Leo George Dion

leogdion@brightdigit.com

brightdigit.com

517.325.3123

Experience

BrightDigit *Server-Side Swift & AWS Cloud Development*

2012-

Official Web Site - https://brightdigit.com

App Store - https://itunes.apple.com/bz/developer/brightdigit/id951071114

YouTube Channel - https://www.youtube.com/channel/UCnl3jF6GHAvlj1MZyb6Ka9g

Podcast - https://www.empowerapps.show

Principal and CEO

Since 2012, BrightDigit specializes in full-stack Swift development, including server-side Swift applications and AWS cloud infrastructure. Currently continuing indie and contract work while employed full-time at Meijer.

- **Server-Side Swift Development**: Architect and maintain backend services and APIs using Vapor and Hummingbird framework
- **AWS Cloud Infrastructure**: Design and implement cloud solutions using AWS services including Lambda, and DynamoDB
- **Cross-Stack Development**: Lead development across iOS, macOS, watchOS, web, and server platforms using Swift, JavaScript, and modern web technologies
- **Remote Team Leadership**: Lead distributed development teams and collaborate with clients across multiple time zones
- **Technical Evangelism**: Present at conferences and produce educational content on server-side Swift and cloud architecture best practices

Meijer *Senior iOS Engineer & Technical Lead*

December 2024-

Modernizing Retail iOS Application

- **Technical Leadership**: Guide team migration from Objective-C/UIKit to modern Swift/SwiftUI architecture

- **Cross-Department Collaboration**: Work with business stakeholders to translate requirements into technical specifications
- **Mentorship**: Train development team on modern iOS patterns and server-side integration best practices

Clients

gBeat *CTO*

Full-Stack Platform Architecture for Multi-Platform Applications

- **Server-Side Swift Leadership**: Architected and led development of scalable backend services using Vapor framework for 10-person distributed team
- **AWS Cloud Infrastructure**: Designed and implemented cloud-based solutions with automated deployment pipelines and continuous integration
- **Cross-Platform API Design**: Built REST APIs and web services supporting iOS, Android, and web clients with shared Swift codebase
- **watchOS XCFramework Development**: Created watchOS XCFrameworks for React Native consumption by client developers, enabling seamless integration of native watchOS functionality
- **Team Management**: Led remote development team across multiple time zones with focus on server-side Swift and cloud architecture
- **DevOps & CI/CD**: Established automated deployment workflows and monitoring systems for production applications

[Client Project - AWS Lambda Integration] *Server-Side Swift*

Server-Side Swift Application with AWS Lambda Backend

- **Server-Side Swift Development**: Built comprehensive backend services using AWS Lambda framework
- **AWS Lambda Integration**: Designed and implemented serverless functions
- **REST API Development**: Created robust APIs for iOS, watchOS, and web client integration with real-time data synchronization
- **Database Design**: Implemented PostgreSQL and DynamoDB solutions for efficient data storage and retrieval

Asset Health *iOS / Swift*

Official Web Site - http://assethealth.info

Track and Integrate with Employers Health Data

- help bring Adobe XD designs to life using Interface Build as well as coded user interfaces in Swift
- work with project managers and lead developers on timelines and the best way to actualize requirements
- communicate with the client team on regular basis on the unique differences of the iOS platform

conferences.io *macOS / Swift*

Real-Time Audience Engagement Application for Presentations

Official Web Site - https://www.conferences.io

- Building and Maintaining a macOS port for conferences.io in Swift
- Communicating on a regular basis with the client's product manager on the capabilities and limitations of macOS, Keynote, Powerpoint, and the AppleScript API
- Automate the process of building and deploying the app to Amazon S3
- Build and design a user interface using Storyboards and AppKit

Jackson National Life *iOS / Objective-C*

Sales Team Application for Showcasing and Presenting Product Material

App Store - https://itunes.apple.com/us/app/jackson/id920821928?mt=8

- Architect and team lead for Jackson National Life's iPad application
- Design an Objective-C API for easy for easy implementation by the client on the Xamarian platform
- Implement user interface designs from the client using Xcode and Interface Builder

Cheeney Media Concepts *iOS / Swift*

CE/CME Genius - Medical Continuing Education Tracking

Official Web Site - https://cmc2.tv

App Store - https://itunes.apple.com/us/developer/cmc-interactive/id1027986564

- work with the client team on porting Android applications to iOS in Swift
- design consistent API and database structure for both Android and iOS
- use CoreData and EventKit API to design a local storage database and the ability to create reminders for users

Organizations and Products

EmpowerApps.Show - Official BrightDigit Podcast

Official Web Site - https://www.empowerapps.show

Apple Podcasts - https://itunes.apple.com/us/podcast/empower-apps/id1437435392?mt=2

Official Producer

2017-

- Arrange and Choose Topics of Interest to the Business and Developer Community centered around Apple products
- Produce and Edit Pre-Show and Post-Show Notes
- Edit and Produce Deliverable Audio and Video for Audience Consumption
- Work with Audio Editor and Outside Marketing Personnel
- Schedule and Communicate with Guests on Show Content and Protocol
- Marketing and Manage Social Media of Episodes and Content

Previous Employers

IDV Solutions

Senior Software Developer

February 2007 - July 2012

- Develop mobile applications for the iPad
- Use .Net and JavaScript to create easy-to-use development tools for businesses

State of Michigan

Software Developer

February 2006 - February 2007

CoreComm

Project Developer

October 2003 – February 2006

Telephone Repair Assistant

July 2001 – October 2003

Education

Michigan State University

M.A., Digital Media and Art Technology

December 2005

B.S., Computer Science

May 2003

Products and Open Source Projects

Bushel

macOS | **SwiftUI** | **Virtualization**

Create Unbreakable macOS Apps

Bushel is the macOS virtual machine app for developers who want to be rigorous and uncompromising in their app testing. Bushel is focused on giving you the complete, native capabilities of the operating system for all your testing requirements.

Heartwitch

Server-Side Swift | **Vapor** | **PostgreSQL** | **Heroku** | **AWS** | **HealthKit** | **Web APIs**

Real-Time Health Data Streaming Platform

Heartwitch is a distributed, serverless application that streams health data from Apple Watch to web browsers in real-time. Built with server-side Swift using Vapor framework and deployed on AWS infrastructure.

Key Technical Achievements:

- **Server-Side Swift Architecture**: Designed and implemented scalable backend services using Vapor framework for real-time data processing
- **Real-Time Web APIs**: Built WebSocket and REST APIs for seamless data transmission between Apple Watch and web clients
- **Database Design**: Architected PostgreSQL schema for efficient health data storage and retrieval
- **Cross-Platform Integration**: Seamless integration between watchOS, iOS, and web platforms using shared Swift codebase

ContributeWordPress

macOS | **linux**

Import your WordPress site into Publish.

ContributeWordPress provides an easy to use API for importing a WordPress site into Publish.

ContributeWordPress uses the exported XML (also referred to as WXR file) to import content into your existing Publish site.

FelinePine

macOS | **iOS** | **watchOS** | **tvOS** | **web**

FelinePine provides an easy-to-use API for setting up logging and log categories across your architecture.

SyntaxKit

Swift | **SwiftSyntax** | **Macros** | **Code Generation**

Generate Swift code programmatically with declarative syntax

SyntaxKit is a Swift package that provides a type-safe, result builder-based API for generating Swift code structures. Designed for macro development, model transformers, and migration utilities where you need to programmatically create Swift code.

Key Features:

- **Swift Macro Development**: Type-safe API for building Swift macros and compiler plugins
- **Result Builder Syntax**: Declarative code generation using Swift's result builder pattern
- **SwiftSyntax Integration**: Built on top of SwiftSyntax for robust AST manipulation
- **Code Generation Tools**: Perfect for generating Swift code from external schemas, APIs, and configurations
- **Developer Tools**: Enables creation of code generators and transformers for complex Swift projects

MistKit

web | **CloudKit**

Swift Package for Service-Side & Command-Line Access to the Apple CloudKit framework

Use MistKit for

- Easily migrate data to and from CloudKit
- Building Command-Line Applications
- Running apps on non-Apple operating systems
- Server-side integration (via Vapor)
- Access serverless computing (via AWS Lambda)
- And more!

NPMPublishPlugin

macOS | **linux**

A Publish plugin that makes it easy to run npm commands for any Publish website.

NPMPublishPlugin allows you to integrate an NPM package into your Publish site. If you require javascript or css to be built for your site, this is the ideal plugin for you.

PackageDSL

macOS | **linux** | **iOS** | **watchOS** | **tvOS**

Simplify the management of your Package.swift file with PackageDSL. I was having a difficult time managing a large Package.swift file. PackageDSL allows you to setup individual targets, products, and dependencies using this DSL and create an easily organized, simplified, and easy to maintain Package for your Swift project.

SimulatorServices

macOS

SimulatorServices provides an easy to use API for managing, querying, and accessing simulators on your Mac.

SimulatorServices allows you to execute subcommands to simctl directly in Swift while offering an easy to use API for parsing and passing arguments.

SimulatorServices uses the SimCtl object to pass subcommands. Each subcommand objects takes custom arguments or property and can parse the standard output into an easy to use Swift object. There are currently two supported subcommands: GetAppContainers and List.

StealthyStash

macOS | **iOS** | **watchOS** | **tvOS**

StealthyStash provides an pluggable easy abstract layer for accessing Keychain data as well as an API for encoding and decoding complex data in the Keychain.

StealthyStash supports the adding, updating, and querying for both generic and internet passwords.

Sublimation

macOS | **iOS** | **watchOS** | **tvOS** | **linux**

Share your local development server easily with your Apple devices.

When you are developing a full stack Swift application, you want to easily test and debug your application on both the device (iPhone, Apple Watch, iPad, etc...) as well as your development server. If you are using simulator then setting your host server to localhost will work but often we need to test on an actual device. You can either be an IT expert your local network's DNS or you can use Sublimation to easily connect your local server to your device.

SundialKit

iOS | **watchOS**

Reactive communications library across Apple platforms.

For easier use in reactive user interfaces, especially with SwiftUI and Combine, I've created a library which abstracts and maps common connectivity APIs. Particularly in my app Heartwitch, I mapped the functionality of WatchConnectivity and Network over to track the user's ability to connect to the Internet as well as the ability for their iPhone to connect to their Apple Watch via WatchConnectivity

SyndiKit

macOS | **iOS** | **watchOS** | **tvOS** | **linux**

Built on top of XMLCoder, SyndiKit can be used to import and read site data whether from a WordPress site, RSS feeds, YouTube channel or podcast.

Features include:

- Import of RSS 2.0, Atom, and JSONFeed formats
- Extensions for iTunes-compatabile podcasts, YouTube channels, as well as WordPress export data
- User-friendly errors
- Abstractions for format-agnostic parsing

ThirtyTo

macOS | **iOS** | **watchOS** | **tvOS** | **linux**

Swift Package for using Base32Crockford Encoding for Data and Identifiers.

ThirtyTo provides a way to encode data and create identifiers which is both efficient and human-readable. While Base64 is more efficient it is not very human-readable with both both upper case and lower case letters as well as punctuation.

Base32Crockford offers the most reasonable compromise when it comes to encoding data. Being a super set of Base16, it uses all ten digits and 22 of the 26 Latin upper case characters.

[TransistorPublishPlugin](#)

macOS | **linux**

A Publish plugin that makes it easy to embed Transistor podcasts in your posts on any Publish website.

Using blockquotes in your markdown, you can easily embed the Transistor podcast player in your website.

Skills

Server-Side Swift & Cloud Development

Vapor Framework | **Server-Side Swift** | **REST APIs** | **WebSocket** | **JSON APIs** | **CloudKit** | **Codable** | **Networking**

AWS Cloud Services

AWS Lambda | **DynamoDB** | **S3** | **EC2**

Languages

Swift | **JavaScript** | **TypeScript** | **Objective-C** | **HTML5** | **CSS3** | **Bash Scripting**

Development Tools & DevOps

Gitlab CI | **Github Actions** | **Fastlane** | **Xcode** | **Docker** | **Homebrew** | **Jazzy** | **Final Cut Pro**

Frameworks and APIs

Apple Development

SwiftUI | **Combine** | **SwiftData** | **UIKit** | **AppKit** | **Cocoa Touch** | **Cocoa** | **Core Data** | **HealthKit** | **WatchKit** | **User Notifications** | **XPC**

Web Technologies

Node.js | **Vue.js** | **React** | **Google Analytics** | **REST APIs** | **WebSocket** | **JSON**

Databases & Data Management

PostgreSQL | **MySQL** | **DynamoDB** | **CloudKit** | **MsSQL** | **Oracle**

- references available on request