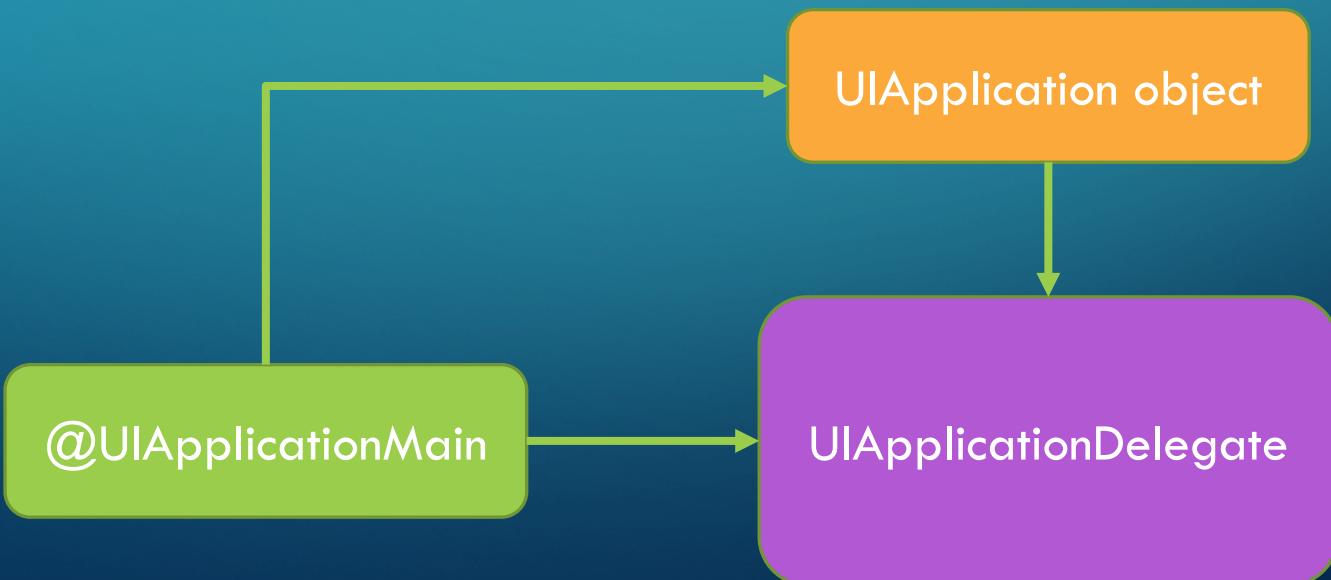


SWIFT: APPLICATION LIFECYCLE, VIEWCONTROLLER LIFECYCLE

@UIApplicationMain

- attribute of AppDelegate.swift
- using this attribute is equivalent to calling the UIApplicationMain function and this call is the main entry point to create a **singleton of application object** and the application delegate and set up the event cycle.



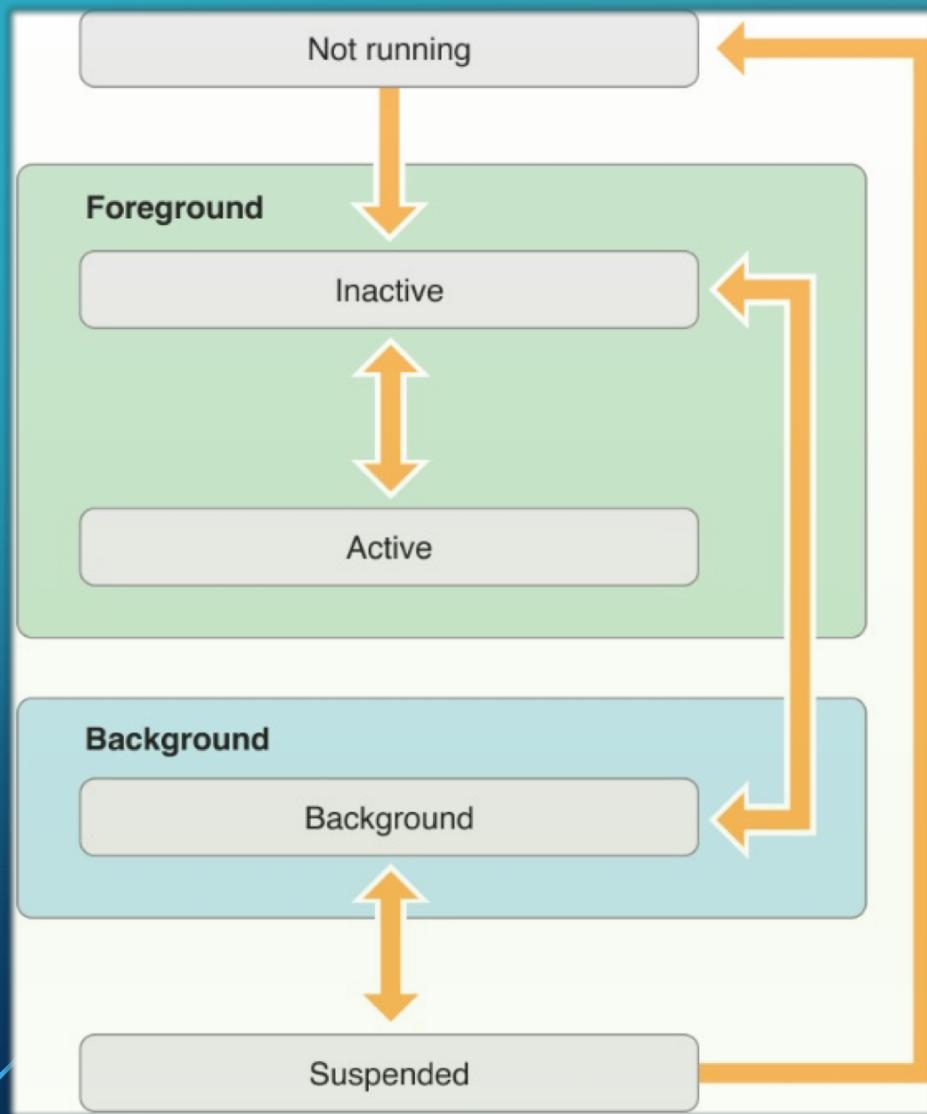
Singleton UIApplication object

- A major role of app's **application object** is to handle the initial routing of incoming user events. It dispatches action messages forwarded to it by control objects to appropriate target objects. The **application object** maintains a list of UIWindow objects and through those can retrieve any of the app's UIView objects.
- The UIApplication class defines a delegate that conforms to UIApplicationDelegate protocol and must implement some of the protocol's methods. The **application object** informs the **delegate** of significant runtime events – for example, app launch, low-memory warnings, and app termination etc.

UIApplicationDelegate

- App delegate is a singleton object and is always present at runtime
- At launch time the app delegate typically creates the app's initial data structures, registers for any required services, tweaks the app's initial user interface based on any available data and provide the root window object for your app
- Main jobs of the app delegate is to respond to state transitions reported by the system. For every state change that occurs, the system calls the appropriate methods of the app delegate.

UIApplicationDelegate: iOS Application lifecycle



- **Not running**: The app has not been launched or was terminated
- **Inactive**: The app is running in the foreground but is not receiving events (UI still is not visible).
- **Active**: The app is running in the foreground and receiving events. This is the normal mode for foreground apps.
- **Background**: The app is executing code but is not visible onscreen. When the user quits an app, the system moves the app to the background state briefly before suspending it. An app in the background state should do as little work as possible.
- **Suspended**: The app is in memory but is not executing code.

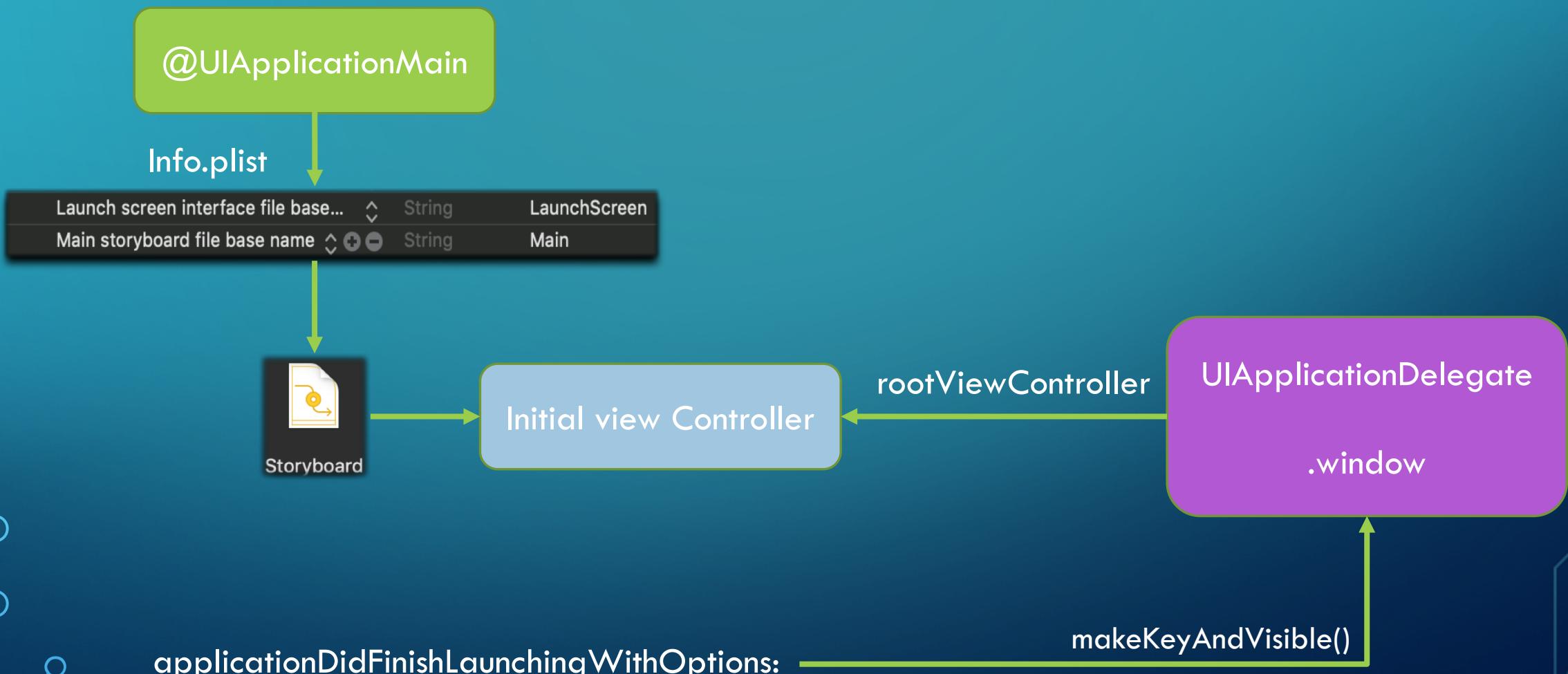
UIApplicationDelegate: transition methods between states:

- Launch time:
 - [application\(:willFinishLaunchingWithOptions:\)](#)
 - [application\(:didFinishLaunchingWithOptions:\)](#)
- Transitioning to the foreground:
 - [applicationDidBecomeActive\(:\)](#)
- Transitioning to the background:
 - [applicationDidEnterBackground\(:\)](#)
- Transitioning to the inactive state:
 - [applicationWillResignActive\(:\)](#) (Called when leaving the foreground state.)
 - [applicationWillEnterForeground\(:\)](#) (Called when transitioning out of the background state.)
- Termination:
 - [applicationWillTerminate\(:\)](#) (Called only when the app is running. This method is not called if the app is suspended.)

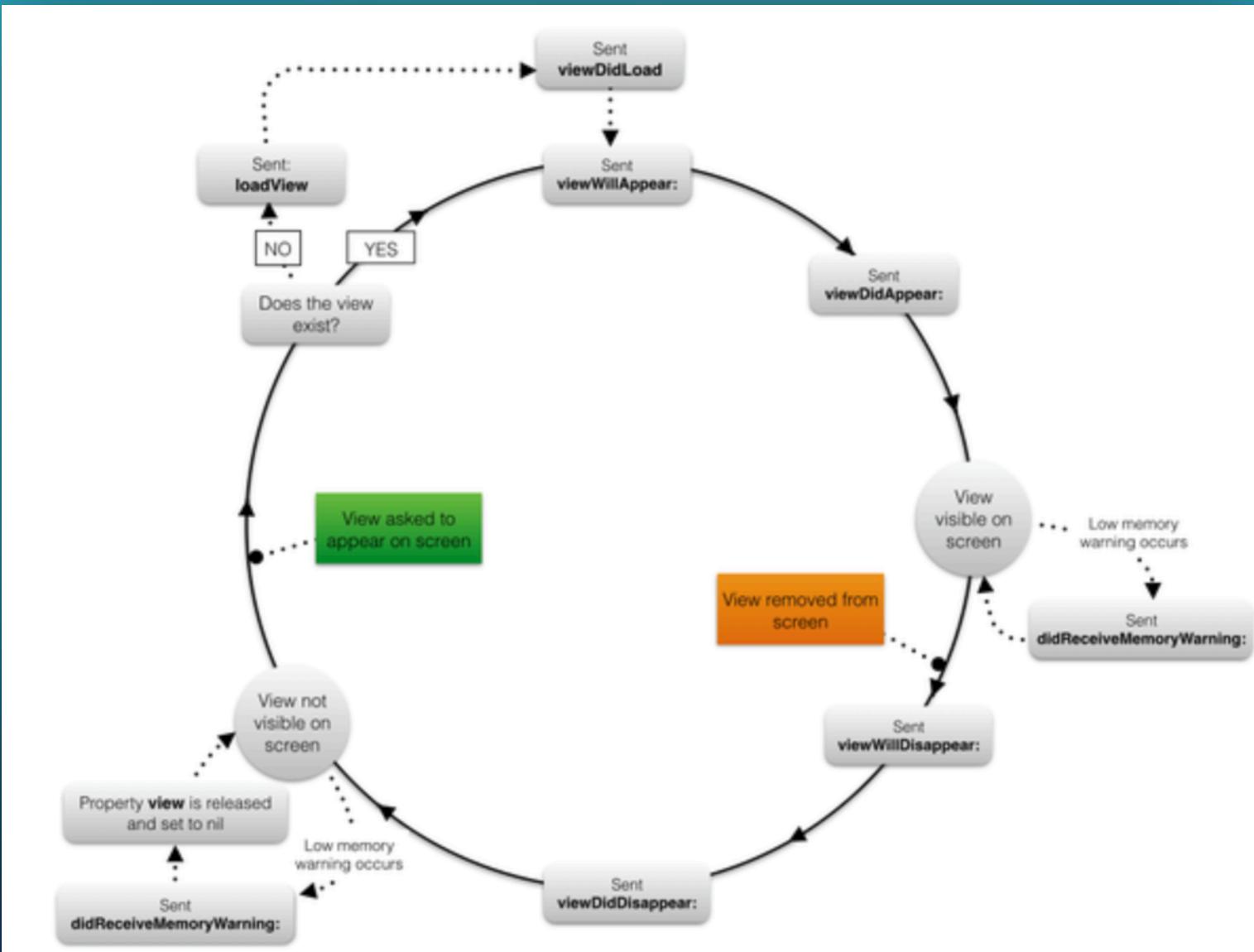
UIWindow

- The backdrop for app's user interface and the object that dispatches events to views
- Every view that appears onscreen is enclosed by a window and events received by your app are initially routed to the appropriate window object, which in turn forwards those events to the appropriate view
- `@UIApplicationMain` reads Storyboard information from `Info.plist` and instantiate an initial view controller. In this time UIWindow is created and associated with `window` property of AppDelegate. Initial view controller is now assigned to `rootViewController` property of the window and therefore its view becomes the current window's content. To show content to user `makeKeyAndVisible` method is called in `didFinishLaunchingWithOptions` (diagram, next slide)

UIWindow



UIViewController



UIViewController

- **init(coder:), init(nibName:bundle:)** – initialize ViewController from a Storyboard using coder/decoder or from a nib file.
- **loadView()** – create the view for ViewController. Shouldn't be called directly
(loadViewIfNeeded()) – load ViewController's view if it hasn't been loaded yet)
- **viewDidLoad()** – when called, the view created and all outlets are in place. Called once during lifetime.
- **viewWillAppear(_:) / viewDidAppear(_:)** – called every time before/after ViewController comes on a screen
- **viewWillDisappear(_:) / viewDidDisappear(_:)** – before/after the transition to the next view controller happens view controller gets removed from screen
- **deinit()** – called before a view controller is removed from memory
- **didReceiveMemoryWarning()** – iOS notifies app about using too much memory