



iP*ios*e

Development

a brief introduction

really, remarkably, very

Pete Goodliffe
pete@goodliffe.net

Pete Goodliffe

*A programmer, a columnist,
an author, a teacher. Someone
who cares about code.*

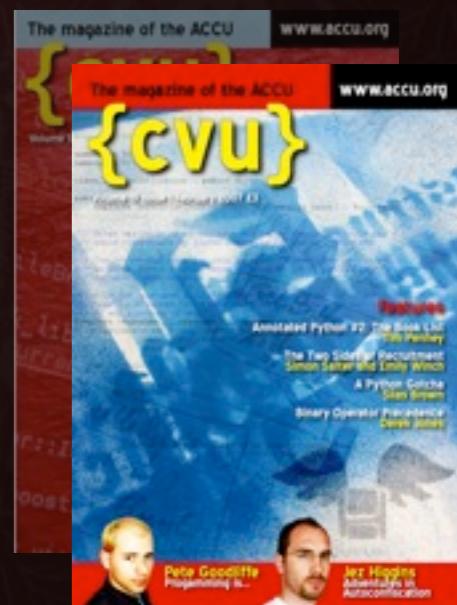
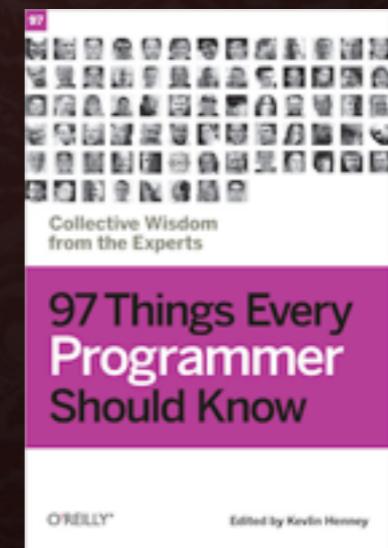
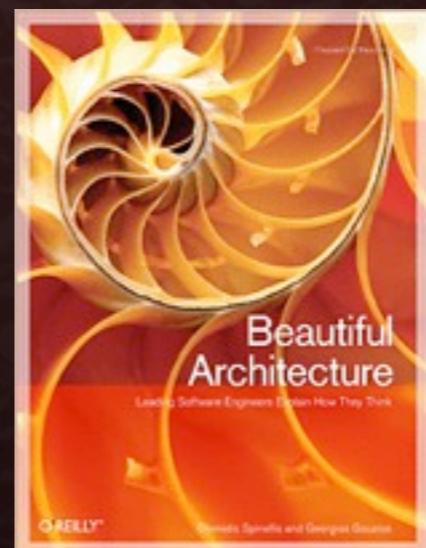
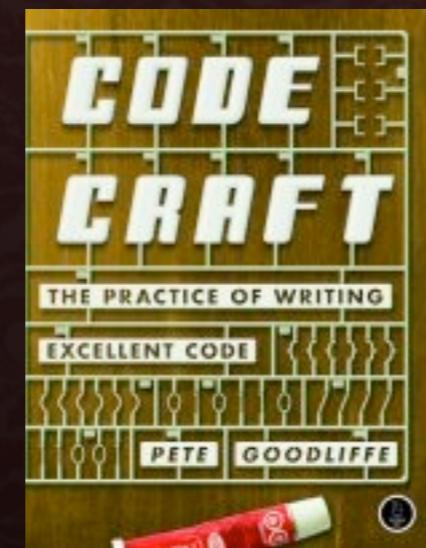
www.goodliffe.net
goodliffe.blogspot.com
[@petegoodliffe](https://twitter.com/petegoodliffe)



Pete Goodliffe

*A programmer, a columnist,
an author, a teacher. Someone
who cares about code.*

www.goodliffe.net
goodliffe.blogspot.com
[@petegoodliffe](https://twitter.com/petegoodliffe)





talk synopsis

An introduction to iPhone/iOS development

An introduction to iPhone development for beginners or those who are casually interested and want a leg up the learning curve.

This talk is an overview of how to get started as an iPhone developer, and an introduction to the core technologies, including:

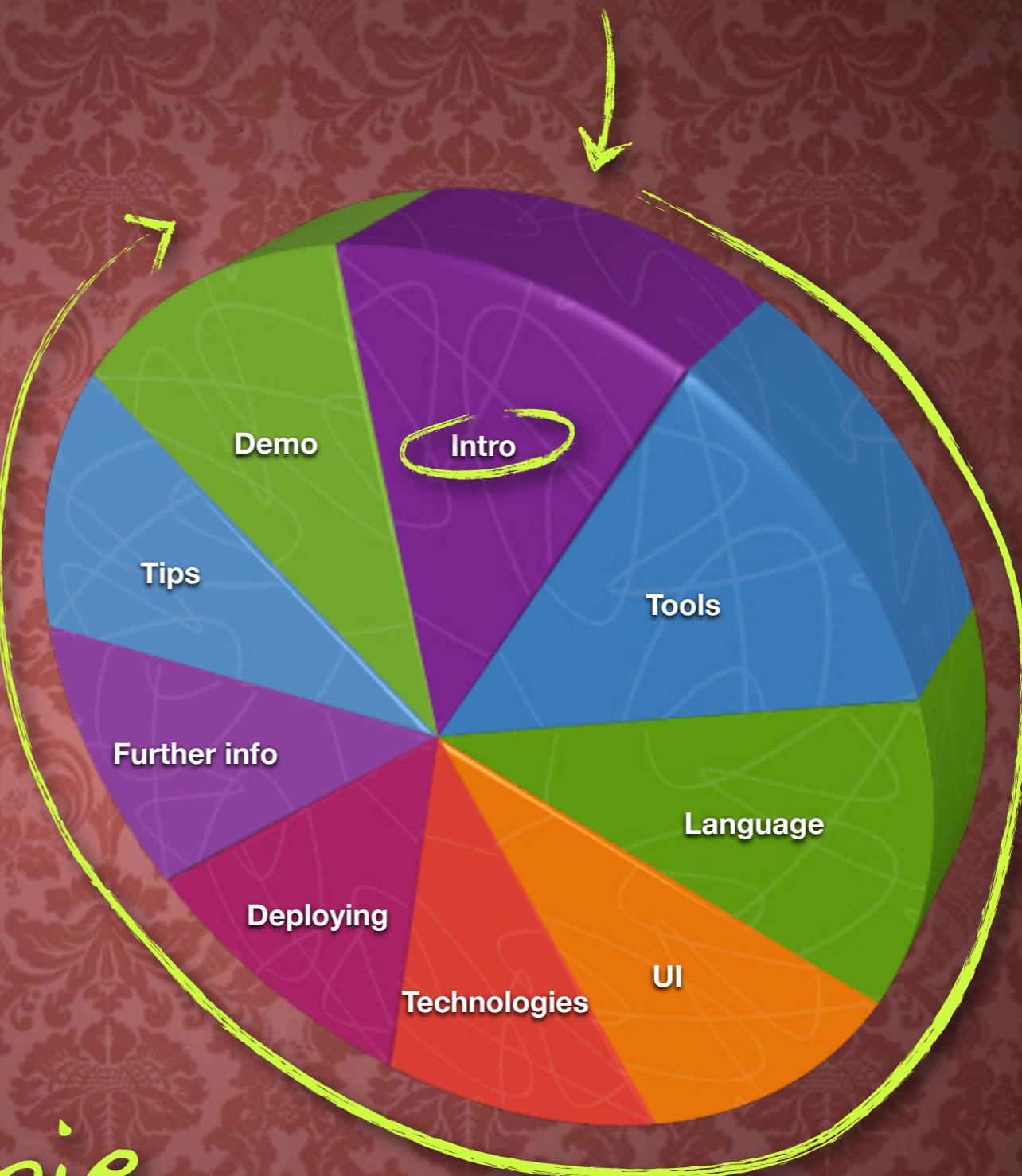
- ▶ *Common iPhone/Mac OS design patterns, idioms, and practices*
- ▶ *Becoming a native: how to "think in iPhone"*
- ▶ *Overview of the libraries and facilities that exist*
- ▶ *Limitations of the development environment*



90 minutes? is that all?!!!

in 90 minutes...

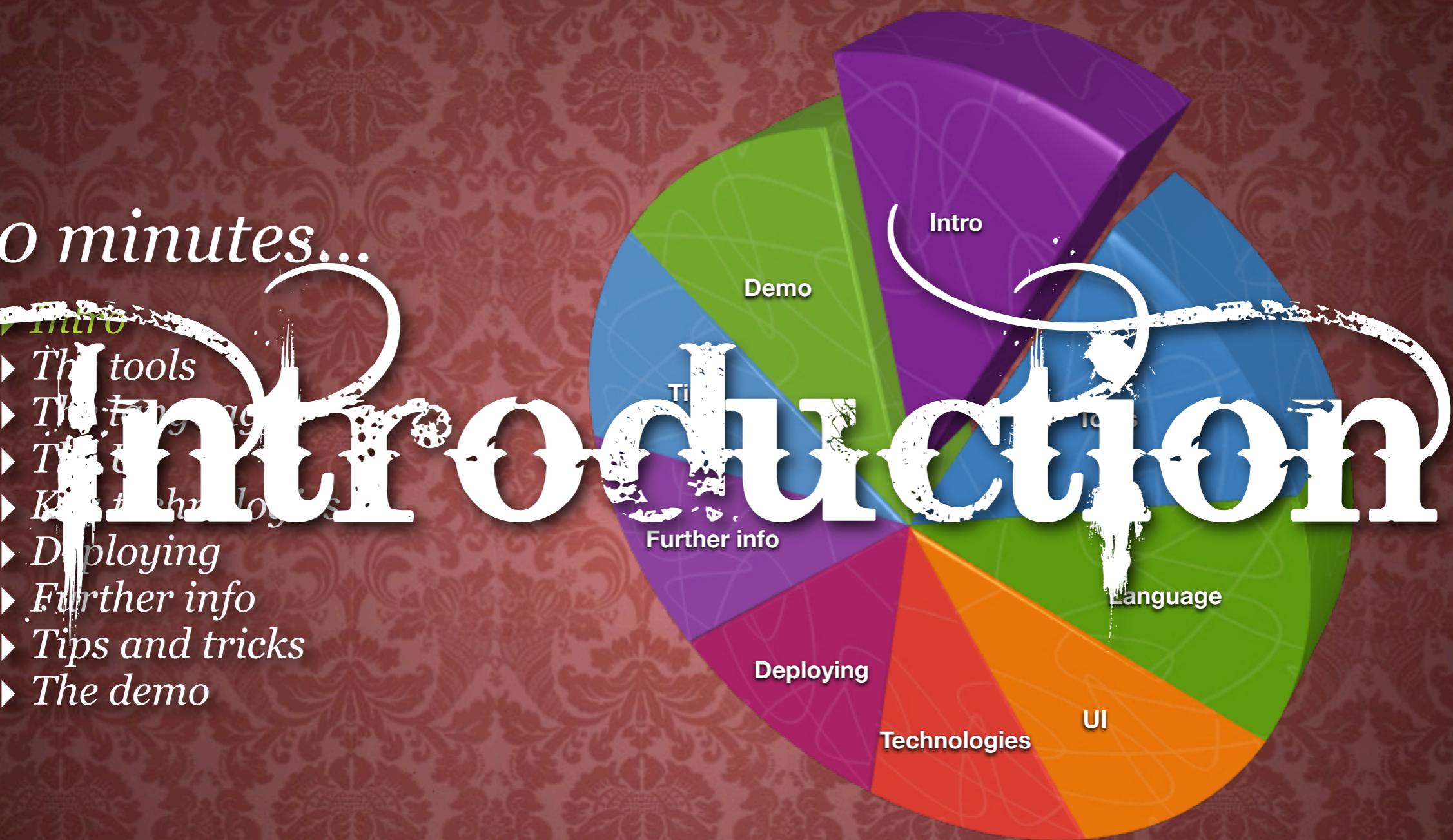
- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



Mmm... pie

in 90 minutes...

- ▶ [Intro](#)
- ▶ [The tools](#)
- ▶ [The technologies](#)
- ▶ [The deployment](#)
- ▶ [Key technologies](#)
- ▶ [Deploying](#)
- ▶ [Further info](#)
- ▶ [Tips and tricks](#)
- ▶ [The demo](#)



who are you?

why develop for the iPhone?

the alternatives

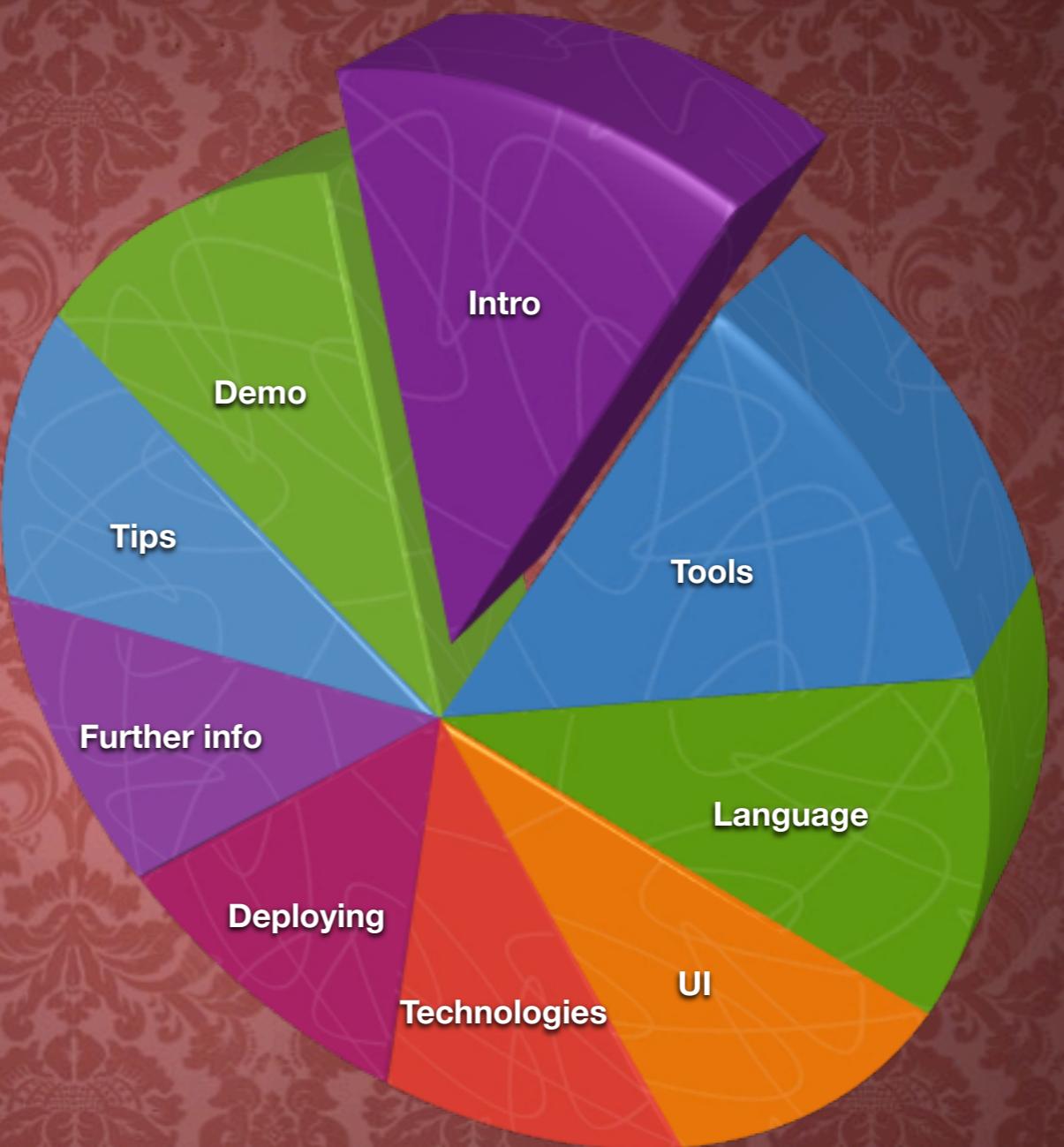


HTML



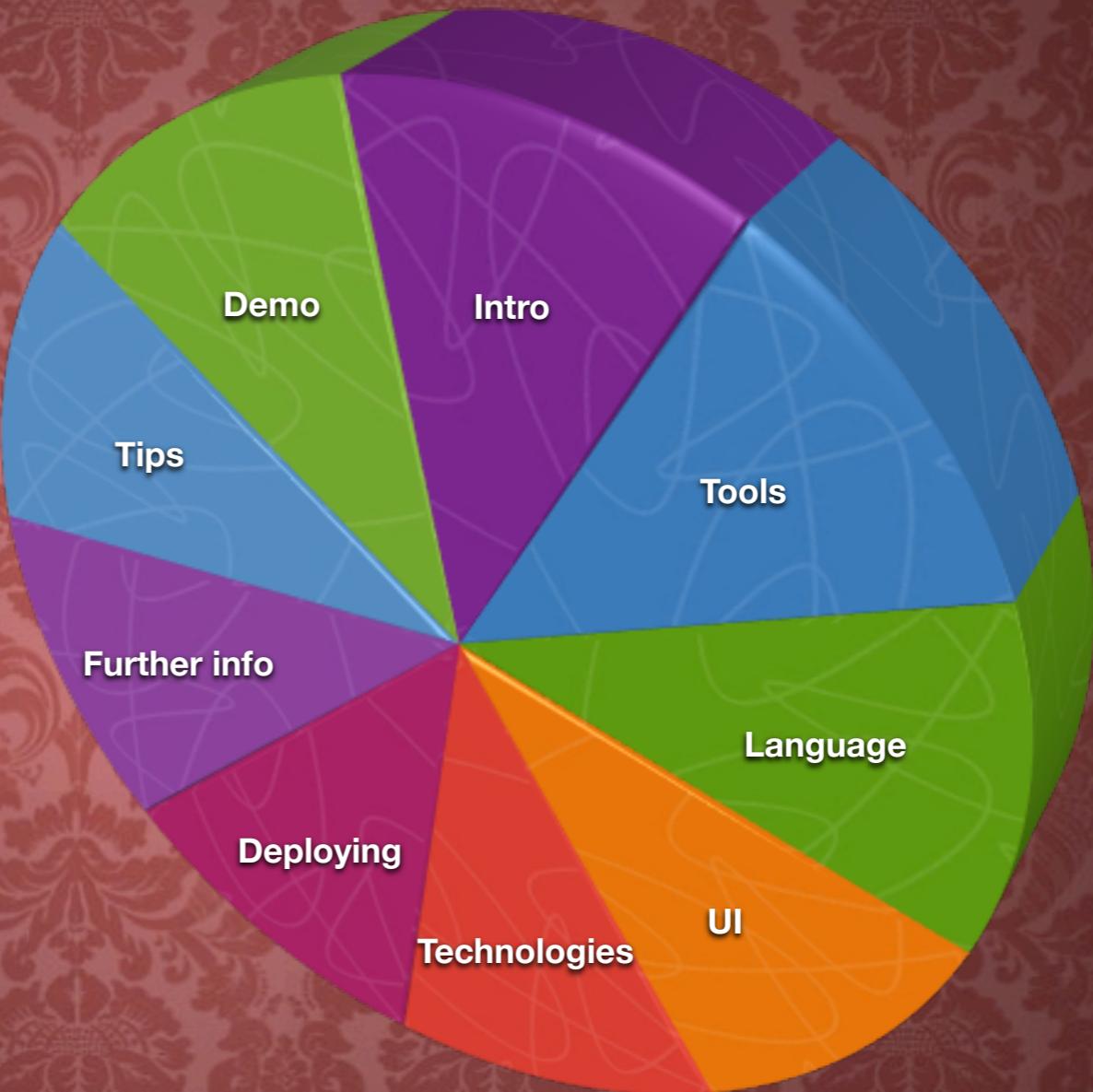
in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

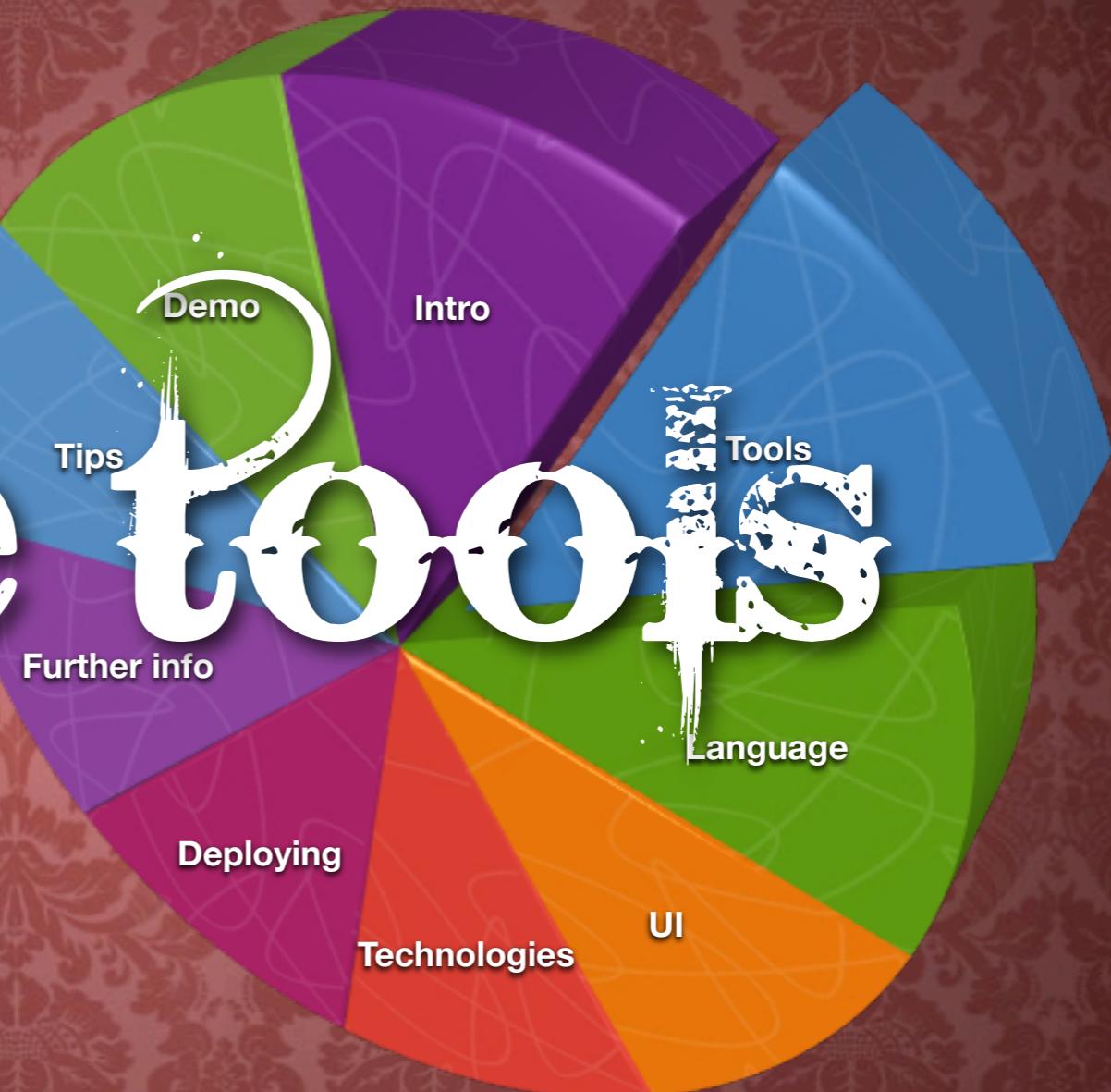
- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*

the tools

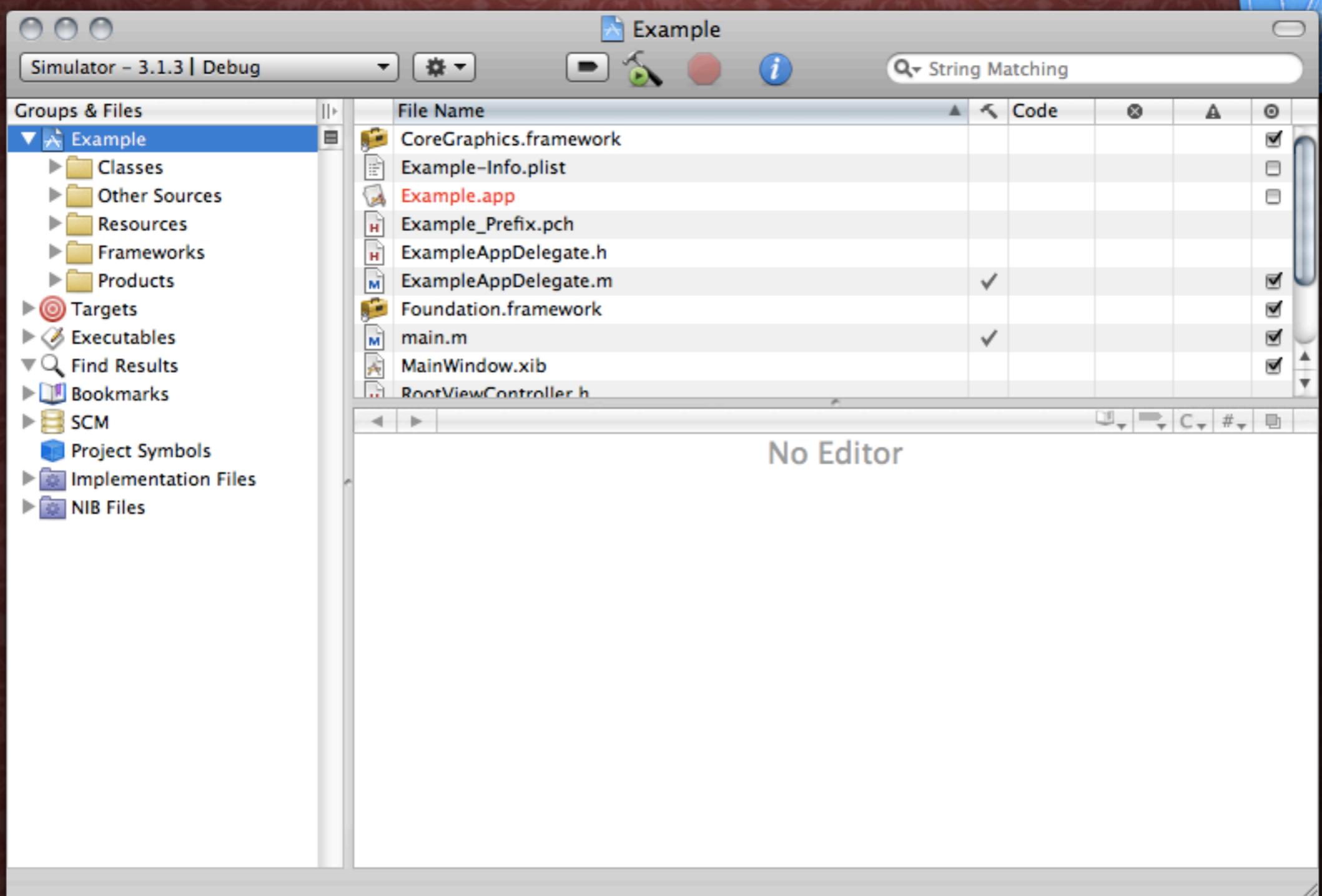


❖ *The cast* ❖

Xcode
iPhone/iPod Touch/iPad
Interface builder
iPhone simulator
Instruments/Shark
iOS Provisioning Portal
Other tools

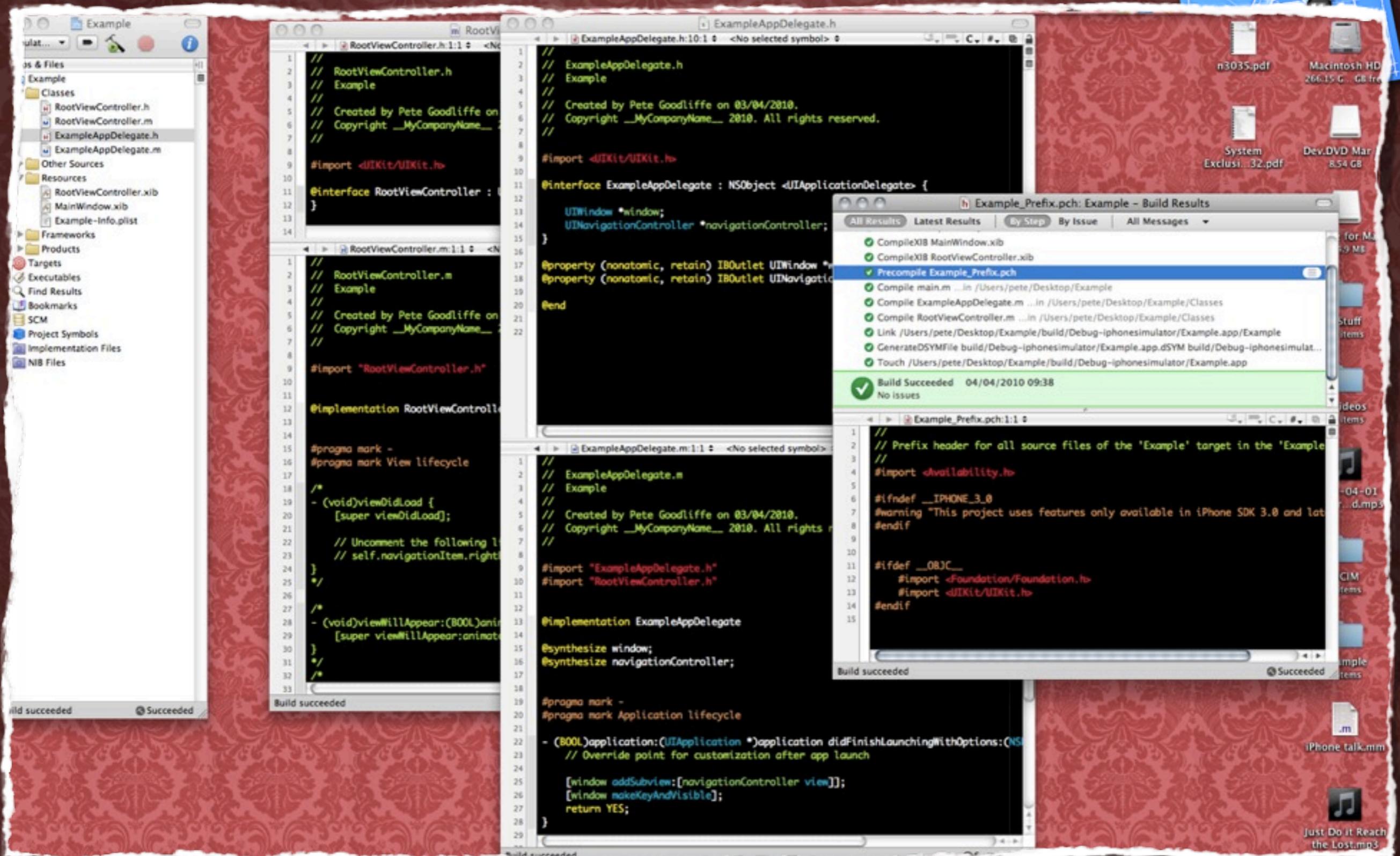


Xcode



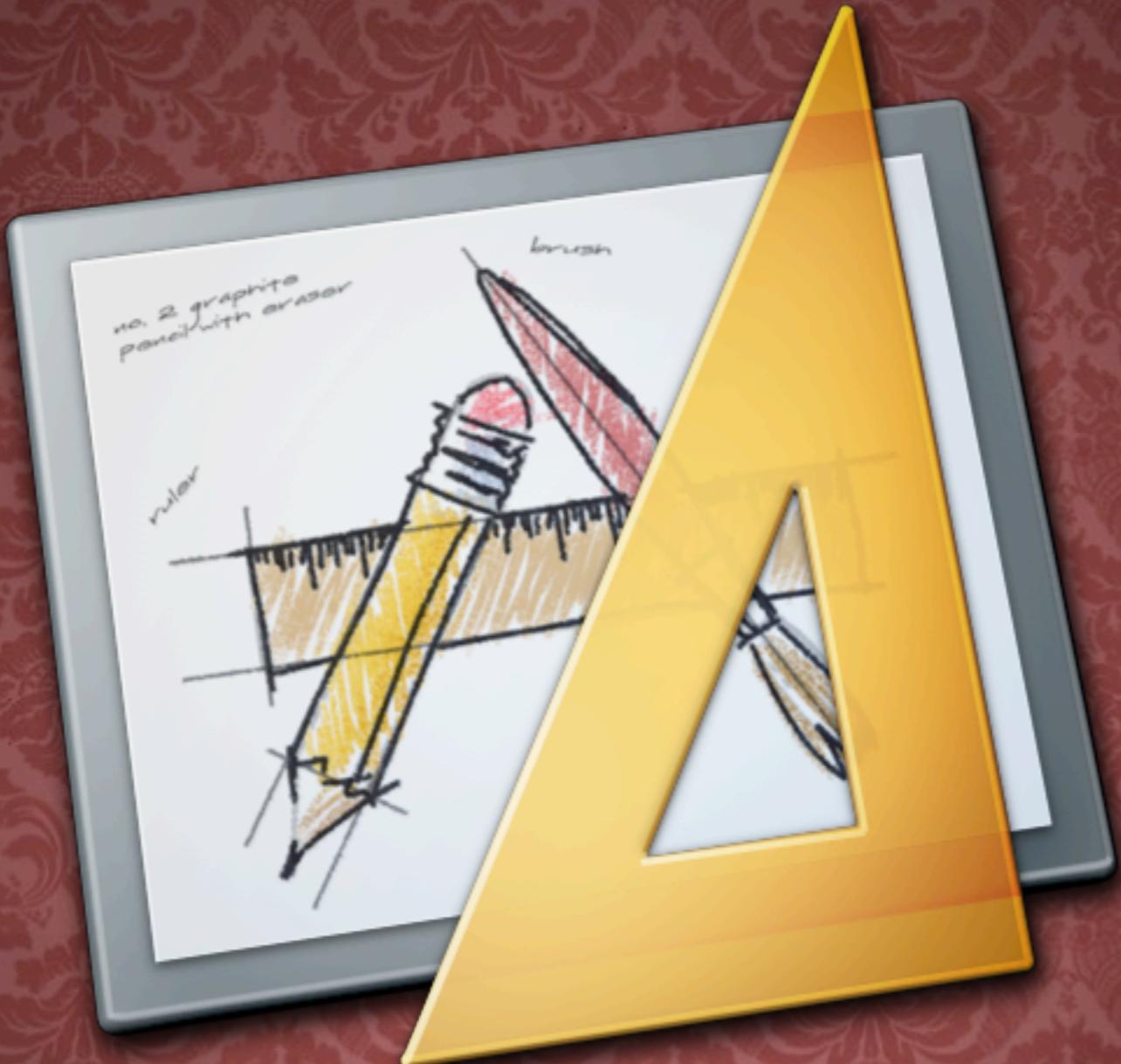
iPhone development a brief introduction

Pete Goodliffe

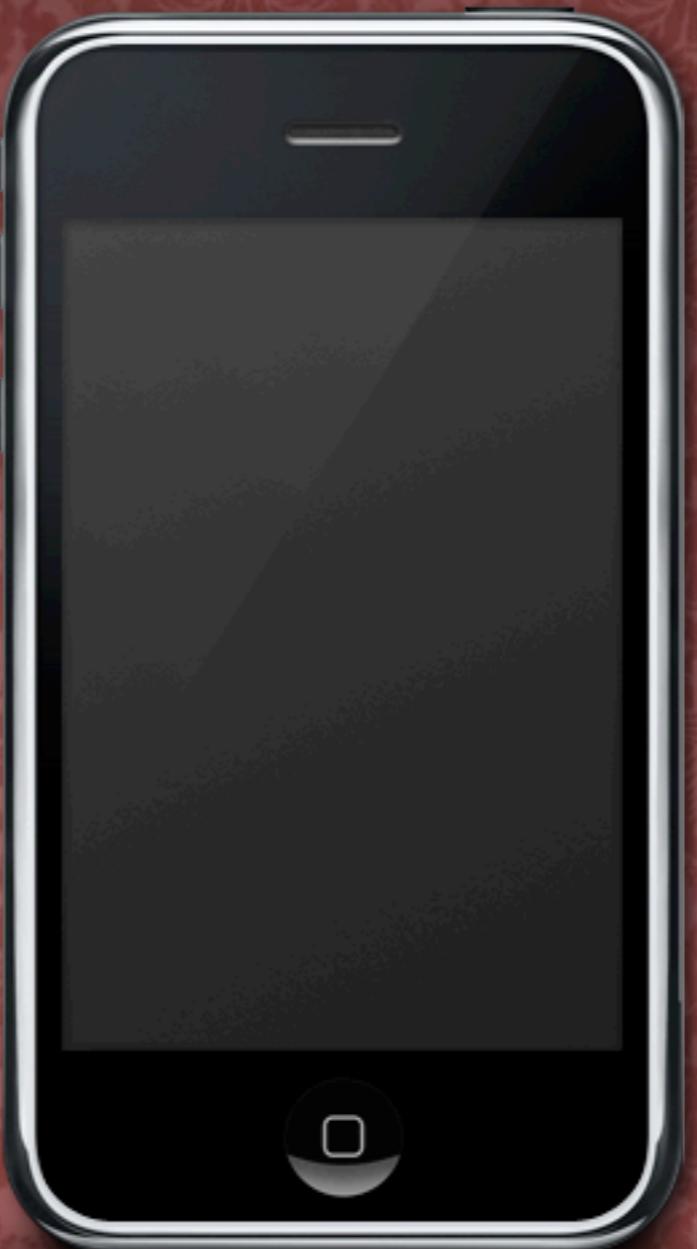




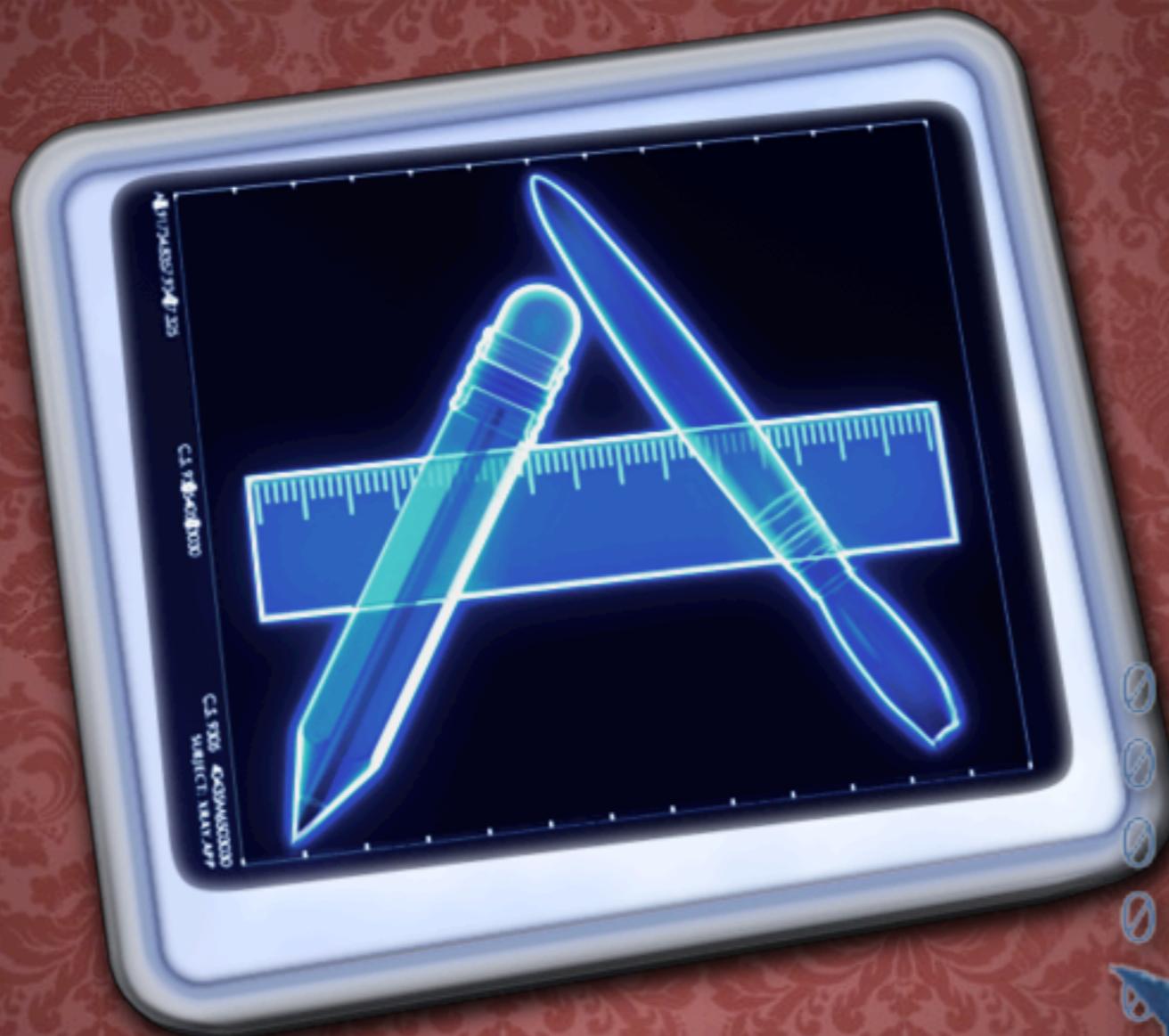
an iPhone



Interface Builder



iPhone simulator



Instruments/Shark



Other tools

Getting set up...

Get a Mac
Get the tools

Install

Get an ADC account \$\$\$

Code

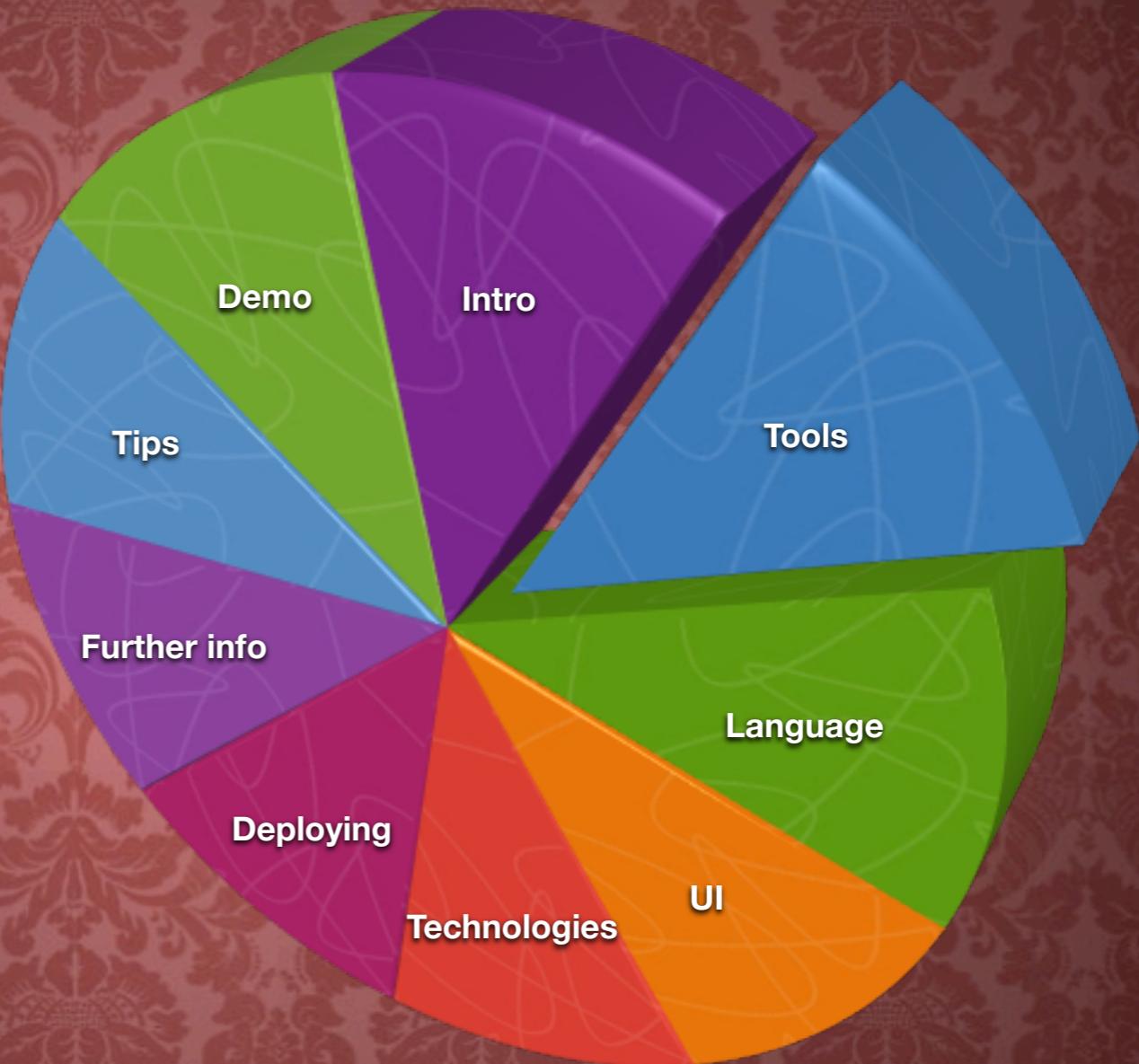
...

Profit



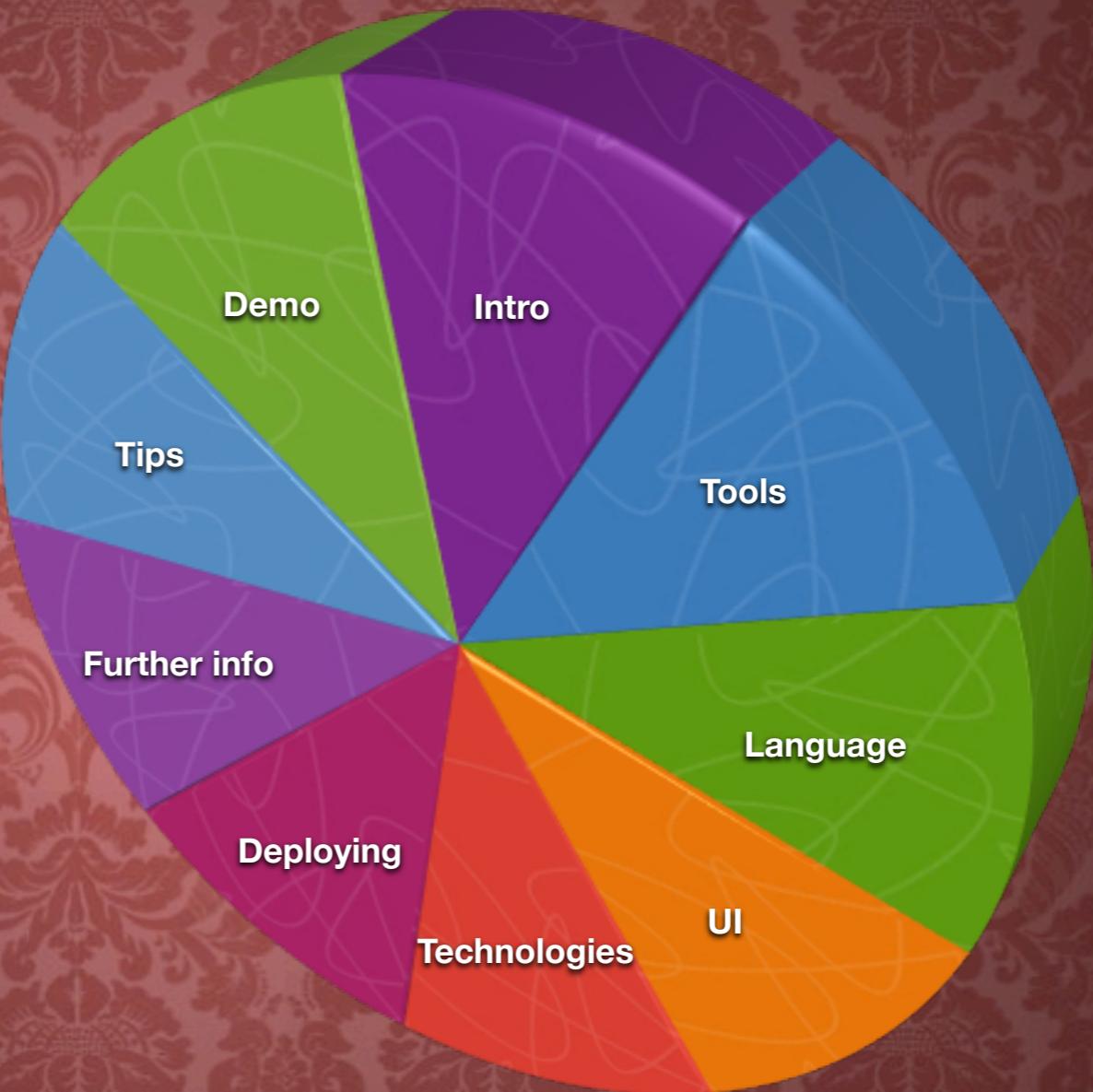
in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

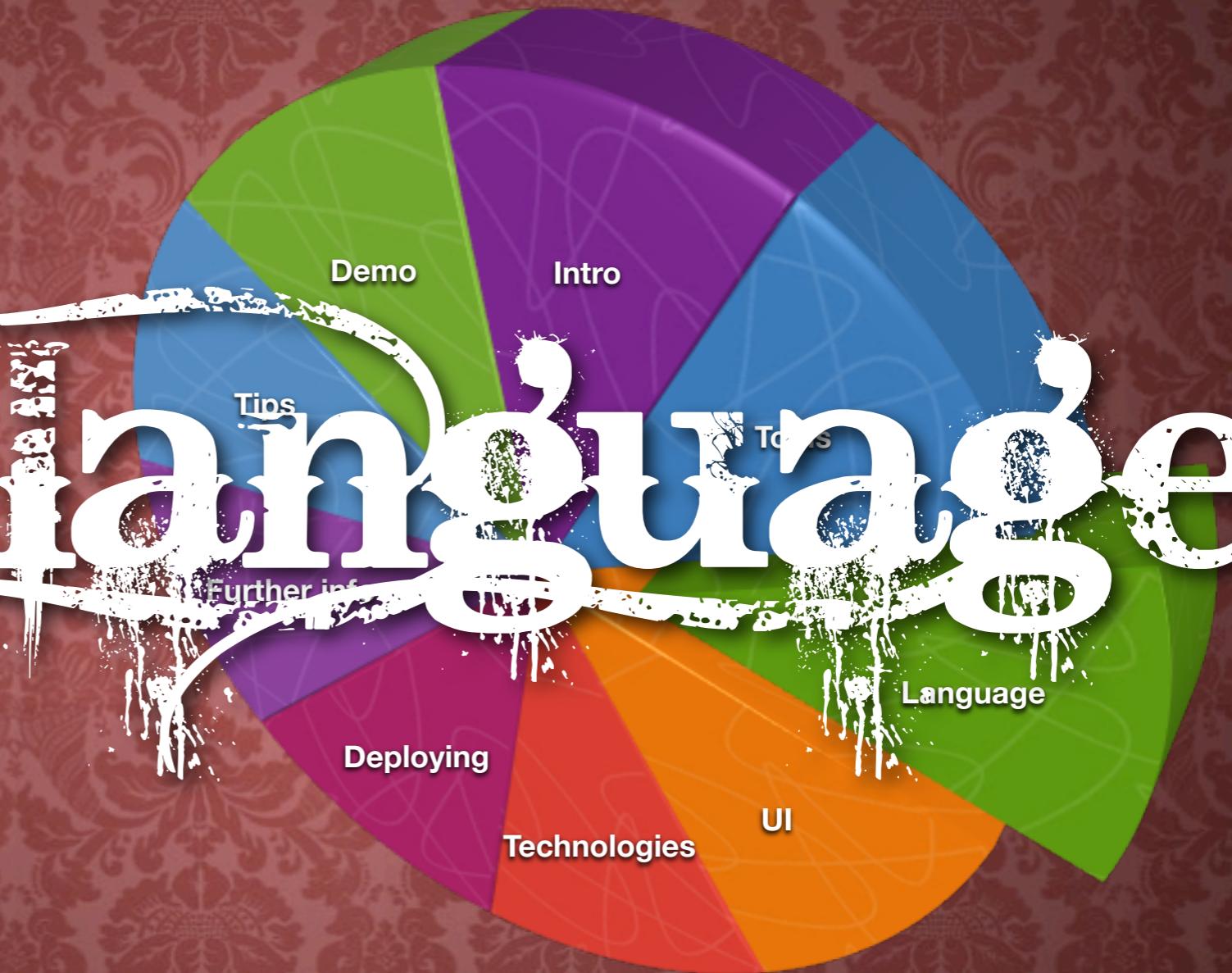
- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*

The language



(well, languages)

C

C++

Objective-C

(well, languages)

C

C++

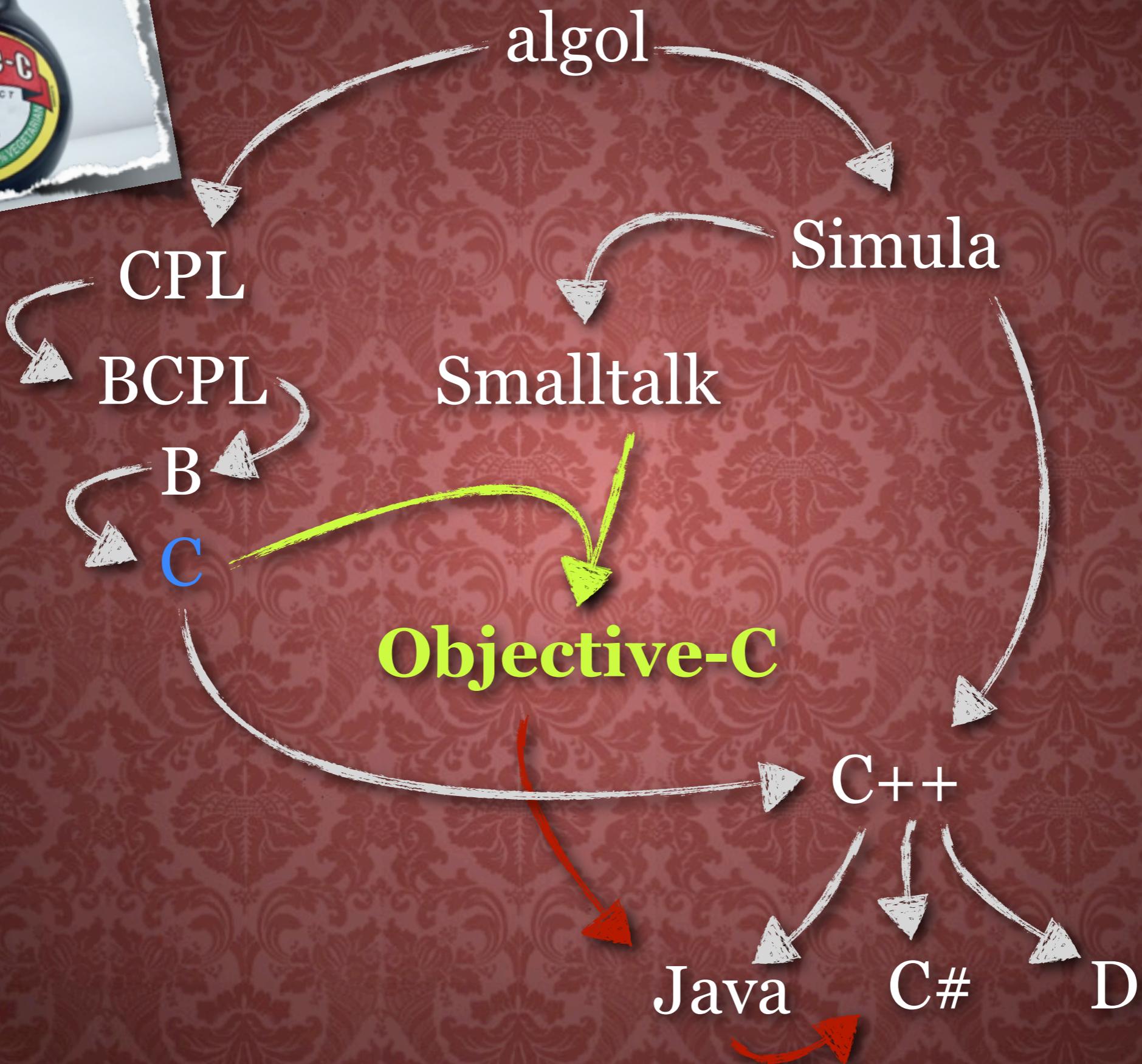
Objective-C





a history of objective-c in 14 lines and 12 words

(give or take)



objective C is...

*dynamic
based on message passing
follows C compilation model
full of [] and @s*

```
- (id) thisIsAMethod:(int)param1  
signature:(float)param2  
{  
    for (int n=0; n<10; ++n)  
    {  
        NSLog(@"N=%d", n);  
        [object callMethod:1 thing:2];  
    }  
}
```

not too
scary!



.m file

@implementation MyClass

```
- (id) thisIsAMethod:(int)param1  
                      signature:(float)param2  
{  
    for (int n = 0; n < 10; ++n)  
    {  
        NSLog(@"%@", n);  
        [object callMethod:1 thing:2];  
    }  
}
```

@end



@implementation MyClass

```
- (id) thisIsAMethod:(int)param1  
@class signature:(float)param2  
@interface  
for @implementation < 10; ++n)  
{ @end  
@NSLog(@"N=%d", n);  
[object callMethod:1 thing:2];  
@private  
}  
@protected
```

```
@end
```



@implementation MyClass

```
- (id) thisIsAMethod:(int)param1  
                      signature:(float)param2  
{  
    for (int n = 0; n < 10; ++n)  
    {  
        NSLog(@"%@", n);  
        [object callMethod:1 thing:2];  
    }  
}  
  
@end
```



```
@implementation MyClass
```

```
- (id) thisIsAMethod:(int)param1  
                      signature:(float)param2  
{  
    for (int n = 0; n < 10; ++n)  
    {  
        NSLog(@"%@", n);  
        [object callMethod:1 thing:2];  
    }  
}
```

```
@end
```



```
@implementation MyClass
```

```
- (id) thisIsAMethod:(int)param1  
                      signature:(float)param2  
{  
    for (int n = 0; n < 10; ++n)  
    {  
        NSLog(@"%@", n);  
        [object callMethod:1 thing:2];  
    }  
}
```

```
@end
```



```
@implementation MyClass
```

```
- (id) thisIsAMethod:(int)param1  
    signature:(float)param2  
{  
    for (int n = 0; n < 10; ) {  
        NSLog(@"%@", n);  
        [object callMethod:1 thing:2];  
    }  
}
```

```
@end
```



*Can still write (and call)
C free functions*

```
@implementation MyClass
```

```
- (id) thisIsAMethod:(int)param1  
                      signature:(float)param2  
{  
    for (int n = 0; n < 10; ++n)  
    {  
        NSLog(@"%@", n);  
        [object callMeWithThing:2];  
    }  
}
```

```
@end
```



```
@implementation MyClass
```

```
- (id) thisIsAMethod:(int)param1  
                      signature:(float)param2  
{  
    for (int n = 0; n < 10; ++n)  
    {  
        NSLog(@"%@", n);  
        [object callMethod:1 thing:2];  
    }  
}
```

```
@end
```



```
@implementation MyClass
```

```
- (id) thisIsAMethod:(int)param1  
    signature:(float)param2  
{  
    for (int n = 0; n < 10; ++n)  
    {  
        NSLog(@"%@", n);  
        [object callMethod:1 thing:2];  
    }  
}  
@end
```

Can call nil



Sending messages



```
[object method];  
int i = [object method];  
[[object method1] method2];  
  
[object method:10];  
[object1 method:[object 2:method]];
```

Method names



object

```
[circle setCenterAtX:110 y:675];
```

method name

parameters

selector name is: **setCenterAtX:y:**

method declared as:

- (void) **setCentreAtX:(float)x y:(float)y**

```
SEL s = @selector(setCenterAtX:y:);  
[circle performSelector:s];
```



parameter types:

*int, float, unsigned, etc
syntax: (return type)
id*

```
@class ForwardDeclaration;  
  
@interface MyClass : NSObject  
{  
    ForwardDeclaration *member;  
    bool otherMember;  
}  
  
- (id) init;  
- (id) initWithItems:(NSArray*)array  
    enabled:(bool)enabled;  
  
+ (id) myClassWithItems:(NSArray*)array;  
  
@end
```



```
@class ForwardDeclaration;  
  
@interface MyClass : NSObject  
{  
    ForwardDeclaration *member;  
    bool otherMember;  
}  
  
- (id) init;  
- (id) initWithItems:(NSArray*)array  
    enabled:(bool)enabled;  
  
+ (id) myClassWithItems:(NSArray*)array;  
  
@end
```



```
@class ForwardDeclaration;  
  
@interface MyClass : NSObject  
{  
    ForwardDeclaration *member;  
    bool otherMember;  
}  
  
- (id) init;  
- (id) initWithItems:(NSArray*)array  
    enabled:(bool)enabled;  
  
+ (id) myClassWithItems:(NSArray*)array;  
  
@end
```



```
@class ForwardDeclaration;  
  
@interface MyClass : NSObject  
{  
    ForwardDeclaration *member;  
    bool otherMember;  
}  
  
- (id) init;  
- (id) initWithItems:(NSArray*)array  
    enabled:(bool)enabled;  
  
+ (id) myClassWithItems:(NSArray*)array;  
  
@end
```



```
@class ForwardDeclaration;  
  
@interface MyClass : NSObject  
{  
    ForwardDeclaration *member;  
    bool otherMember;  
}  
  
- (id) init;  
- (id) initWithItems:(NSArray*)array  
    enabled:(bool)enabled;  
  
+ (id) myClassWithItems:(NSArray*)array;  
  
@end
```



```
@class ForwardDeclaration;  
  
@interface MyClass : NSObject  
{  
    ForwardDeclaration *member;  
    bool otherMember;  
}  
  
- (id) init; (redundant)  
- (id) initWithItems: (NSArray*)array  
    enabled: (bool)enabled;  
  
+ (id) myClassWithItems: (NSArray*)array;  
  
@end
```



```
@class ForwardDeclaration;  
  
@interface MyClass : NSObject  
{  
    ForwardDeclaration *member;  
    bool otherMember;  
}  
  
- (id) init;  
- (id) initWithItems:(NSArray*)array  
    enabled:(bool)enabled;  
  
+ (id) myClassWithItems:(NSArray*)array;  
  
@end
```





Objective C++

In .mm files

Classes can contain C++ objects

May use C++ constructs in functions

*You can use RAI
C++ libraries*

Properties

```
@interface MyClass  
{  
    int harold;  
}  
@property (nonatomic) int harold;  
@end
```

In MyClass.m...

```
@interface MyClass  
@synthesize harold;  
@end
```



This will become important when we look at creating UIs



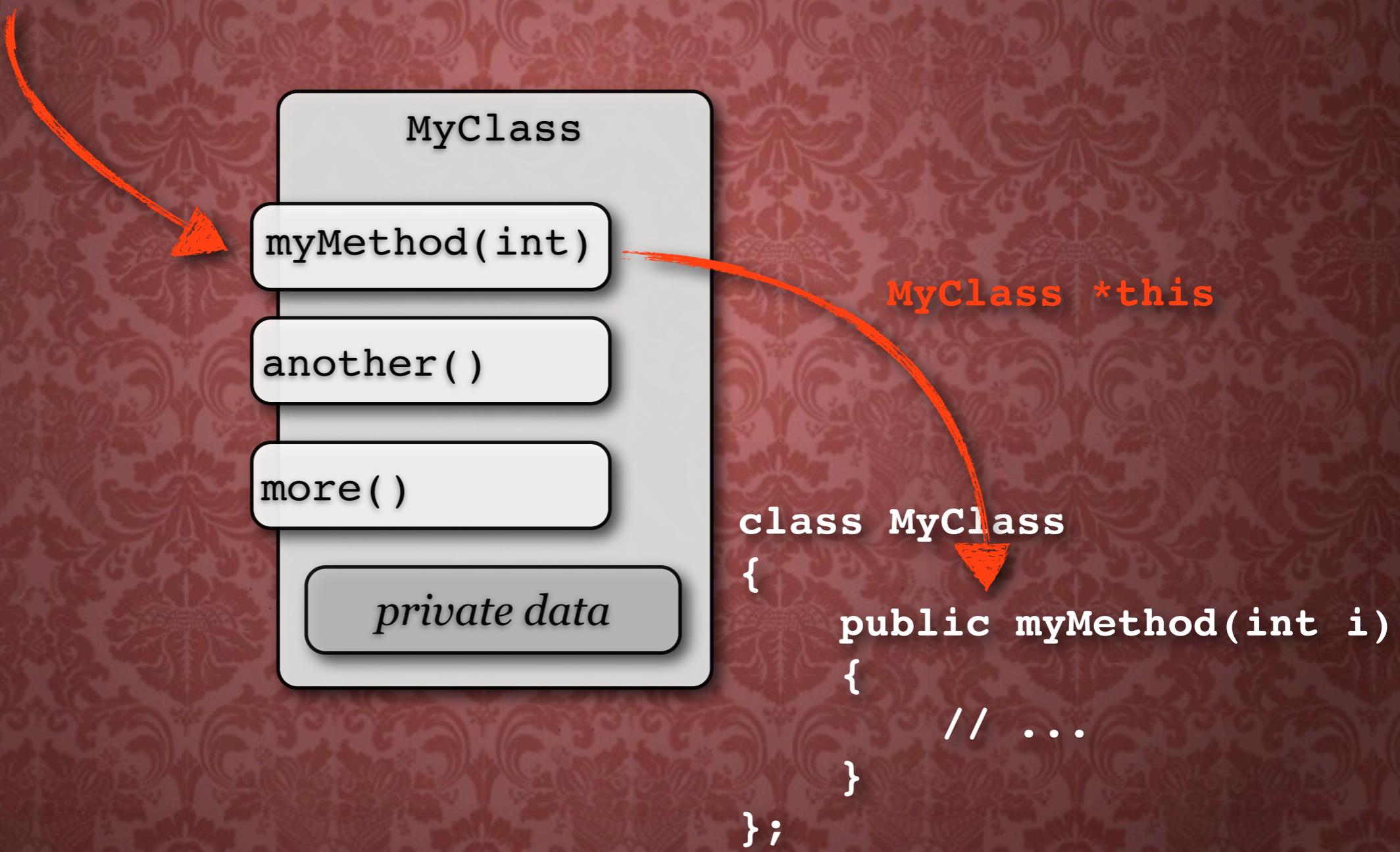
The tao of objects

*You don't call methods
You pass messages to objects*

[theObject sendMessage:parameter];

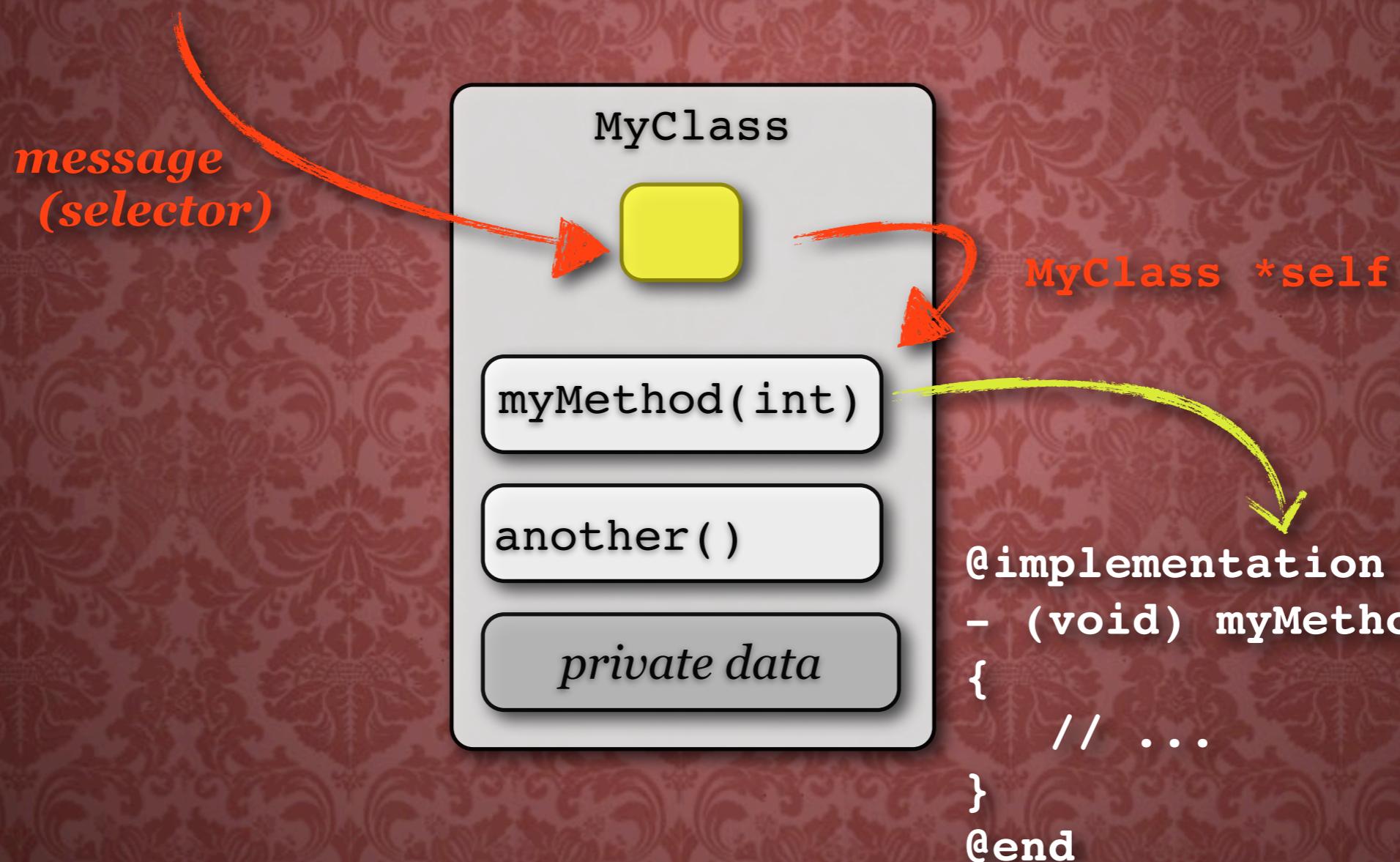
Java, C++, C#

`object.myMethod(1066);`



Objective-C

```
[object myMethod:1066];
```



Categories



```
@interface NSObject (Foo)  
- (void) extraFooMethod;  
@end
```

```
NSArray *array = [NSArray array];  
[array extraFooMethod];
```



Object lifetime

All objects are reference counted
(the retain count)

```
NSArray *array1 = [[NSArray alloc] init];
```

```
NSArray *array2 = [NSArray array];
```

```
[array1 release];
```

```
[array2 release]; // wrong
```

```
Circle *circle = [[Circle alloc] initAtCenterX:100  
                                y:100  
                           radius:200];
```

Action	Retain count
<code>obj=[[ClassType alloc] init];</code>	1
<code>[obj retain];</code>	2
<code>[obj release];</code>	1
<code>[obj release];</code>	0

`[object dealloc];`



The usual suspects

NSObject

NSString

NSArray

NSNumber

NSThread

NSDictionary **NSMutableDictionary**

NSApplication

NSData

NSMutableData

NSURL

NSSet

NSMutableSet

NSValue



Miscellany

*Statically typed my arse
#import
Constructor kludgery*



Idioms

*Delegates
Key-value coding
Protocol (interfaces)
formal and informal*



The libraries

Objective C



C

Cocoa Touch

UIKit, etc

Media

*Core Audio, OpenGL ES,
Quartz, Core Animation*

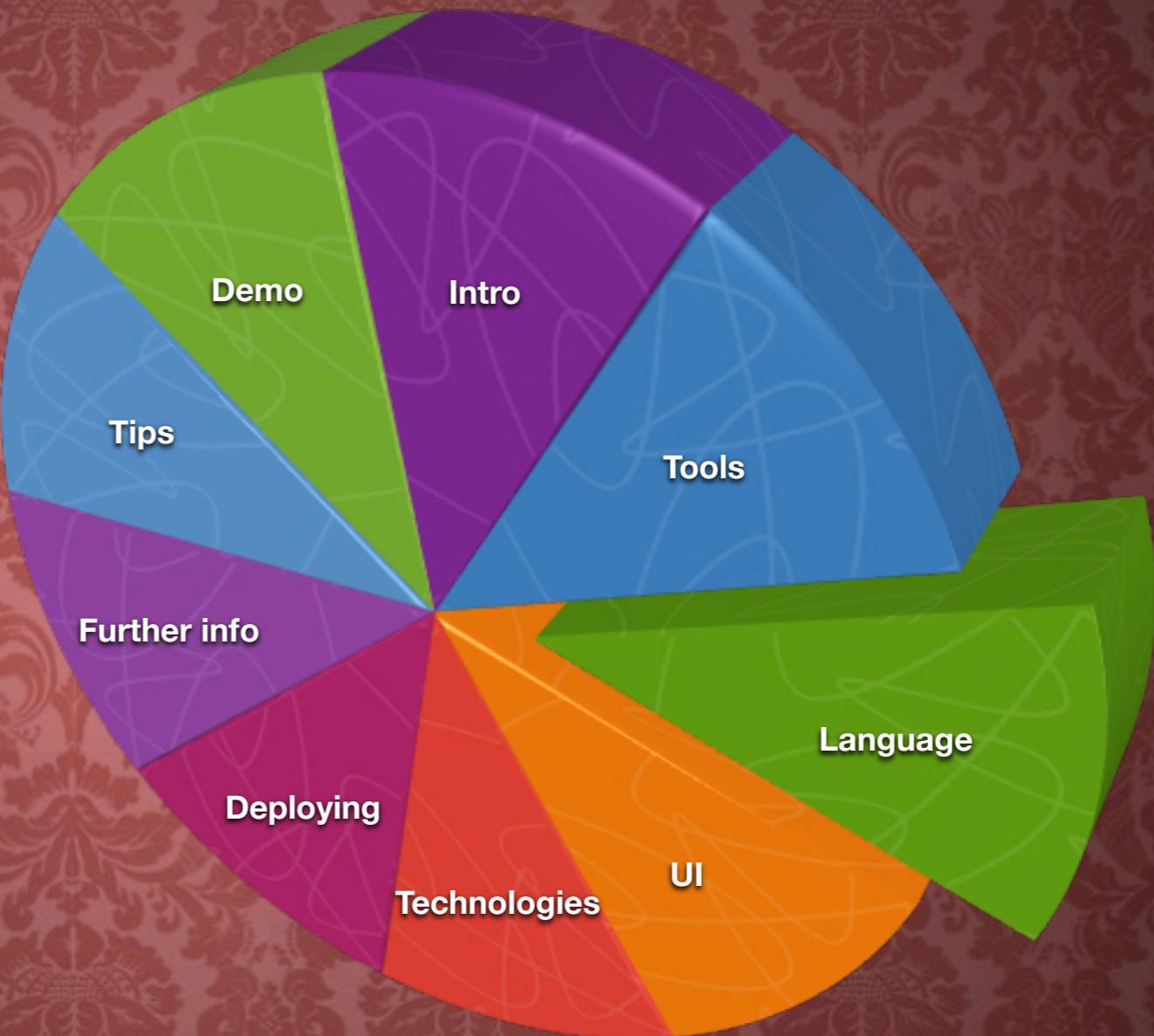
Core Services

*Posix, SQLite, Core
Foundation, BSD*

Core OS

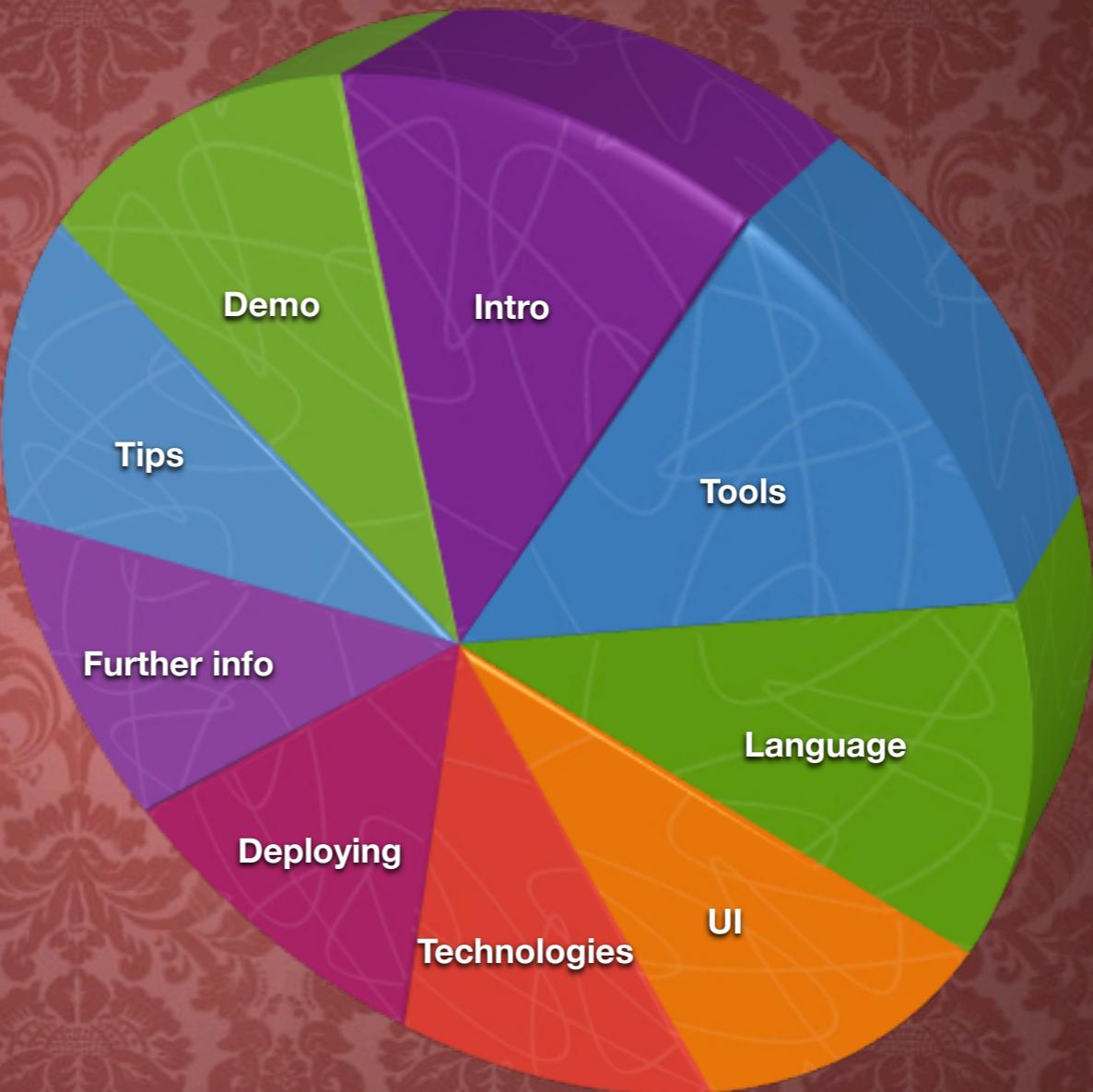
in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



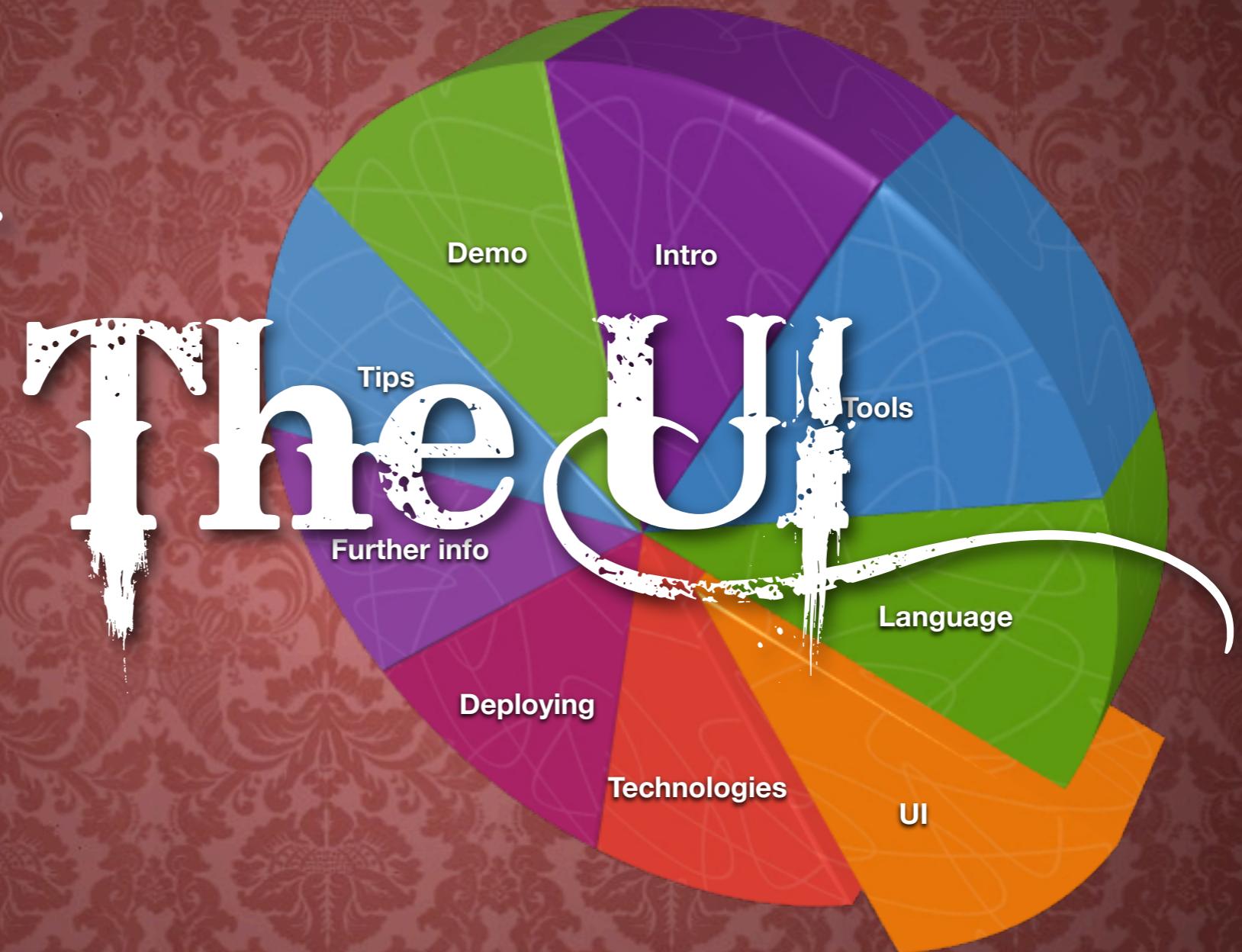
in 90 minutes...

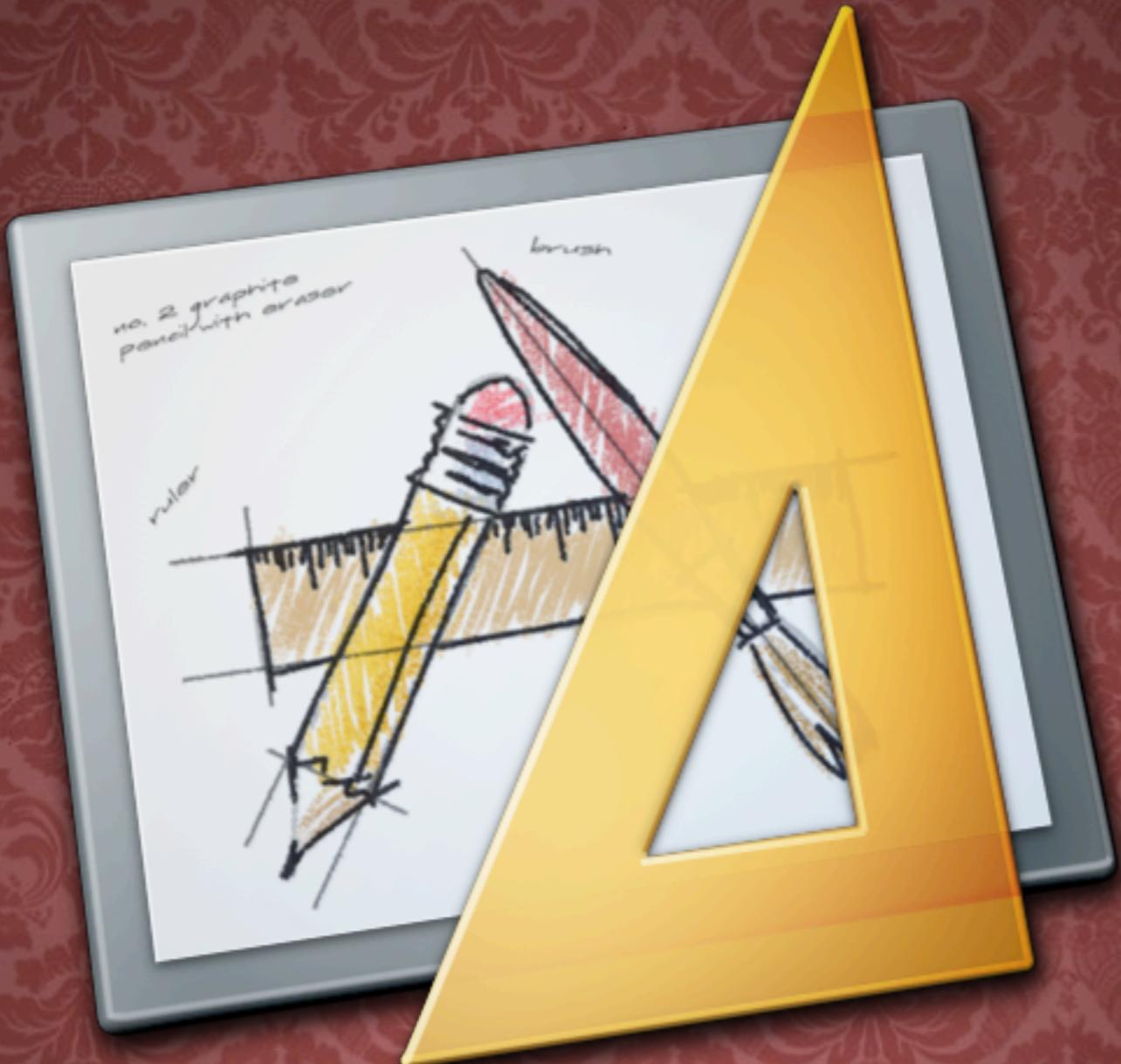
- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*





Interface Builder

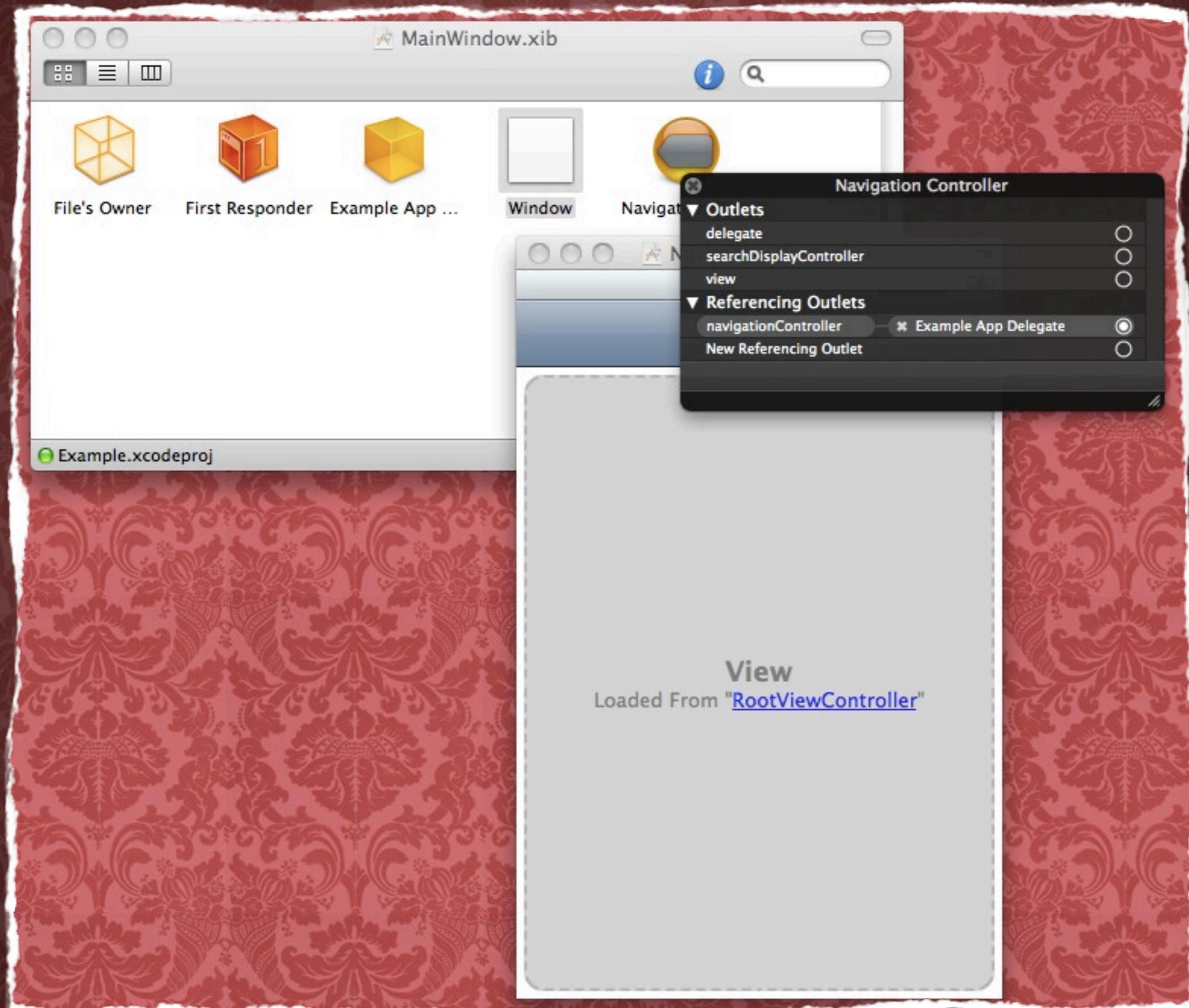


***Held in .xib files
(Compiled NIBs)***

***Loaded into application
(semi-automatically)***

***IB is tightly integrated with
Xcode***

UIKit framework

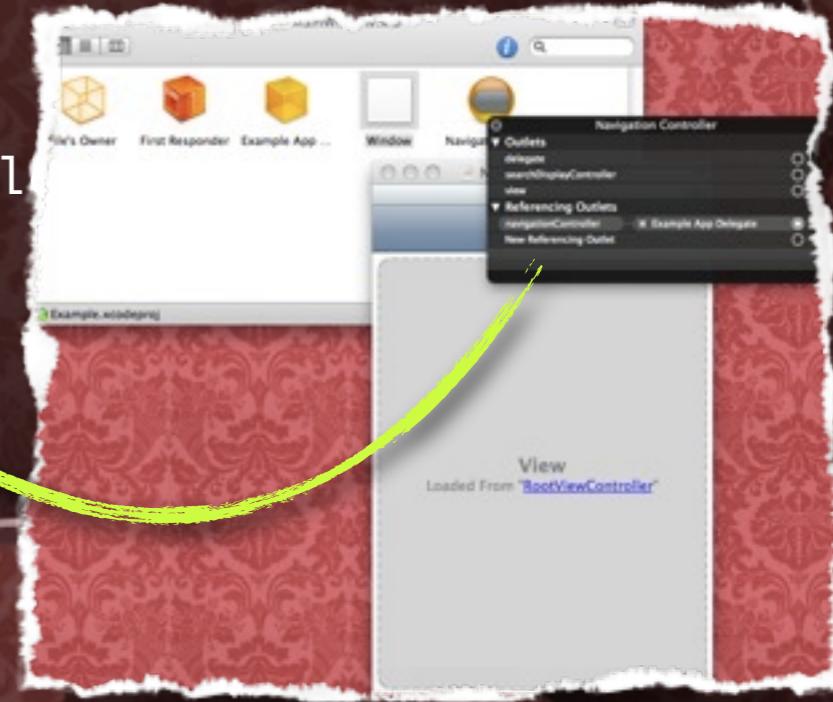


Create UI elements

*Link to an object reference in
the XIB through properties
marked as IBOutlet*

*Link actions to methods
defined IBAction*

```
//  
// ExampleAppDelegate.h  
// Example  
//  
// Created by Pete Goodliffe on 03/04/2010.  
// Copyright __MyCompanyName__ 2010. All rights reserved.  
  
#import <UIKit/UIKit.h>  
  
@interface ExampleAppDelegate : NSObject <UIApplicationDelegate> {  
  
    UIWindow *window;  
    UINavigationController *navigationController;  
}  
  
@property (nonatomic, retain) IBOutlet UIWindow *window;  
@property (nonatomic, retain) IBOutlet UINavigationController *navigationController;  
  
@end
```



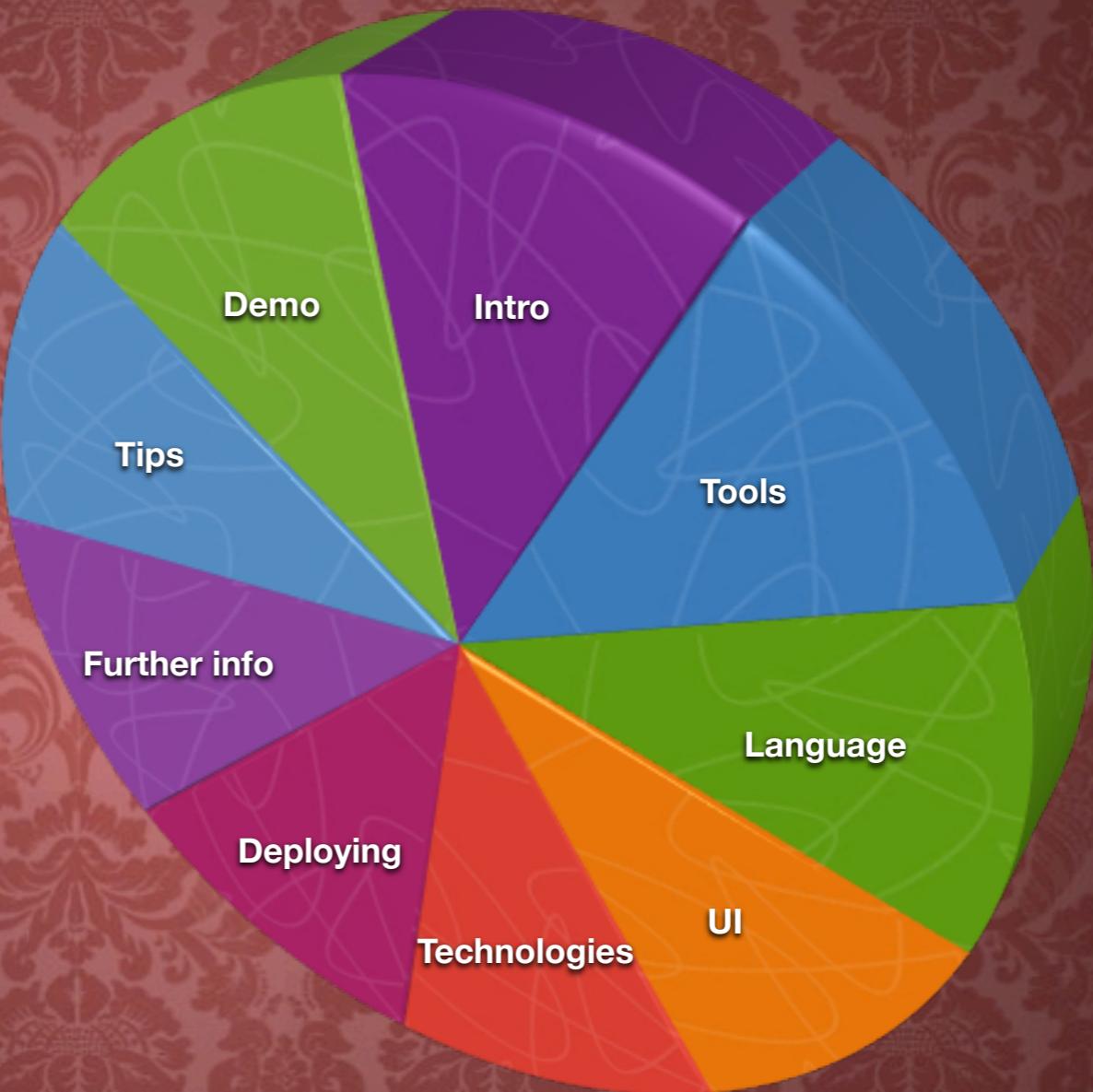
in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

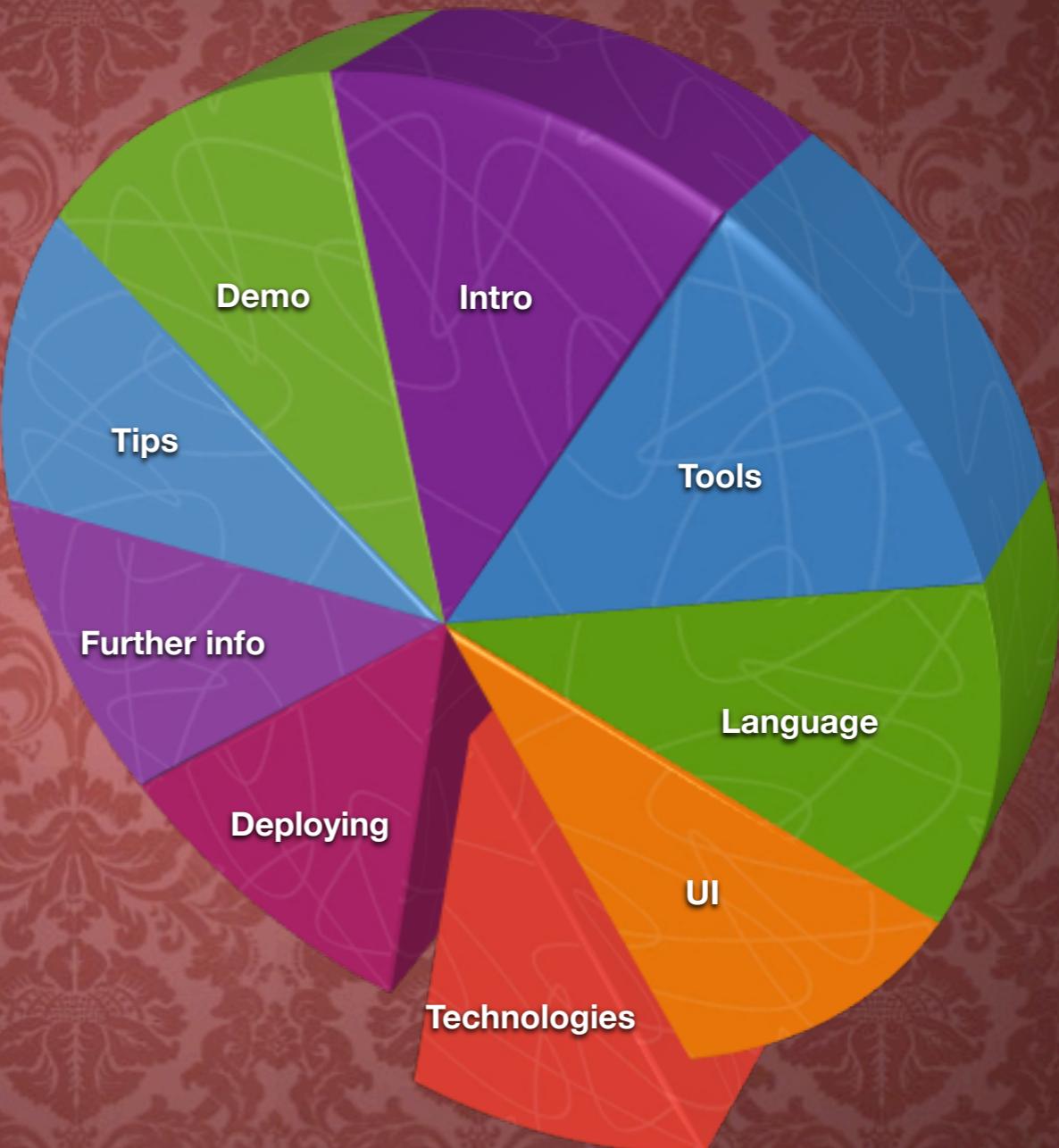
- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deployment*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



UIKit
Core Graphics
Core Audio
Media Core / AVFoundation
MapKit / Core Location
CoreData (or sqlite)
OpenGL ES
External Accessory

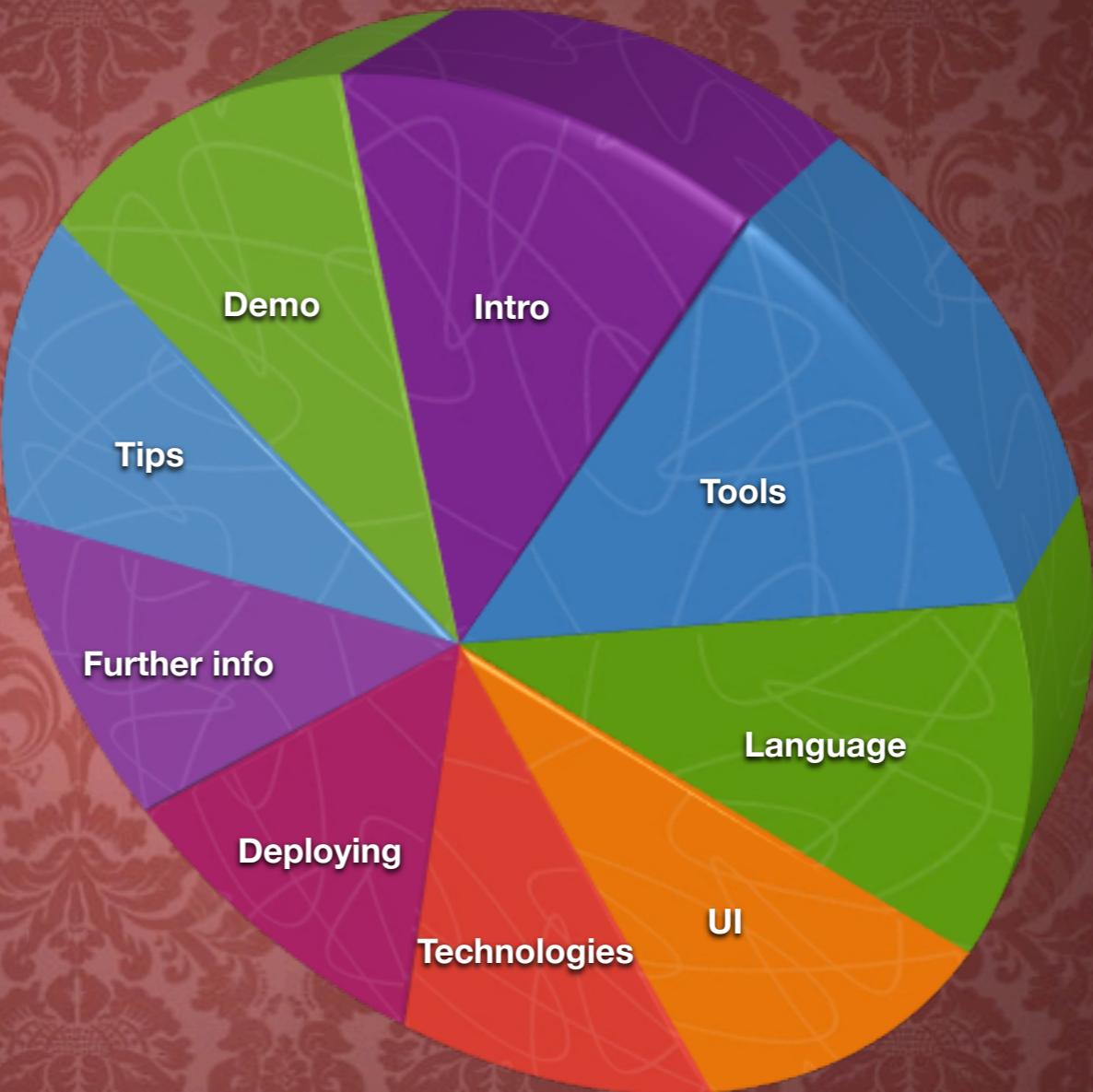
in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ ***Key technologies***
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

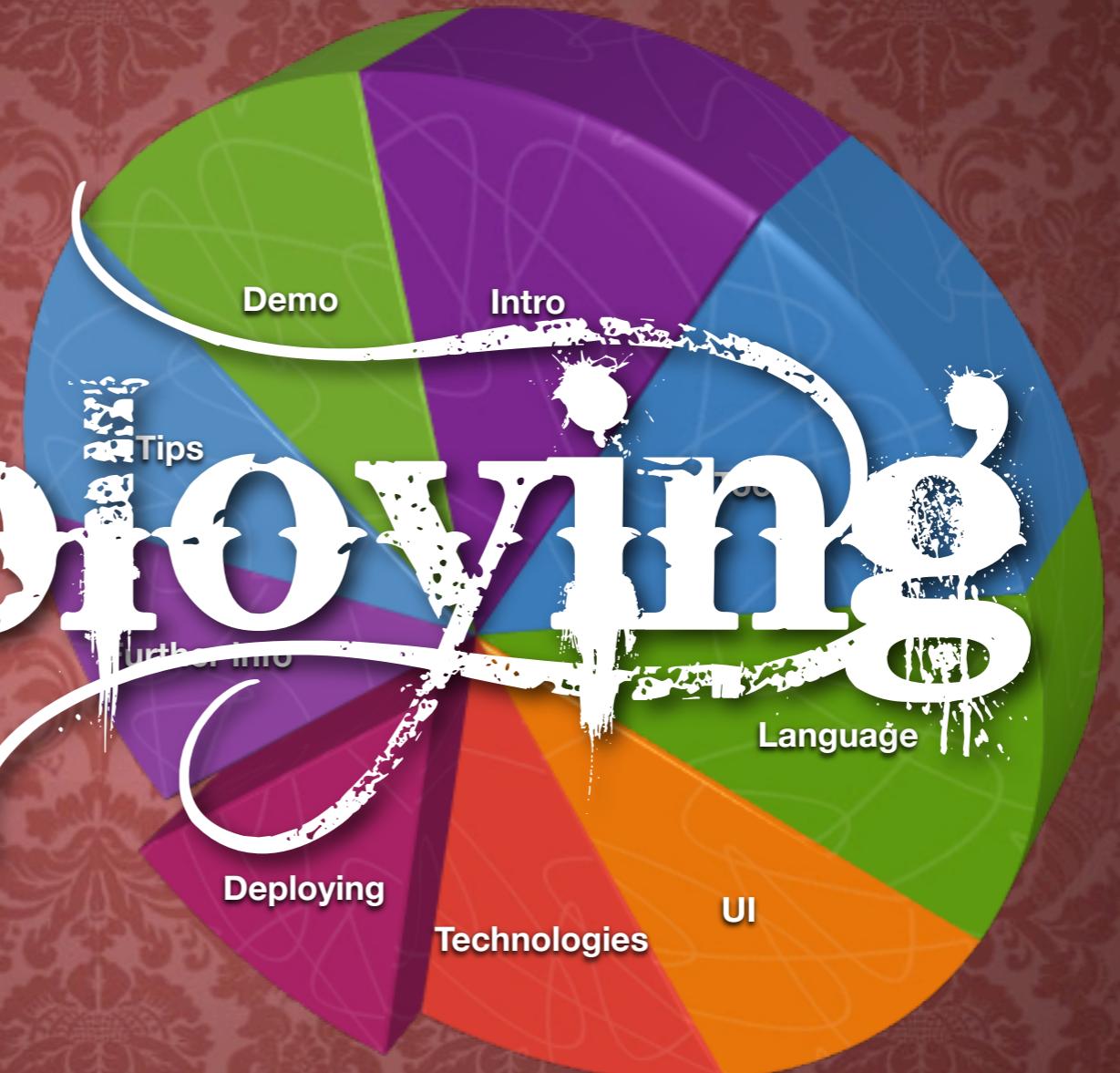
- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



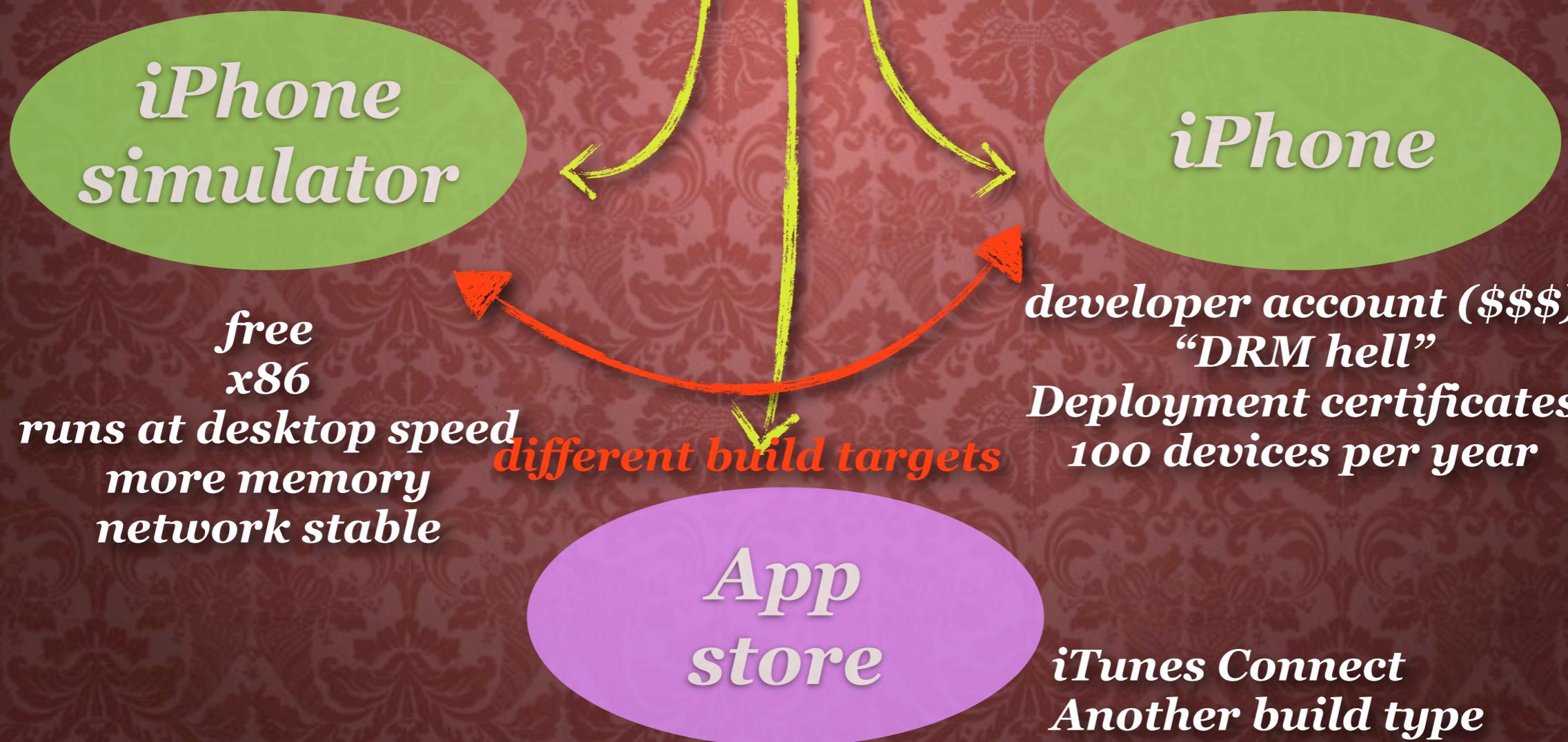
in 90 minutes...

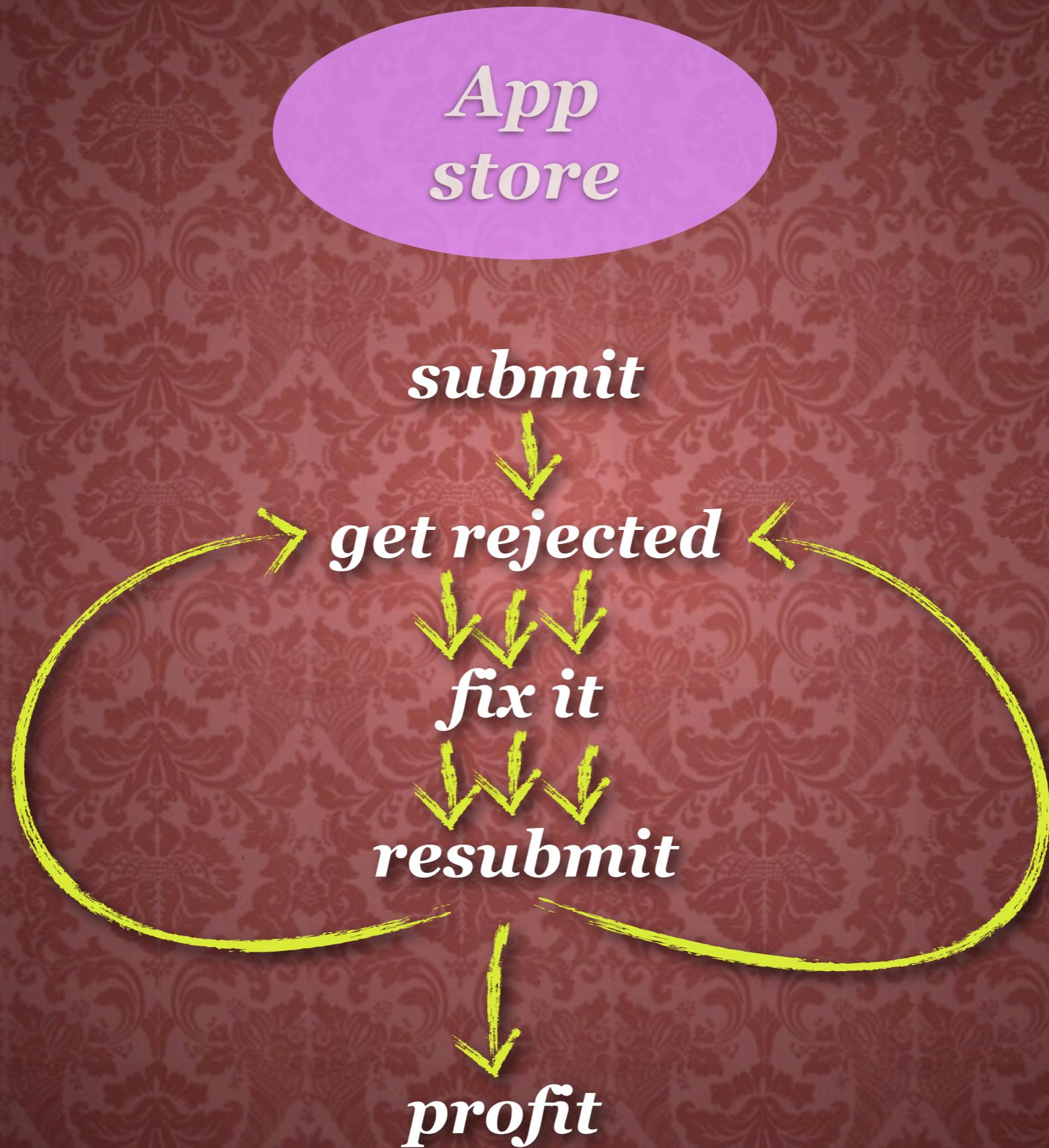
- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*

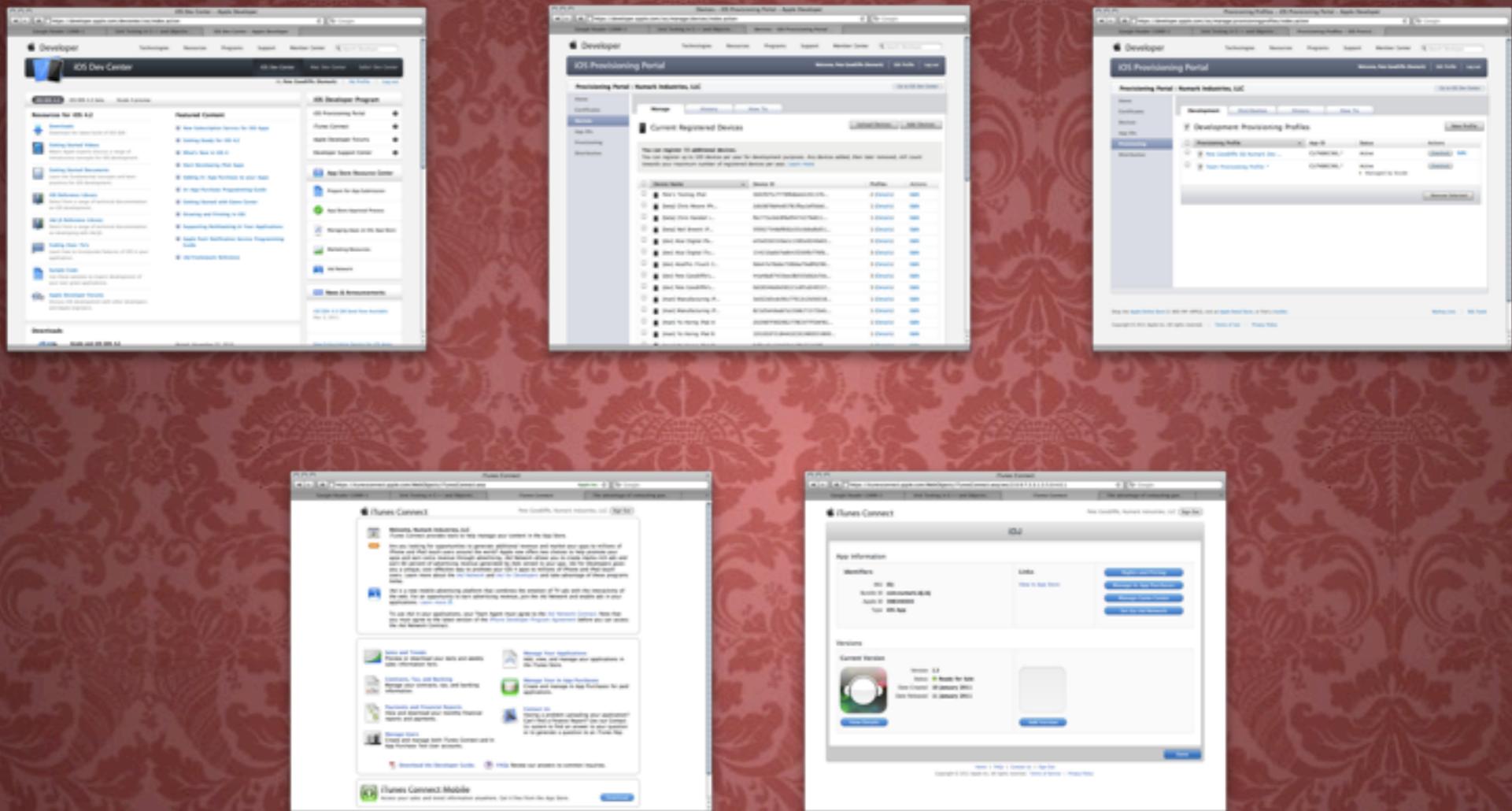
Deploying



Deploying







The screenshot shows the iOS Dev Center - Apple Developer website. The top navigation bar includes links for Google Reader (1000+), Unit Testing in C++ and Objective-C, and the current page, iOS Dev Center - Apple Developer. The main menu features sections for Technologies, Resources, Programs, Support, Member Center, and a search bar. A sidebar on the right provides links to the iOS Developer Program, App Store Resource Center, and News & Announcements. The central content area displays 'Resources for iOS 4.2' and 'Featured Content' sections, along with a 'Downloads' section at the bottom.

iOS Dev Center – Apple Developer

Google Reader (1000+) Unit Testing in C++ and Objective-C... iOS Dev Center – Apple Developer

Developer Technologies Resources Programs Support Member Center Search Developer

iOS Dev Center iOS Dev Center Mac Dev Center Safari Dev Center

Hi, Pete Goodliffe (Numark) | My Profile | Log out

iOS SDK 4.2 iOS SDK 4.3 beta Xcode 4 preview

Resources for iOS 4.2

- Downloads**
Download the latest build of iOS SDK
- Getting Started Videos**
Watch Apple experts discuss a range of introductory concepts for iOS development.
- Getting Started Documents**
Learn the fundamental concepts and best-practices for iOS development.
- iOS Reference Library**
Select from a range of technical documentation on iOS development.
- iAd JS Reference Library**
Select from a range of technical documentation on developing with iAd JS.
- Coding How-Tos**
Learn how to incorporate features of iOS in your application.
- Sample Code**
Use these samples to inspire development of your own great applications.
- Apple Developer Forums**
Discuss iOS development with other developers and Apple engineers.

Featured Content

- New Subscription Service for iOS Apps
- Getting Ready for iOS 4.2
- What's New in iOS 4
- Start Developing iPad Apps
- Adding In-App Purchase to your Apps
- In-App Purchase Programming Guide
- Getting Started with Game Center
- Drawing and Printing in iOS
- Supporting Multitasking In Your Applications
- Apple Push Notification Service Programming Guide
- iAd Framework Reference

iOS Developer Program

- iOS Provisioning Portal
- iTunes Connect
- Apple Developer Forums
- Developer Support Center

App Store Resource Center

- Prepare for App Submission
- App Store Approval Process
- Managing Apps on the App Store
- Marketing Resources
- iAd Network

RSS News & Announcements

- iOS SDK 4.3 GM Seed Now Available
Mar 3, 2011
- New Subscription Service for iOS Apps

Devices – iOS Provisioning Portal – Apple Developer
https://developer.apple.com/ios/manage/devices/index.action

Google Reader (1000+) Unit Testing in C++ and Objective-C Devices – iOS Provisioning Portal ...

Developer Technologies Resources Programs Support Member Center Search Developer

Welcome, Pete Goodliffe (Numark) Edit Profile Log out

iOS Provisioning Portal Go to iOS Dev Center

Provisioning Portal : Numark Industries, LLC

Home Certificates Devices App IDs Provisioning Distribution

Manage History How To

Current Registered Devices

You can register 72 additional devices.

You can register up to 100 devices per year for development purposes. Any devices added, then later removed, still count towards your maximum number of registered devices per year. [Learn more](#)

Device Name Device ID Profiles Actions

Device Name	Device ID	Profiles	Actions
Pete's Testing iPod	3bfcf975c7778f68deb2201376...	2 (Details)	Edit
[beta] Chris Moore iPh...	2db3878d4e83781ffa1ef56dd...	1 (Details)	Edit
[beta] Chris Randall i...	fbc773a3dc8f8af9474279e811...	1 (Details)	Edit
[beta] Neil Brewitt IP...	5f0927548df692d35c0dbd6d51...	1 (Details)	Edit
[dev] Akai Digital iPa...	e45e93b554be1c1385e9240e65...	3 (Details)	Edit
[dev] Akai Digital iTo...	154210a6b7ed641f200fb776fd...	3 (Details)	Edit
[dev] AkaiPro iTouch 3...	9de47e78abe738bba79a8f0298...	3 (Details)	Edit
[dev] Pete Goodliffe's...	44a48a87433eec8b503db2e7de...	3 (Details)	Edit
[dev] Pete Goodliffe's...	0d2834bb9d38121c8f1d24f237...	3 (Details)	Edit
[man] Manufacturing IP...	3e022d3cdc90c77612c2b56018...	1 (Details)	Edit
[man] Manufacturing IP...	821d5e4dea67a15b8c71572bdc...	1 (Details)	Edit
[man] Ya Horng iPad A	202987F9ED9E277BC977FDCAF81...	1 (Details)	Edit
[man] Ya Horng iPad B	1D10D2F3184A32C81980D55800...	1 (Details)	Edit

Provisioning Profiles – iOS Provisioning Portal – Apple Developer

https://developer.apple.com/ios/manage/provisioningprofiles/index.action

Google Reader (1000+) Unit Testing in C++ and Objective-C Provisioning Profiles – iOS Provi...

Developer Technologies Resources Programs Support Member Center Search Developer

iOS Provisioning Portal Welcome, Pete Goodliffe (Numark) Edit Profile Log out

Provisioning Portal : Numark Industries, LLC Go to iOS Dev Center

Home Certificates Devices App IDs Provisioning Distribution History How To

Development Provisioning Profiles New Profile

Provisioning Profile	App ID	Status	Actions
Pete Goodliffe IDJ Numark Dev ...	CU7NB6C96L.*	Active	Download Edit
Team Provisioning Profile: *	CU7NB6C96L.*	Active • Managed by Xcode	Download

Remove Selected

Shop the [Apple Online Store](#) (1-800-MY-APPLE), visit an [Apple Retail Store](#), or find a [reseller](#).

Mailing Lists RSS Feeds

Copyright © 2011 Apple Inc. All rights reserved. | [Terms of Use](#) | [Privacy Policy](#)

Provisioning Profile	App ID	Status	Actions
Pete Goodliffe IDJ Numark Dev ...	CU7NB6C96L.*	Active	Download Edit
Team Provisioning Profile: *	CU7NB6C96L.*	Active • Managed by Xcode	Download

iTunes Connect

https://itunesconnect.apple.com/WebObjects/iTunesConnect.woa

Apple Inc. Google

Google Reader (1000+) Unit Testing in C++ and Objective... iTunes Connect The advantage of computing go...

Pete Goodliffe, Numark Industries, LLC Sign Out

iTunes Connect

Welcome, Numark Industries, LLC

iTunes Connect provides tools to help manage your content in the App Store.

NEW

Are you looking for opportunities to generate additional revenue and market your apps to millions of iPhone and iPod touch users around the world? Apple now offers two choices to help promote your apps and earn extra revenue through advertising. iAd Network allows you to create media-rich ads and earn 60 percent of advertising revenue generated by iAds served to your app. iAd for Developers gives you a unique, cost-effective way to promote your iOS 4 apps to millions of iPhone and iPod touch users. Learn more about the [iAd Network](#) and [iAd for Developers](#) and take advantage of these programs today.

iAd is a new mobile advertising platform that combines the emotion of TV ads with the interactivity of the web. For an opportunity to earn advertising revenue, join the iAd Network and enable ads in your applications. [Learn more](#).

To use iAd in your applications, your Team Agent must agree to the [iAd Network Contract](#). Note that you must agree to the latest version of the [iPhone Developer Program Agreement](#) before you can access the iAd Network Contract.

Sales and Trends
Preview or download your daily and weekly sales information here.

Contracts, Tax, and Banking
Manage your contracts, tax, and banking information.

Payments and Financial Reports
View and download your monthly financial reports and payments.

Manage Users
Create and manage both iTunes Connect and In App Purchase Test User accounts.

Manage Your Applications
Add, view, and manage your applications in the iTunes Store.

Manage Your In App Purchases
Create and manage In App Purchases for paid applications.

Contact Us
Having a problem uploading your application? Can't find a Finance Report? Use our Contact Us system to find an answer to your question or to generate a question to an iTunes Rep.

Download the Developer Guide. FAQs Review our answers to common inquiries.

iTunes Connect Mobile
Access your sales and trend information anywhere. Get it free from the App Store.

iTunes Connect

https://itunesconnect.apple.com/WebObjects/iTunesConnect.woa/wo/2.0.9.7.3.3.1.5.5.0.4.0.1

Google Reader (1000+) Unit Testing in C++ and Objectiv... iTunes Connect The advantage of computing go... +

Pete Goodliffe, Numark Industries, LLC [Sign Out](#)

iDJ

App Information

Identifiers SKU iDJ Bundle ID com.numark.idj.idj Apple ID 398348303 Type iOS App	Links View in App Store	Rights and Pricing Manage In App Purchases Manage Game Center Set Up iAd Network
---	---	---

Versions

Current Version  View Details	Version 1.3 Status Ready for Sale Date Created 10 January 2011 Date Released 11 January 2011	Add Version
---	---	-----------------------------

[Done](#)

[Home](#) | [FAQs](#) | [Contact Us](#) | [Sign Out](#)
Copyright © 2011 Apple Inc. All rights reserved. [Terms of Service](#) | [Privacy Policy](#)

TestFlight | iOS beta testing on the fly

http://testflightapp.com/

Google Reader (1000+) Unit Testing in C++ and Objectiv... TestFlight | iOS beta testing on th...

Log In Support Blog Follow us

TestFlight
iOS beta testing on the fly

Free over-the-air beta distribution
Quick. Painless. Magical*.

Register Now

or try a demo below

1 Drag your IPA here

2 Email yourself and others

3 Real-time counter

001 Apps installed

Watch it!

By dragging a file you agree to TOS and PP

* Thanks Apple, for building the tools to make magic possible

"Neven and I have been using TestFlight as a verb today. TestFlight it when you can. TestFlighting it now. I love it."

Matt Comi - Big Bucket Software

"Oh. My. God. All I did was tap the link. I love you."

Loren Brichter - Creator of Tweetie

"TestFlight will change the way every legit iOS dev does testing."

Adam Huda - Posterous

The screenshot shows the TestFlight website homepage. At the top, there's a navigation bar with tabs for 'Google Reader (1000+)', 'Unit Testing in C++ and Objectiv...', and 'TestFlight | iOS beta testing on the fly'. Below the header, the TestFlight logo is displayed with the tagline 'iOS beta testing on the fly'. To the right are links for 'Log In', 'Support', 'Blog', and 'Follow us'. The main banner features the text 'Free over-the-air beta distribution' and 'Quick. Painless. Magical*.' Below this is a large blue button labeled 'Register Now'. Underneath the banner, there's a section with three numbered steps: 1. 'Drag your IPA here' with a placeholder for an IPA file, accompanied by the handwritten note 'Seriously, try it!'; 2. 'Email yourself and others' with an input field for email addresses and a 'submit' button; 3. 'Real-time counter' showing '001 Apps installed' with a handwritten note 'Watch it!'. At the bottom, there are three user quotes: 'Neven and I have been using TestFlight as a verb today. TestFlight it when you can. TestFlighting it now. I love it.' attributed to Matt Comi - Big Bucket Software; 'Oh. My. God. All I did was tap the link. I love you.' attributed to Loren Brichter - Creator of Tweetie; and 'TestFlight will change the way every legit iOS dev does testing.' attributed to Adam Huda - Posterous.

in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The basics*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



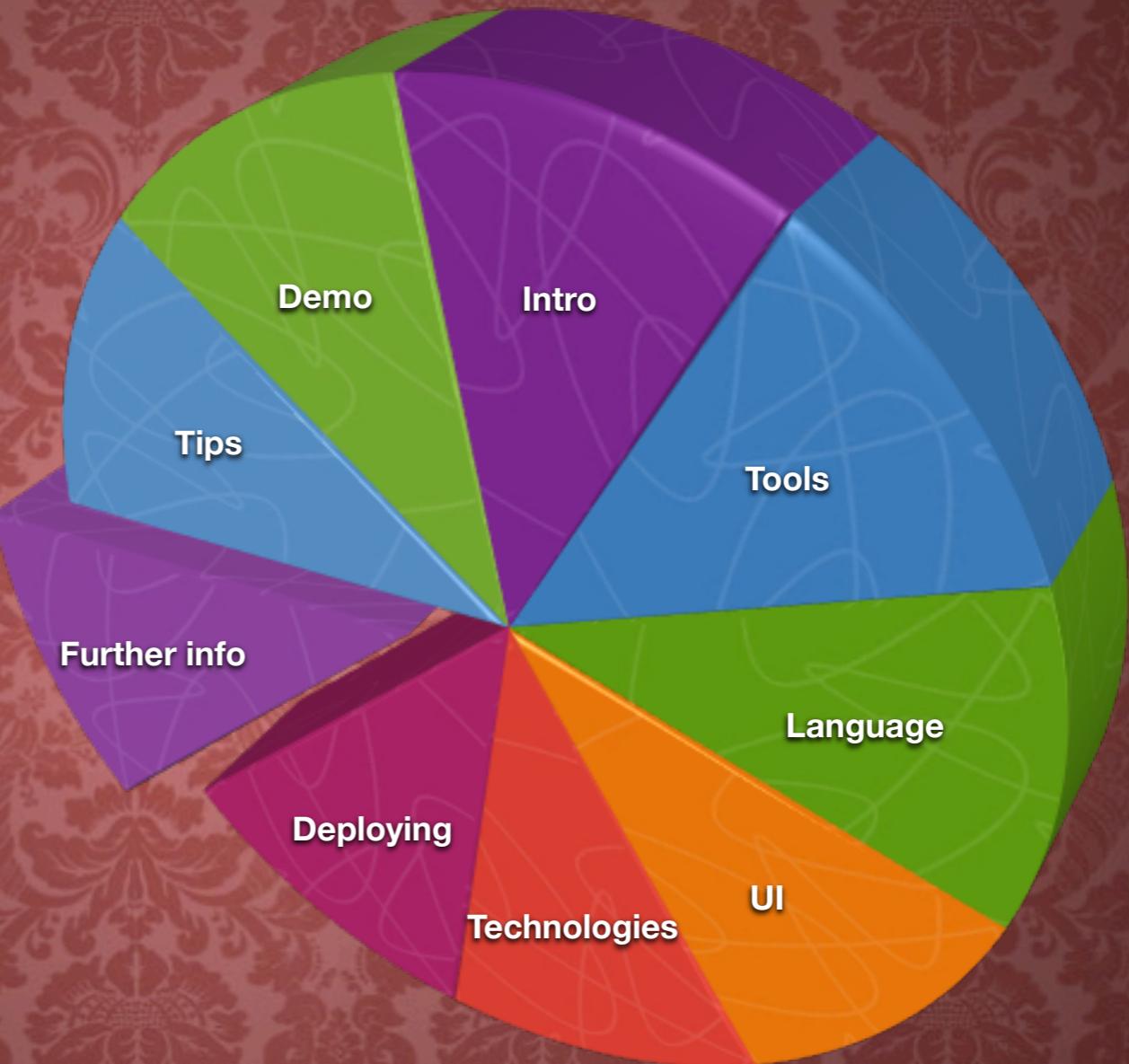
*Xcode documentation
developer.apple.com/iphone
Books
Websites*

Stack Overflow, CocoaDev, ADC

*Videos on iTunes U
<http://itunes.stanford.edu/>*

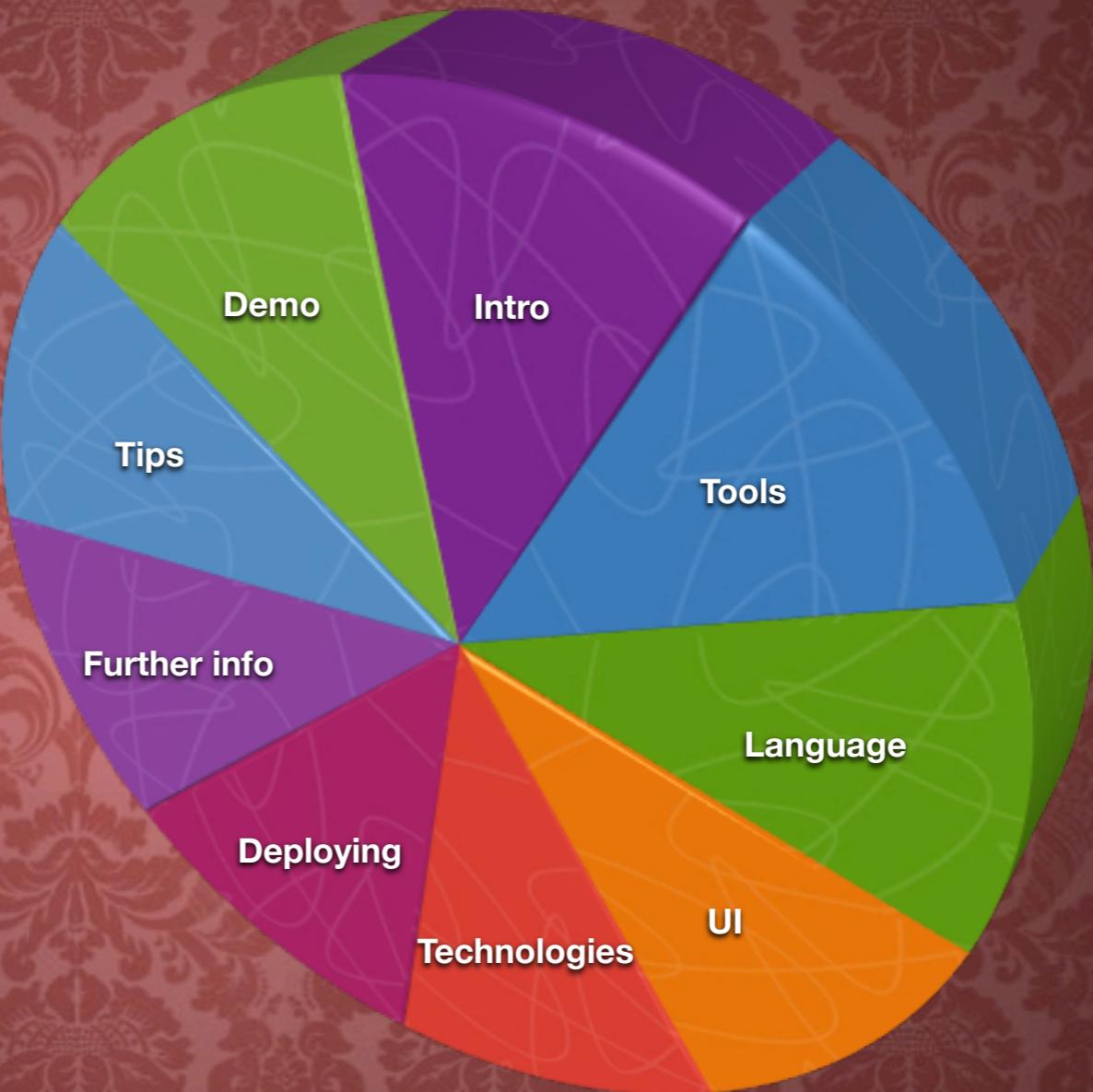
in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ **Tips and tricks**
- ▶ *The demo*



Thinking iPhone

Get the UI right

*HIG
resource conservation
(alpha blending, prerendering)
asynchrony*

***The network is not always
available***

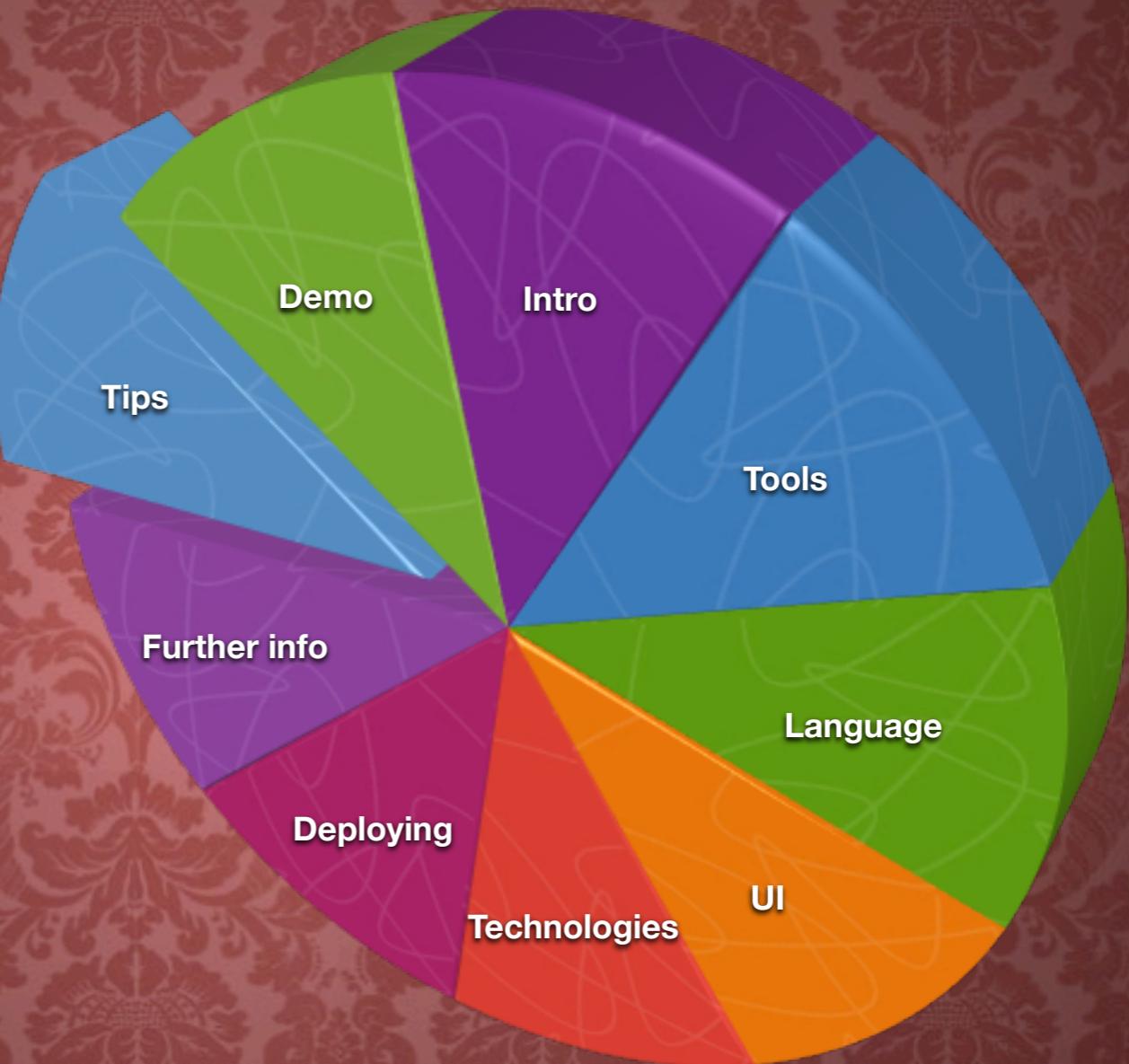
***Do not write to disk
(often)***

Conserve resources

*RAM
CPU
filesystem*

in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ ***Tips and tricks***
- ▶ *The demo*



in 90 minutes...

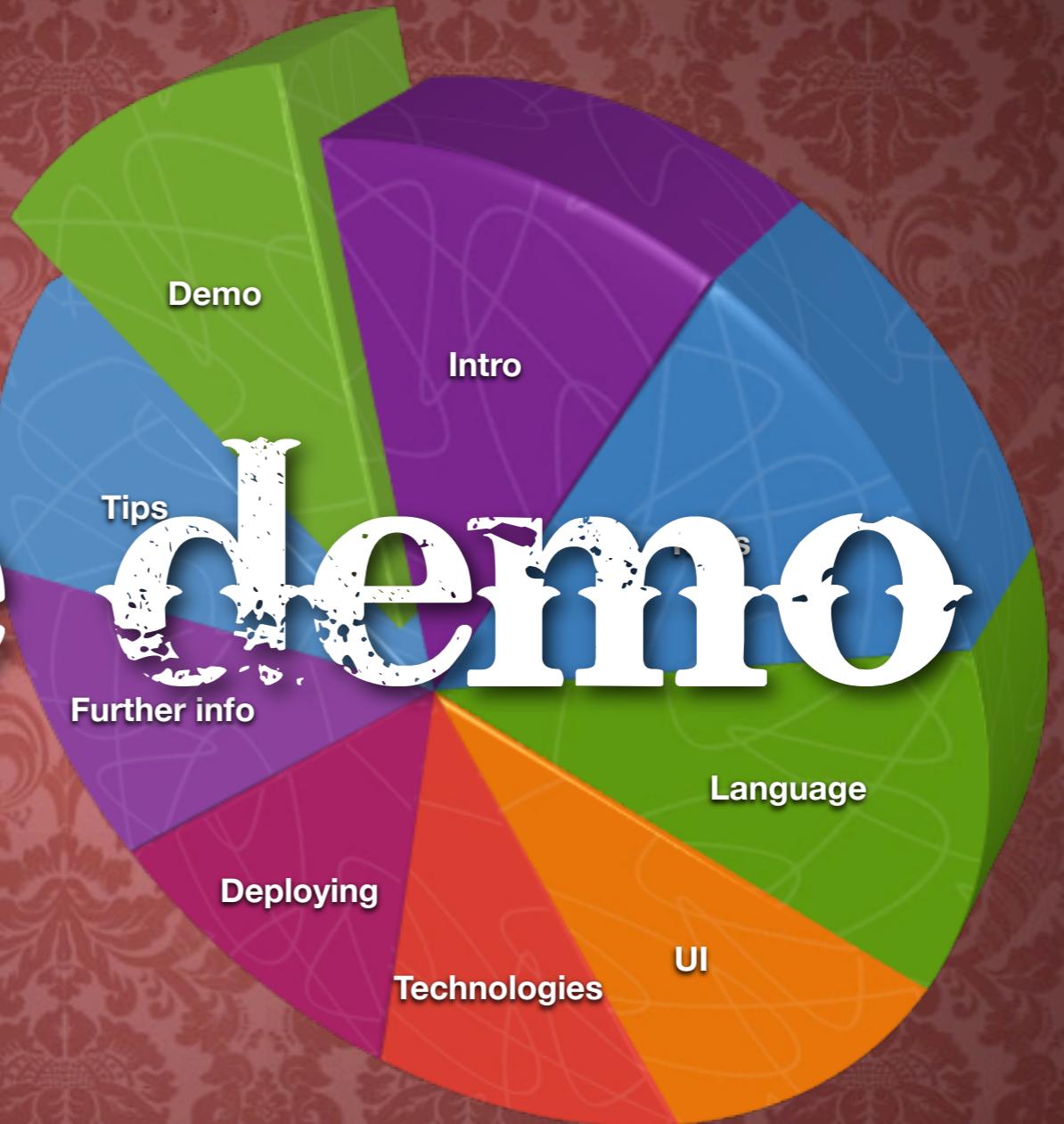
- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*



in 90 minutes...

- ▶ *Intro*
- ▶ *The tools*
- ▶ *The language*
- ▶ *The UI*
- ▶ *Key technologies*
- ▶ *Deploying*
- ▶ *Further info*
- ▶ *Tips and tricks*
- ▶ *The demo*

the demo



~~(*wish me luck*)~~

*wasn't that
marvellous?!*



KEEP
CALM
AND
CARRY
ON

Pete Goodliffe
pete@goodliffe.net