

The Swift Programming Language
Kevin Roark
Columbia University

This talk aims to be an overview of Swift not for eager Apple developers, but from the perspective of new programming language design, compiler advancements, interesting runtimes, etc. I hope to first touch on Swift's legacy: the good and bad parts of Apple's other language, the 30-year old Objective-C; in many ways Swift can be seen as a reactionary language. I'll then dive into the language's other influences, its interesting features, and its inventiveness. Then I'll talk about what makes Swift's compiler (Apple's LLVM), target language, and runtime somewhat compelling. This should transition well into some of the language design constraints of Swift. Swift is certainly imperfect (and still changing), so I'll also touch on its most obvious bad parts. Finally, we can discuss the potentially unique role of Swift compared to similarly-philosophical new-paradigm languages like Rust and Go.