QCon LONDON

#qconlondon

Swift@IBM

The Case for Bringing Swift to the Server

Patrick Bohrer, Chris Bailey

IBM Cloud

Agenda

- Why does this matter ?
- Swift Background
- What does it take to really support Swift on Linux ?
- What does it take to really support Swift on the Cloud ?
- Developer Resources for Innovators and Early Adopters

Why do we care if a mostly mobile dev language is now available on the server ?

Hint: Building new mobile experiences drive new cloud requirements

A little bit about me

Systems → Mobile → Cloud → Swift

Pre-2008: My Systems Years

C language

Functional Verification

Multithreading

CPUs/Threads

Hypervisors

Debuggers

Compilers

Performance Modeling

Memory

Latency vs BW

CPU vs Memory

Caching

JITs

Operating Systems

System Simulators

Then This Happened

iPhone

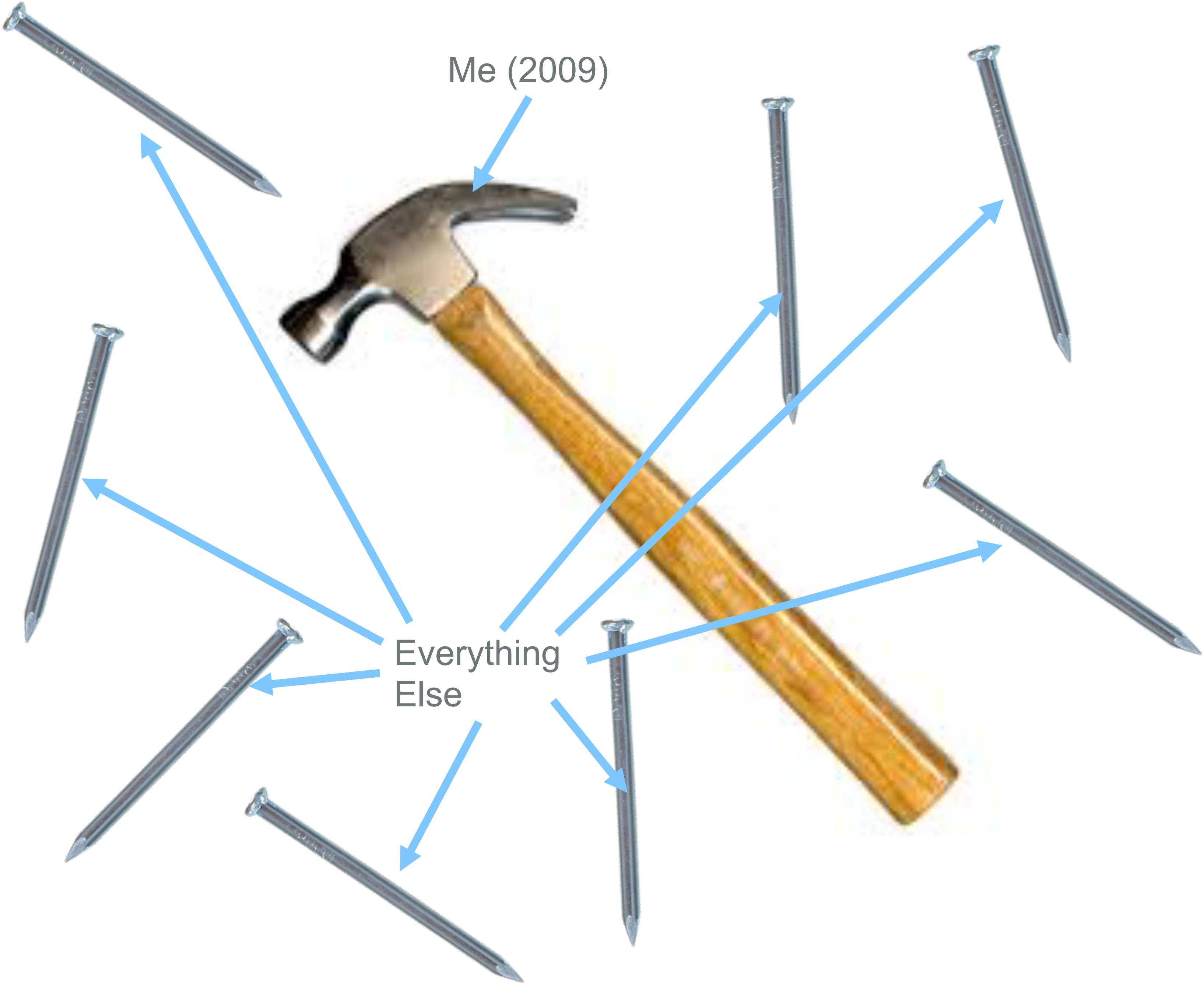
The iPhone App Store
opened on July 10, 2008

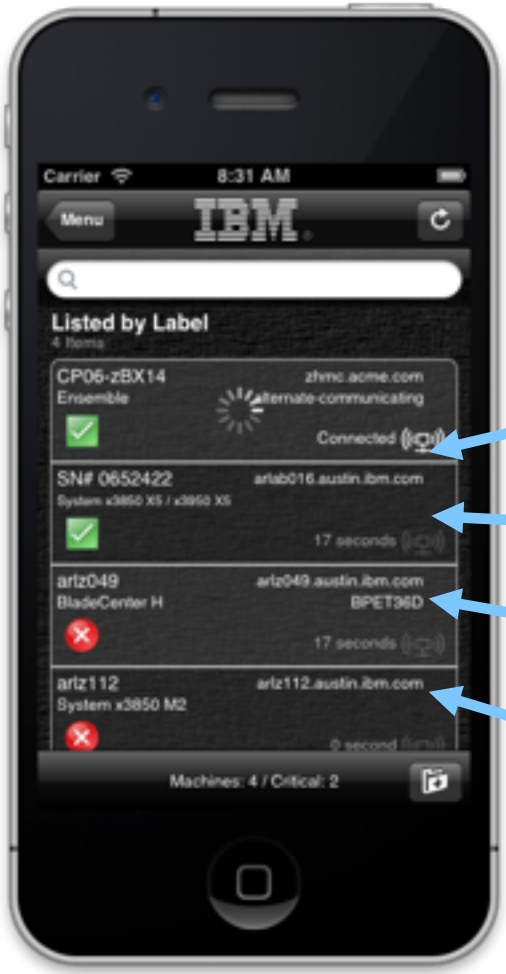
ANNOUNCED:
January 9, 2007

RELEASED:
June 29, 2007

KEY FEATURES:
3.5-inch diagonal screen; 320 x 480 pixels at 163 ppi; 2-megapixel camera

PRICING:
4GB model, $499; 8GB version, $599 (with a two-year contract)


Me (2009)
Everything Else



Carrier 8:31 AM

Menu IBM

Listed by Label
4 Items

CP06-zBX14 zhmc.acme.com
Ensemble alternate-communicating
✓ Connected

SN# 0652422 arlab016.austin.ibm.com
System x3850 X5 / x3950 X5
✓ 17 seconds

arlz049 arlz049.austin.ibm.com
BladeCenter H BPET36D
✗ 17 seconds

arlz112 arlz112.austin.ibm.com
System x3850 M2
✗ 0 second

Machines: 4 / Critical: 2

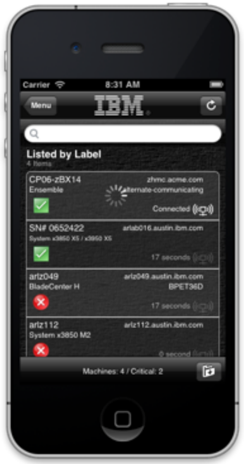
New Experience exposed Capabilities Mismatch

Mobile Usage:

- Frequent Use
- Short (~30 sec) Usage Cycle
- Low Latency
- Precious Network BW
- Prioritize alerting flow
- Drill down data requirements

Legacy Sys Mgmt Usage:

- Infrequent Use
- 20 minute usage minimum
- High latency
- Unlimited Network
- Large data payloads
- Sensor/trigger data treated same as other data



Listed by Label
4 Items

CP06-zBX14 zhmc.acme.com
Ensemble alternate-communicating
✓ Connected

SN# 0652422 arlab016.austin.ibm.com
System x3850 X5 / x3950 X5
✓ 17 seconds

arlz049 arlz049.austin.ibm.com
BladeCenter H BPET36D
✗ 17 seconds

arlz112 arlz112.austin.ibm.com
System x3850 M2
✗ 0 second

Machines: 4 / Critical: 2

Client layer:

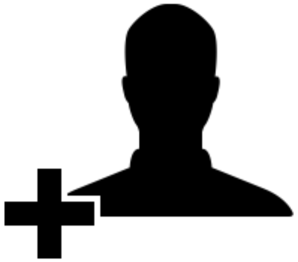
- Latency Hiding
- Smart Caching
- Lazy Loading
- Add Liveness
- Async Everything

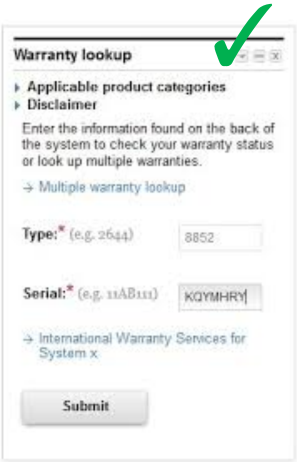
ANDROID APP ON Google play

Available on the App Store

Guess what ?   They Love it and Want More!

- Share across my devices
- Slightly different experience for each device
- Background monitoring
- Collaborate with co-workers (social)
- Warranty Lookup
- Notification of firmware updates
- …





Warranty lookup

- Applicable product categories
- Disclaimer

Enter the information found on the back of the system to check your warranty status or look up multiple warranties.

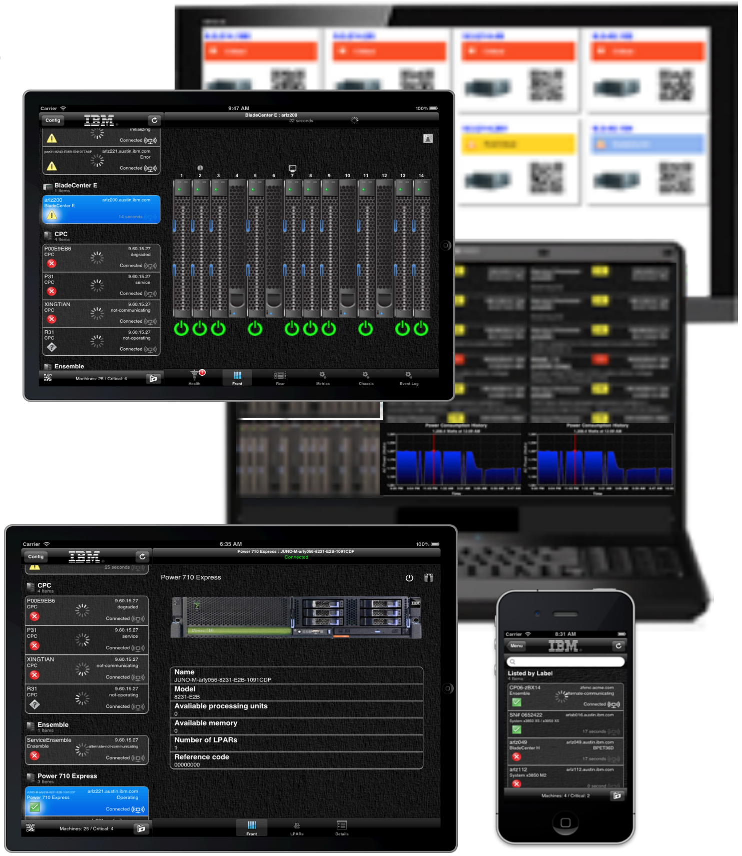
→ Multiple warranty lookup

Type:* (e.g. 2644) 8852

Serial:* (e.g. 11AB111) KQYMHRY

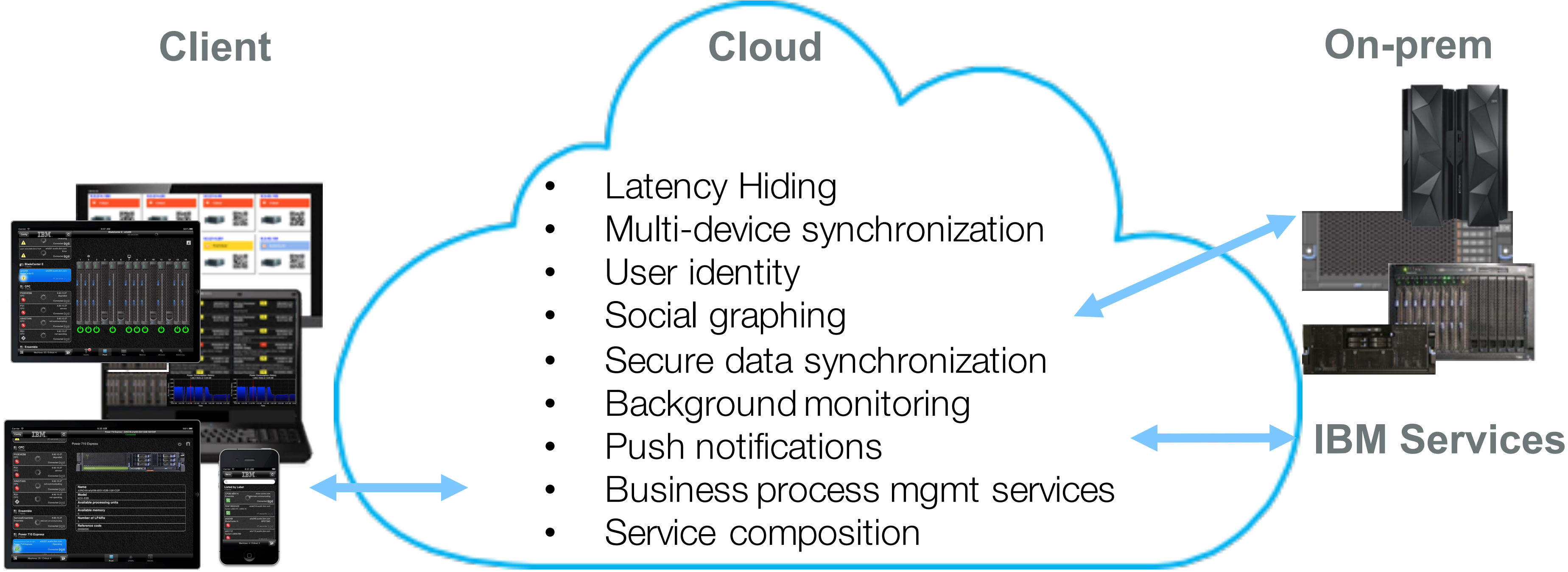
→ International Warranty Services for System x

Submit



Compelling Mobile Experiences Create Need for new Cloud Services

- Simple mobile experience led to much more involved solution
- Middle tier services built to deliver demands of new mobile experience
- I knew what my mobile clients needed and had to switch roles/gears to delivery capabilities across stack/languages (Obj-C → NodeJS → C/Java)



Client

Cloud

- Latency Hiding
- Multi-device synchronization
- User identity
- Social graphing
- Secure data synchronization
- Background monitoring
- Push notifications
- Business process mgmt services
- Service composition

On-prem

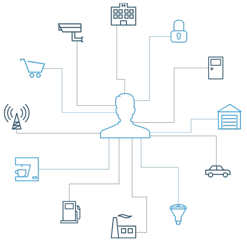
IBM Services

User Experience Across Devices

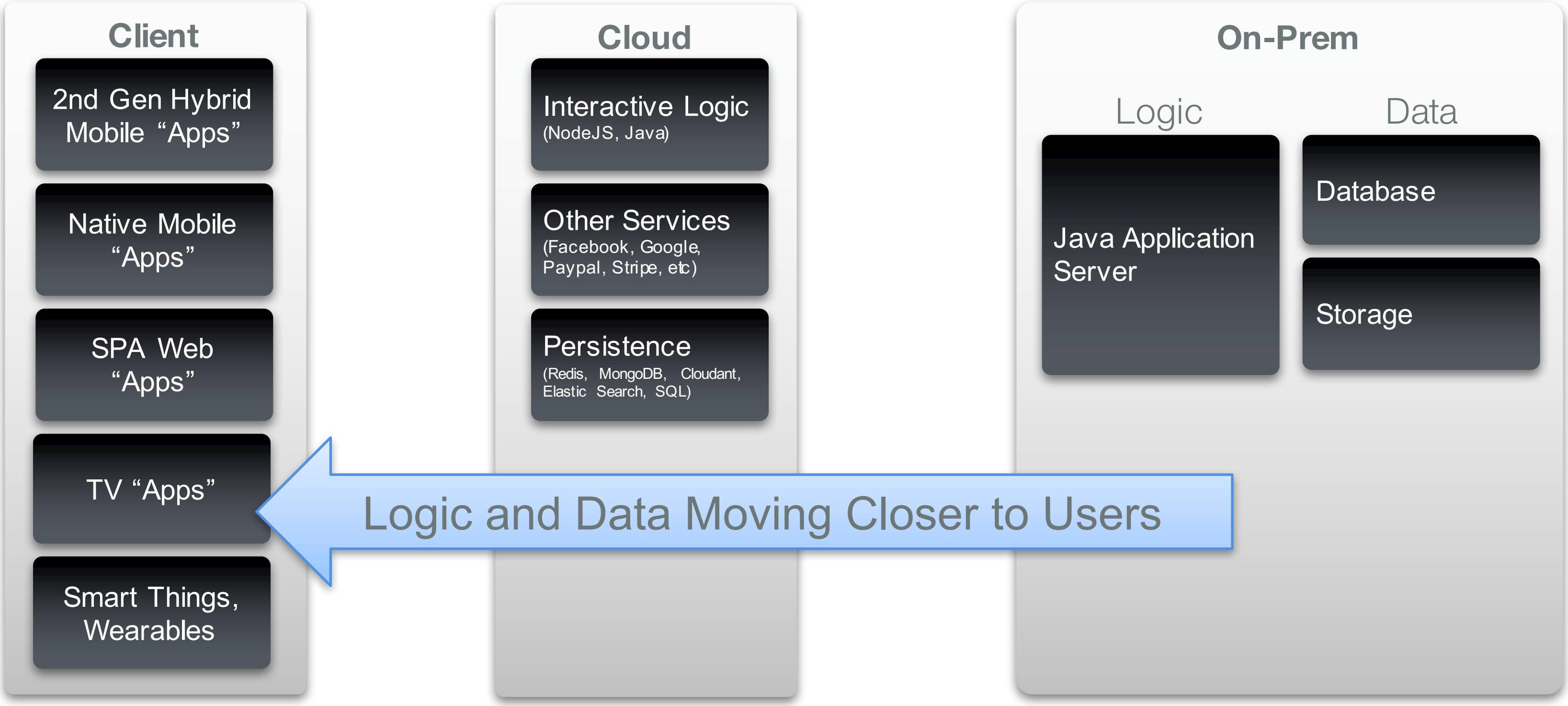
Always-on Cloud

Legacy Services

Evolution of Enterprise Applications: Data Model Contrast



New Experiences are drawing logic and data closer to users



	Client	Cloud	On-Prem
Tier Attributes	• State-full • Significant App Logic • Real-time • Eventual Consistency • Disconnected Use • Sometimes on • Limited CPU/Mem/BW	• Event-driven • Low latency • High volume • Request composition • API Driven • Always On • Infinite CPU/Mem/Network	• Transactional • Session based • Lower Volume
Dev Skills	• Objective-C • Swift • Java • Javascript	• Javascript • Java • Python • NoSQL, In-Mem DBs, Obj Store, etc	• Java • J2EE • SQL DB

| Swift@IBM

Not unique to me or my app

Mobile Drives Cloud Demand

We have seen this same pattern emerge in the apps we have built since these days. Many of these new apps (including the Apple/IBM Partnership apps) are now written in ***Swift***

Swift Background

Created by Apple

Created by team at Apple by Chris Lattner (creator of LLVM) and team to replace Objective C. Strongly influenced from languages like "Objective-C, Rust, Haskell, Ruby, Python, C#, CLU, and far too many others to list".

Strongly typed, concise syntax, and modern language features.

Swift Programming Language

Apple released the Swift language for iOS and OS X development at WWDC 2014. Swift is the strategic language for the future of Apple ecosystem development. Swift is industrial-quality systems programming language that blends the expressiveness and ease of found in scripting languages. Within a year, Swift has emerged as one of the top programming languages. Swift, released in June 2014, just broke the top 20, now at 18, on RedMonk's Programming Language Rankings: June 2015.

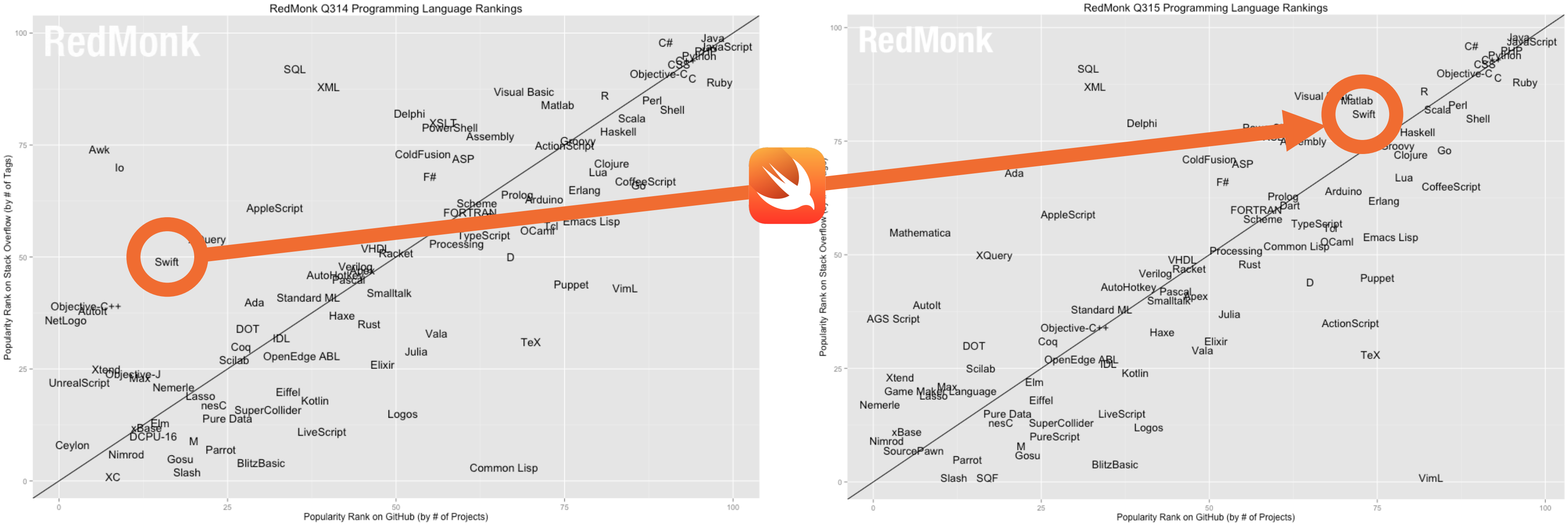
Technical Background

Swift leverages Apple's language architecture. The language parser is built on the LLVM compiler infrastructure. LLVM is the layered backend compiler infrastructure used for intermediate code generation, profiling, debugging, and instrumentation. All Apple supported languages including C, C++, Objective-C, and JavaScript are based on LLVM.

Open Source Swift

Apple open sourced Swift on December 3, 2015. It includes support for Linux, Swift language parser and integrated LLVM infrastructure, integrated LLVM LLDB debugger, integrated Read/Eval/Print Loop (REPL) interactive command line tool, Swift Package Manager, Libdispatch (Grand Central Dispatch), Foundation and the Swift Standard Library.

Growth in Swift Popularity

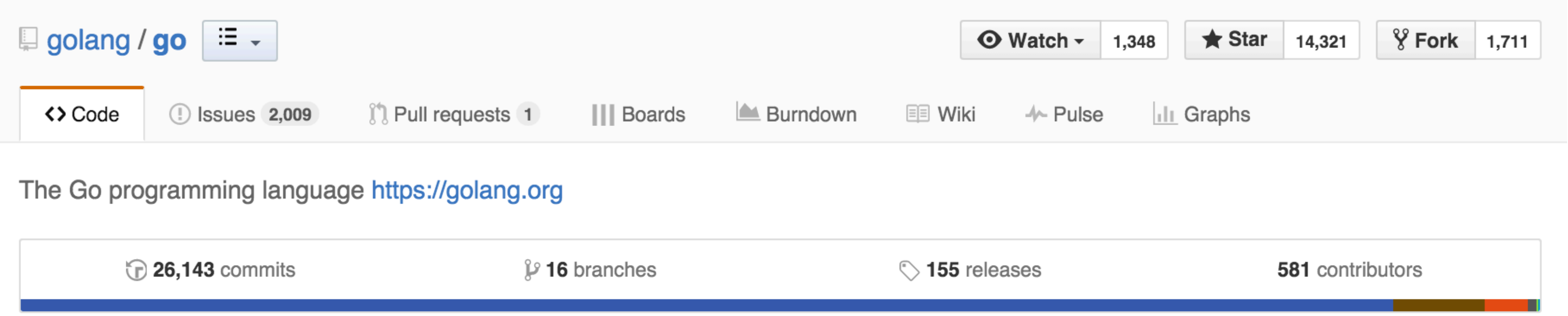


2014

2015

Surging Github Popularity

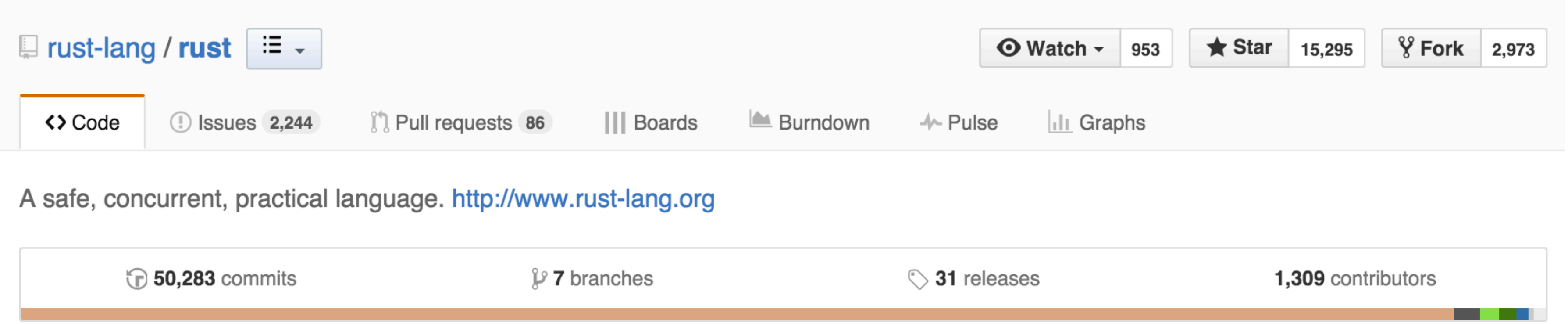
within 2 months compared to other popular languages



golang / go

The Go programming language https://golang.org

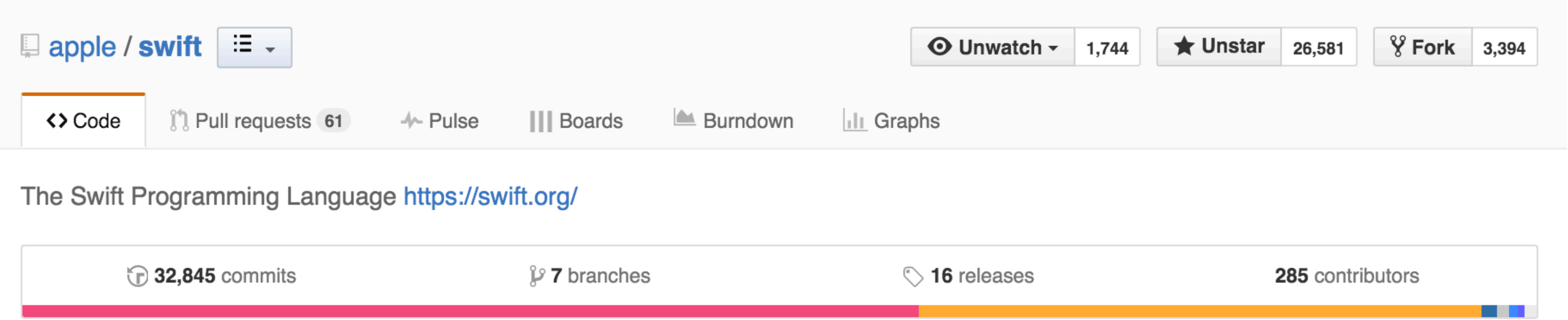
26,143 commits
16 branches
155 releases
581 contributors



rust-lang / rust

A safe, concurrent, practical language. http://www.rust-lang.org

50,283 commits
7 branches
31 releases
1,309 contributors



apple / swift

The Swift Programming Language https://swift.org/

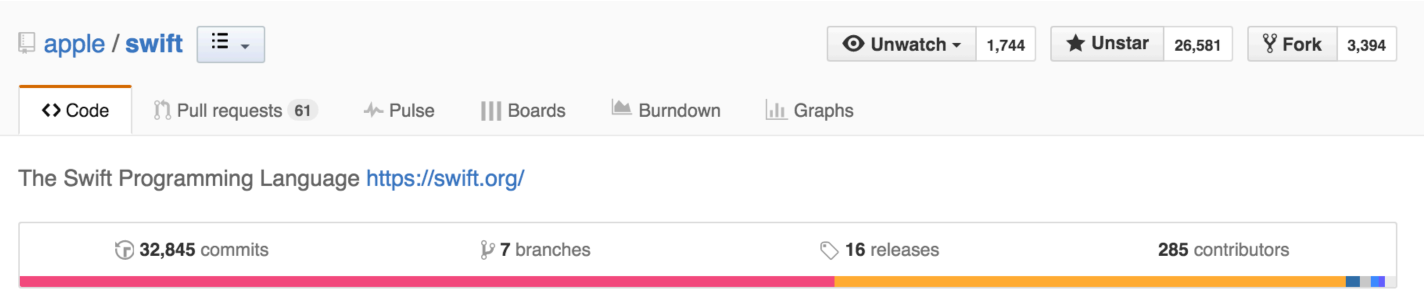
32,845 commits
7 branches
16 releases
285 contributors

Developer Empowerment in the Cloud

Swift caters to an incredibly affluent and growing community of developers. These developers are creating applications that are literally changing the way we all live our lives.

These applications are dependent upon the Cloud to delivery these experiences.

The open sourcing of the language and will now open opportunities around language adoption across servers and other client platforms.



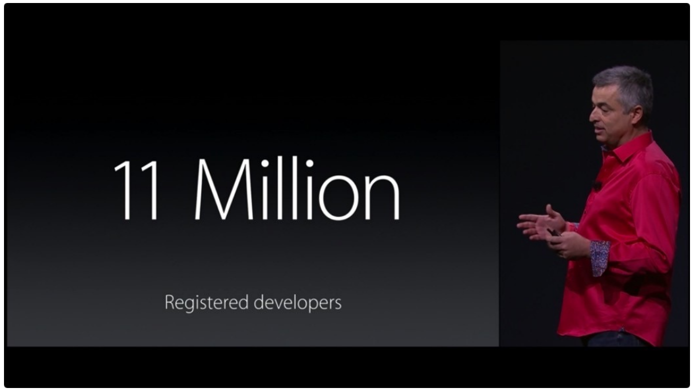
apple / swift

Unwatch 1,744 | Unstar 26,581 | Fork 3,394

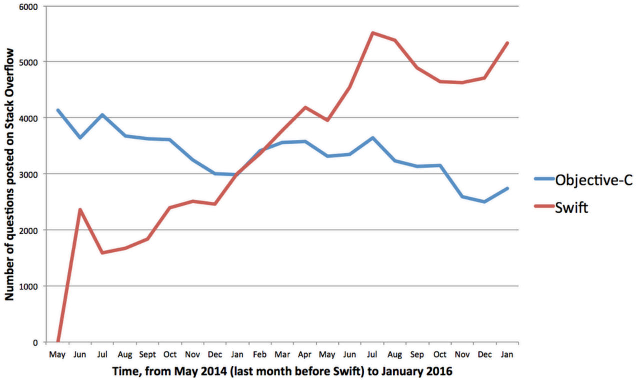
Code | Pull requests 61 | Pulse | Boards | Burndown | Graphs

The Swift Programming Language https://swift.org/

32,845 commits | 7 branches | 16 releases | 285 contributors

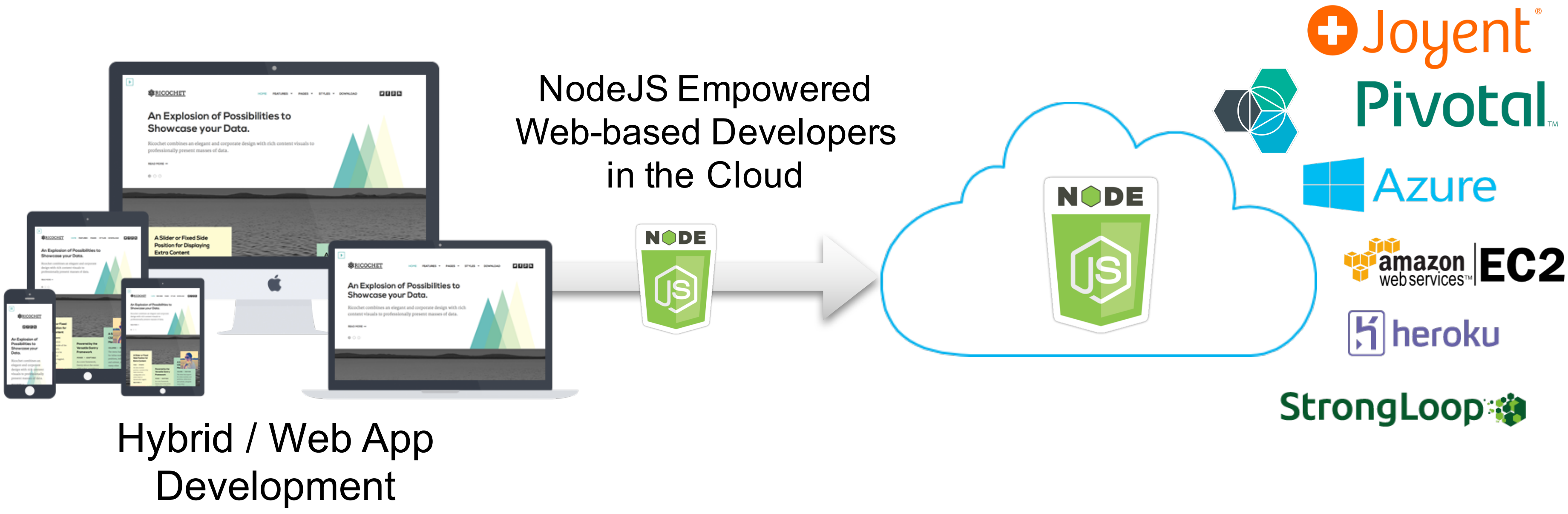


11 Million Registered Apple Developers in 2015



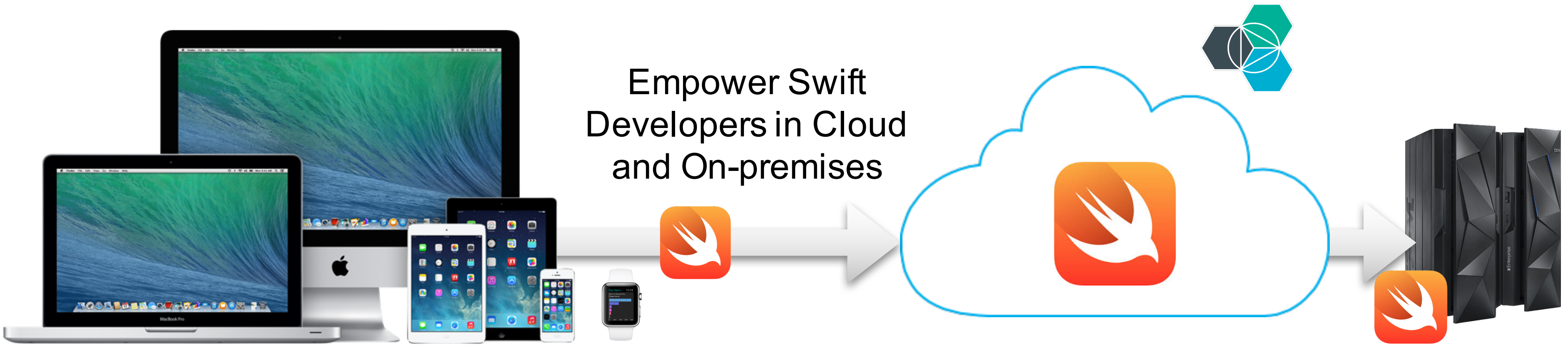
Stackoverflow Questions Hint at Conversion Trend from ObjC → Swift

Empowering New Client-side Development Community



NodeJS Empowered Web-based Developers in the Cloud

Hybrid / Web App Development



Empower Swift Developers in Cloud and On-premises

Swift App Development

What does it take to bring a new language to the Server ?

Hint: Base Runtime,
Foundation, Concurrency,
Debugging, Instrumentation,
Performance, IDE support,
→ Viable Runtime

IBM Runtime Technologies

developerWorks Open

Senior Technical Staff Member

chrisbaileyibm

@seabaylea

@Chris__Bailey

About Me

About Me

About Me

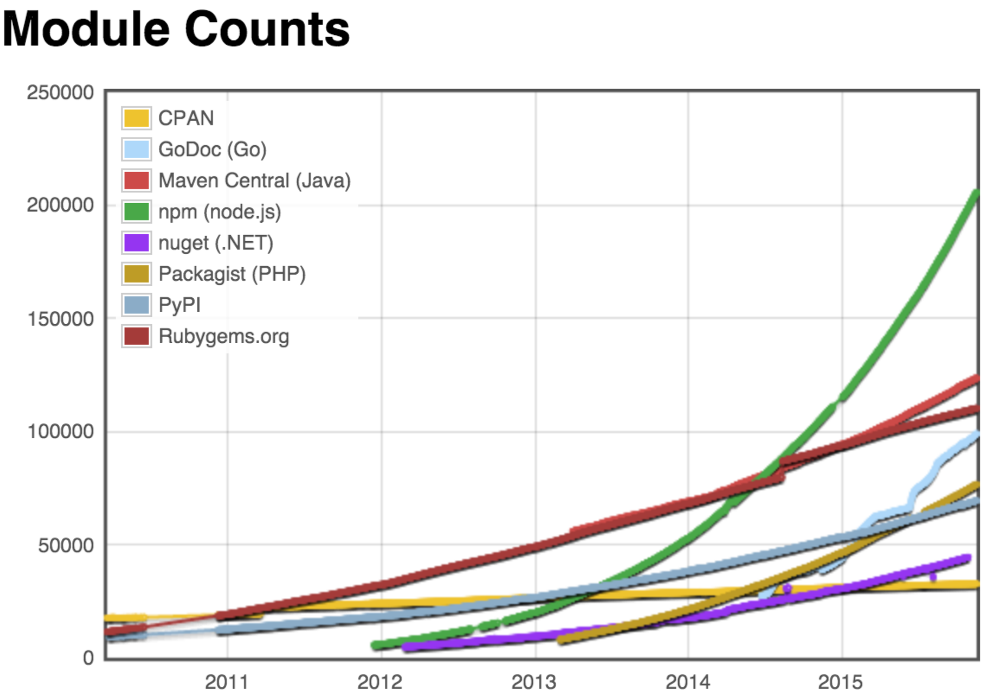
About Me

Lessons to be learned from NodeJS Timeline

V8 Release (2008)	+ libuv (concurency) + foundation = NodeJS (2009)	+ Express: beta→1.0 (Web Framework) (2009-2010)	+ npm (2009-2011)	Package Growth (2012)	Initial Popularity (2012-2013)	Mainstream Usage (2014-2015)
Swift Release (Late 2015)	+ libdispatch (concurency) + Foundation + web foundation (Kitura) = ?? (Early 2016)	+ Kitura: alpha→1.0 (Web Framework) (2016-?)	+ swift pkg mgr + catalog (? 2016)	Package Growth (2016-2017)	Initial Popularity (??)	Mainstream Usage (??)

We are here

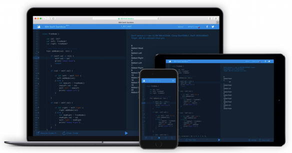
Many differences in the languages but insights can be gained by looking at a similar community driven development timeline.



Lessons to be learned from NodeJS Timeline

V8 Release (2008)	+ libuv (concurency) + foundation = NodeJS (2009)	+ Express: beta→1.0 (Web Framework) (2009-2010)	+ npm (2009-2011)	Package Growth (2012)	Initial Popularity (2012-2013)	Mainstream Usage (2014-2015)
Swift Release (Late 2015)	+ libdispatch (concurency) + Foundation + web foundation (Kitura) = ?? (Early 2016)	+ Kitura: alpha→1.0 (Web Framework) (2016-?)	+ swift pkg mgr + catalog (? 2016)	Package Growth (2016-2017)	Initial Popularity (??)	Mainstream Usage (??)

Swift.org Contributions



Sandbox

KITURA™

| Swift@IBM

Status Quo

Still early days

- Language evolution (1.0 → 2.0 → 3.0 → 4.0) currently at 2.2 → 3.0

New:

- Swift Package Manager:
  - Being developed by Max Howell (creator of Homebrew on Mac)

New on Linux (Still in progress)

- Libdispatch (Concurrency)
- Foundation (Objective-C library on iOS/OSX → Pure Swift)

Concurrency Implementation

Grand Central Dispatch (“Dispatch”)

- Efficiently provides execution services, resource management, QoS, event sources, etc.
- *Dispatch Queues:* serial or concurrent execution queues
- *Dispatch Sources:* register callbacks to execute on system events
- *Dispatch Groups:* allows tasks to be grouped and joined when complete

```
for task in 0...NTASKS-1 {
   let f = Fannkuchredux(n: n);
   f.runTask(task, Fact: Fact, CHUNKZ: CHUNKZ);
   chkSums[task] = f.chkSums;
   maxFlips = f.maxFlips;
}
```

Concurrency Implementation

Grand Central Dispatch (“Dispatch”)

- Efficiently provides execution services, resource management, QoS, event sources, etc.
- *Dispatch Queues:* serial or concurrent execution queues
- *Dispatch Sources:* register callbacks to execute on system events
- *Dispatch Groups:* allows tasks to be grouped and joined when complete

```
import Dispatch

let dq = dispatch_queue_create(“tasks”,
           DISPATCH_QUEUE_CONCURRENT)
```

```
for task in 0...NTASKS-1 {
   let f = Fannkuchredux(n: n);
   f.runTask(task, Fact: Fact, CHUNKZ: CHUNKZ);
   chkSums[task] = f.chkSums;
   maxFlips = f.maxFlips;
}
```

Concurrency Implementation

Grand Central Dispatch (“Dispatch”)

- Efficiently provides execution services, resource management, QoS, event sources, etc.
- *Dispatch Queues:* serial or concurrent execution queues
- *Dispatch Sources:* register callbacks to execute on system events
- *Dispatch Groups:* allows tasks to be grouped and joined when complete

```
import Dispatch

let dq = dispatch_queue_create(“tasks”,
           DISPATCH_QUEUE_CONCURRENT)
```

```
for task in 0...NTASKS-1 {
   let f = Fannkuchredux(n: n);
   f.runTask(task, Fact: Fact, CHUNKZ: CHUNKZ);
   chkSums[task] = f.chkSums;
   maxFlips = f.maxFlips;
}
```

```
dispatch_apply(dq, NTASKS { task in
   let f = Fannkuchredux(n: n);
   f.runTask(task, Fact: Fact, CHUNKZ: CHUNKZ);
   chkSums[task] = f.chkSums;
   maxFlips = f.maxFlips;
}
```

Concurrency Implementation

Grand Central Dispatch (“Dispatch”)

- Efficiently provides execution services, resource management, QoS, event sources, etc.
- *Dispatch Queues:* serial or concurrent execution queues
- *Dispatch Sources:* register callbacks to execute on system events
- *Dispatch Groups:* allows tasks to be grouped and joined when complete

```
import Dispatch

let dq = dispatch_queue_create(“tasks”,
           DISPATCH_QUEUE_CONCURRENT)
```

```
for task in 0...NTASKS-1 {
   let f = Fannkuchredux(n: n);
   f.runTask(task, Fact: Fact, CHUNKZ: CHUNKZ);
   chkSums[task] = f.chkSums;
   maxFlips = f.maxFlips;
}
```

```
dispatch_apply(dq, NTASKS { task in
   let f = Fannkuchredux(n: n);
   f.runTask(task, Fact: Fact, CHUNKZ: CHUNKZ);
   chkSums[task] = f.chkSums;
   maxFlips = f.maxFlips;
}
```

Language	Duration (s)	CPU Time	CPU Load
Swift (Dispatch)	13.60	51.44	100% 86% 100% 93%
Swift (Serial)	51.00	50.99	1% 0% 0% 100%

Concurrency Implementation

Grand Central Dispatch (“Dispatch”)

- Efficiently provides execution services, resource management, QoS, event sources, etc.
- *Dispatch Queues:* serial or concurrent execution queues
- *Dispatch Sources:* register callbacks to execute on system events
- *Dispatch Groups:* allows tasks to be grouped and joined when complete

```
import Dispatch

let dq = dispatch_queue_create(“tasks”,
           DISPATCH_QUEUE_CONCURRENT)
```

```
for task in 0...NTASKS-1 {
   let f = Fannkuchredux(n: n);
   f.runTask(task, Fact: Fact, CHUNKZ: CHUNKZ);
   chkSums[task] = f.chkSums;
   maxFlips = f.maxFlips;
}
```

```
dispatch_apply(dq, NTASKS { task in
   let f = Fannkuchredux(n: n);
   f.runTask(task, Fact: Fact, CHUNKZ: CHUNKZ);
   chkSums[task] = f.chkSums;
   maxFlips = f.maxFlips;
}
```

Language	Duration (s)	CPU Time	CPU Load
Swift (Dispatch)	13.60	51.44	100% 86% 100% 93%
Swift (Serial)	51.00	50.99	1% 0% 0% 100%

IBM has contributed 37 or 40 pull requests to Dispatch

Swift Standard Library and Foundation

Swift and the Swift Standard Library:

- Dispatch integration as a first class citizen
- Support for Linux PPC64 LE
- print() performance

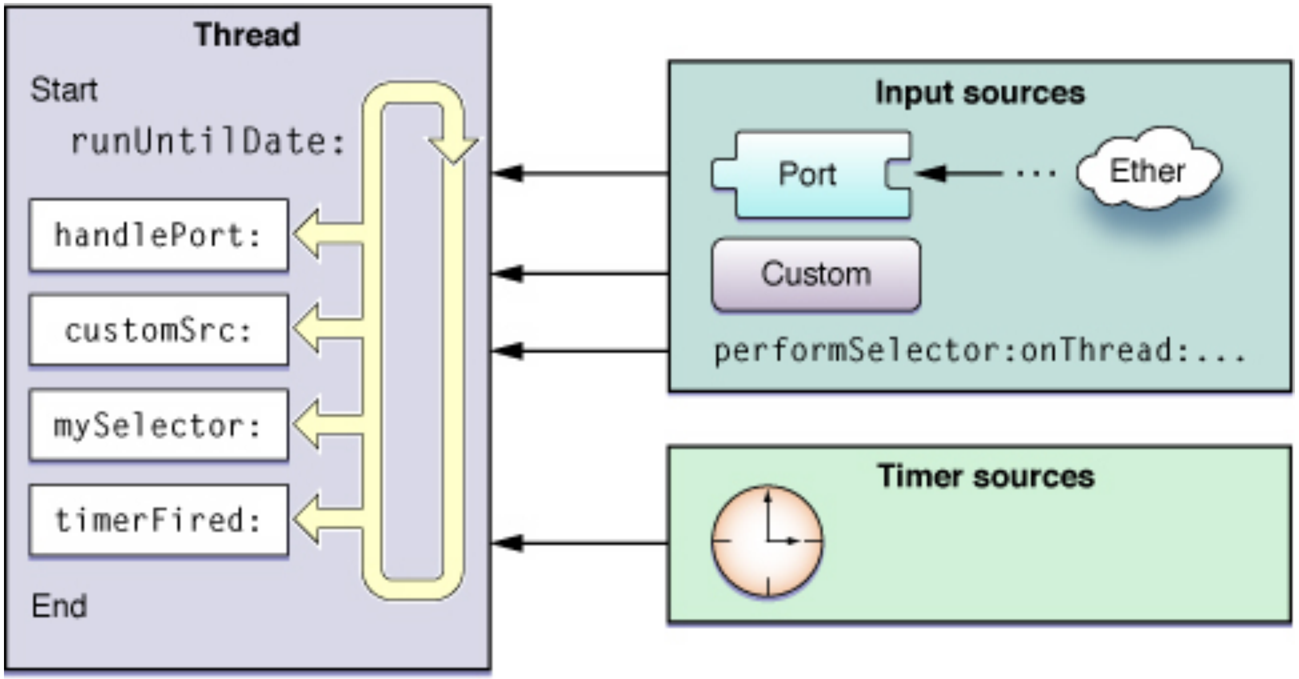
Core Foundation and Foundation:

- NSJSONSerialization
- NSRegularExpression
- NSBundle
- NSNumber
- NSNumberFormatter
- NSJSONSerialization
- CFRunLoop and NSRunLoop

NSRunLoop

Higher level abstraction for asynchronous handling of events from:

- *Input Sources:* System events or custom events
- *Timer Sources:* Scheduled events



What Next?

Continue to contribute Swift implementations of Foundation APIs

- NSOperationQueue
- NSURLSession
- NSStream

Foundational "Server" APIs

- Server sockets
- HTTP parsing
- SSL and TLS support

Enterprise requirements

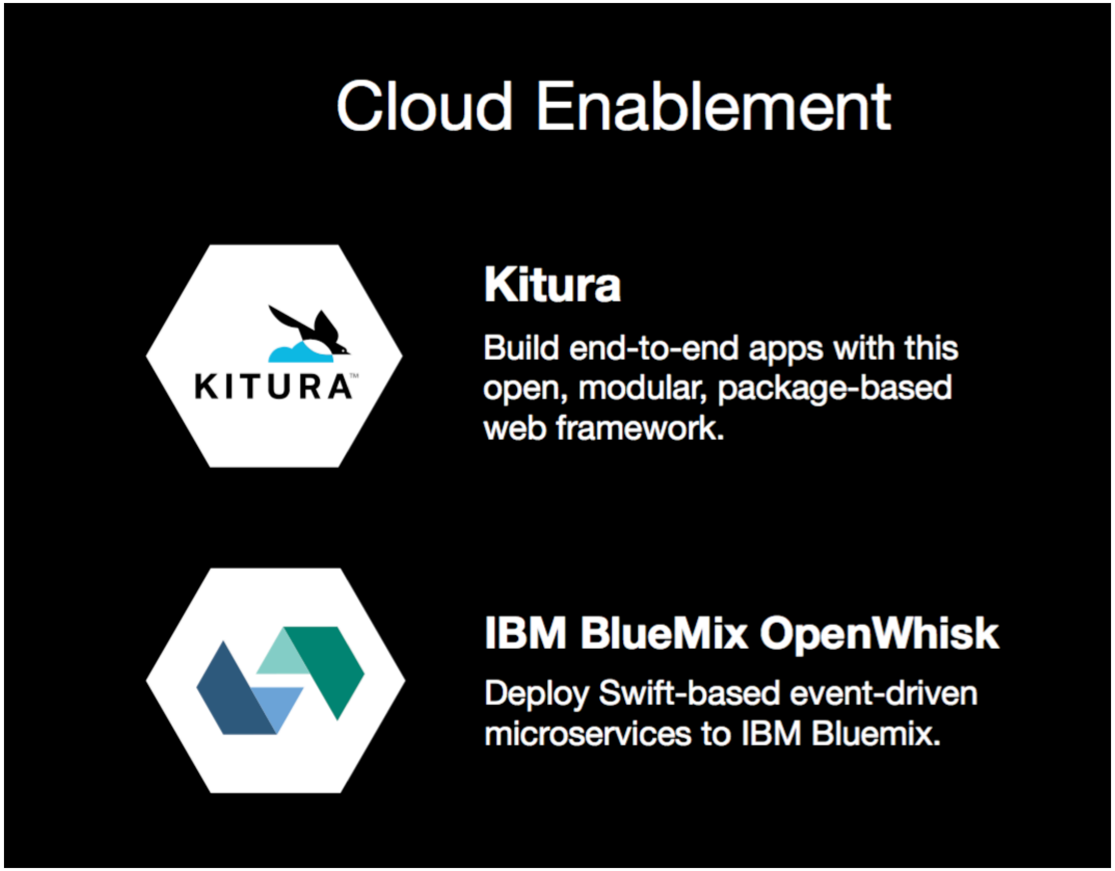
- FIPS compliance
- Transaction processing framework(s)

Enterprise non-functional requirements

- Runtime monitoring, clustering and auto-scaling
- Post-failure diagnostics
- Performance and scalability
- Alternative memory management algorithms

What does it take to bring a new language to the Cloud ?

Hint: Server Support + Web Foundation, Web Frameworks, Cloud Runtimes, Rich Package Ecosystem, Enhanced Developer Experience, Activity Dev Community
→ Developer Value & Buy-in



Cloud Enablement

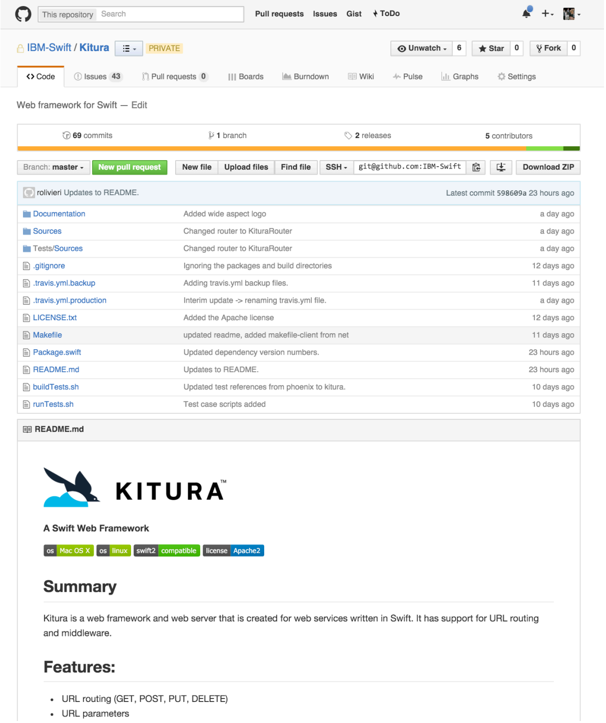
Kitura

Build end-to-end apps with this open, modular, package-based web framework.

IBM BlueMix OpenWhisk

Deploy Swift-based event-driven microservices to IBM Bluemix.

Kitura Web Framework



Web framework for Swift — Edit

rolivieri Updates to README. Latest commit 598609a 23 hours ago

Documentation	Added wide aspect logo	a day ago
Sources	Changed router to KituraRouter	a day ago
Tests/Sources	Changed router to KituraRouter	a day ago
.gitignore	Ignoring the packages and build directories	12 days ago
.travis.yml.backup	Adding travis.yml backup files.	11 days ago
.travis.yml.production	Interim update -> renaming travis.yml file.	a day ago
LICENSE.txt	Added the Apache license	12 days ago
Makefile	updated readme, added makefile-client from net	11 days ago
Package.swift	Updated dependency version numbers.	23 hours ago
README.md	Updates to README.	23 hours ago
buildTests.sh	Updated test references from phoenix to kitura.	10 days ago
runTests.sh	Test case scripts added	10 days ago

KITURA™

A Swift Web Framework

os Mac OS X | os linux | swift2 compatible | license Apache2

Summary

Kitura is a web framework and web server that is created for web services written in Swift. It has support for URL routing and middleware.

Features:

- URL routing (GET, POST, PUT, DELETE)
- URL parameters

What is it?

New, modular, package-based web framework written in Swift

Why is this cool?

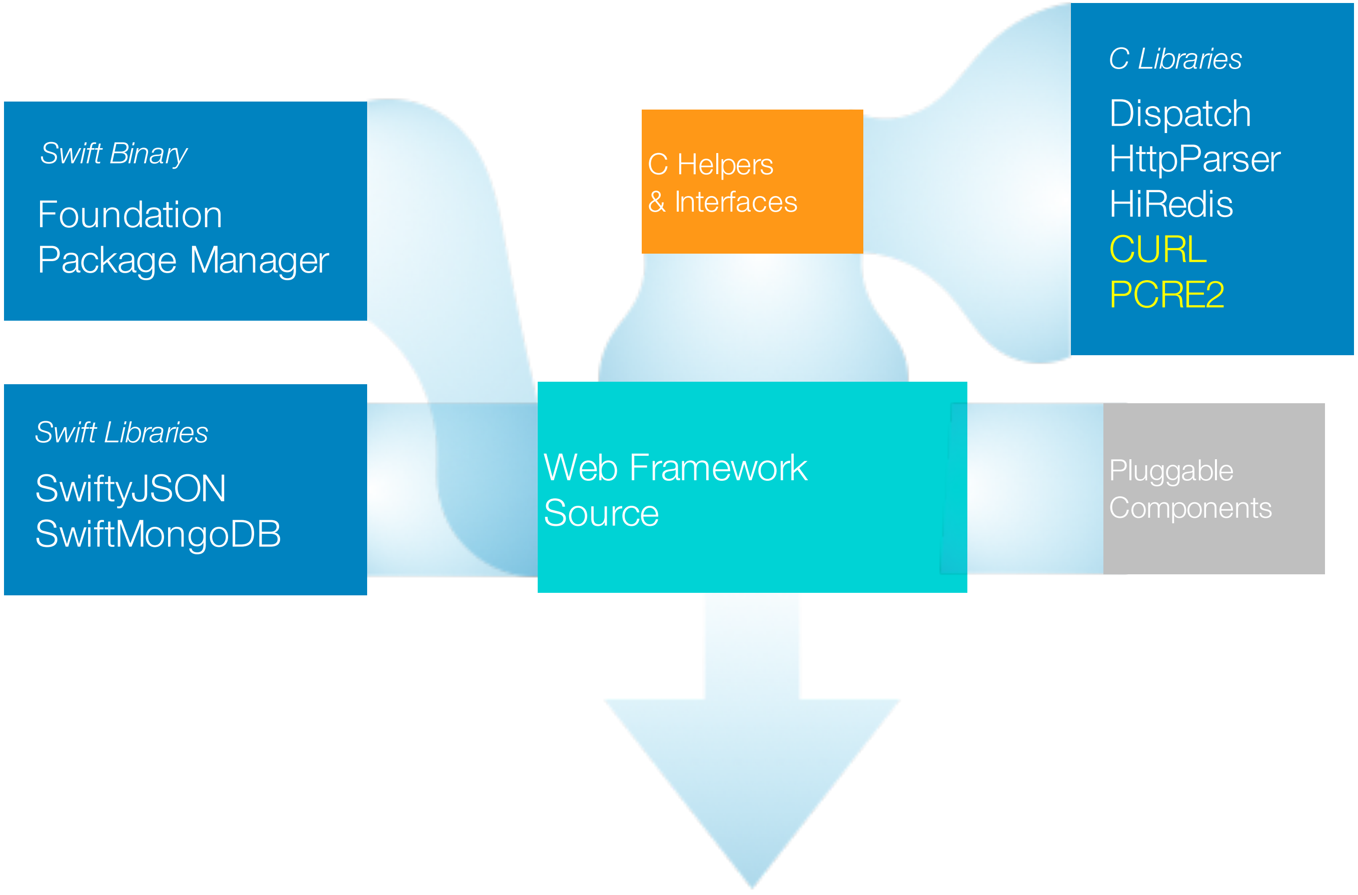
Empower a new generation of native mobile developers to write and deploy code into the Cloud.

Developer Benefits ?

Delivers core technologies needed to stand up enterprise apps on the server

Enables developers to create a web application in Swift and deploy these servers on Linux and the Cloud.

http://github.com/ibm-swift/kitura



Swift Binary

Foundation
Package Manager

Swift Libraries

SwiftyJSON
SwiftMongoDB

C Helpers
& Interfaces

C Libraries

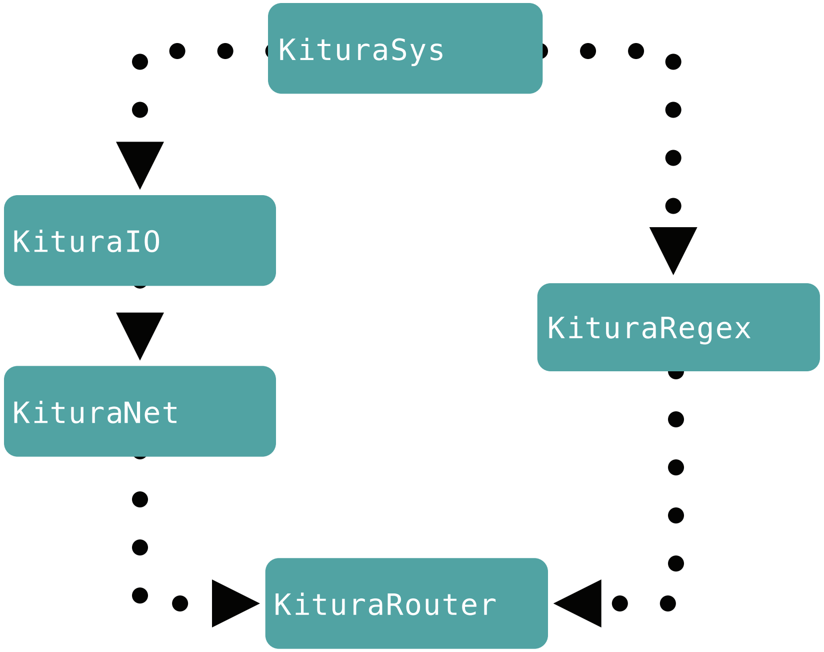
Dispatch
HttpParser
HiRedis
CURL
PCRE2

Web Framework
Source

Pluggable
Components

Web Framework

Web Framework Core Modules

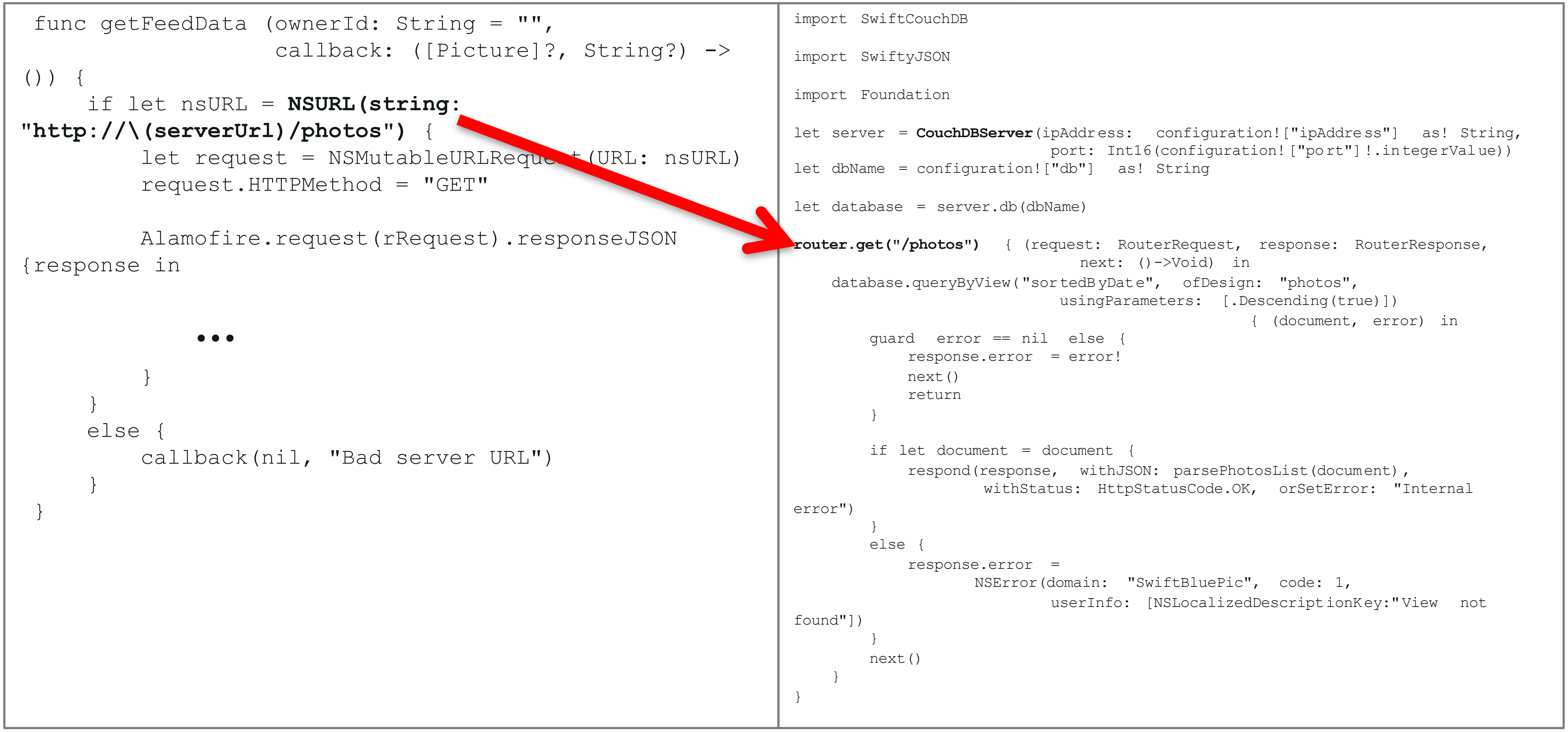


Web Framework App Sample Code

```
router.get("/hello") { (request: RouterRequest,
                        response: RouterResponse,
                        next: () -> Void) in

   let json = JSON([“world”: “From Swift”, “version”: 1])
   response.status(HttpStatusCode.OK).sendJson(json)
   do {
      try response.end()
   }
   catch {}
   next()
}
```

End to end Swift (Client and Server)



```swift
 func getFeedData (ownerId: String = "",
                      callback: ([Picture]?, String?) ->
()) {
      if let nsURL = NSURL(string:
"http://\(serverUrl)/photos") {
           let request = NSMutableURLRequest(URL: nsURL)
           request.HTTPMethod = "GET"

           Alamofire.request(rRequest).responseJSON
{response in

                  ...

           }
      }
      else {
           callback(nil, "Bad server URL")
      }
 }
```

```swift
import SwiftCouchDB

import SwiftyJSON

import Foundation

let server = CouchDBServer(ipAddress: configuration!["ipAddress"] as! String,
                           port: Int16(configuration!["port"]!.integerValue))
let dbName = configuration!["db"] as! String

let database = server.db(dbName)

router.get("/photos") { (request: RouterRequest, response: RouterResponse,
                           next: ()->Void) in
    database.queryByView("sortedByDate", ofDesign: "photos",
                          usingParameters: [.Descending(true)])
                                              { (document, error) in
        guard error == nil else {
            response.error = error!
            next()
            return
        }

        if let document = document {
            respond(response, withJSON: parsePhotosList(document),
                   withStatus: HttpStatusCode.OK, orSetError: "Internal
error")
        }
        else {
            response.error =
                   NSError(domain: "SwiftBluePic", code: 1,
                           userInfo: [NSLocalizedDescriptionKey:"View not
found"])
        }
        next()
    }
}
```

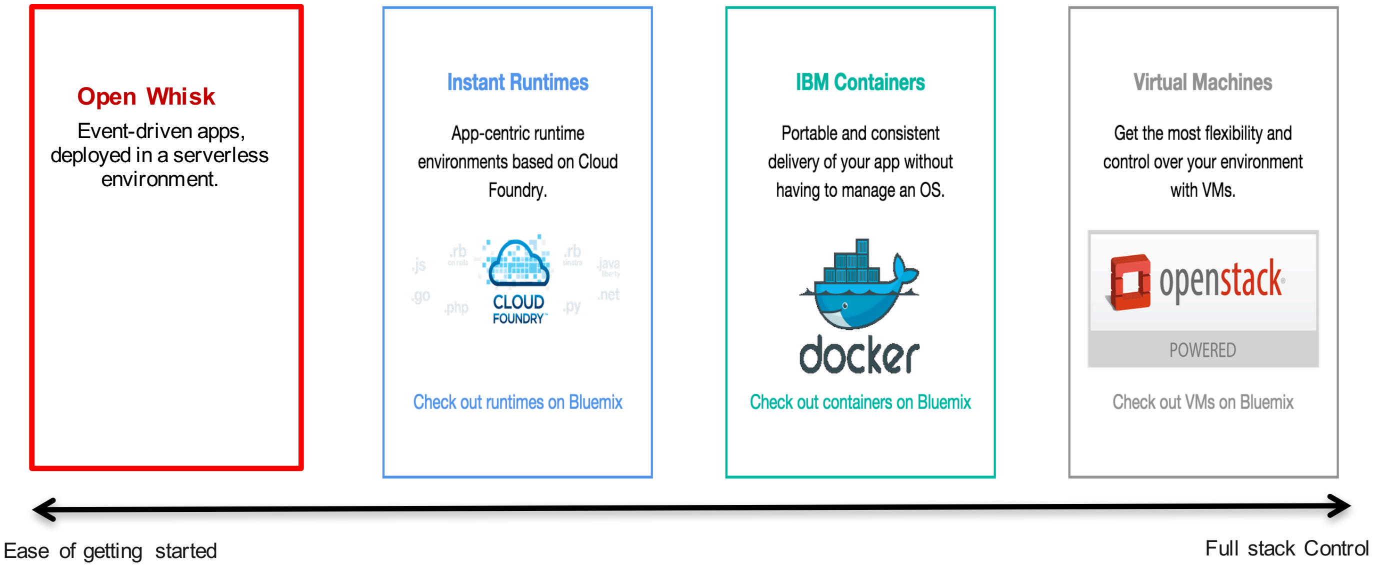
iOS Swift Code (Calling REST API)

Cloud Swift Code (Fetching data from DB)

Bluemix Workloads

Build your apps, your way.

Use a combination of the most prominent open-source compute technologies to power your apps. Then, let Bluemix handle the rest.



Open Whisk

Event-driven apps, deployed in a serverless environment.

Instant Runtimes

App-centric runtime environments based on Cloud Foundry.

Check out runtimes on Bluemix

IBM Containers

Portable and consistent delivery of your app without having to manage an OS.

Check out containers on Bluemix

Virtual Machines

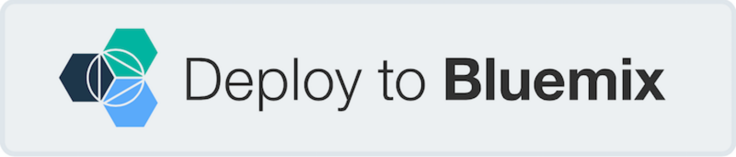
Get the most flexibility and control over your environment with VMs.

Check out VMs on Bluemix

Ease of getting started

Full stack Control

The Swift Runtime
on IBM Bluemix

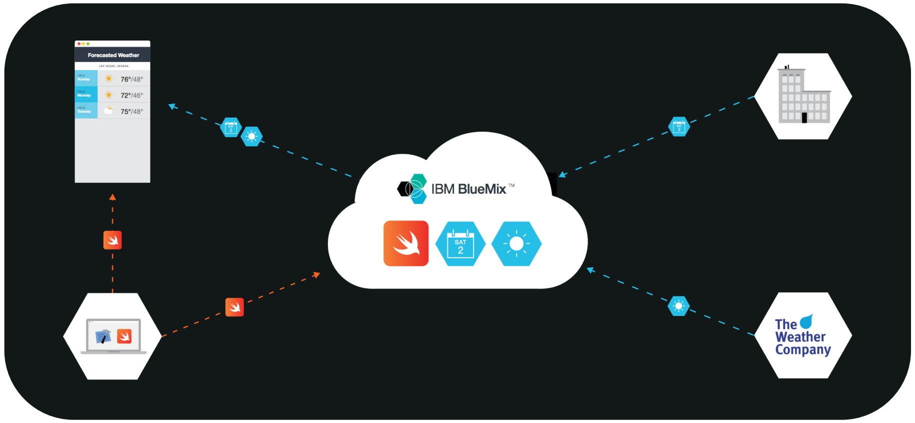


Key Features

- The Swift runtime on IBM Bluemix is powered by the Cloud Foundry buildpack for Swift
- Cloud Foundry buildpacks provide the runtime required to execute your applications on the Bluemix cloud.
- When you push your application, Bluemix automatically detects which buildpack should be used. The buildpack then inspects artifacts in your application to find out what dependencies should be downloaded.
- This is exciting news since you can now push Swift applications to Bluemix that follow the structure and conventions required by the Swift Package Manager.

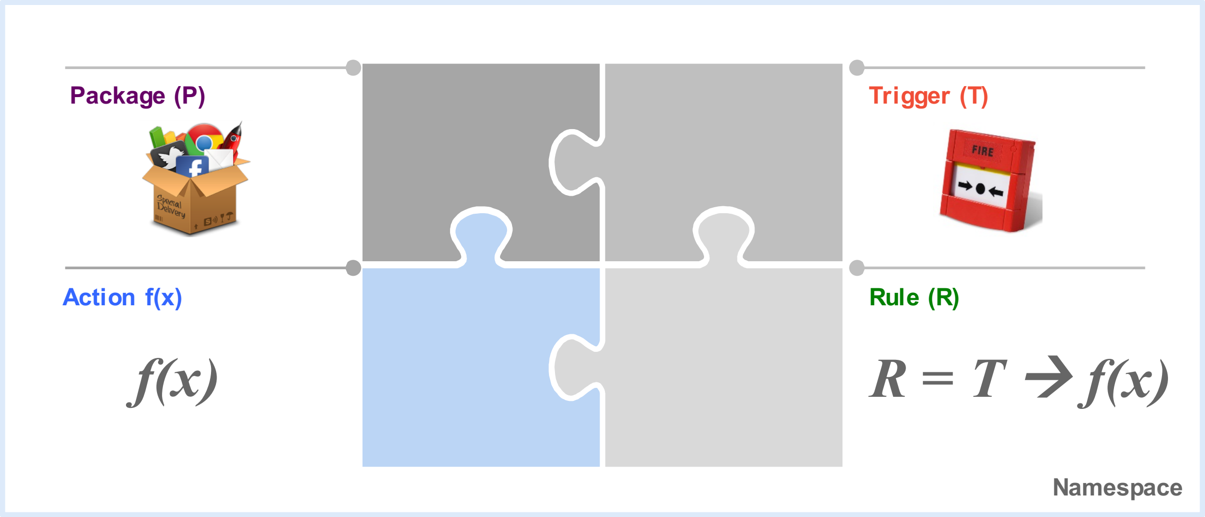
Open Whisk + Swift = iOS Mobile Developer Value

- Mobile developers rely on the cloud for <u>*data and events*</u>
- They rely on multiple cloud APIs to feed their apps
- Typically require others for backend changes



- **Open Whisk + Swift breaks down barrier between Front-end & Backend**
- Seamless language support across front-end & back-end (Swift)
- No worry of provisioning, scaling and monitoring
- Open Whisk “Sequences” allow developers to augment existing backend logic

Open Whisk Programming Model



Package (P)

Trigger (T)

Action f(x)

$f(x)$

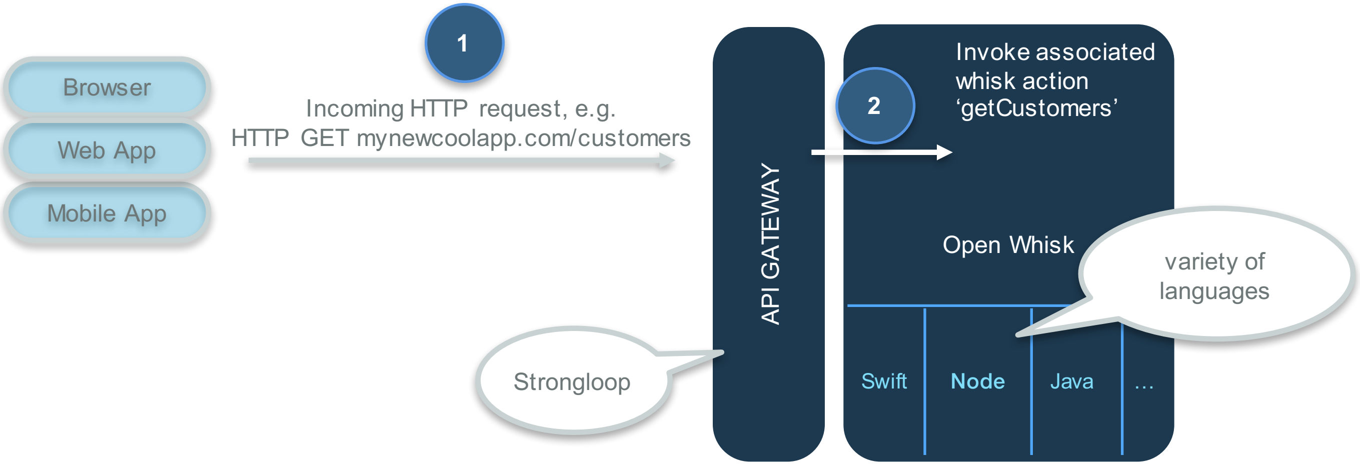
Rule (R)

$R = T \rightarrow f(x)$

Namespace

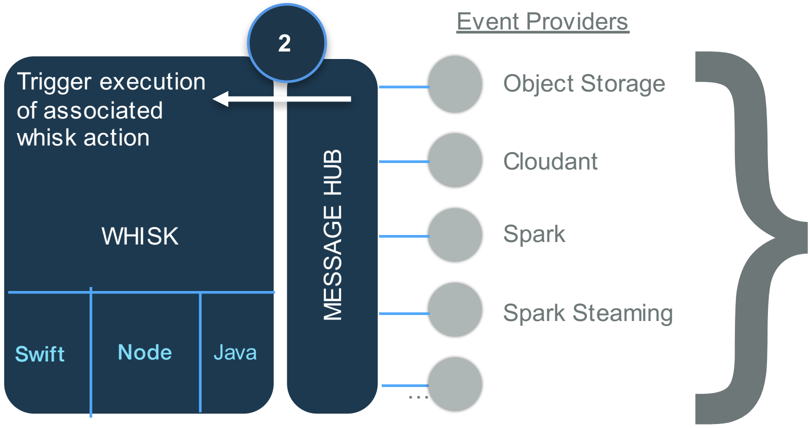
Motivation and Introduction

Open Whisk is a distributed compute service that allows to execute application logic in response to... requests coming in from web or mobile apps…


Browser
Web App
Mobile App
1
Incoming HTTP request, e.g.
HTTP GET mynewcoolapp.com/customers
API GATEWAY
Strongloop
2
Invoke associated whisk action 'getCustomers'
Open Whisk
variety of languages
Swift
Node
Java
…

Motivation and Introduction

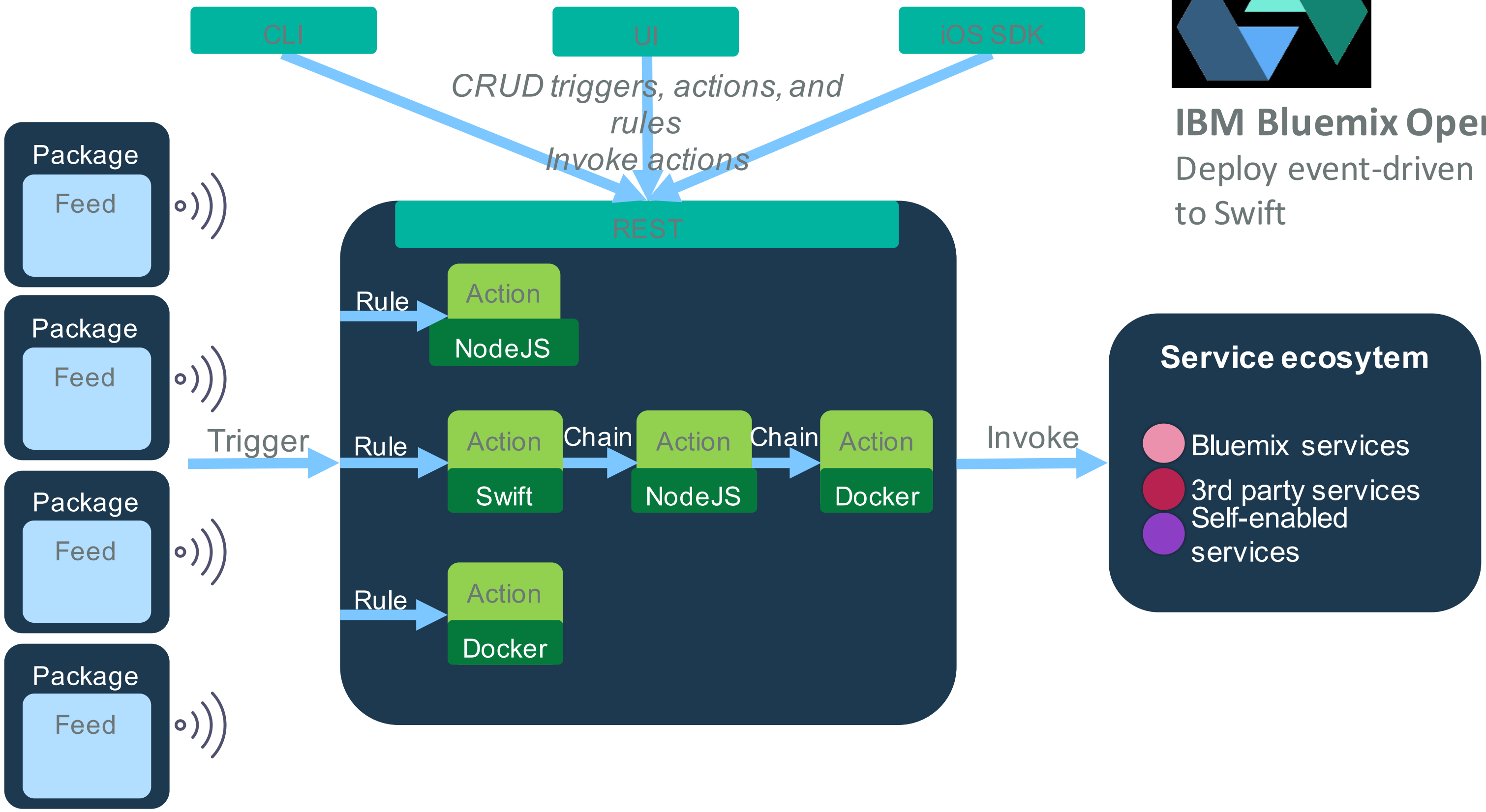
Whisk is a distributed compute service that allows to execute application logic in response to… events other services emit…



1

Data event occurs, e.g.
- CRUD operation on on Object Storage
- CRUD operation in Cloudant
- Spark reports trend detection

OpenWhisk



CLI

UI

iOS SDK

CRUD triggers, actions, and rules
Invoke actions

Package
Feed

Package
Feed

Package
Feed

Package
Feed

Trigger

REST

Rule
Action
NodeJS

Rule
Action
Swift
Chain
Action
NodeJS
Chain
Action
Docker

Rule
Action
Docker

Invoke

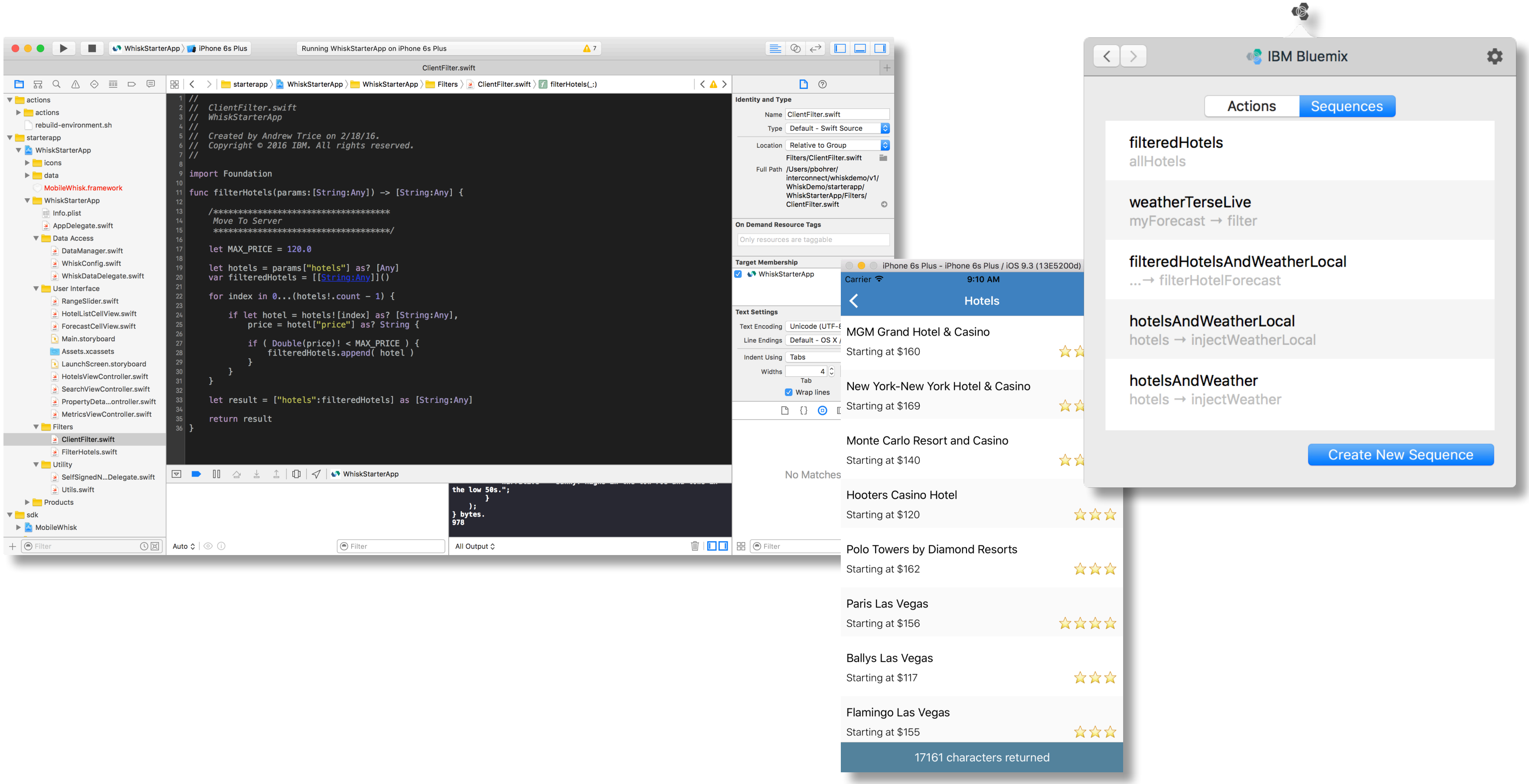
Service ecosytem

● Bluemix services
● 3rd party services
● Self-enabled services

IBM Bluemix OpenWhisk

Deploy event-driven micro-services to Swift

Preview: Xcode to Swift on OpenWhisk Flow



Running WhiskStarterApp on iPhone 6s Plus

ClientFilter.swift

starterapp 〉 WhiskStarterApp 〉 WhiskStarterApp 〉 Filters 〉 ClientFilter.swift 〉 filterHotels(_:)

```
//
//  ClientFilter.swift
//  WhiskStarterApp
//
//  Created by Andrew Trice on 2/18/16.
//  Copyright © 2016 IBM. All rights reserved.
//

import Foundation

func filterHotels(params:[String:Any]) -> [String:Any] {

    /*************************************
     Move To Server
     *************************************/

    let MAX_PRICE = 120.0

    let hotels = params["hotels"] as? [Any]
    var filteredHotels = [[String:Any]]()

    for index in 0...(hotels!.count - 1) {

        if let hotel = hotels![index] as? [String:Any],
            price = hotel["price"] as? String {

            if ( Double(price)! < MAX_PRICE ) {
                filteredHotels.append( hotel )
            }
        }
    }

    let result = ["hotels":filteredHotels] as [String:Any]

    return result
}
```

```
the low 50s.";
        }
    );
} bytes.
978
```

Identity and Type

Name: ClientFilter.swift
Type: Default - Swift Source
Location: Relative to Group
Filters/ClientFilter.swift
Full Path: /Users/pbohrer/interconnect/whiskdemo/v1/WhiskDemo/starterapp/WhiskStarterApp/Filters/ClientFilter.swift

On Demand Resource Tags

Only resources are taggable

Target Membership

☑ WhiskStarterApp

Text Settings

Text Encoding: Unicode (UTF-8)
Line Endings: Default - OS X
Indent Using: Tabs
Widths: 4 Tab
☑ Wrap lines

No Matches

iPhone 6s Plus - iPhone 6s Plus / iOS 9.3 (13E5200d)

Carrier 9:10 AM

Hotels

MGM Grand Hotel & Casino	Starting at $160
New York-New York Hotel & Casino	Starting at $169
Monte Carlo Resort and Casino	Starting at $140
Hooters Casino Hotel	Starting at $120
Polo Towers by Diamond Resorts	Starting at $162
Paris Las Vegas	Starting at $156
Ballys Las Vegas	Starting at $117
Flamingo Las Vegas	Starting at $155

17161 characters returned

IBM Bluemix

Actions | Sequences

filteredHotels
allHotels

weatherTerseLive
myForecast → filter

filteredHotelsAndWeatherLocal
...→ filterHotelForecast

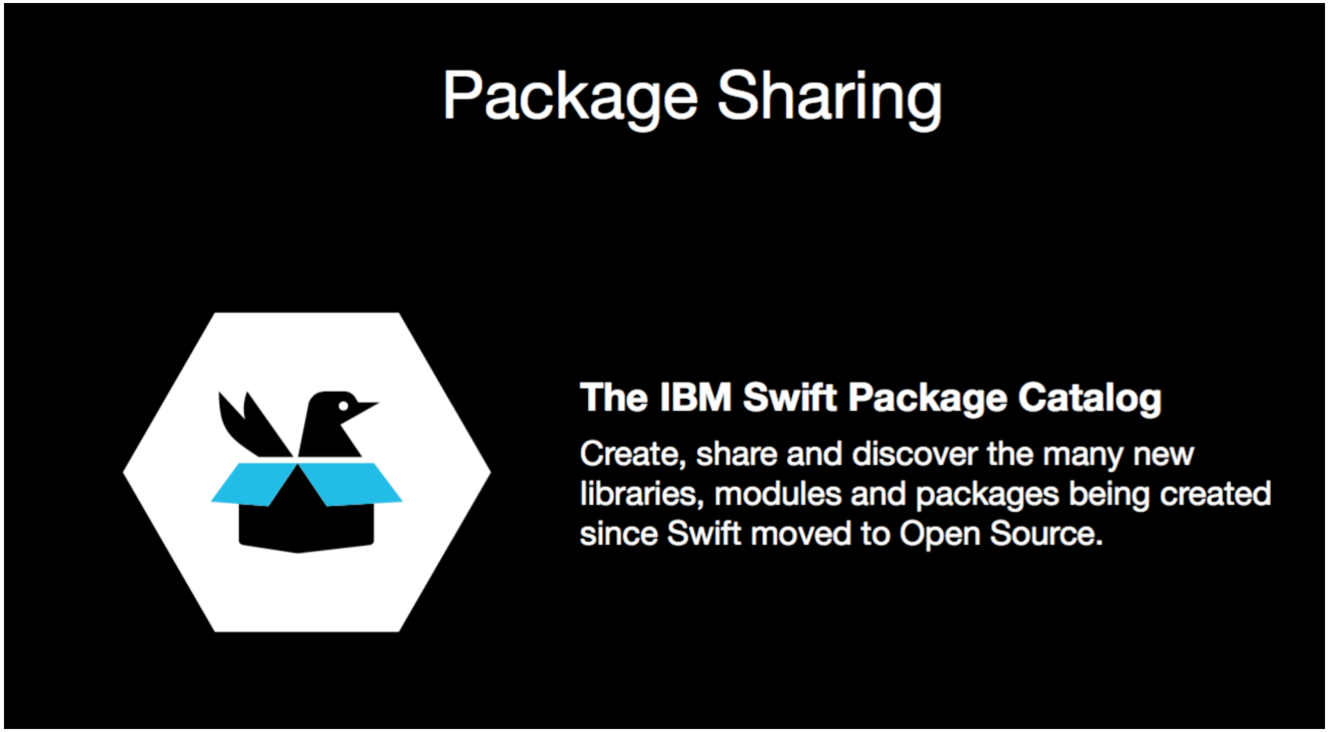
hotelsAndWeatherLocal
hotels → injectWeatherLocal

hotelsAndWeather
hotels → injectWeather

Create New Sequence

Developer Resources

Package Sharing
Swift Sandbox
Developer Portals



Package Sharing

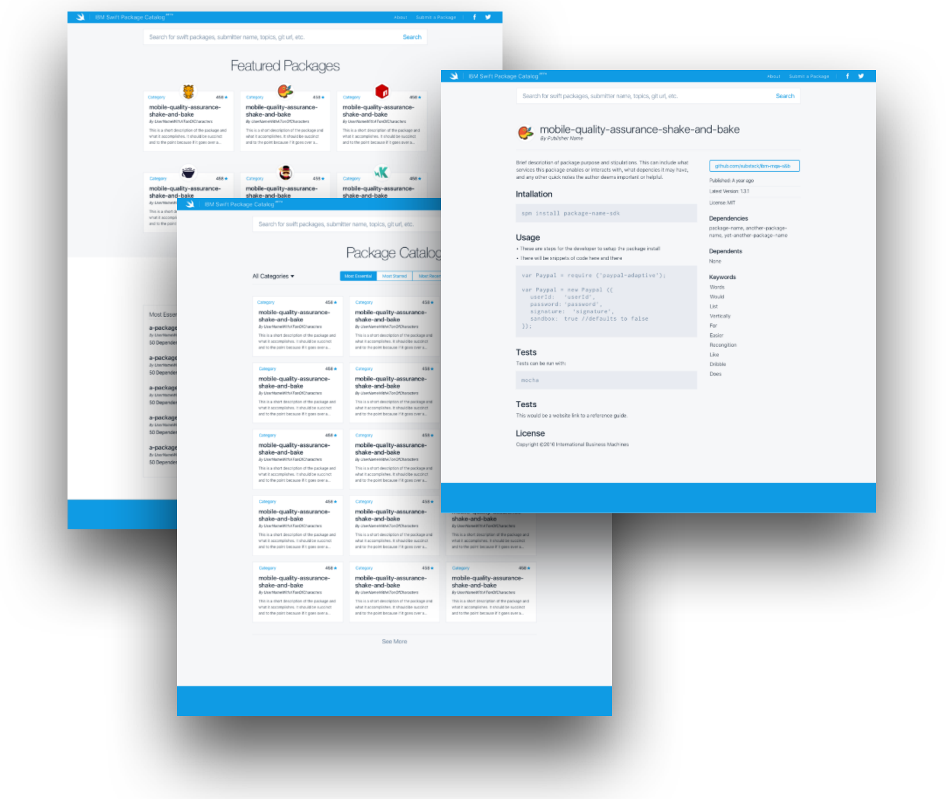
The IBM Swift Package Catalog

Create, share and discover the many new libraries, modules and packages being created since Swift moved to Open Source.

Swift.org Swift Package Manager

- The newly released package manager allows developers to specify build targets and dependencies for their libraries and/or applications
- The Swift Package Manager reads these specification files (Package.swift) and pulls down any dependencies and builds those along with building the targets that you specify
- The is new for Swift 3.0
- Based on a decentralized design, so there is no one clearinghouse for all packages. Typically end up pointing to git repositories in Github.
- Early days but many projects are already popping up with support for being built with the Swift Package Manager

The IBM Swift Package Catalog



IBM Swift Package Catalog

About | Submit a Package

Search for swift packages, submitter name, topics, git url, etc. Search

Featured Packages

Package Catalog

mobile-quality-assurance-shake-and-bake

By Publisher Name

Brief description of package purpose and stipulations. This can include what services this package enables or interacts with, what depencies it may have, and any other quick notes the author deems important or helpful.

Intallation

```
spm install package-name-sdk
```

Usage

- These are steps for the developer to setup the package install
- There will be snippets of code here and there

```
var Paypal = require ('paypal-adaptive');

var Paypal = new Paypal ({
  userId:   'userId',
  password: 'password',
  signature:  'signature',
  sandbox:  true //defaults to false
});
```

Tests

Tests can be run with:

```
mocha
```

Tests

This would be a website link to a reference guide.

License

Copyright ©2016 International Business Machines

github.com/substack/ibm-mqa-s&b

Published: A year ago

Latest Version: 1.3.1

License: MIT

Dependencies

package-name, another-package-name, yet-another-package-name

Dependents

None

Keywords

Words

Would

List

Vertically

For

Easier

Recongition

Like

Dribble

Does

What is it?

Create, share and discover the many new libraries, modules and packages being created since Swift moved to Open Source.

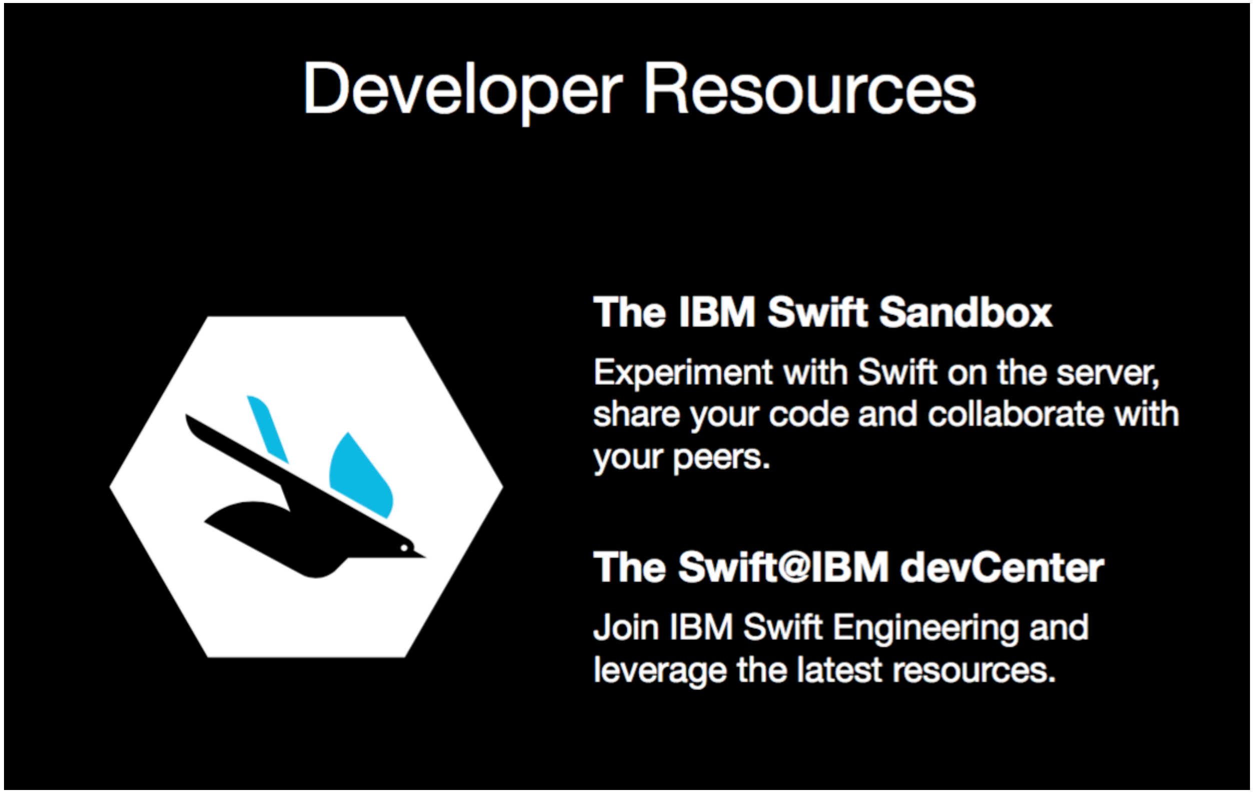
Why is this cool?

Brings the benefits of a catalog to the community enabling the rapid sharing and discovery of new submissions.

Developer Benefits ?

Greatly reduce the effort in finding and integrating new code into your Swift applications

http://swiftpkgs.ng.bluemix.net



Developer Resources

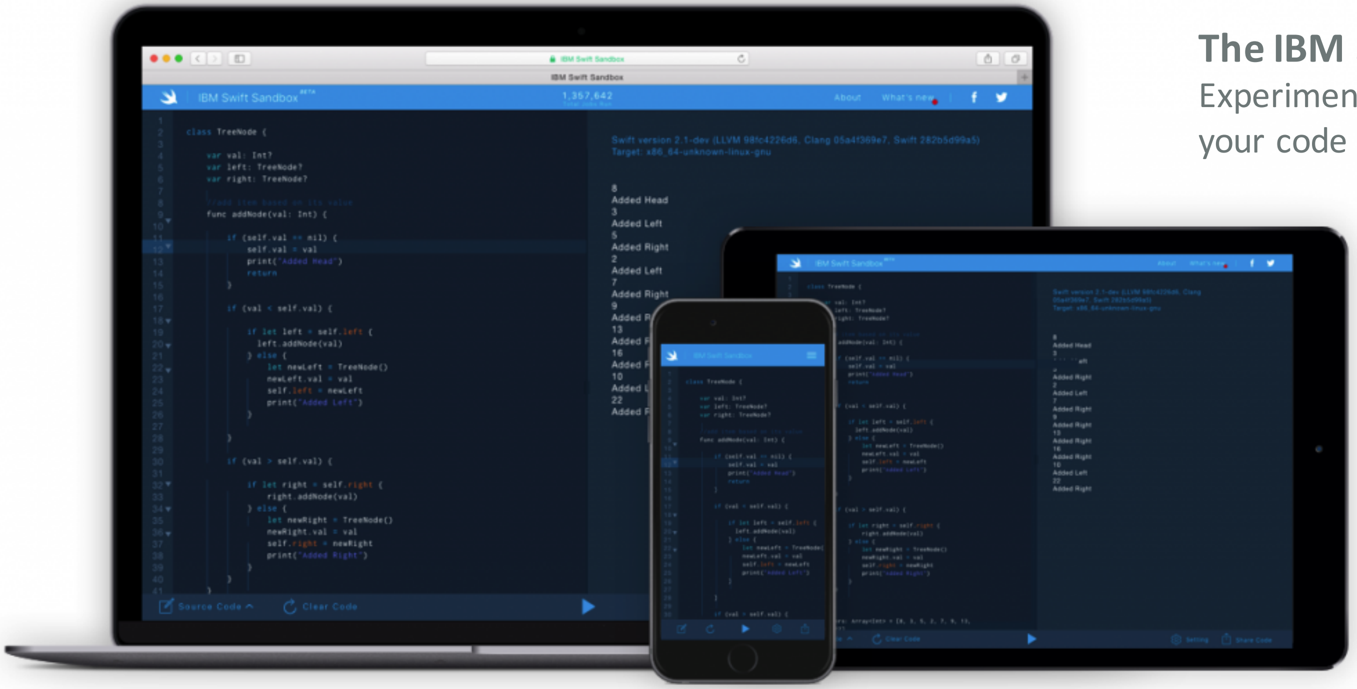
The IBM Swift Sandbox

Experiment with Swift on the server, share your code and collaborate with your peers.

The Swift@IBM devCenter

Join IBM Swift Engineering and leverage the latest resources.

IBM Swift Sandbox



```
class TreeNode {

    var val: Int?
    var left: TreeNode?
    var right: TreeNode?

    //add item based on its value
    func addNode(val: Int) {

        if (self.val == nil) {
            self.val = val
            print("Added Head")
            return
        }

        if (val < self.val) {

            if let left = self.left {
              left.addNode(val)
            } else {
                let newLeft = TreeNode()
                newLeft.val = val
                self.left = newLeft
                print("Added Left")
            }

        }

        if (val > self.val) {

            if let right = self.right {
                right.addNode(val)
            } else {
                let newRight = TreeNode()
                newRight.val = val
                self.right = newRight
                print("Added Right")
            }
        }
    }
```

```
Swift version 2.1-dev (LLVM 98fc4226d6, Clang 05a4f369e7, Swift 282b5d99a5)
Target: x86_64-unknown-linux-gnu


8
Added Head
3
Added Left
5
Added Right
2
Added Left
7
Added Right
9
Added R
13
Added F
16
Added F
10
Added L
22
Added F
```

The IBM Swift Sandbox

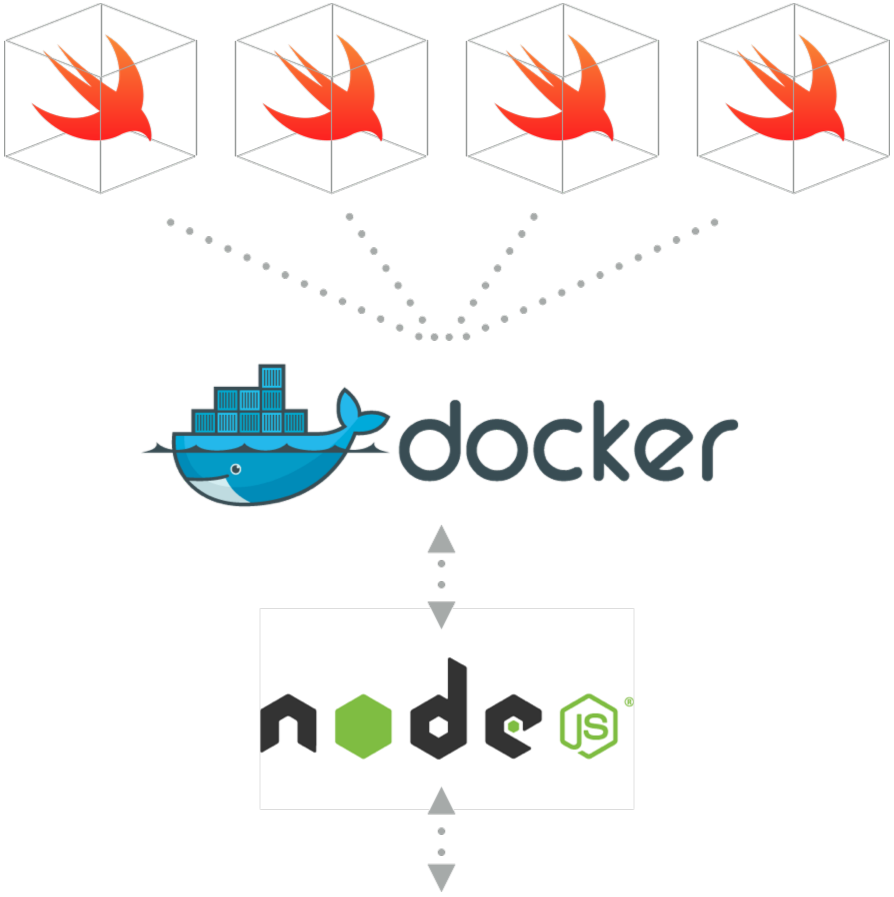
Experiment with Swift on the server, share your code and collaborate with your peers

Features

- Mobile UI & Auto Saving Draft
- Code Snapshots & Sharing, UI Themes, Social
- Social Sharing

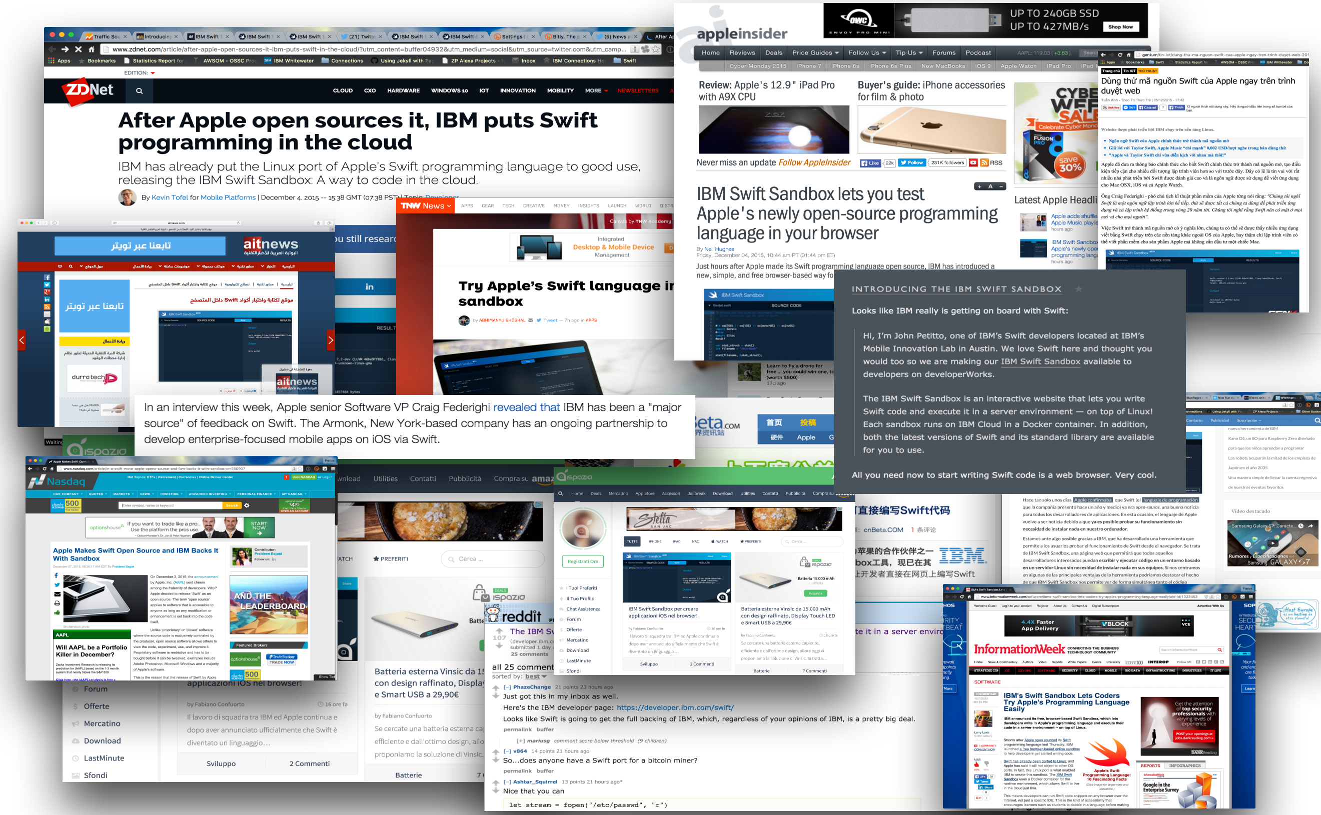
http://swiftlang.ng.bluemix.net

Docker Instances Launched for every Run Request in the Web Console



To client…

Global Excitement around Swift



After Apple open sources it, IBM puts Swift programming in the cloud

IBM has already put the Linux port of Apple's Swift programming language to good use, releasing the IBM Swift Sandbox: A way to code in the cloud.

By Kevin Tofel for Mobile Platforms | December 4, 2015 -- 15:38 GMT (07:38 PST)

In an interview this week, Apple senior Software VP Craig Federighi revealed that IBM has been a "major source" of feedback on Swift. The Armonk, New York-based company has an ongoing partnership to develop enterprise-focused mobile apps on iOS via Swift.

IBM Swift Sandbox lets you test Apple's newly open-source programming language in your browser

Try Apple's Swift language in sandbox

INTRODUCING THE IBM SWIFT SANDBOX

Looks like IBM really is getting on board with Swift:

Hi, I'm John Petitto, one of IBM's Swift developers located at IBM's Mobile Innovation Lab in Austin. We love Swift here and thought you would too so we are making our IBM Swift Sandbox available to developers on developerWorks.

The IBM Swift Sandbox is an interactive website that lets you write Swift code and execute it in a server environment — on top of Linux! Each sandbox runs on IBM Cloud in a Docker container. In addition, both the latest versions of Swift and its standard library are available for you to use.

All you need now to start writing Swift code is a web browser. Very cool.

Dùng thử mã nguồn Swift của Apple ngay trên trình duyệt web

Apple Makes Swift Open Source and IBM Backs It With Sandbox

IBM's Swift Sandbox Lets Coders Try Apple's Programming Language Easily

Just got this in my inbox as well.

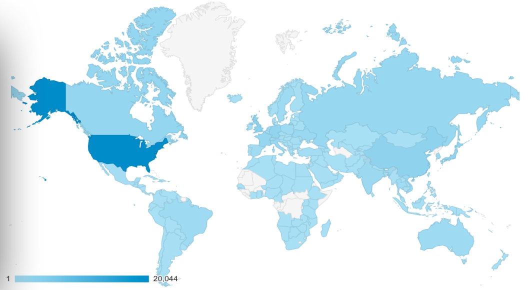
Here's the IBM developer page: https://developer.ibm.com/swift/

Looks like Swift is going to get the full backing of IBM, which, regardless of your opinions of IBM, which is a pretty big deal.

So...does anyone have a Swift port for a bitcoin miner?

Nice that you can

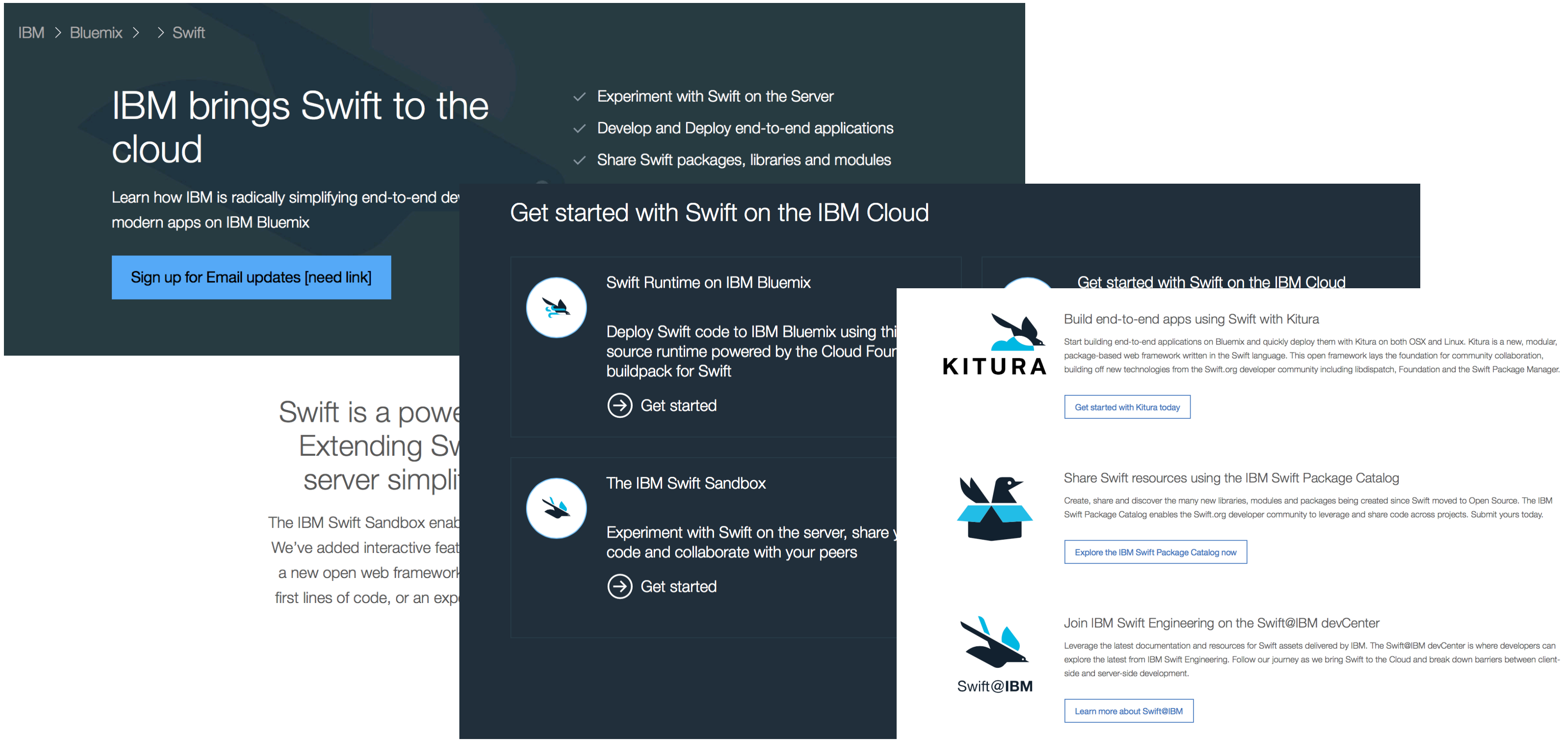
```
let stream = fopen("/etc/passwd", "r")
```



178 Countries
US Traffic ~25%

Translations across over 450 tech blogs

Swift on the IBM Cloud



IBM > Bluemix > > Swift

IBM brings Swift to the cloud

Learn how IBM is radically simplifying end-to-end de
modern apps on IBM Bluemix

Sign up for Email updates [need link]

✓ Experiment with Swift on the Server
✓ Develop and Deploy end-to-end applications
✓ Share Swift packages, libraries and modules

Swift is a powe
Extending Sw
server simpli

The IBM Swift Sandbox enab
We've added interactive feat
a new open web framework
first lines of code, or an exp

Get started with Swift on the IBM Cloud

Swift Runtime on IBM Bluemix

Deploy Swift code to IBM Bluemix using thi
source runtime powered by the Cloud Foun
buildpack for Swift

→ Get started

The IBM Swift Sandbox

Experiment with Swift on the server, share y
code and collaborate with your peers

→ Get started

Get started with Swift on the IBM Cloud

KITURA

Build end-to-end apps using Swift with Kitura

Start building end-to-end applications on Bluemix and quickly deploy them with Kitura on both OSX and Linux. Kitura is a new, modular, package-based web framework written in the Swift language. This open framework lays the foundation for community collaboration, building off new technologies from the Swift.org developer community including libdispatch, Foundation and the Swift Package Manager.

Get started with Kitura today

Share Swift resources using the IBM Swift Package Catalog

Create, share and discover the many new libraries, modules and packages being created since Swift moved to Open Source. The IBM Swift Package Catalog enables the Swift.org developer community to leverage and share code across projects. Submit yours today.

Explore the IBM Swift Package Catalog now

Swift@IBM

Join IBM Swift Engineering on the Swift@IBM devCenter

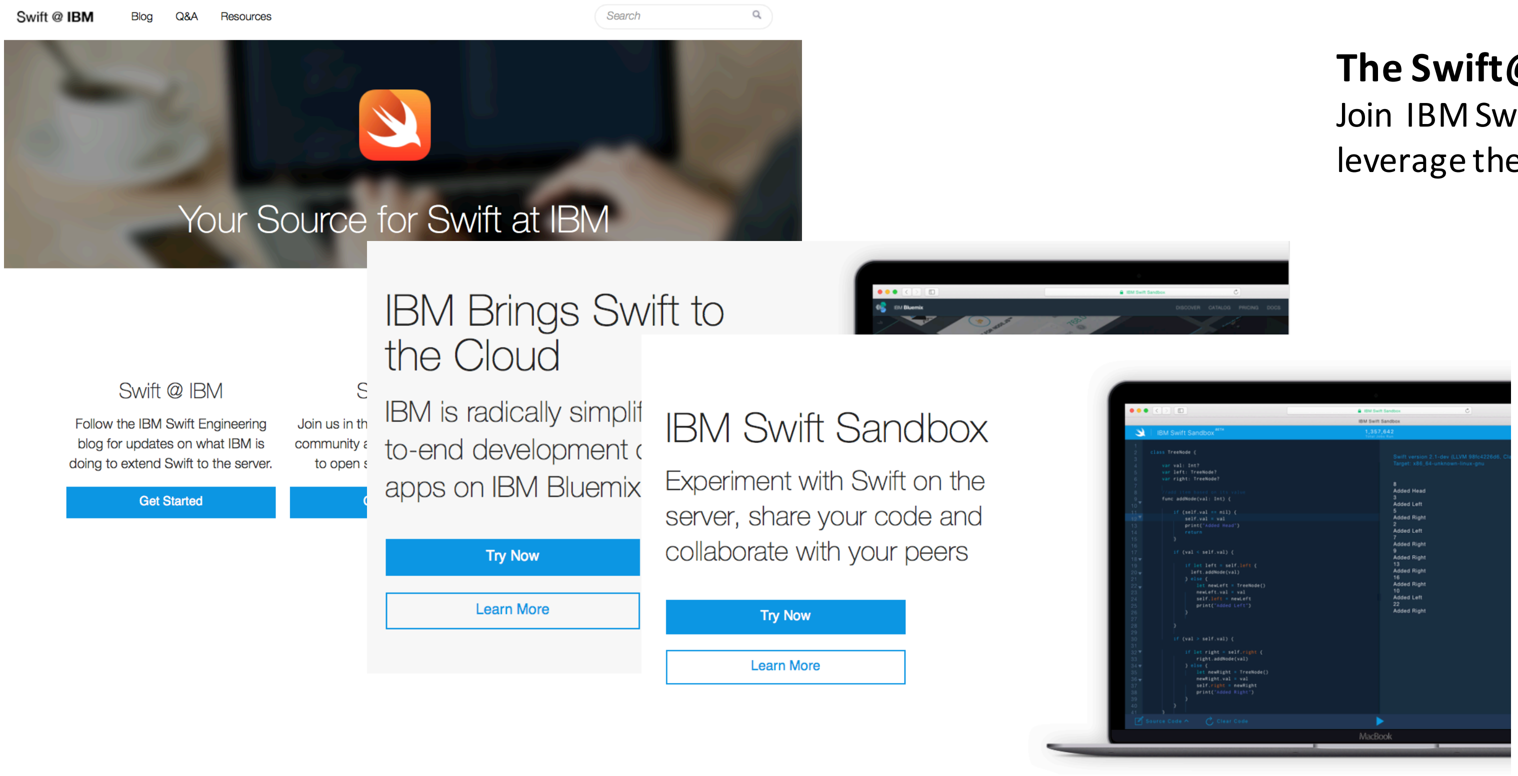
Leverage the latest documentation and resources for Swift assets delivered by IBM. The Swift@IBM devCenter is where developers can explore the latest from IBM Swift Engineering. Follow our journey as we bring Swift to the Cloud and break down barriers between client-side and server-side development.

Learn more about Swift@IBM

Swift@IBM - Developer Resources

| Swift@IBM

The Swift@IBM devCenter
Join IBM Swift Engineering and leverage the latest resources



Your Source for Swift at IBM

Swift @ IBM

Follow the IBM Swift Engineering blog for updates on what IBM is doing to extend Swift to the server.

S

Join us in th
community
to open s

IBM Brings Swift to the Cloud

IBM is radically simplif
to-end development
apps on IBM Bluemix

IBM Swift Sandbox

Experiment with Swift on the server, share your code and collaborate with your peers

https://developer.ibm.com/swift/

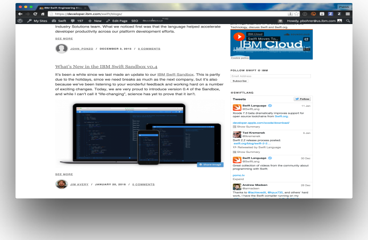
Technical Blog Threads on Swift@IBM

Swift (General)

- Why I'm Excited about Swift (12/3)
- Running Swift within Docker (12/15)
- Introducing the (beta) IBM Watson iOS SDK! (12/18)

Swift Sandbox

- Introducing Swift Sandbox (12/3)
- Hello Swift! IBM Swift Sandbox Day 1 Wrapup (12/5)
- #HourofCode: Learn Swift in three easy steps today! (12/8)
- Introduction to Swift Tutorial using the IBM Swift Sandbox (12/8)
- What's new in the IBM Swift Sandbox v0.3 (12/21)
- Exploring Swift on Linux (12/28)
- What's new in the IBM Swift Sandbox v0.4 (1/20)


IBM Swift Engineering Blog
https://developer.ibm.com/swift/blogs/
My Sites
Swift
197
New
Edit Page
SEO
Howdy, pbohrer@us.ibm.com
Patrick

Industry Solutions team. What we noticed first was that the language helped accelerate developer productivity across our platform development efforts.

SEE MORE

JOHN_PONZO / DECEMBER 3, 2015 / 0 COMMENTS

What's New in the IBM Swift Sandbox v0.4

It's been a while since we last made an update to our IBM Swift Sandbox. This is partly due to the holidays, since we need breaks as much as the next company, but it's also because we've been listening to your wonderful feedback and working hard on a number of exciting changes. Today, we are very proud to introduce version 0.4 of the Sandbox, and while I can't call it "life-changing", science has yet to prove that it isn't.

SEE MORE

JIM AVERY / JANUARY 20, 2016 / 0 COMMENTS

Technology, discuss Swift and Swift.org.

FOLLOW SWIFT @ IBM

Email Address

Subscribe

@SWIFTLANG

Tweets

Follow

Swift Language @SwiftLang 11 Jan
Xcode 7.3 beta dramatically improves support for open source toolchains from Swift.org.
developer.apple.com/xcode/download/
Show Summary

Ted Kremenek @tkremenek 5 Jan
Swift 2.2 release process posted:
swift.org/blog/swift-2-2...
Retweeted by Swift Language
Show Summary

Swift Language @SwiftLang 30 Dec
Great collection of videos from the community about programming with Swift:
pomo.tv
Expand

Andrew Madsen @armadsen 26 Dec
Thanks to @lachievedit, @hpux735, and others' hard work, I have the Swift compiler running on my

Swift (General)

- Swift on POWER Linux (2/1)
- Seven Swift Snares & How to Avoid Them (1/27)

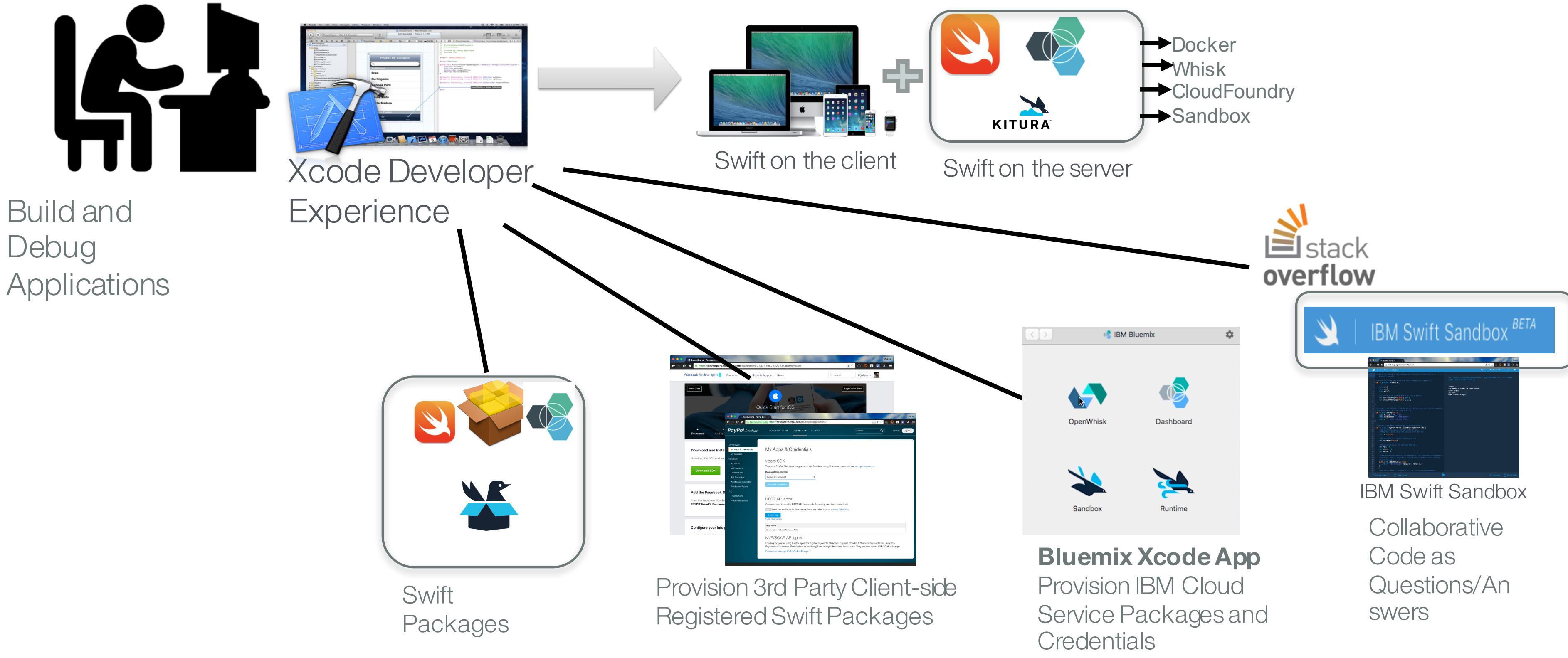
Interconnect 2016

- Build End-to-End Cloud Apps using Swift with Kitura (2/21)
- Introducing the Swift Package Catalog (2/21)
- Talking about Swift Concurrency on Linux (2/21)
- Explore the IBM Swift Sandbox 1-2-3 (2/21)
- Using the Cloud Foundry Buildpack for Swift on Bluemix (2/21)
- 10 Steps To Running a Swift App in an IBM Container (2/21)
- Build End-to-End Cloud Apps using Swift with Kitura (2/21)

Drumbeat of Blogs/Announcements from IBM Swift Engineering Community

https://developer.ibm.com/swift/blogs

End-to-end developer experience



Build and Debug Applications

Xcode Developer Experience

Swift on the client

Swift on the server

- Docker
- Whisk
- CloudFoundry
- Sandbox

Swift Packages

Provision 3rd Party Client-side Registered Swift Packages

Bluemix Xcode App
Provision IBM Cloud Service Packages and Credentials

IBM Swift Sandbox

Collaborative Code as Questions/Answers

Innovators & Early Adopters Needed

Get involved now at
swift.org
&
developer.ibm.com/swift

Swift@IBM

QCon LONDON

Thank you for listening and Thank you QCON London for a Great Conference!

Patrick Bohrer
@pbohrer

Chris Bailey
@Chris__Bailey

Get Started Today!
developer.ibm.com/swift