



CF

Download a PDF of this presentation and a ZIP file of code from <http://blog.joshuaadams.com>

Josh Adams

Adobe Systems Incorporated  
Sr. SE, ColdFusion Specialist

[joadams@adobe.com](mailto:joadams@adobe.com)

# iPhone Apps + Adobe® ColdFusion®

Josh Adams

Adobe Systems Incorporated  
Sr. SE, ColdFusion Specialist

[joadams@adobe.com](mailto:joadams@adobe.com)



# What This All About?

- § What you need to know to get **started** developing **native** apps for iPhone...
- § ...by writing JavaScript, HTML, and CSS
- § ...without directly writing Objective-C

Question:  
What's iPhone?

Response:  
You're kidding, right?

# What's iPhone?

- § From wikipedia: "The iPhone is an Internet-connected, multimedia GSM smartphone designed and marketed by Apple Inc."
- § It's more (much more) than a phone!
- § It runs applications (a.k.a. apps)
  - § Official Apple iPhone apps
    - § Including Safari browser
  - § Third-party apps
- § It can establish wireless data connections
  - § Carrier networks
  - § Wi-Fi networks

# Why Do We Care about iPhone?

- § We've already established the capabilities of the platform...
- § So what about the size of the user-base?
  - § 21.4 million units sold worldwide through March 2009
  - § 6.4 million active units in the US alone
- § It's cool

# Why We Care about iPhone: Opportunities for Developers

## § Expand existing revenue streams

- § Connect with existing customers via apps and web apps
- § Acquire and connect with new customers via apps and web apps

## § Create new revenue streams

- § Earn revenues from ads in apps and web apps
- § Sell apps through iTunes and the iPhone App Store

# What about iPod Touch?

- § Everything we're talking about today applies to iPod Touch as well
- § iPod Touch has limited capabilities as compared to iPhone
  - § No camera
  - § No GPS
  - § No phone
  - § Data connections only available via Wi-Fi



# Developing for Mobile Clients

- § Three kinds of apps
  - § Native
  - § Player or virtual machine
  - § Browser-based
- § What? No Players? Adobe—what gives? We want Flash Player!

# Flash Player on iPhone

§ It's a complicated situation J

§ Official(ish) statement:

*We would like to work with Apple to bring Flash to the iPhone, meanwhile we continue to work towards bringing the full Flash Player to Symbian, Windows Mobile, Android, and other smartphones enabling a more complete web experience.*

# Native Apps

## § Native

- § Apps that run natively on the mobile device's OS

- § Code is resident on the client device

# Browser-based apps

## § Browser-based

- § Apps that run in a browser on top of the mobile device's OS
  - § May also run in a player or virtual machine browser plug-in
- § Mobile browsers vary widely in the features they support
- § Code is not resident on the client device; it is delivered from the server

# Web Services (the) Enterprise Hub

## Clients



Flash Player

Flash Remoting

Data Services



AIR

Flash Remoting

Data Services



Web Browser  
HTML / Ajax

HTTP



Adobe Reader  
Documents / Forms

HTTP



Mobile Devices  
HTML / SMS

HTTP



Web Services  
SOAP / REST

HTTP

```
<cfquery name="myQuery" ...>
...
</cfquery>
<html>
  <body>
    <cfoutput>#myQuery.aField#
  </cfoutput>
  </body>
</html>
```

**myPage.cfm**

Web Server

GUI:  
JavaScript  
& HTML

Data

```
<cfcomponent>
  <cffunction name="myMethod"
    access="remote">
    <cfquery name="myQuery" ...>
    ...
  </cfquery>
  <cfreturn myQuery>
</cffunction>
</cfcomponent>
```

**myComp.cfc**

Java



COM  
CORBA  
Web Services



RDBMS  
XML



Email  
FTP  
LDAP



.NET  
Exchange  
Share Point



## Enterprise Data and Infrastructure

# iPhone + ColdFusion: What's Where?

## § iPhone = client device

- § iPhone is a client device that can make requests over HTTP/HTTPS

- § Client application is an iPhone app

## § ColdFusion = application server

- § ColdFusion can handle requests and send responses over HTTP/HTTPS

## § So...ColdFusion is capable of powering both custom-developed apps and web apps that run in Safari browser

- § ColdFusion doesn't care!

- § In fact, ColdFusion doesn't know: it just knows it's getting a request

# Can I Use ColdFusion To Create Native iPhone Apps?

§ Short answer: no

§ Longer answer: nooooo

§ Remember, ColdFusion is on the server responding to requests from the client application

§ If the client application is an app, ColdFusion's responses will contain only data (at least in the vast majority of cases)

§ If the client application is Safari browser, ColdFusion's response will contain UI in addition to or instead of data (at least much more commonly)

# Native Apps vs Browser-based Apps

## § Native Apps

1. Access to accelerometer, camera, GPS (on iPhones that contain it), phone functionality, and media
2. Full (well, at least as full as it gets) UI control
3. Application UI resides on client
  - A. Less bandwidth: only need to transmit data
  - B. Updates are more involved: users must proactively update their apps
4. Application must be approved by Apple
  - A. Have to wait on Apple to approve the app—and they may not
5. Application available via the app store
6. Must pay to join iPhone Developer Program in order to deploy apps

## § Browser-based Apps

1. No access to anything listed to the left
2. UI constrained to and by HTML, CSS, and JavaScript
3. Application UI must be loaded from server
  - A. More bandwidth: must transmit UI and data
  - B. Easy to update: just deploy a newer version of the UI to the server
4. Application uploaded directly to deployment web server
  - A. Can upload it on own schedule
5. Application only available by visiting site where app is deployed
6. No program to join



# What's This about Paying To Join the iPhone Developer Program?

- § First: don't get ahead of yourself here: you don't have to pay Apple to develop iPhone apps...
  - § But you do have to sign up as a Registered iPhone Developer
- § ...but you do have to pay Apple if you actually want to deploy an iPhone app to an iPhone
  - § True even for testing an app on an iPhone!
  - § Must enroll in the iPhone Developer Program

# iPhone Developer Program Options

## § Standard

§ One-time \$99 fee

§ Individual: single developer

§ Company: team of developers

## § Enterprise

§ One-time \$299 fee

§ Team of developers

# App Distribution Types

## § Public: Standard iPhone Developer Program only

- § Deployment via App Store and iTunes

- § If app costs money, installation requires purchase

- § Important note: apps must be approved by Apple before they are made available

## § In-house: Enterprise iPhone Developer Program only

- § Deployment to authorized iPhones, via iTunes

## § Selected: both Standard and Enterprise iPhone Developer Programs

- § Ad Hoc deployment to authorized iPhones, via iTunes

- § Limited to 100 users

# What about “Jailbreaking”

- § Jailbreaking an iPhone is the process of releasing it from Apple’s totalitarian control
- § Allows free install of apps on the device without iTunes or the App Store
- § What about developing for both the App Store and for jailbroken iPhones?
  - § I’ve been told (not from any source with Apple) that Apple takes a rather dim view of this
- § Statements made, on slides other than this one, about developing for the iPhone are made in the context of developing apps for non-jailbroken iPhones
  - § They may or may not apply to developing apps for jailbroken iPhones

# Okay, I Get It and I Want To Create an iPhone App—Now What?

- § Log into the iPhone Dev Center
- § Download the iPhone SDK
- § Install the iPhone SDK on a Mac
- § Develop and test the app
- § Make the app available

# Get Ready...Get Set...

## § Have a Mac? Guess what? You will!

- § You need a Mac running OS X

## § The iPhone SDK

- § Available in the iPhone Dev Center

- § Standard Mac install process

- § Installs several applications

- § Xcode: the IDE for iPhone app (and more) development

- § Interface Builder: the visual designer for iPhone app UIs (and more)

- § iPhone Simulator: the Mac OS testing environment for iPhone apps

- § More!

# Go!

## § Develop and test app

- § Will use Xcode, maybe Interface Builder, probably the iPhone Simulator

- § Xcode to write and compile code

- § Interface Builder to assist with the writing of UI

- § iPhone Simulator to run compiled app

## § Make app available

- § We covered this earlier

The logo consists of the letters 'C' and 'F' in a bold, blue, sans-serif font. The 'C' is on the left and the 'F' is on the right, both with a slight 3D effect. The logo is set against a white square background that has a subtle gradient and is reflected on a dark surface below it.

# Demo

Xcode, Interface Builder, and iPhone Simulator



# Coding Native Apps the iPhone

## § Apple's way: Objective-C

§ Pros: full (well, at least as full as it gets) capabilities

§ Cons: new language to learn (if you don't already know it)

## § My way for this presentation: JavaScript, HTML, and CSS

§ Pros: I already know these languages

§ Cons: limited capabilities: I only get the capabilities of these languages + what the underlying framework for them exposes to me

§ Either approach produces native apps that can be submitted to Apple for consideration for inclusion in the App Store!

# Frameworks: How to Create Apps for iPhone without Objective-C

- § Effectively abstract away Objective-C
- § Allow programming with one or more other languages
- § Expose iPhone APIs into the other language(s)
- § Tons available...

# Frameworks Comparison

Name	Languages	OS Targets	Support	Price
<a href="#"><u>NimbleKit</u></a>	JavaScript, HTML, and CSS	iPhone	Included	Free to try \$99 to buy
<a href="#"><u>PhoneGap</u></a>	JavaScript, HTML, and CSS	iPhone, Android, and Blackberry OS	Nothing official	Free
<a href="#"><u>quickconnect</u></a>	JavaScript, HTML, and CSS	iPhone, Android, and Symbian	Nothing official	Free
<a href="#"><u>LiquidGear</u></a>	JavaScript, HTML, and CSS	iPhone	Nothing official	Free
<a href="#"><u>Appcelerator Titanium Mobile</u></a>	JavaScript, HTML, and CSS	iPhone	Included	? (currently in beta)
<a href="#"><u>Rhomobile Rhodes</u></a>	Ruby and HTML	iPhone, Android, Blackberry OS, Symbian, and Windows Mobile	Via Google Group	\$500 per app for unlimited users
<a href="#"><u>Anasca Corona</u></a>	Lua	iPhone	Unclear	? (currently in beta)

The logo consists of the letters 'C' and 'F' in a bold, blue, sans-serif font. The 'C' is on the left and the 'F' is on the right, both rendered in a consistent blue color. The logo is set against a white square background that has a subtle gradient and is reflected on a dark surface below it.

# Demo

Using a Framework: NimbleKit

CF

Q&A



# Summary

§ Don't need to know Objective-C to write native apps for iPhone!

§ Can use JavaScript, HTML, and CSS and/or other languages

§ Getting started

§ Log into the iPhone Dev Center

§ Download the iPhone SDK

§ Install the iPhone SDK on a Mac

§ If using a framework:

§ Download the framework

§ Install the framework

§ Develop and test app

§ Make app available

# Additional Details: Player or Virtual Machine Applications

- § Apps that run in a player or virtual machine on top of the mobile device's OS
  - § Flash Player
  - § JVM
  - § More
- § Code is resident on the client device
- § iPhone does not currently support applications for players or virtual machines

# Additional Details: Web Apps

§ Apple provides resources for optimizing web applications for the iPhone

§ Check out “Getting Started with iPhone Web Apps”:

[http://developer.apple.com/safari/library/referencelibrary/GettingStarted/GS\\_iPhoneWebApp/index.html#//apple\\_ref/doc/uid/TP40008134](http://developer.apple.com/safari/library/referencelibrary/GettingStarted/GS_iPhoneWebApp/index.html#//apple_ref/doc/uid/TP40008134)

§ Apple provides a development environment, installed as part of the iPhone SDK, which can be used to create web applications: Dashcode

§ Check out “Dashcode User Guide: Web Application Tutorial”:

[http://developer.apple.com/safari/library/documentation/AppleApplications/Conceptual/Dashcode\\_UserGuide/Contents/Resources/en.lproj/MakingaWebApp/MakingaWebApp.html#//apple\\_ref/doc/uid/TP40004692-CH18](http://developer.apple.com/safari/library/documentation/AppleApplications/Conceptual/Dashcode_UserGuide/Contents/Resources/en.lproj/MakingaWebApp/MakingaWebApp.html#//apple_ref/doc/uid/TP40004692-CH18)

§ Deployment requires putting these web apps on a web server

§ There are libraries available to help the development/deployment of web apps that will support multiple mobile browsers

§ WURFL, UA Prof, and so on



# Additional Details: Apps in iPhone App Store

- § You get 70% of sales price; Apple gets 30%
- § Cannot create a trial app or an app that times out—so customers must pay up front for your app; there is no “try before you buy” approach possible
  - § Often, apps are released in multiple versions (generally 2):
    - § Free edition: often called something like “lite version;” will have limited features and/or ads
      - § Note that Apple may reject apps that are considered upselling apps and apps with limited features as compared to the full version can cause Apple to consider them upselling apps
    - § Full edition: will have full features and generally no ads
- § Ad networks
  - § Google AdSense (now in public beta)
  - § AdMob
  - § PinchMedia
  - § More

# Additional Details: iPhone Dev Center

§ <http://developer.apple.com/iphone>

§ Requires you to have signed up as a Registered iPhone Developer

§ iPhone SDK Download

§ <http://developer.apple.com/iphone/index.action#downloads>

# Additional Details: Registered iPhone Developer Program

§ <http://developer.apple.com/iphone/program/start/register>

§ Free

§ Requires you to provide some information, create a login (if you don't already have one)

# Additional Details: iPhone Developer Program Registration

§ <http://developer.apple.com/iphone/program/start/enroll>

§ There are additional requirements to register an organization for the iPhone Developer Program through the Standard Corporate Program or through the Enterprise Program

§ For both Standard Corporate Program and Enterprise Program

§ Must have legal authority to bind organization to the iPhone Developer Program License Agreement

§ Must provide a whole bunch of business documents so Apple can verify the business

§ For Enterprise Program only

§ Company must have at least 500 employees and must provide the number of employees

§ Dun & Bradstreet Number (DUNS)

# Additional Details: Deployment to Authorized iPhones

## § Deployment to authorized iPhones

§ Ad Hoc and Enterprise deployment to authorized iPhones requires deployment to client machines and from there via iTunes to authorized iPhones

§ iTunes is required!

# Additional Details: App Distribution

§ <http://developer.apple.com/iphone/program/distribute.html>

## § App Approval for App Store

§ Buzz is this takes 5 days on the low end, 20 days on the high end—and there's no guarantee app will be accepted at all!

## § Ad Hoc Distribution

§ Must pre-register users: for each, must obtain iPhone's UUID, create a mobile provisioning file, provide it to user

§ More details on Ad Hoc Distribution app installation process

§ <http://codemorphic.com/blog/2009/04/13/installing-an-iphone-application-using-ad-hoc-distribution>

# Additional Details: Frameworks

## § NimbleKit

- § Must purchase in order to submit an app using NimbleKit for consideration for the App Store
- § I noticed that when I put it in my shopping cart then navigated away from the page, I got pop-up window offering me 15% off if I bought the product
- § Support: claims questions generally answered within 1 hour

## § PhoneGap, quickconnect, LiquidGear are community-backed projects and though there is no official support, there are resources for connecting with the community for mutual assistance

## § Appcelerator Titanium Mobile

- § Python and Ruby may come into play here; they do for some other products from Appcelerator and it isn't clear whether or not they do for Titanium Mobile or not

## § Rhomobile Rhodes

- § Develop online with online IDE RhoHub

# Additional Details: Apple Approval of Apps Using Frameworks

- § The buzz is that Apple has chosen not to give approval to some apps because they used PhoneGap
  - § The story is that if you go through and rename the PhoneGap classes to your own unique names, Apple won't be able to tell you're using those classes and therefore will not refuse approval of your app for using PhoneGap
  - § What about other frameworks? It seems like Apple would have no issue with frameworks geared specifically towards working with iPhone, that the OS-agnostic approach of certain frameworks is what raises their ire
    - § However, use a framework at your own risk!



# Additional Details: SSL in Native Apps

- § Apps using SSL must have certification from the Department of Commerce
  - § Getting this certification is an involved and time-consuming process
  - § Check out “Mass Market Encryption Commodity Classification for iPhone Applications in 8 Easy Steps”: <http://www.zetetic.net/blog/2009/08/03/mass-market-encryption-commodity-classification-for-iphone-applications-in-8-easy-steps>

# Additional Details: Jailbreaking

- § Apple does not approve: they can void the warranties on jailbroken iPhones
- § There are alternative apps to the iPhone App Store that can be accessed on jailbroken iPhones to find 3<sup>rd</sup> party apps not available in the iPhone App Store
- § Developing for both the App Store and for jailbroken iPhones?
  - § There is a Registered iPhone Developer Agreement you must accept in order to register for the program; consult legal counsel if you have questions
    - § [http://developer.apple.com/iphone/terms/registered\\_iphone\\_developer\\_20081020.pdf](http://developer.apple.com/iphone/terms/registered_iphone_developer_20081020.pdf)
    - § Check also the agreement you must accept for the iPhone Developer Program

# Additional Details: ColdFusion

- § ColdFusion has a built-in web server which can handle requests and send responses to clients, however...
- § ...if using an external web server (which is recommended), requests are made to it; it hands off requests to the ColdFusion server, which hands responses back to the external web server which relays them on to the client

# Resources

§ Wikipedia's iPhone entry

§ <http://en.wikipedia.org/wiki/IPhone>

§ My blog post "Flash Player on Mobile Devices":

§ <http://blog.joshuaadams.com/index.cfm/2009/8/13/Flash-Player-on-Mobile-Devices>

§ 6 Steps to Building a Better iPhone App

§ <http://mashable.com/2009/06/10/build-iphone-app>

§ Andy Matthews's Notes from cf.Objective() 2009 iPhone Application Birds of a Feather

§ <http://www.andymatthews.net/read/2009/05/15/iPhone-Application-Development-BoF-notes---cf.Objective>

§ If you are interested in learning Objective-C, Andy provides links to learning resources

# Join the Social



twitter

- § @ColdFusion
- § @BenForta
- § @Adrocknaphobia - Adam Lehman
- § @TPRyan - Terry Ryan
- § @JLAdams97 - Josh Adams
- § @cfemea - Claude Englebert

# Adobe Developer eSeminars

## § Adobe ColdFusion eSeminar Series

§ Next Session: Centaur and Bolt: Accelerated ColdFusion Development

August, 19, 2009

11:00 AM - 12:00 noon PT

§ [More Information + Registration](#)

## § Adobe Flash Platform eSeminar series for Adobe AIR and Flex developers

§ [More Information + Registration](#)

## § [Adobe ColdFusion OnDemand eSeminars](#)



**Save US\$200** off the regular price of a full conference pass (offer ends 8/31)

§ Connect with peers who are already pushing the boundaries of what's possible

§ Choose from more than 300 sessions and labs

§ Learn directly from industry and Adobe experts

**ADOBE MAX 2009**  
CONNECT. DISCOVER. INSPIRE.

**October 4–7, 2009**

Los Angeles Convention Center  
and Nokia Theatre L.A. LIVE

Los Angeles, California

[Register now at max.adobe.com](http://max.adobe.com)



# Josh Adams

Adobe Systems Incorporated  
Sr. SE, ColdFusion Specialist  
(678) 701-7051

[joadams@adobe.com](mailto:joadams@adobe.com)

<http://blog.joshuaadams.com>

<http://www.facebook.com/jladams97>

<http://www.twitter.com/jladams97>

## Adobe ColdFusion Sales

[cfsales@adobe.com](mailto:cfsales@adobe.com)





# [To Do]

- § In future versions of this presentation, I plan to:
  - § Do cooler things in the code such as interact with a local DB
  - § Include a demo of an app written with Objective-C