

iPhone Application Programming

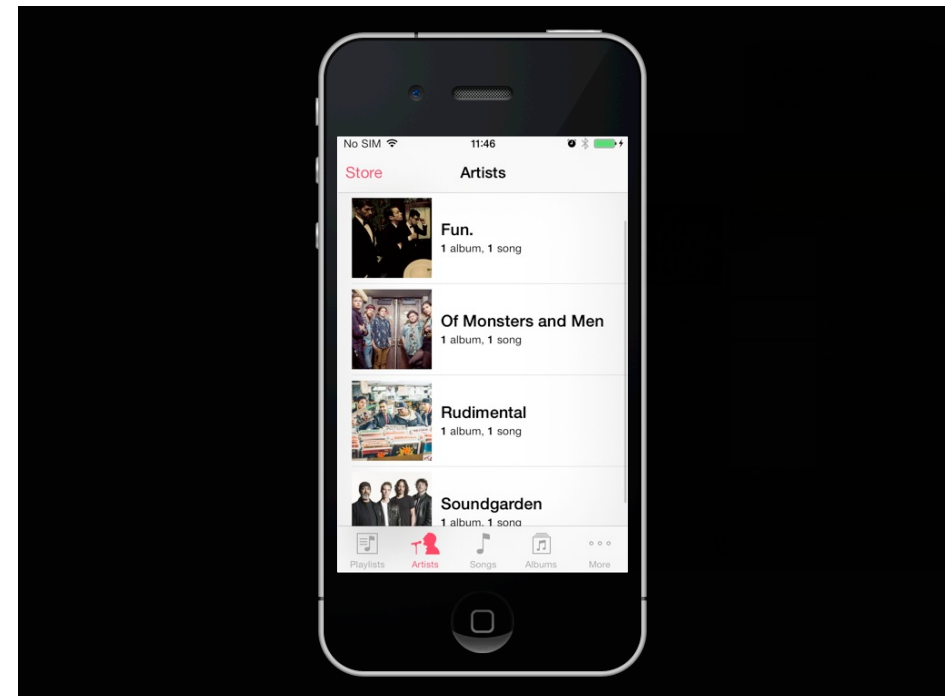
Lecture 5:View Controllers

Chat Wacharamanotham
Media Computing Group
RWTH Aachen University
Winter Semester 2013/2014
<http://hci.rwth-aachen.de/iphone>

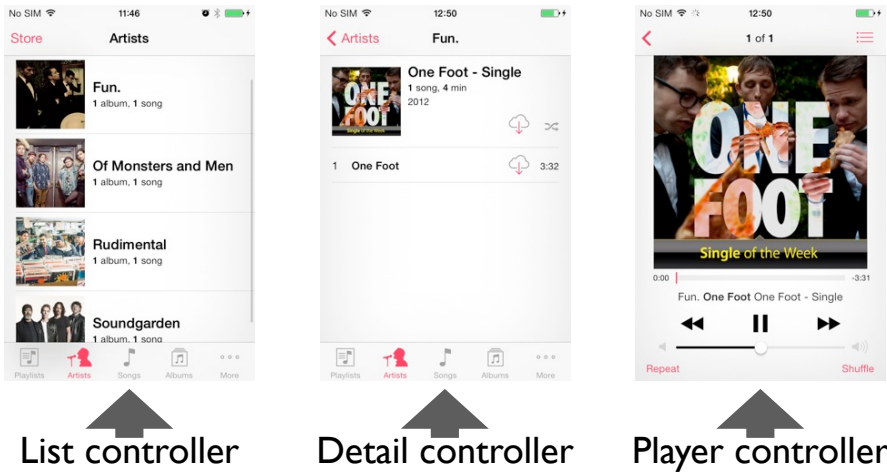
View Controllers

Windows, Views, ViewControllers

- Window
 - Every application has at least one window
 - Blank canvas to host views
- View
 - Draw and animate content
 - Layout subviews
 - Receive and forward events
- ViewController
 - Manages a self-contained view hierarchy



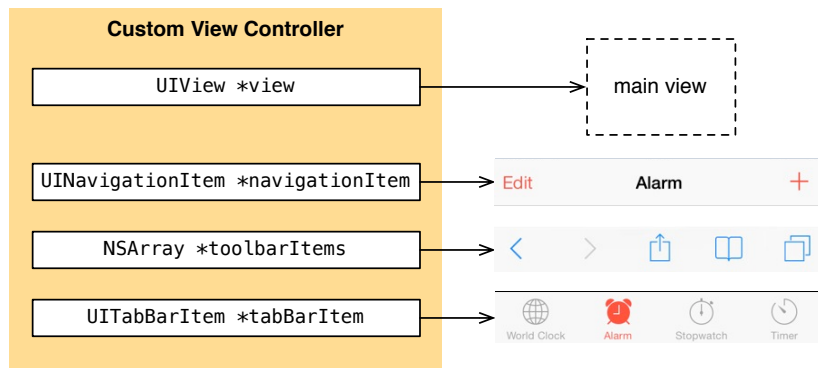
Screens



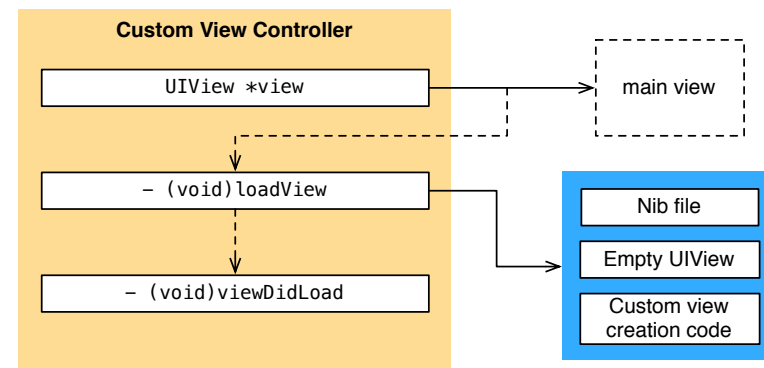
View Controllers

- Every screen should have its own controller
- **UIViewController** implements much of the iOS standard interface behavior
 - Loading nib file
 - Working with navigation, tab, and tool bars
 - Composing multiple view controllers
 - Handling events and memory warnings
 - Managing interface orientation change

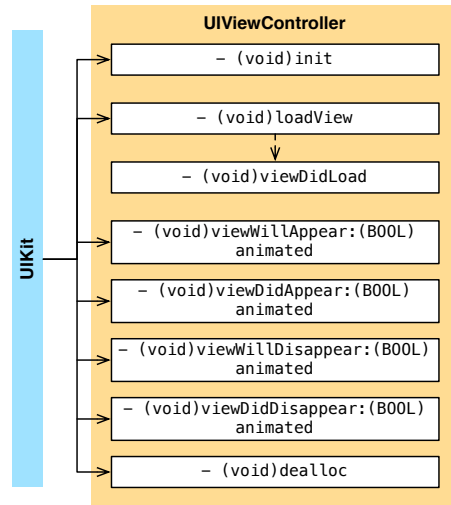
View Controller Anatomy



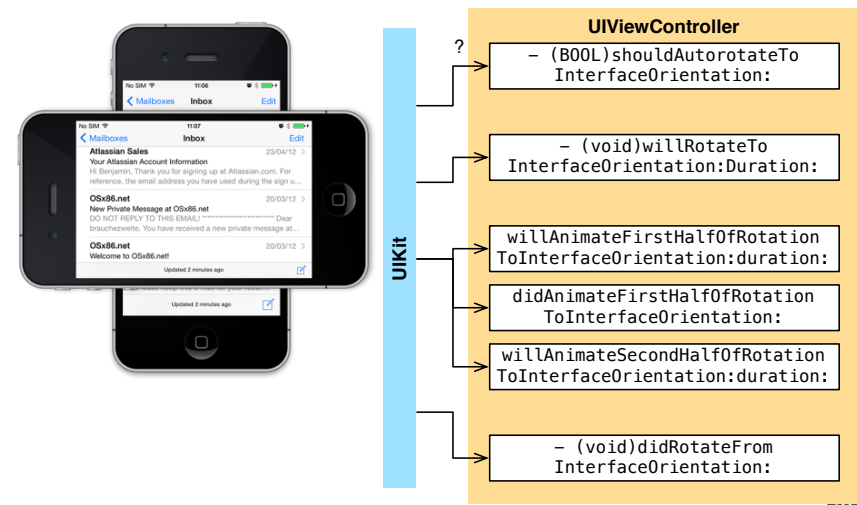
View Management Cycle



View Controller Life Cycle



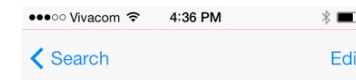
Interface Orientation



Types of View Controllers

- Custom view controller
 - Directly express content on the screen
- Container view controller
 - Manage other view controllers
 - Usually does not express content directly
- Modal view controller
 - Any view controller can be presented modally

Navigation Bar



John Appleseed

home
012 345678



FaceTime



home
johnappleseed@icloud.com

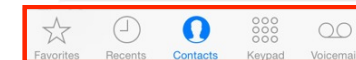
Notes

Send Message

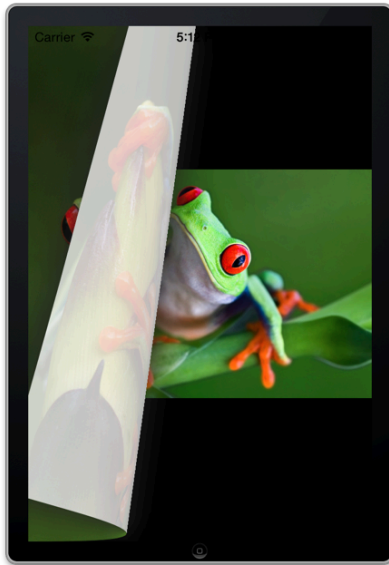
Modal View

Table View

Tab Bar



Page View Controller



13 iPhone Application Programming • Prof. Jan Borchers



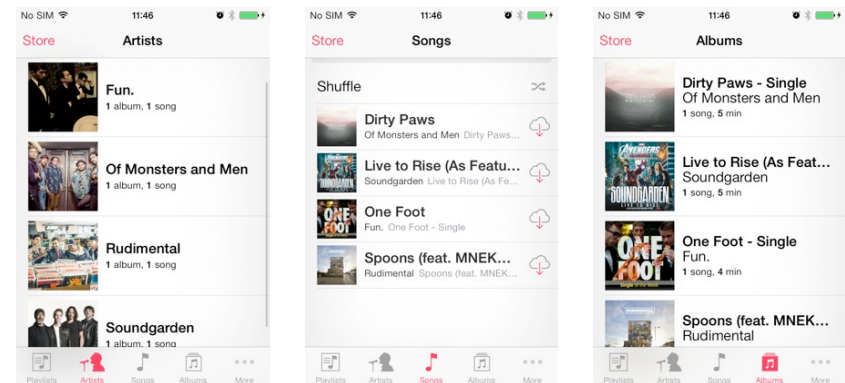
View Controller Showcase

- Tab Bar Controller
- Navigation Controller
- Table View Controller

14 iPhone Application Programming • Prof. Jan Borchers



Tab Bar Controller



15 iPhone Application Programming • Prof. Jan Borchers

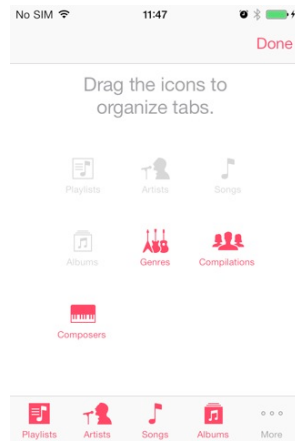


16 iPhone Application Programming • Prof. Jan Borchers



More Tab Items

- More than 5 view controllers cannot be shown
- “More” tab is shown automatically
 - User can navigate remaining view controllers and customize the order



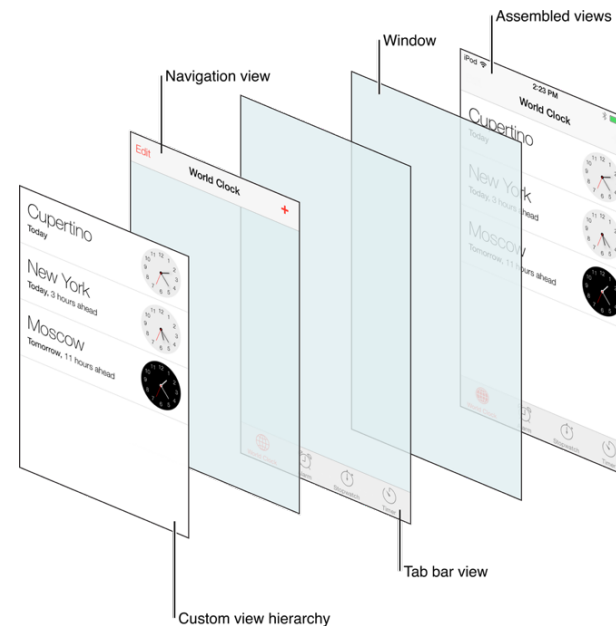
Creating a Tab Bar Controller

```
// Create a tab bar controller
tabBarController = [[UITabBarController alloc] init];

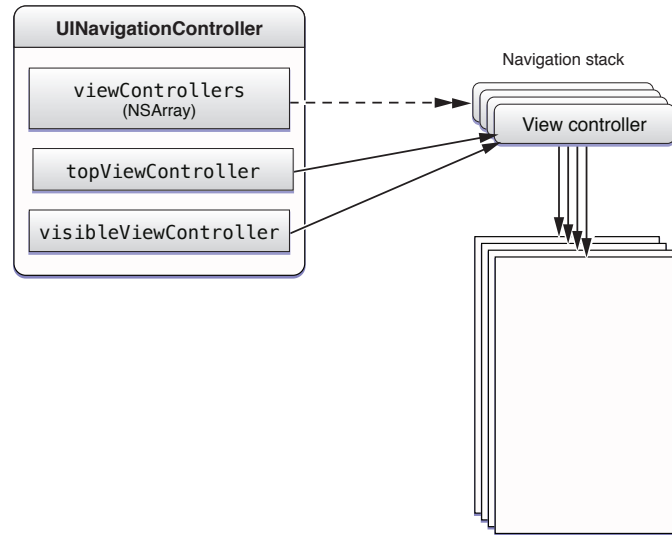
// Set the array of view controllers
NSArray *myViewControllers = ...
tabBarController.viewControllers = myViewControllers;

// Add the tab bar controller's view to the window
[window addSubview:tabBarController.view];
```

Navigation View Controller



Creating the Navigation Controller



[View Controller Programming Guide for iOS]

```
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {

    // set up the root view controller
    rootViewController = [[MyViewController alloc] init];

    // set up the navigation controller
    navigationController = [[UINavigationController alloc]
        initWithRootViewController:rootViewController];

    // Add the view controller's view to the window and display.
    [self.window addSubview:navigationController.view];
    [self.window makeKeyAndVisible];

    return YES;
}
```

22 iPhone Application Programming • Prof. Jan Borchers

Working with the Navigation Stack

```
// load the view controller
UIViewController *myViewController = [UIViewController
alloc];[myViewController initWithNibName:@"MyNib"
bundle:nil];

// push it to the navigation stack
[self.navigationController
pushViewController:myViewController
                animated:YES];

// we can release the view controller now
[myViewController release];

...

// pop the top view controller from the stack
[self.navigationController popViewControllerAnimated:YES];
```

23 iPhone Application Programming • Prof. Jan Borchers

Table View Controller

24 iPhone Application Programming • Prof. Jan Borchers

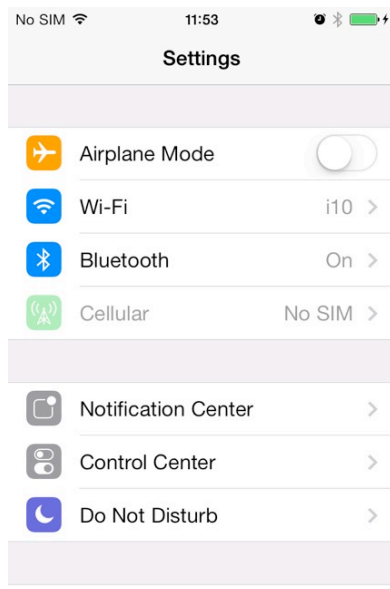
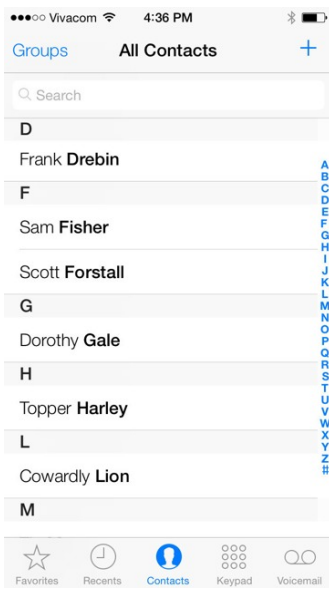


Table Views

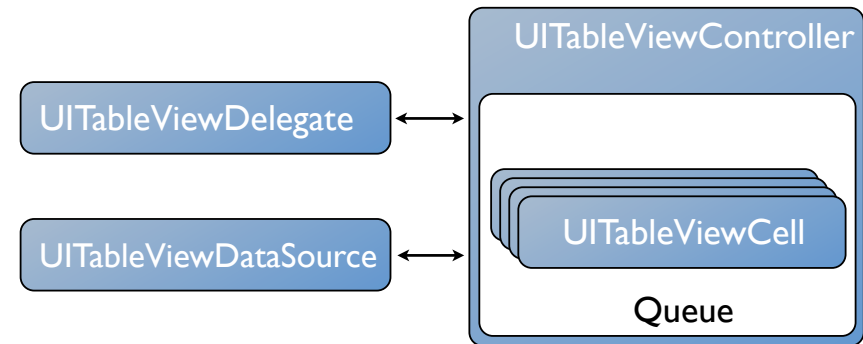


Table View Delegate

- Conform to `UITableViewDelegate` protocol
- Respond to selection
 - `tableView:didSelectRowAtIndexPath:`
 - `tableView:accessoryButtonTappedForRowWithIndexPath:`
- Manage header and footer views

Table View Data Source

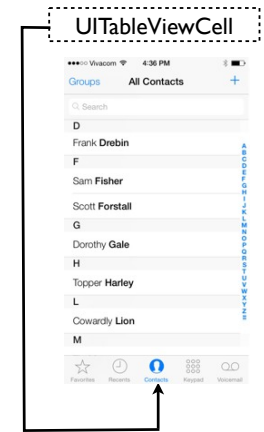
- Conform to `UITableViewDataSource` protocol
- Provide size of data
 - `numberOfSectionsInTableView:`
 - `tableView:numberOfRowsInSection:`
- Provide data
 - `tableView:cellForRowAtIndexPath:`
- Manage editing
 - `tableView:canEditRowAtIndexPath:`
 - `tableView:commitEditingStyle:forRowAtIndexPath:`

UITableViewCell Styles

- `UITableViewCellStyleDefault` Title
- `UITableViewCellStyleValue1` Title Detail
- `UITableViewCellStyleValue2` Title Detail
- `UITableViewCellStyleSubtitle` Title Detail

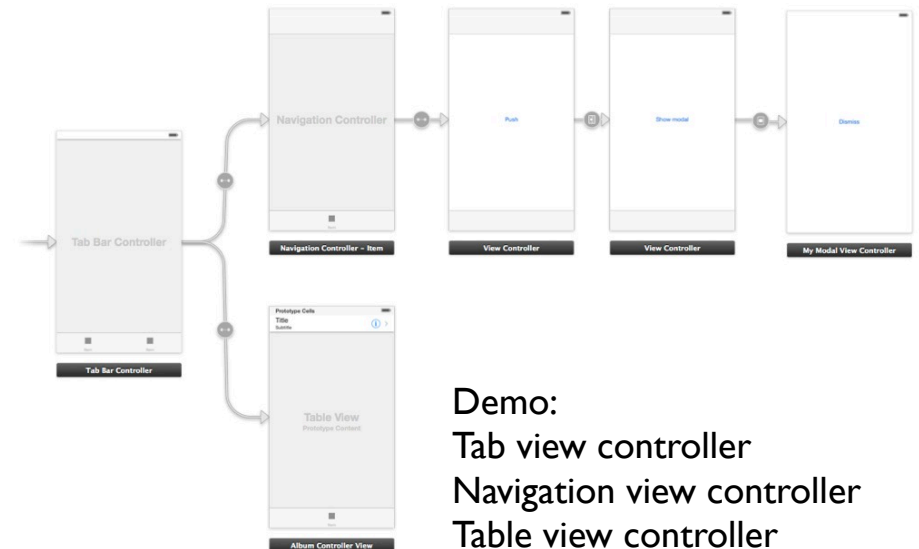
Memory Management

- Views are reused
 - `dequeueReusableCellWithIdentifier:`
 - Use different identifiers for different cell styles



UITableViewController

- `UIViewController` for table views
 - Delegate and data source
- Does not need a Nib file
 - `initWithStyle:`
- Takes care of some standard operations
 - Calls `reloadData` on appearing
 - Deselect rows in combination with navigation controller
 - Flashes scroll indicators
 - Powerful combination with Core Data (later)



Demo:
 Tab view controller
 Navigation view controller
 Table view controller

Application Components

Application Life Cycle

main.m

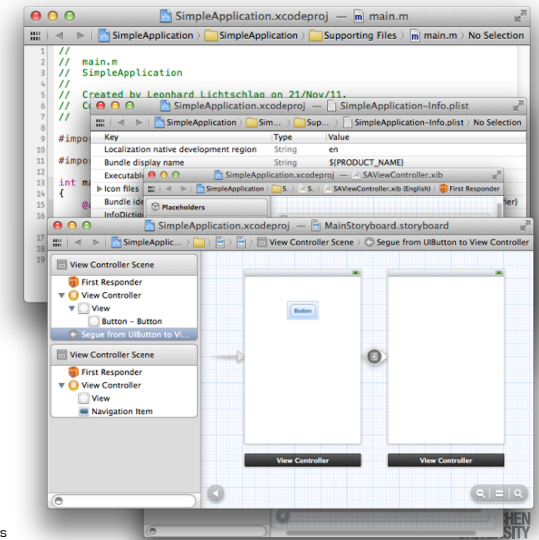
- UIApplication
- CFRunLoop

Info.plist

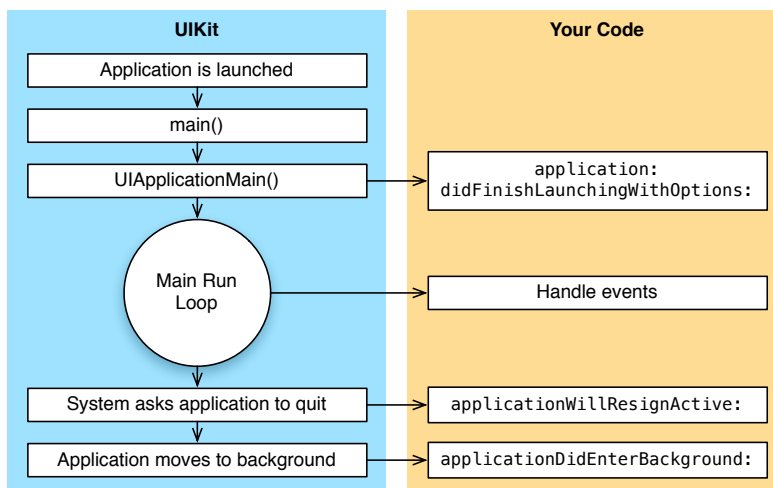
MainWindow.xib

- UIApplicationDelegate
- UIWindow

Main.storyboard

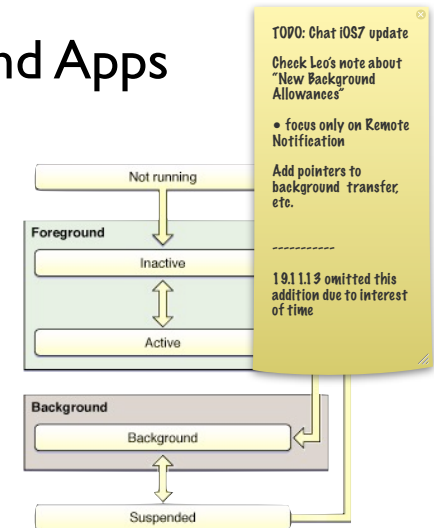


Application Life Cycle

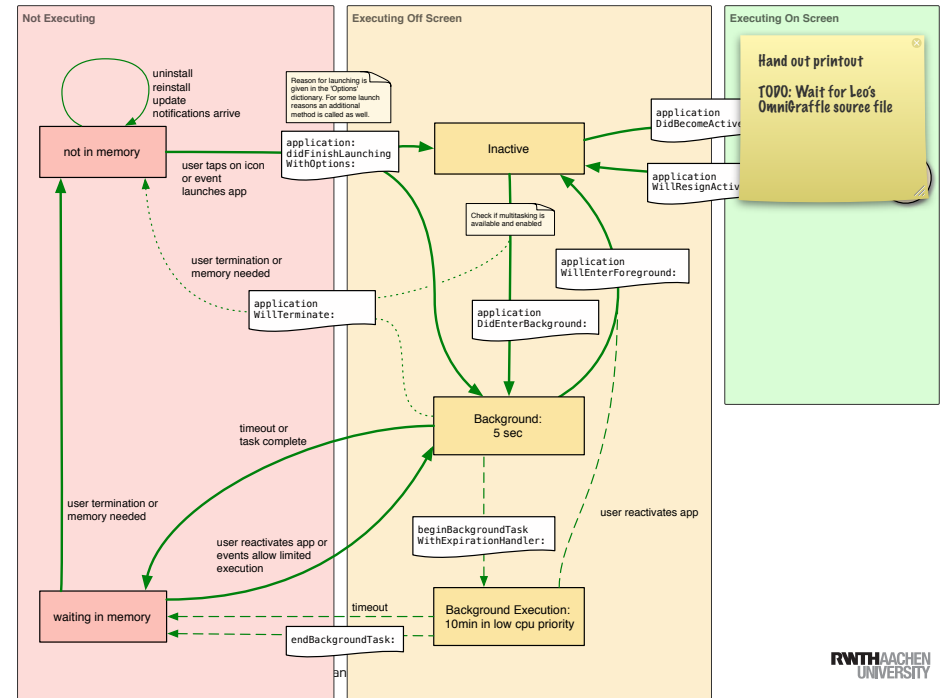
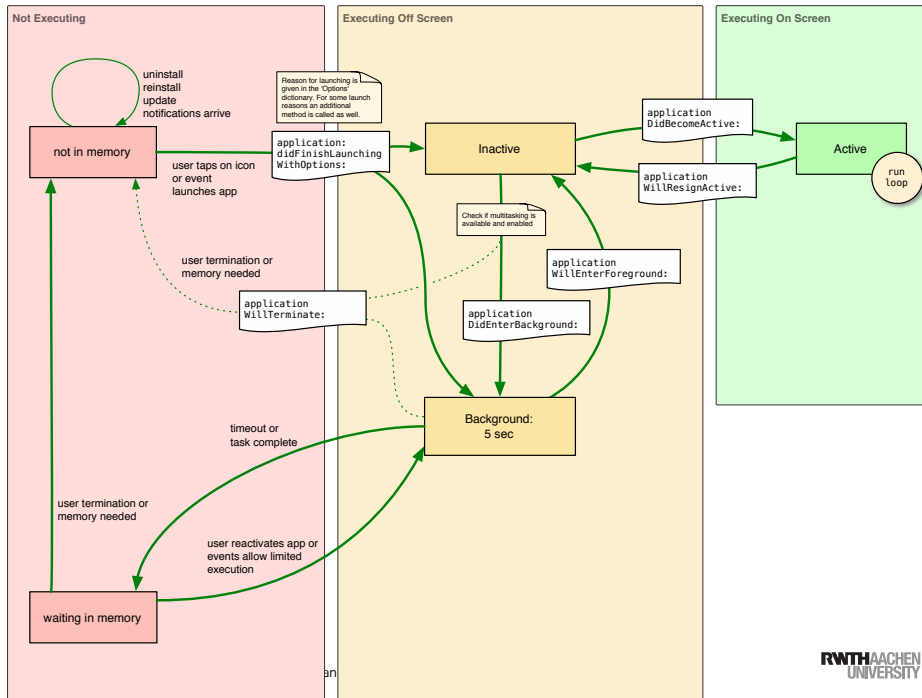


Background Apps

- Complete long-running tasks
- Schedule local notifications
- Receive location updates
- Play background audio
- Implement VoIP application
- Newsstand apps



Check `[[UIApplication sharedApplication] applicationState]` and `[[UIDevice currentDevice] isMultitaskingSupported]`



Summary

- View Controllers
 - Tab Bar Controllers
 - Navigation Controller
 - TableView Controller
- Storyboard demos
- Application Life Cycle
- Reading assignment:
 - 📖 View Controller Programming Guide
 - 📖 iPhone Application Programming Guide



Demo