Mobile Application Development

MAS 490: Theory and Practice of Mobile Applications

Professor John F. Clark

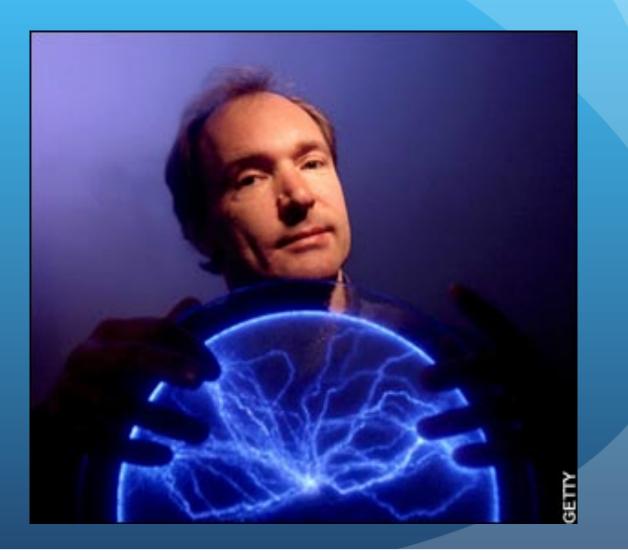
What is Interface Builder?

- Interface Builder is a software development application for Apple's Mac OSX operating system.
- It is part of Xcode, the Apple developer's toolkit
- Interface Builder allows Cocoa and Carbon developers to create interfaces for applications using a graphical user interface (GUI).
- The resulting interface is stored as a .nib file, short for NeXT Interface Builder
- As of Xcode 4, Interface Builder is no longer a stand-alone application and is fully integrated into Xcode

History of Interface Builder

- Interface Builder dates back to 1986. It was originally written in LISP, which is the second-oldest high-level programming language still in use. It was deeply integrated into the Macintosh toolbox in the days of Mac OS 8.
- It was introduced to Steve Jobs when he was running NeXT. By 1988 it was incorporated in NeXTSTEP.
- It was the first commercial application that allowed buttons, menus, and windows to be placed in an interface using a mouse.

Tim Berners-Lee



Tim uses a NeXT computer to design the first web browser



NeXTSTEP Desktop

The Part									
info	. P.							Hal	boors
File									
Edt	P.		Disk E3	E	File Viewe			create	
Disk			Eject #	-	161			Cataghile	
View			Initialize	285	1		72	eof	
Tools	P.		Check for Disks				m l	MagicCap MagicDay	
Windo	_	a	2	paul	Imp	Loc	alApps	netrifo-tail	к.
Servic							1000	nextdoom	
Hide	8		10					nerð admir nerð-class	
LogO	ut q		$ \circ$					next-icon	
			Celeta	an and pushake on re	note data		533.34	next-jobs	
		44	35 messages (60.2ME) - 3	10	A4			nerd mana Next-PPI	igers P
			A4399 Jan 18 To		· 45	- P		nerd prog	
			4402 Jan 18 El 4403 Jan 18 El			REN	U.,	nifp-hom	ebneur
		6	4404 Jan 18 El	Polande -	paul	Adv.Net	40.47	nugi nuukvic	
	1	Attributes inspecto	X0 19 10			+		Own/Web	
	Amount of		19 To	A Didavid	E R Adv Network			pdo	
		ASIOUNS 4	roughly	hard_disk.hdf	AppMan diag			ph7 procmail	
	Const.	1	100gray	mailtest	P Apps P A SoftPC local			PRite	
	1000	Adv.NetInfo.rtf	1 ago, 51	the news-archive	r Barclays	1 B		quickbase Radium	£
	\$633		low he's conture	The second se	F. Dargains.Prop	06		-Thompso	
	Patte /t	iome/paul	- and a second	paul	 Calendar CISMall 	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1		SafetyNet	
	10000000		ought N		CISnews			SarNet uk-ne-6-ar	
	10000		much as for relea		T Cmos.nam			uk-nav6-ut	
	Link To				piedy". Rhapsody will be	e based initially on		uk-riders	
	1200				th NeXT, and with a deve			WebObier	che .
	Size:	30.2KB	any of e		aded by Avadis Tevaniar	n, formerly head of		E E	CII)
	Owner:	paul	12 E						_
	Come I	internet and		I run on all current Power PC Macintoshes. The initial release for developers				Name: Onn/H	Veb.mbcx
	Group: wheel			penStep API (Application Programming Interface), which is part of the spending system, and the user interface also from NeXCTSTEP, with minor				Delete	C COLUMN
	_	Permissions	Changed ; plan for	n the Macintosh System	7 interface. The full relea	ise of Rhapsedy		Leen	
				her adaptations of the user interface, will include additional tookits from				Open	Trans
	Read	Y X X	121	a and the treat	11.12.1			-	
	100000	Owner Group Other	56						
			read .						
		Revet O	e 1						
	-								





1100

How does it work?

- Interface Builder provides collections (or palettes) of user interface objects (text fields, data tables, sliders, and pop-up menus, for example) to the Objective-C programmer.
- The palettes are extensible, meaning you can customize and develop new objects that can be added to new or existing palettes.
- To build an interface, the developer just drags and drops objects into a window or menu. Actions that the objects can perform are connected to targets in the code and outlets (pointers) declared in the code are connected back to objects.

How does it work? Part II

- Interface Builder saves an application's interface as a bundle that contains the interface objects and relationships used in the application.
- These objects are archived into either an XML file or a NeXT-style property list file with a .nib extension.
- Upon running an application, the proper NIB objects are unarchived, connected into the binary of their owning application, and awakened.
- NIBs are often referred to as *freeze dried* because they contain the archived objects themselves, ready to run.

Basic Tools

Xcode developer environment
For writing code

- Interface Builder
 - GUI for designing interfaces
- Connections:
 - Xcode has IBOutlet and IBAction types to connect Interface Builder elements to code objects.

Some User Interface (UI) Elements

NavigationController (optional)
ViewController
View
Image
Label
Button
TextField

Varieties of "C" Code

- Objective-C basics:
 - Simple method, no parameters:
 - [robot stand];
 - robot.stand();
 - Method with one parameter:
 - [robot walkDistance:(int)distance];
 - robot.walk(int distance);
 - Method with two parameters:
 - [robot walkDistance:(int)distance inDirection: (float)direction];
 - robot.walk(int distance, int direction);

View Controller

 Most interface logic belongs in a view controller subclass

• Event Driven

- Init/InitWithNibName:Bundle:
- ViewDidLoad
- ViewWillAppear:
- ViewDidAppear:
- ViewWillDisappear:
- ViewDidDisappear:

Common Usage

ViewDidLoad

- Called once when view is finished initializing and is added to the view stack
- Use this for code that you want to run only once before the user sees anything
- Example:
 - Set the title of the view
 - [self setTitle:@"My View"];

Common Usage, Part II

• ViewWillAppear

- Called just before the view becomes visible, can be called multiple times (back button)
- Use this for code that you want to run every time the view is displayed
- Example:
 - Reload a page's dynamic contents
 - [self setViewCounter:viewCounter + 1];
 - self.counterLabel.text = self.viewCounter;

Common Usage, Part III

- ViewDidDisappear
 - Called after the view becomes invisible, can be called multiple times
 - Use this for code that you want to run every time the view is gone
 - Example
 - Stop refreshing a timer
 - [self stopMyTimer];

A More Complicated Example

- Let's suppose a View Controller contains six labels.
 - The content of three of the labels is dynamic and could be longer than one line.
 - The code must resize the labels and move down the remaining labels.

```
// Category for UIView
@interface UIView (UIKitExtensions)
- (void)moveDown:(int)pixels;
- (void)updateSize:(CGSize)newSize;
@end
```

```
@implementation UIView (UIKitExtensions)
```

```
- (void)moveDown:(int)pixels {
    self.frame = CGRectMake
(self.frame.origin.x, self.frame.origin.y +
pixels, self.frame.size.width, self.frame.size.height);
}
```

```
- (void)updateSize:(CGSize)newSize {
    self.frame = CGRectMake
(self.frame.origin.x, self.frame.origin.y, newSize.width,
newSize.height);
}
```

@end

int labelWidth = challenges.frame.size.width; int labelHeight = challenges.frame.size.height; int extraHeight = 0; int labelMaxHeight = 2000; int originalHeight; UIFont *font = challenges.font; CGSize size;

challenges.text = issue.challenges; contributingFactors.text = issue.contributingFactors; consequences.text = issue.consequences;

// update next label
extraHeight += size.height - originalHeight;
[contributingFactorsHeader moveDown:extraHeight];
[contributingFactors moveDown:extraHeight];

//...continued

// update next label
extraHeight += size.height - originalHeight;
[consequencesHeader moveDown:extraHeight];
[consequences moveDown:extraHeight];