Going Server-side with Swift Open Source

Session 415

Philippe Hausler  Frameworks Engineer
John Ponzo  IBM MobileFirst CTO
Patrick Bohrer  Swift Technical Lead, IBM
Agenda

Why Server-side Swift
Architecture
What’s Included
Examples
Contribution
Server-side Swift

Benefits
Server-side Swift

Benefits

Same code runs in both places
Server-side Swift

Benefits

Same code runs in both places
Reduce development time by sharing code
Server-side Swift

Benefits

Same code runs in both places
Reduce development time by sharing code
Leverage great APIs on the serve
Architecture

System Libraries/Frameworks
Darwin

System Libraries
Linux
Architecture

System Libraries/Frameworks

Darwin

Standard Library

System Libraries

Linux
Architecture

System Libraries/Frameworks

1. Standard Library
2. System Libraries/Frameworks
3. Overlay

Darwin

Core Libraries

1. Standard Library
2. System Libraries

Linux
What’s In

Included projects

Welcome to Swift.org

Swift is now open source!

We are excited by this new chapter in the story of Swift. After Apple unveiled the Swift programming language, it quickly became one of the fastest growing languages in history. Swift makes it easy to write software that is incredibly fast and safe by design. Now that Swift is open source, you can help make the best general purpose programming language available everywhere.

For students, learning Swift has been a great introduction to modern programming concepts and best practices. And because it is now open, their Swift skills will be able to be applied to an even broader range of platforms, from mobile devices to the desktop to the cloud.

Welcome to the Swift community. Together we are working to build a better programming language for everyone.

— The Swift Team
What’s In

Included projects

Linux and Mac Platform support
What’s In
Included projects

Linux and Mac Platform support
Standard Library
What’s In

Included projects

Linux and Mac Platform support

Standard Library

Foundation, Dispatch, and XCTest
What’s In

Included projects

- Linux and Mac Platform support
- Standard Library
- Foundation, Dispatch, and XCTest
- Compiler Command Line Tools
Your Code Goes Here

Developer opportunities
Your Code Goes Here
Developer opportunities
New platforms
Developer opportunities

New platforms
User interface
Your Code Goes Here

Developer opportunities

New platforms
User interface
Deployment
Your Code Goes Here

Developer opportunities

New platforms
User interface
Deployment
Your applications
Real-world Server-side Swift

John Ponzo
IBM MobileFirst CTO
Swift @ IBM

IBM MobileFirst for iOS
Swift @ IBM

Bringing Swift to the server
Swift @ IBM

Bringing Swift to the server

Community enablement

• Swift Sandbox
• Swift Package Catalog
Swift @ IBM

Bringing Swift to the server

Community enablement
• Swift Sandbox
• Swift Package Catalog

Consistent developer experience
• Kitura Web Framework
• Core swift.org contributions
Swift @ IBM
Bringing Swift to the server

Community enablement
• Swift Sandbox
• Swift Package Catalog

Consistent developer experience
• Kitura Web Framework
• Core swift.org contributions

Cloud enablement
• IBM Cloud Deployment
• IBM Cloud Tools for Swift
Community Enablement

IBM Swift Sandbox

```swift
// Sample that demonstrates classes and protocols in Swift using a baseball example.

// Define variables and functions that classes must conform to
public protocol Hittable {
    func hit()
    func out()
    func walk()
}

// A variable of type Double with only a getter
var battingAverage: Double {
    get
}

var onBasePercentage: Double {
    get
}

// An enum lists defines discrete values. A raw value can also be provided
enum PlayerType: String {
    case AllStar = "All Star"
    case BenchWarmer = "Bench Warmer"
    case MVP = "Most Valuable Player"
}

// Player class that conforms to Hittable protocol
public class Player: Hittable, CustomStringConvertible {
    // Immutable variables are defined with let...
    // They don't have to be initialized when defined.
    let name: String
}
```

http://swiftlang.ng.bluemix.net
Community Enablement

IBM Swift Sandbox

http://swiftlang.ng.bluemix.net
Community Enablement
IBM Swift Sandbox

http://swiftlang.ng.bluemix.net
Community Enablement
IBM Swift Sandbox

Cross-platform Interest
http://swiftlang.ng.bluemix.net

iOS/ Mac 44%
Other 56%
Community Enablement
IBM Swift Sandbox

Worldwide Interest
Community Enablement
IBM Swift Package Catalog

http://swiftpkgs.ng.bluemix.net
Community Enablement
IBM Swift Package Catalog

http://swiftpkg.ng.bluemix.net
Community Enablement
IBM Swift Package Catalog

1,500+
IBM Swift Package Manager Packages

http://swiftpkg.ng.bluemix.net
Consistent Developer Experience
Kitura web framework
Consistent Developer Experience
Kitura web framework

Open sourced in February 2016
Consistent Developer Experience
Kitura web framework

Open sourced in February 2016
Modular package-based web framework
Consistent Developer Experience
Kitura web framework

Open sourced in February 2016
Modular package-based web framework
Leverages libdispatch and Foundation
Open sourced in February 2016
Modular package-based web framework
Leverages libdispatch and Foundation
Driving IBM’s contributions into swift.org

Consistent Developer Experience
Kitura web framework
Consistent Developer Experience
Kitura web framework

Open sourced in February 2016
Modular package-based web framework
Leverages libdispatch and Foundation
Driving IBM’s contributions into swift.org
Used at IBM for Swift server-side applications
Consistent Developer Experience
Kitura web framework

Open sourced in February 2016
Modular package-based web framework
Leverages libdispatch and Foundation
Driving IBM’s contributions into swift.org
Used at IBM for Swift server-side applications
Available at github.com/ibm-swift/kitura

Consistent Developer Experience
Kitura web framework
Cloud Enablement
IBM Cloud Deployment
Cloud Enablement
IBM Cloud Deployment

![Cloud Enablement logo](image1)

![IBM Cloud Deployment icon](image2)
Cloud Enablement
IBM Cloud Tools for Swift

http://cloudtools.bluemix.net
Let’s Jump In!

Patrick Bohrer
Swift Technical Lead, IBM
Demo
Simple, Swift Web Server running on Linux
End-to-end Swift

BluePic
End-to-end Swift

BluePic

Sample photo sharing app
End-to-end Swift

BluePic

Sample photo sharing app
Rich client interface written in Swift
End-to-end Swift

BluePic

Sample photo sharing app

Rich client interface written in Swift

Swift Server built with Kitura, IBM Cloud, Watson,
End-to-end Swift

BluePic

Sample photo sharing app
Rich client interface written in Swift
Swift Server built with Kitura, IBM Cloud, Watson, and OpenWhisk
End-to-end Swift

BluePic

Sample photo sharing app

Rich client interface written in Swift

Swift Server built with Kitura, IBM Cloud, Watson, and OpenWhisk

Available at github.com/IBM-Swift/BluePic
End-to-end Swift
BluePic design pattern
End-to-end Swift
BluePic design pattern

iOS App

User Interaction
End-to-end Swift
BluePic design pattern

iOS App
- Views
- Controllers
- Model
  - Request
  - Response

Application Server
- Routing
- Service Logic

User Interaction

Client and Service Integration
End-to-end Swift
BluePic design pattern

iOS App
- Views
- Controllers
- Model
  - Request
  - Response

Application Server
- Routing
- Service Logic

Other Services
- Cloudant
- Object Storage
- Watson Image Recognition

User Interaction
Client and Service Integration
Core Cloud Services
End-to-end Swift
BluePic design pattern

Users
Photos
Image Insights = Tags
Push Notifications
Application Server

Client and Service Integration

Meta Data
Mountain, Sunset, River
Other Services

Core Cloud Services
End-to-end Swift

Application tiers

Clients

Application Server

Cloud

Meta Data

Mountain, Sunset, River

Other Services
End-to-end Swift
BluePic development setup

Develop and Debug
macOS

Application Server

Cloud

Other Services
- Meta Data
- Mountain, Sunset, River
End-to-end Swift
BluePic development setup

Develop and Debug
macOS

Develop and Debug

Other Services

Cloud

Meta Data

Mountain, Sunset, River
End-to-end Swift
BluePic development setup

IBM Cloud Tools for Swift (Mac App)
IBM Cloud Tools for Swift
End-to-end Swift Development

Develop and Debug
Develop and Debug

macOS + Cloud

Meta Data
Mountain, Sunset, River

Other Services
Demo
End-to-end Swift Application
Get Started Today
IBM Swift tools and tech

Swift@IBM
Breaking down barriers between client and server all over the world

New: Try the new IBM Cloud Tools for Swift (Beta)
For developers interested in creating Swift applications that span both client and server-side code, the newest tools from IBM can help simplify the management and deployment of server-side assets. Extend your mobile apps and connect them to IBM Bluemix Cloud services with a local development environment that complements the productive Xcode environment with the IBM Cloud Tools for Swift (beta) now available on IBM Bluemix.

http://developer.ibm.com/swift
Real-world Server-side Swift
IBM review

Open source contributions
Helping to grow the community
Building tools for a consistent world
Contribution

Chipping in

Implementations

API Design, guidelines and language Features

Your Packages
More Information

## Related Sessions

<table>
<thead>
<tr>
<th>Session</th>
<th>Location</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swift API Design Guidelines</td>
<td>Presidio</td>
<td>Tuesday 10:00AM</td>
</tr>
<tr>
<td>What's New in Foundation for Swift</td>
<td>Mission</td>
<td>Tuesday 4:00PM</td>
</tr>
<tr>
<td>Understanding Swift Performance</td>
<td>Mission</td>
<td>Friday 11:00AM</td>
</tr>
<tr>
<td>Concurrent Programming with GCD in Swift 3</td>
<td>Pacific Heights</td>
<td>Friday 4:00PM</td>
</tr>
</tbody>
</table>
## Labs

<table>
<thead>
<tr>
<th></th>
<th>Developer Tools Lab A</th>
<th>Friday 9:00AM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Swift Open Hours</strong></td>
<td>Frameworks Lab A</td>
<td>Friday 1:00PM</td>
</tr>
<tr>
<td><strong>Cocoa Lab</strong></td>
<td>Frameworks Lab D</td>
<td>Friday 5:00PM</td>
</tr>
<tr>
<td><strong>GCD Lab</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---