Současné trendy vývoje aplikací pro platformu iOS

Petr Zvoníček
@zvonicek

27. 11. 2014
PV226

Proč je vývoj pro iOS zajímavý?

- uzavřený ekosystém
- silná open source komunita
- roční vývojový cyklus major verzí
- rychlá adopce nových verzí OS

Osnova

- Historické kořeny iOS
- Proměny programování pro iOS
- Swift

iOS

“iPhone runs OS X! Why would we want to run such a sophisticated OS on a mobile device? It's got everything we need. Multitasking, networking, power management, security, video, graphics, audio, core animation.”

Steve Jobs, 2007

NeXTSTEP

- Unix-like OS vyvynutý společností NeXT roku 1989
- Přímý předchůdce OS X a iOS
- Vývoj aplikací: jazyk Objective-C a NeXT API (Cocoa Framework)
- Firma NeXT v roce 1997 koupena Applem

Gorm
- Info
- Document
- Edit
- Tools
- Windows
- Services
- Hide h
- Quit q

Tools
- Inspector...
- Palettes...
- Load Palette...

NSWindow Inspector

Attributes

Title: LinuxFocus

Options

Visible at launch

Palettes

GNU

Text

Title

Button

PopUp1

PullDown1

LFtest Menu
- Info h
- Quit q

LinuxFocus

GNUstep test for LinuxFocus

It takes less than 5 minutes to create the window, the info panel and the menu

Ok

test.nib.gorm...ystem/Apps/exam

Objects Images Sounds Classes

NSOwner NSFirst NSMenu My Window

Panel

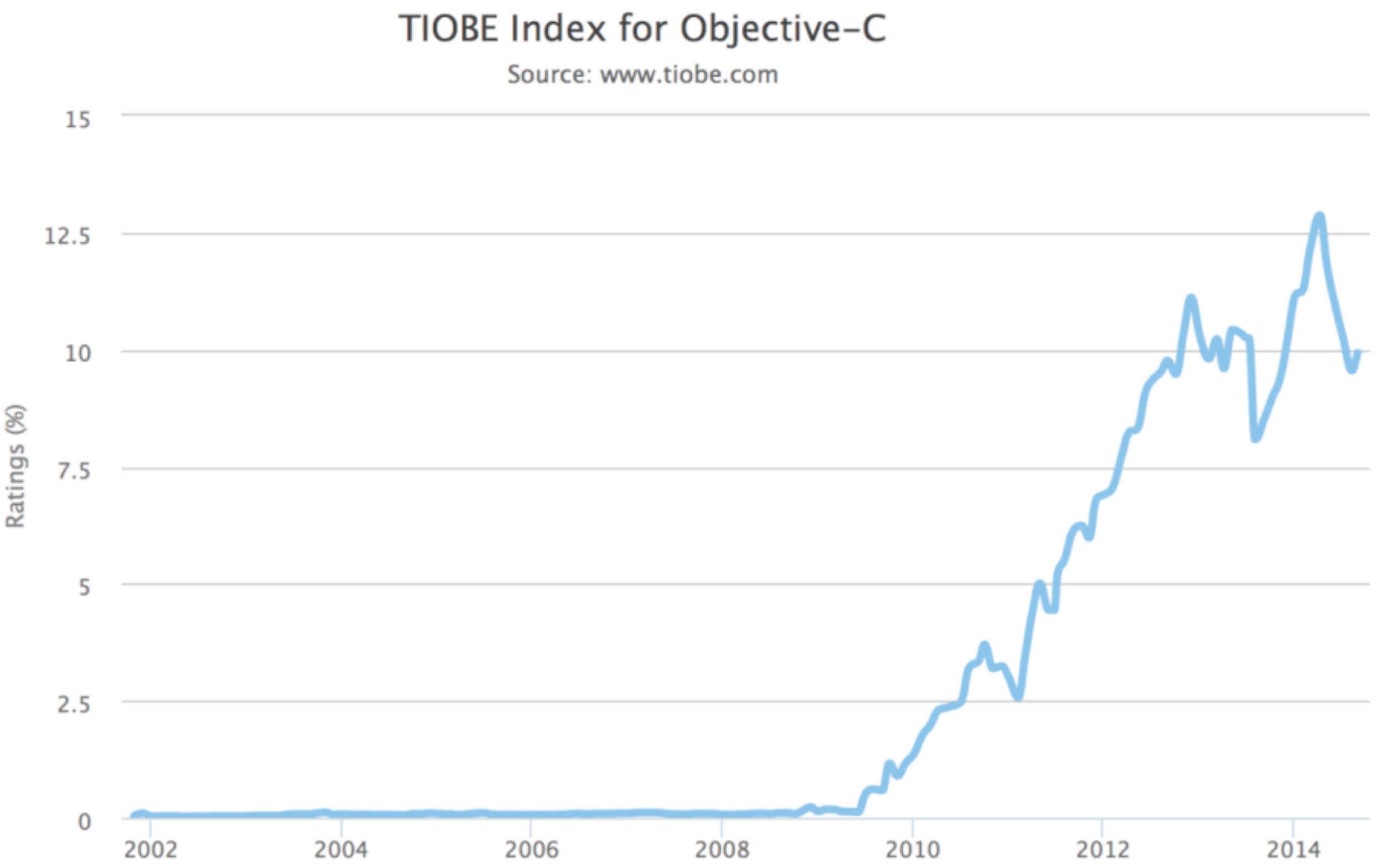
Info Pane

Test GNUstep for LinuxFocus

georges.t@linuxfocus.org

Objective-C

- objektově orientovaný jazyk, představen r. 1983
- výrazná popularizace až s NeXT, OS X a iOS
- rozšíření jazyka C o OOP
- OOP založeno na zasílání zpráv (na rozdíl od volání metod u C++) po vzoru Smalltalku
- třída – 2 soubory (class.m a class.h)


TIOBE Index for Objective-C
Source: www.tiobe.com
Ratings (%)
15
12.5
10
7.5
5
2.5
0
2002
2004
2006
2008
2010
2012
2014

C++:

```
obj->method(argument);
```

Objective-C:

```
[obj method:argument];
```

C++:

```
int method (int i)
{
    return square_root(i);
}
```

Objective-C:

```
- (int)method:(int)i
{
    return [self square_root:i];
}
```

iPhone OS

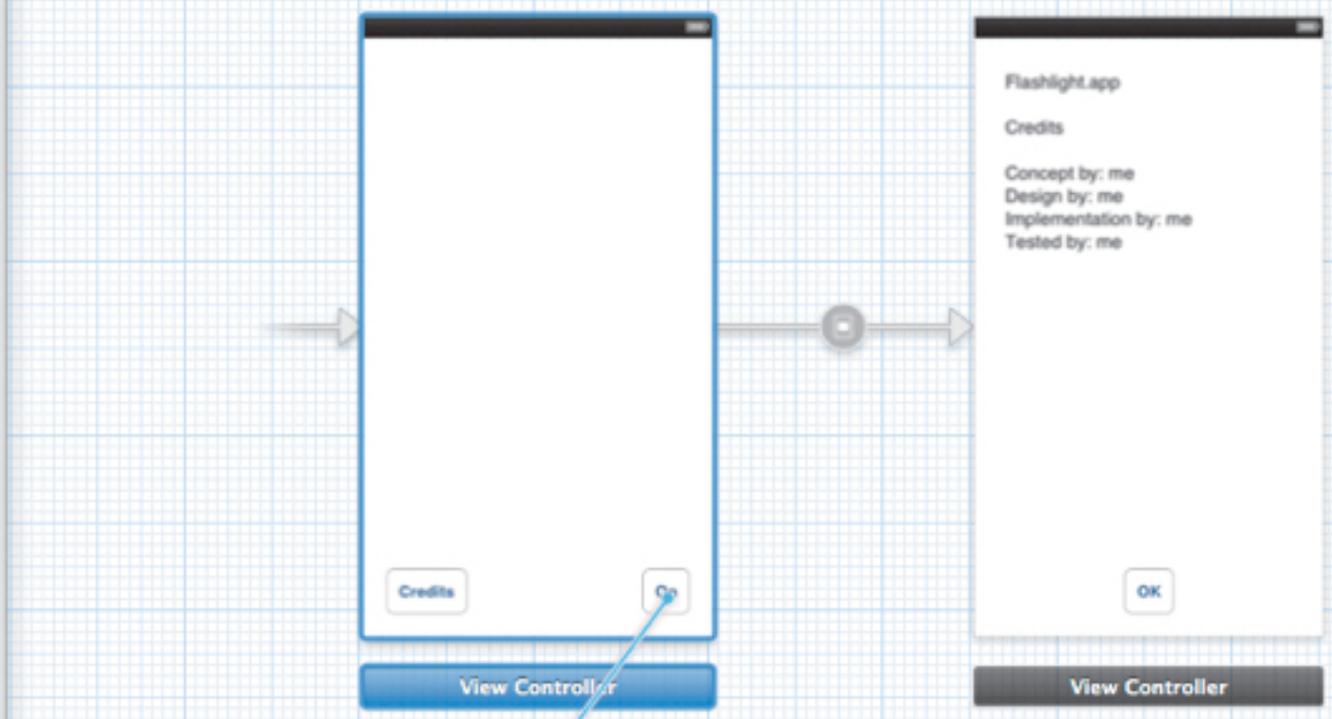
- dědictví z NeXTstep a OS X
  - Darwin (BSD-like základ NeXTstep a OS X)
  - Objective-C
  - Cocoa Framework
  - Xcode + Interface Builder
- memory management

Lighting Suite.xcworkspace — MainStoryboard_iPhone.storyboard

Run | Stop | Flashlight 〉 iPhone 6.1 Simulator (Scheme) | Breakpoints | Build Flashlight: **Succeeded** | Yesterday at 11:52 | Editor | View | Organizer

Flashlight 〉 Flashlight 〉 MainStoryboard_iPhone.storyboard 〉 MainStoryboard_iPhone.storyboard (English) 〉 View Controller Scene 〉 View Controller

- View Controller Scene
  - View Controller
    - View
      - Button – On
      - Button – Credits
      - Constraints
        - Horizontal Space – Button – Credit...
        - Vertical Space – View – Button – C...
        - Vertical Space – View – Button – On
        - Horizontal Space – View – Button – On
    - First Responder
    - Exit
    - Modal segue from Credits to View Controller
- View Controller Scene
  - View Controller
    - View
      - Text View
      - Button – OK
      - Constraints
        - Horizontal Space – Text View – View
        - Vertical Space – Text View – View
        - Horizontal Space – View – Text View
        - Center X Alignment – Button – OK...
        - Vertical Space – Button – OK – Tex...



Automatic 〉 LSViewController.h 〉 No Selection

```
1  //
2  //  LSViewController.h
3  //  Flashlight
4  //
5  //  Created by Maurice Kelly on 16/03/2013.
6  //  Copyright (c) 2013 Lighting Suite. All rights reserved.
7  //
8
9  #import <UIKit/UIKit.h>
10
11 @interface LSViewController : UIViewController
12
13 @property (weak, nonatomic) IBOutlet UIButton *button;
14
15 - (IBAction)toggleSwitch:(id)sender;
16
17 @end
18
```

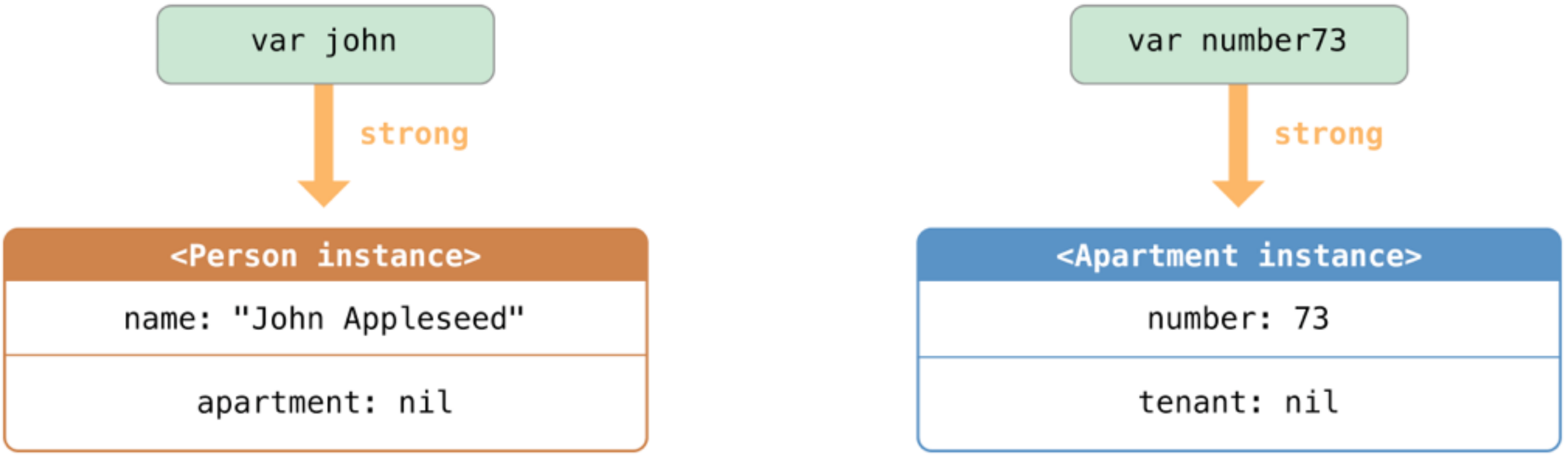
Connect Outlet

Počátky

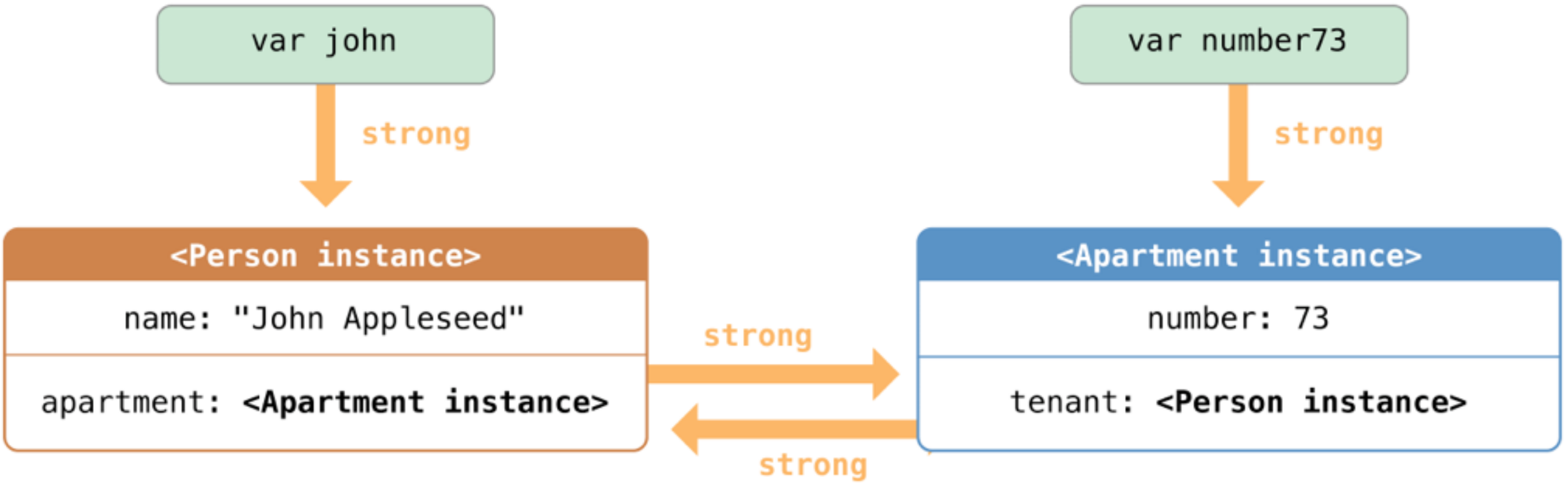
- Leden 2007 – představení iPhone a iPhone OS 1.0
- Červen 2008 – iPhone OS 2.0, App Store
- Červen 2009 – iPhone OS 3.0
- Červen 2010 – iOS 4.0
- …
- Červen 2014 – iOS 8.0

Automatic Reference Counting

- alternativa ke garbage collectoru
- založen na statické analýze kódu a udržování počtu referencí na instanci
- deterministický
- uvolnění paměti zajišťuje Apple LLVM kompilátor
- problém retain cyklů



zdroj: developer.apple.com



zdroj: developer.apple.com

var john

var number73

<Person instance>

name: "John Appleseed"

apartment: **<Apartment instance>**

strong

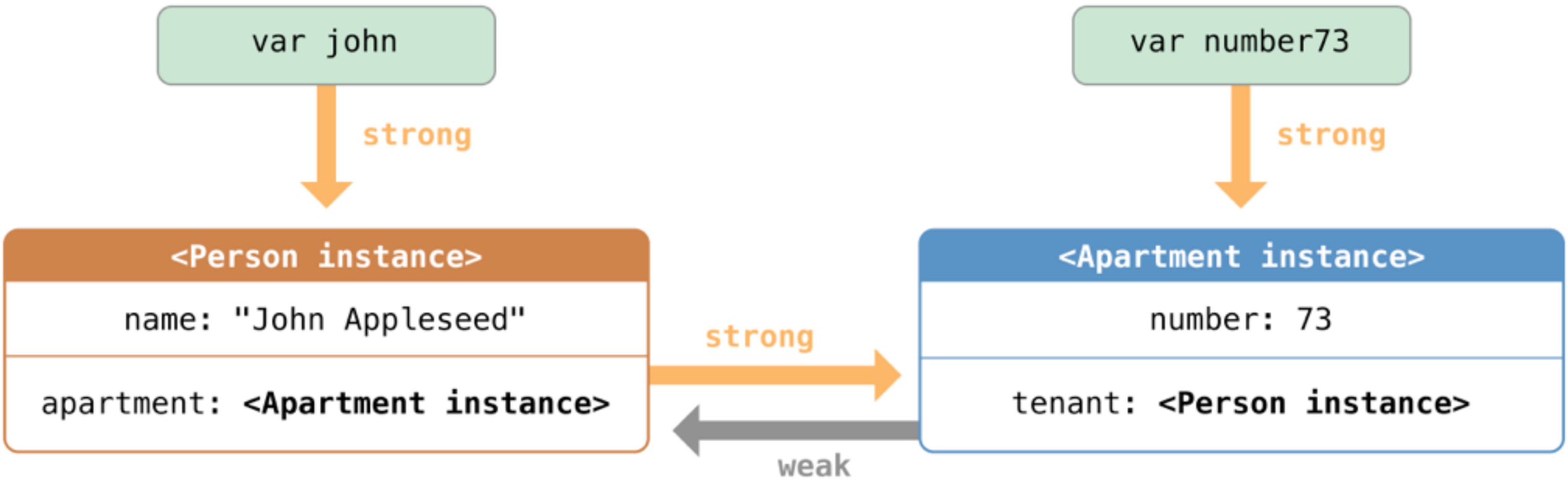
strong

<Apartment instance>

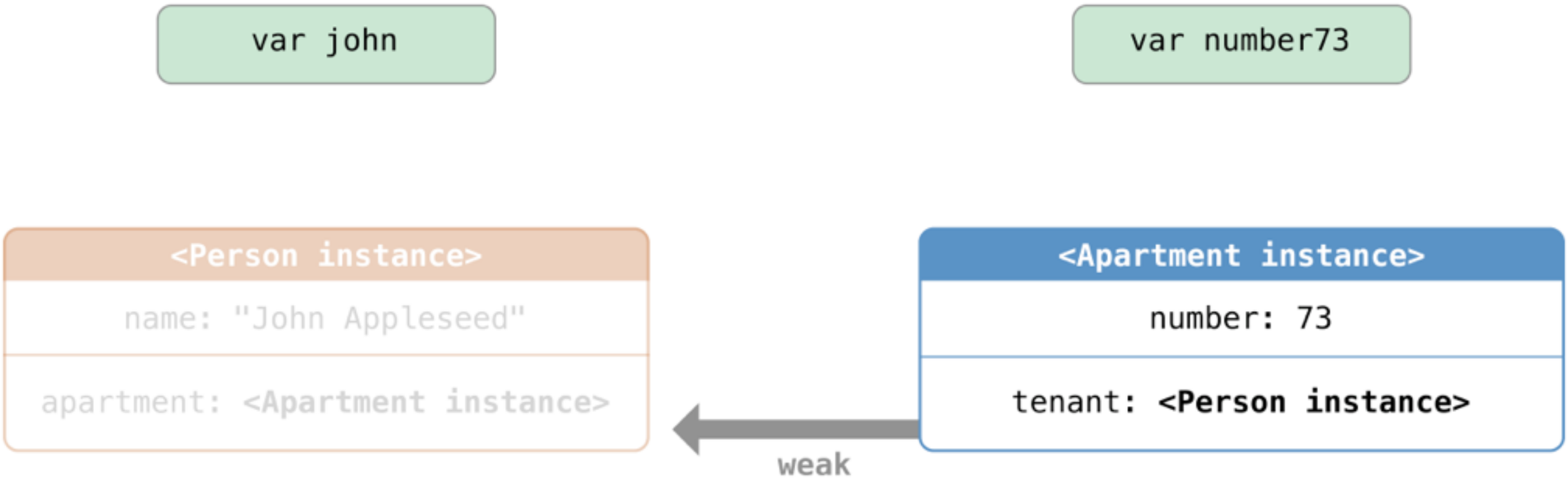
number: 73

tenant: **<Person instance>**

zdroj: developer.apple.com



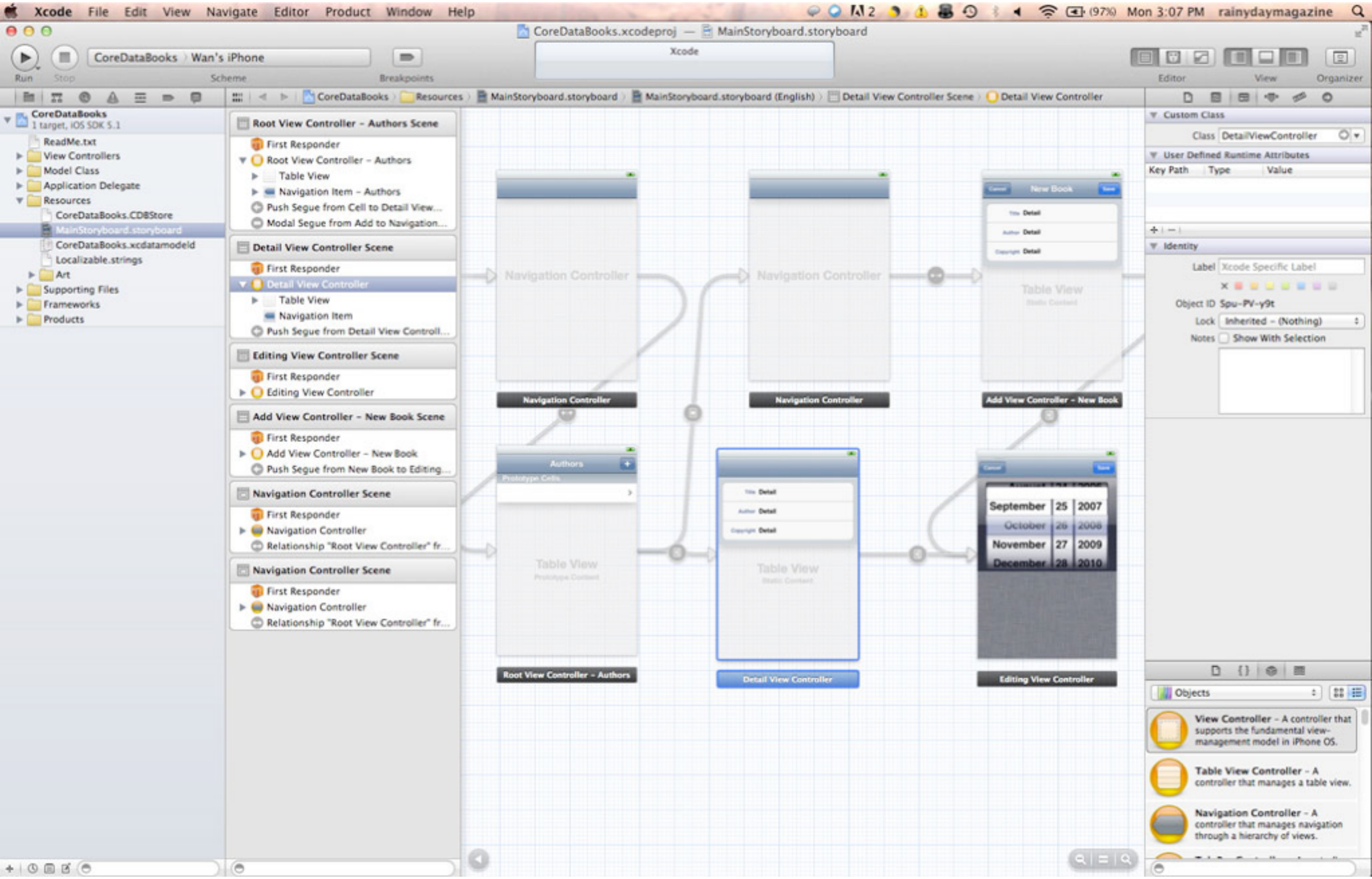
zdroj: developer.apple.com



zdroj: developer.apple.com

Storyboarding

- Návrh UI formou flow chartu
- Snadný prototyping
- Přizpůsobení rozhraní pro zařízení iPhone a iPad


Xcode File Edit View Navigate Editor Product Window Help
Mon 3:07 PM rainydaymagazine
CoreDataBooks.xcodeproj — MainStoryboard.storyboard
Run Stop CoreDataBooks › Wan's iPhone Scheme Breakpoints
Editor View Organizer
CoreDataBooks 1 target, iOS SDK 5.1
ReadMe.txt
View Controllers
Model Class
Application Delegate
Resources
CoreDataBooks.CDBStore
MainStoryboard.storyboard
CoreDataBooks.xcdatamodeld
Localizable.strings
Art
Supporting Files
Frameworks
Products
Root View Controller – Authors Scene
First Responder
Root View Controller – Authors
Table View
Navigation Item – Authors
Push Segue from Cell to Detail View...
Modal Segue from Add to Navigation...
Detail View Controller Scene
First Responder
Detail View Controller
Table View
Navigation Item
Push Segue from Detail View Controll...
Editing View Controller Scene
First Responder
Editing View Controller
Add View Controller – New Book Scene
First Responder
Add View Controller – New Book
Push Segue from New Book to Editing...
Navigation Controller Scene
First Responder
Navigation Controller
Relationship "Root View Controller" fr...
Navigation Controller Scene
First Responder
Navigation Controller
Relationship "Root View Controller" fr...
CoreDataBooks › Resources › MainStoryboard.storyboard › MainStoryboard.storyboard (English) › Detail View Controller Scene › Detail View Controller
Navigation Controller
Navigation Controller
Add View Controller – New Book
Authors
Table View
Prototype Content
Root View Controller – Authors
Table View
Detail View Controller
September 25 2007
October 26 2008
November 27 2009
December 28 2010
Editing View Controller
Custom Class
Class DetailViewController
User Defined Runtime Attributes
Key Path Type Value
Identity
Label Xcode Specific Label
Object ID 5pu-PV-y9t
Lock Inherited – (Nothing)
Notes Show With Selection
Objects
View Controller – A controller that supports the fundamental view-management model in iPhone OS.
Table View Controller – A controller that manages a table view.
Navigation Controller – A controller that manages navigation through a hierarchy of views.

iOS 7


8:41
3G
Monday
10
Calendar
Photos
Camera
Videos
Messages
Clock
Maps
280
Weather
Reminders
Stocks
Game Center
Passbook
Notes
App Store
iTunes Store
Newsstand
Settings
Compass
Mail
Safari
Music

iOS 8

- App Extensions
  - Action
  - Today
  - Keyboard
- Handoff


Swift

Swift

- multi-paradigmatický jazyk
- inspirován jazyky Objective-C, Rust, Haskell, Ruby, Python, C#
- Snaha eliminovat programátorské chyby
- Moderní prvky: generics, closures, type inference
- Funkcionální prvky (map, reduce, filter, …)
- Rychlejší než Objective-C

Variables

```
var languageName: String = "Swift"
```

Constants

```
let languageName: String = "Swift"
```

Type inferrence

```
let languageName = "Swift"   // String
let introduced = 2014         // Int
let isAwesome = true          // Bool
```

Unicode names

```
let 🐶🐮 = "dogcow"
```

Arrays and dictionaries

```
var names = ["Anna", "Alex", "Brian", “Jack"]
```

```
var numberOfLegs = ["ant": 6, "snake": 0, "cheetah": 4]
```

Optionals

```
let numberOfLegs = ["ant": 6, "snake": 0, "cheetah": 4]

let possibleLegCount: Int? = numberOfLegs["aardvark"]
```

```
if possibleLegCount == nil {
    println("Aardvark wasn't found")
} else {
    let legCount = possibleLegCount!
    println("An aardvark has \(legCount) legs")
}
```

Optionals

```
let numberOfLegs = ["ant": 6, "snake": 0, "cheetah": 4]

let possibleLegCount: Int? = numberOfLegs["aardvark"]
```

```
if let legCount = possibleLegCount {
    println("An aardvark has \(legCount) legs")
}
```

Functions

```
func buildGreeting(name: String = "World") -> String
{
    return "Hello " + name
}

let greeting: String = buildGreeting()
```

Closures

```
let greetingPrinter: () -> () = {
    println("Hello World!")
}
```

```
greetingPrinter()    //Hello world
```

Closures as Parameters

```
func repeat(count: Int, task: () -> ()) {
    for i in 0..count {
    task() }
}

repeat(2, {
    println("Hello!")
})
```

Trailing closures

```
func repeat(count: Int, task: () -> ()) {
    for i in 0..count {
    task() }
}

repeat(2) {
    println("Hello!")
}
```

Closures

```
struct Array<String> {
    func sort(order: (String, String) -> Bool)
}

clients.sort({(a: String, b: String) -> Bool in
    return a < b
})
```

Closures

```
struct Array<String> {
    func sort(order: (String, String) -> Bool)
}

clients.sort({ a, b in
    return a < b
})
```

Closures

```
struct Array<String> {
    func sort(order: (String, String) -> Bool)
}

clients.sort({ $0 < $1 })
```

Properties

```
class Vehicle {

    var description: String {
        get {
            return "Speed: \(speed)"
        }
        set { }
    }

    var speed: Double = 0.0  {
        willSet {
            if newValue > 90.0 {
                println("Careful now.")
            }
        }
    }
}
```

Structures

```
struct Rect {
    var width: Double
    var height: Double

    var area: Double {
        return width * height
    }

    func isBiggerThanRect(other: Rect) -> Bool {
        return self.area > other.area
    }
}
```

Generics

```
struct Stack<T> {
    var elements = [T]()

    mutating func push(element: T) {
        elements.append(element)
    }

    mutating func pop() -> T {
        return elements.removeLast()
    }
}

var intStack = Stack<Int>()
intStack.push(50)
let lastIn = intStack.pop()
```

Spolupráce mezi Swiftem a Objective-C

- Problém
  - Systémové frameworky napsány v Objective-C (Cocoa)
  - externí knihovny, legacy kód
- Řešení
  - Objective-C -> Swift – ručně: bridging header
  - Swift -> Objective-C – automaticky: generovaný header

Playgrounds

- Interaktivní prostředí pro vyzkoušení si kódu

- Možnost kombinace kódu s HTML textem (interaktivní učebnice)

Playgrounds demo

Díky!

Otázky?