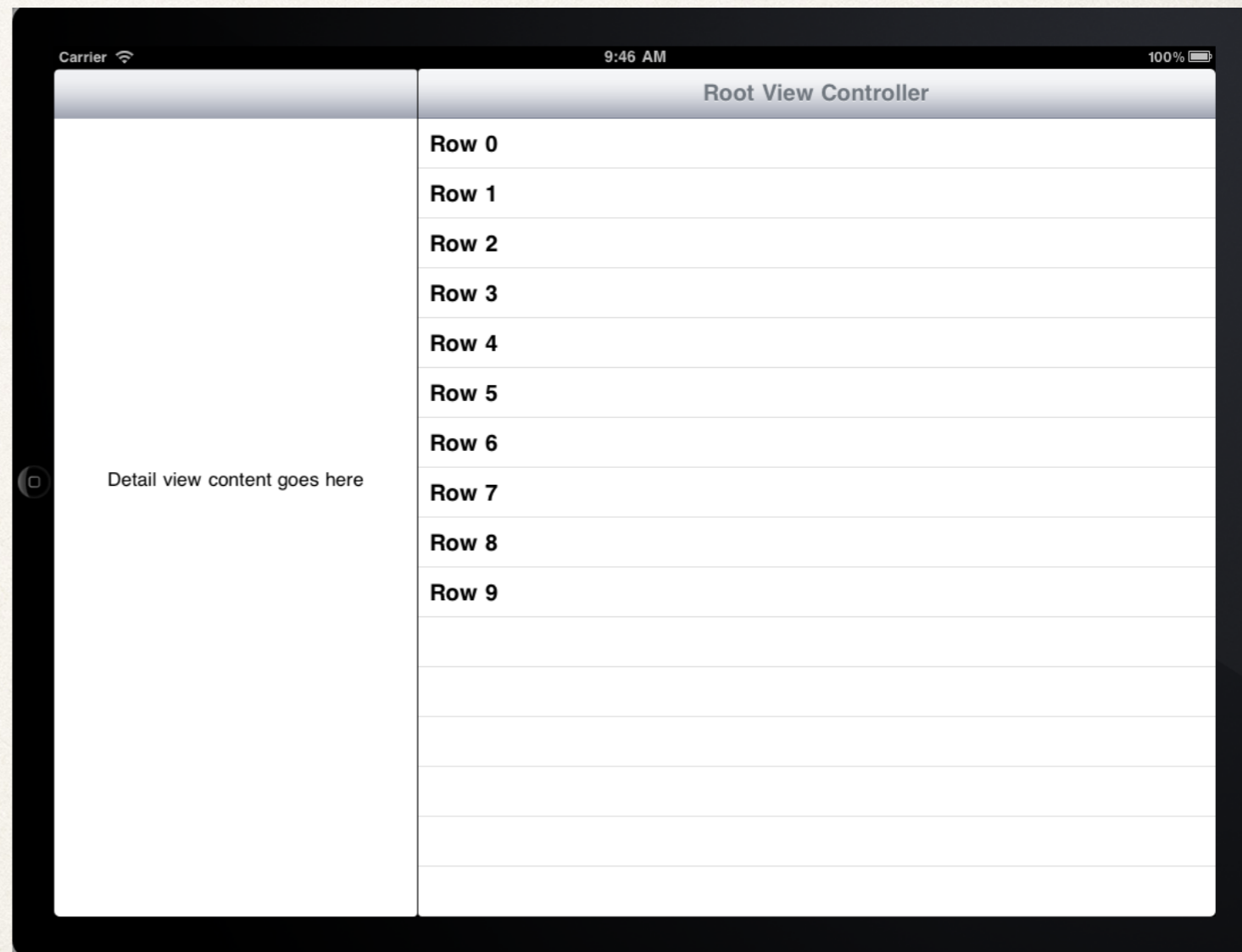# Show Segue

- Used with view controllers in a Navigation Controller container

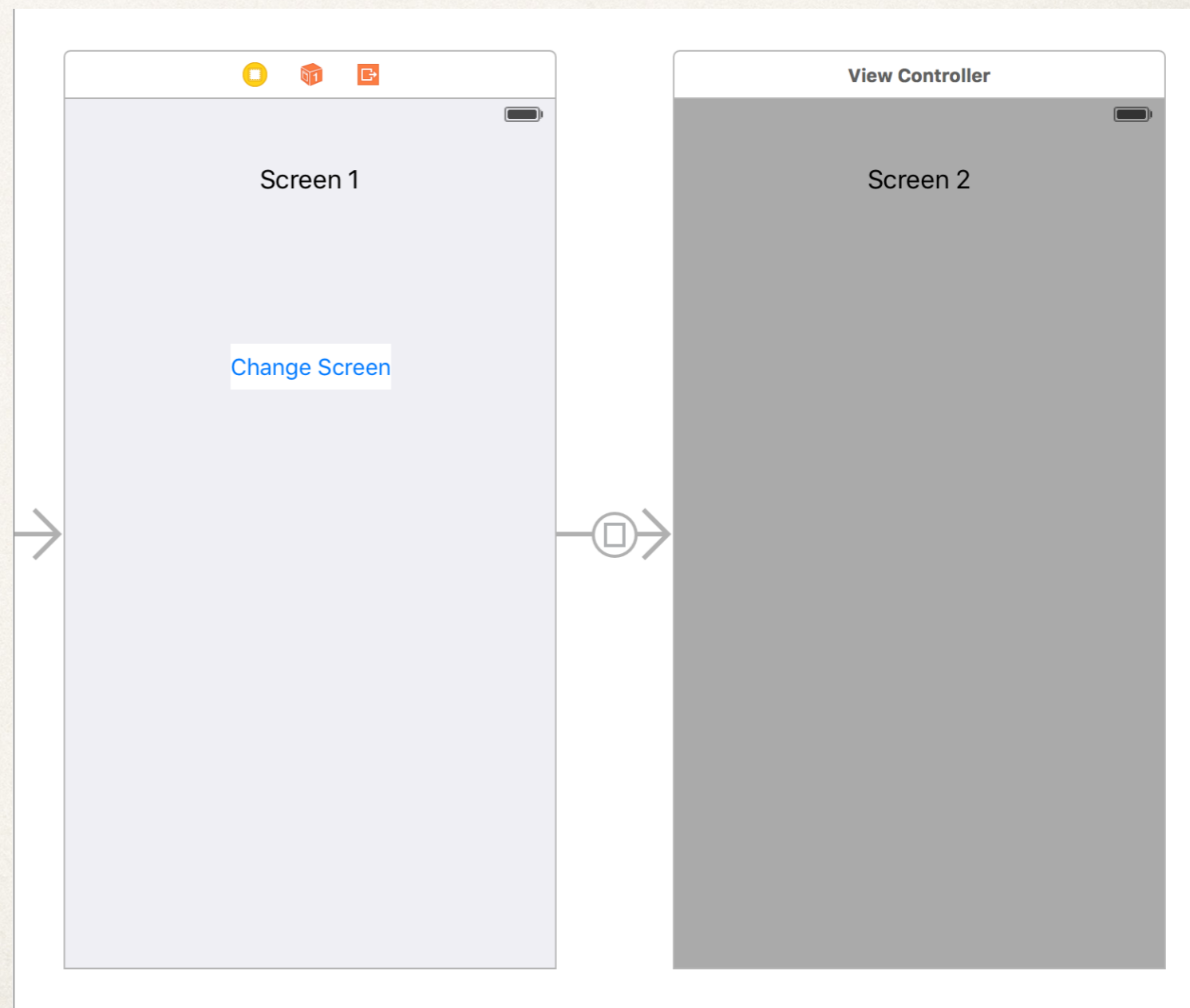- Allows for back button in Navigation Bar to return user to previous screen

Show Detail Segue

- Used with view controllers in a Split View Controller container

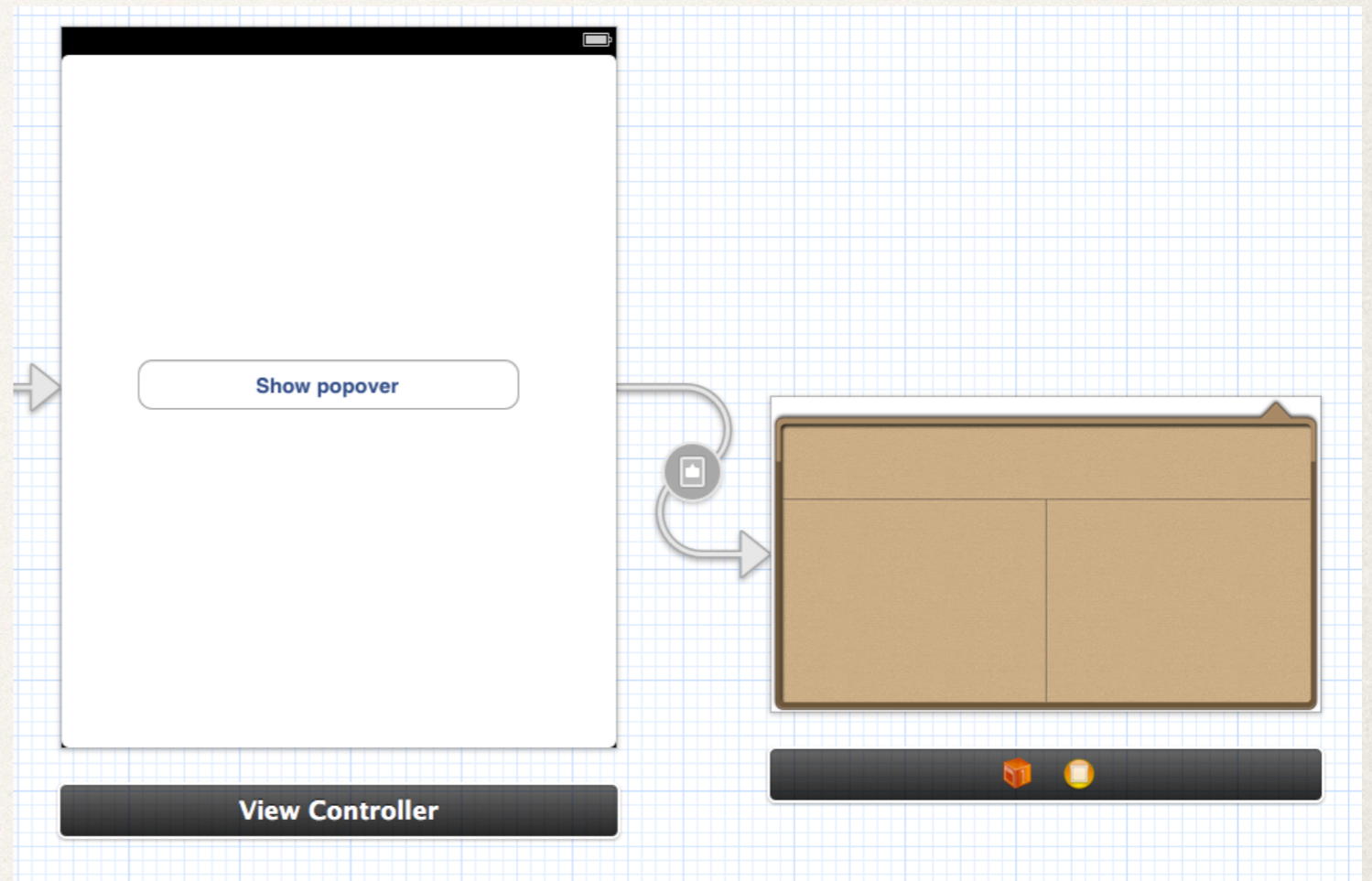- Replaces detail view side of Split View with next view controller

Present Modally Segue

- Used with separate view controllers outside of a controller container

- Shows destination view controller in front of previous one

- Prevents leaving partway through an action

Popover Presentation Segue

✤ Used to present new view controller as a smaller box over the current view controller

✤ Designed for the iPad

✤ Default on iPhone similar to "Present Modally" but can be customized

Custom Segue

✤ Used to make a custom segue style and transition

✤ Allows for deeper customization of animations and behaviors

✤ Builds on UIStoryboardSegue class to implement `perform` method

(www.appcoda.com)

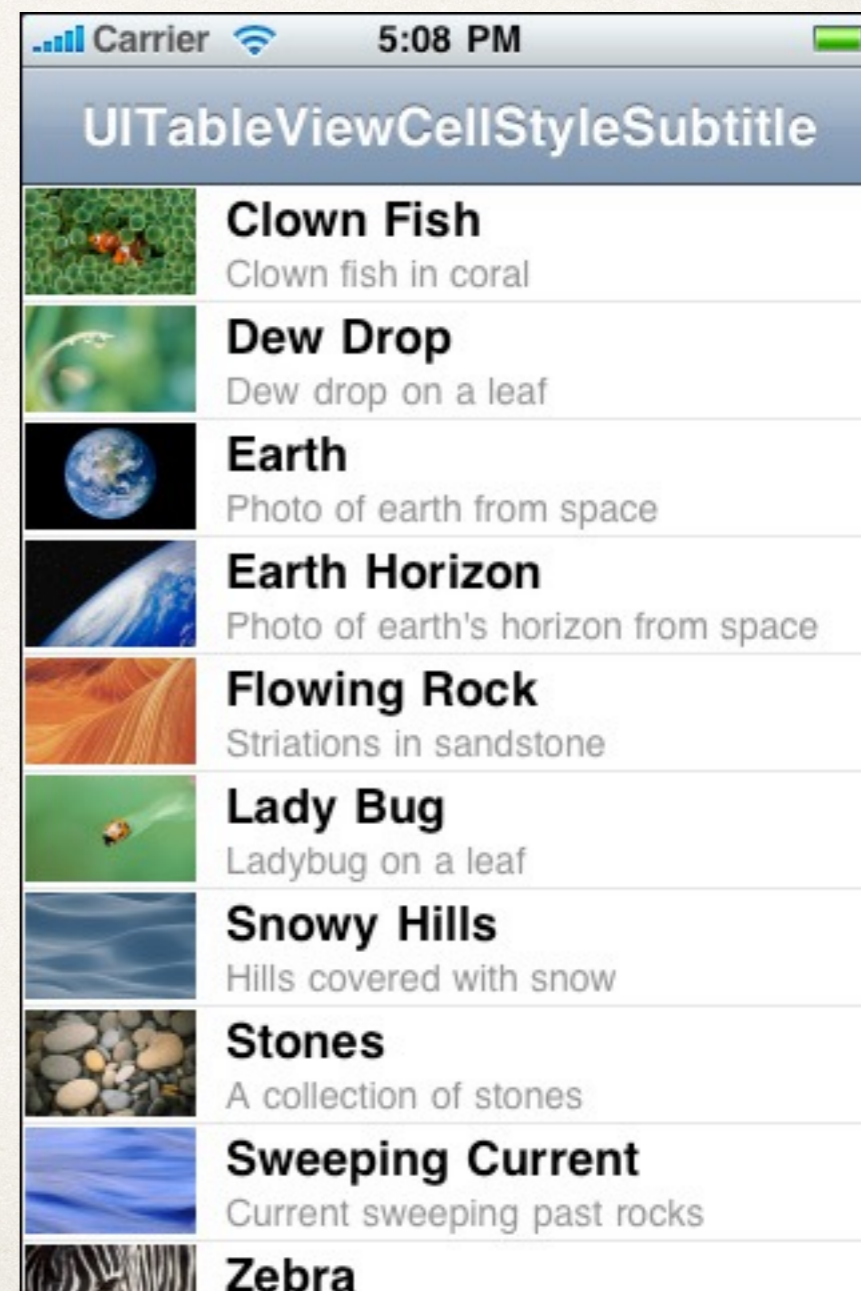# Segue Demo

Table Views

Table View Controller

✤ Derive new class from UITableViewController

   ✤ Generates method signatures for required methods and (commented out) optional methods

   ✤ Fleshing out necessary methods allows for table to display data

✤ Programmer must consider:

   ✤ How to structure the data model for table view to access it

   ✤ How to display the data using table view

UITableViewController

- Assumes most of the view consists of a Table View

  - Automatically loads the Table View into the storyboard (accessible through tableView property)

  - Automatically sets the Table View data source and delegate to `self`

  - Handles first appearance/loading of table view (`ViewWillAppear` and `ViewDidAppear`)

  - Handles resizing of table view when keyboard is activated

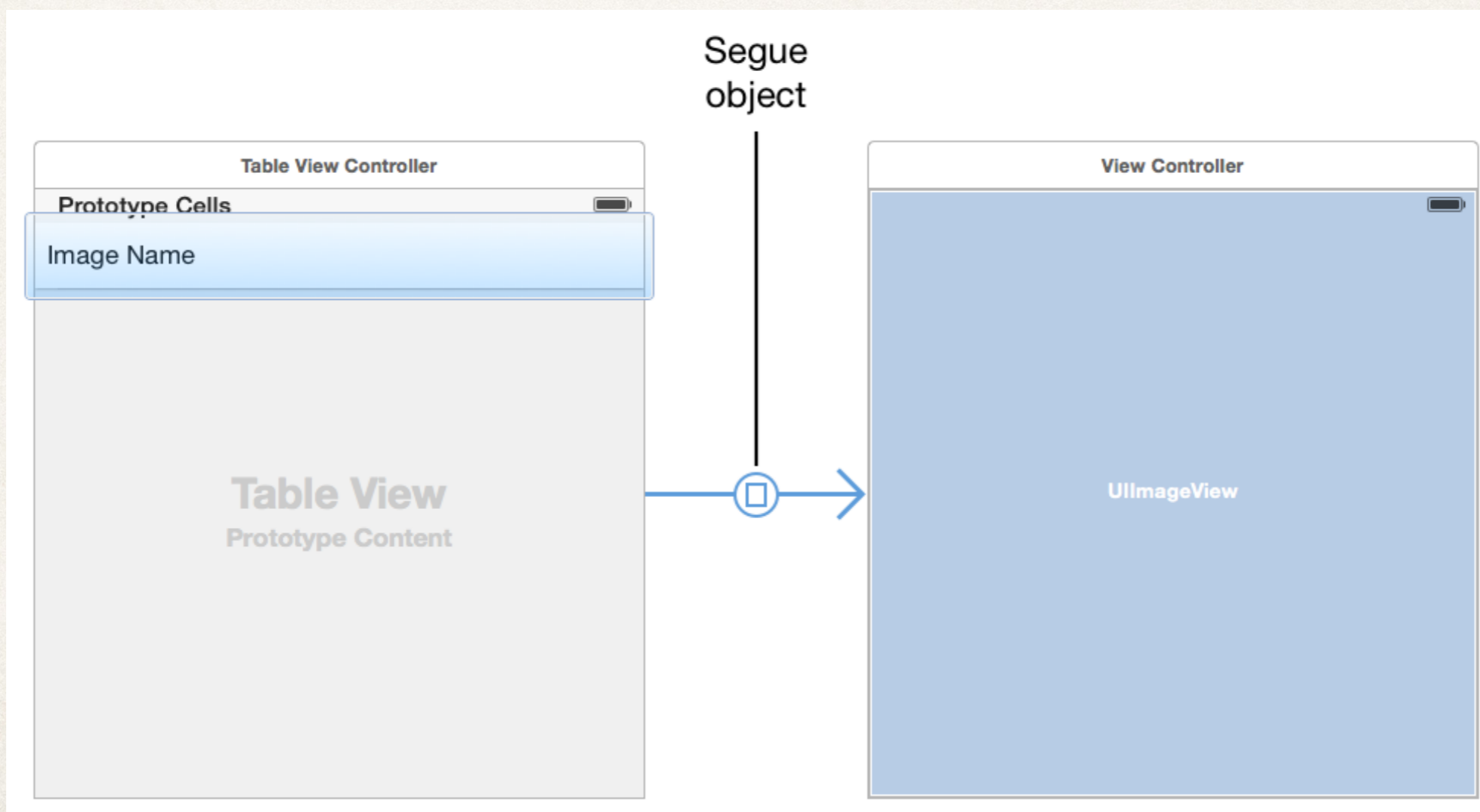- Use a standard UIViewController and manually add in a Table View if more flexibility is needed

TableView Cells

✤ TableView generates list of cells to display data

✤ Provides basic predefined cell styles

✤ Several styles available depending on basic uses

✤ Customization also possible and almost always used

Combining Transitions

✤ Table views, navigation controllers, and segues can be combined in various ways

✤ What's happening in this example?

Quiz Question!

✤ Which of these segues is designed to present a view controller outside of the controller container?

A. Show Segue

B. Show Detail Segue

C. Present Modally Segue

D. Popover Presentation Segue