Sun*

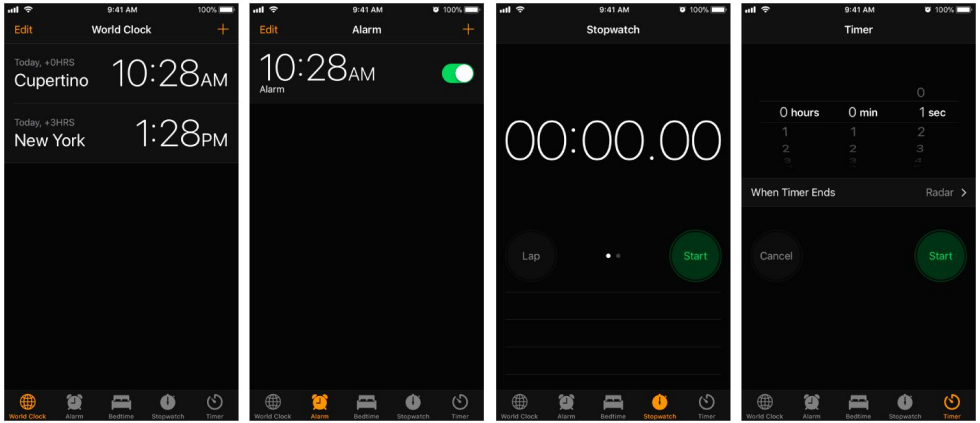
UITabbarController

Outline

1. Overview
2. The Views of a Tab Bar Controller
3. The More Navigation Controller
4. State Preservation

1. Overview

- The tab bar interface displays tabs at the bottom of the window for selecting between the different modes and for displaying the views for that mode. This class is generally used as-is, but may also be subclassed.
- Each tab of a tab bar controller interface is associated with a custom view controller. When the user selects a specific tab, the tab bar controller displays the root view of the corresponding view controller, replacing any previous views.



1. Overview (cont)

- You should never access the tab bar view of a tab bar controller directly. To configure the tabs of a tab bar controller, you assign the view controllers that provide the root view for each tab to the `viewControllers` property.

- The order in which you specify the view controllers determines the order in which they appear in the tab bar. When setting this property, you should also assign a value to the `selectedViewController` property to indicate which view controller is selected initially. (You can also select view controllers by array index using the `selectedIndex` property.).

- When you embed the tab bar controller's view (obtained using the inherited `view` property) in your app window, the tab bar controller automatically selects that view controller and displays its contents, resizing them as needed to fit the tab bar interface.

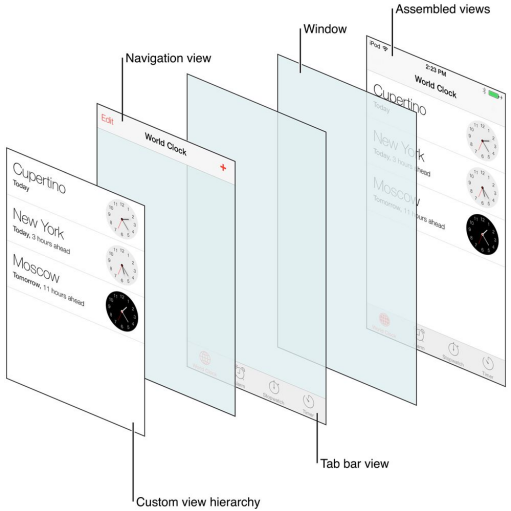
1. Overview (cont)

- Tab bar items are configured through their corresponding view controller. To associate a tab bar item with a view controller, create a new instance of the `UITabBarItem` class, configure it appropriately for the view controller, and assign it to the view controller's `tabBarItem` property.

- If you don't provide a custom tab bar item for your view controller, the view controller creates a default item containing no image and the text from the view controller's `title` property.

2. The Views of a Tab Bar Controller

- Because the `UITabBarController` class inherits from the `UIViewController` class, tab bar controllers have their own view that is accessible through the `view` property. The view for a tab bar controller is just a container for a tab bar view and the view containing your custom content. The tab bar view provides the selection controls for the user and consists of one or more tab bar items.



3. The More Navigation Controller

- The tab bar has limited space for displaying your custom items. If you add six or more custom view controllers to a tab bar controller, the tab bar controller displays only the first four items plus the standard More item on the tab bar. Tapping the More item brings up a standard interface for selecting the remaining items.

- The interface for the standard More item includes an Edit button that allows the user to reconfigure the tab bar. By default, the user is allowed to rearrange all items on the tab bar. If you do not want the user to modify some items, though, you can remove the appropriate view controllers from the array in the `customizableViewControllers` property.

4. State Preservation

- In iOS 6 and later, if you assign a value to this view controller's `restorationIdentifier` property, it preserves a reference to the view controller in the selected tab. At restore time, it uses the reference to select the tab with the same view controller.

- When preserving a tab bar controller, assign unique restoration identifiers to the child view controllers you want to preserve. Omitting a restoration identifier from a child view controller causes that tab to return to its default configuration. Although the tab bar controller saves its tabs in the same order that they are listed in the `viewControllers` property, the save order is actually irrelevant.

- The state preservation system restores the contents of each tab based on the assigned restoration identifier, not based on the position of the tab.

Question & Answer?