

SwiftPMのプラグイン機能を 活用する

自己紹介

- 宇佐見 公輔（うさみこうすけ） / @usamik26
- 株式会社ゆめみ / iOSテックリード
- 最近、本を書いたりiOSの記事を寄稿したりしています。



Swift Package Managerの 更新

Swift 5.6リリース

- 2022-03-15（日本時間）リリース
- 同じ日にリリースされたXcode 13.3に付属
- 言語機能の更新もあったが、Swift Package Managerの更新もあった

Swift Package Manager

- SwiftPM 5.6 Release Notes
- SwiftPMにプラグイン機能が新しく追加された
- SE-0303 ビルドツールプラグイン：`swift build` のとき外部ツールを実行できる
- SE-0332 コマンドプラグイン：`swift package` コマンドを拡張して外部ツールを実行できる

コマンドプラグイン

- ``swift package`` コマンドを拡張して外部ツールを実行できる
- ``Package.swift`` で使いたいプラグインを指定するとコマンドが拡張される
- コマンドは ``swift package do-something`` という形式になる

Examples

- `apple/swift-docc-plugin`
- これを導入すると以下のコマンドが使えるようになる
- ``swift package generate-documentation``
- ``swift package preview-documentation``
- 先週のSwift愛好会で話しました (Swift 5.6 で使えるようになった `swift-docc-plugin` を試してみる)

ビルドツールプラグイン

- ``swift build`` のとき外部ツールを実行できる
- ``Package.swift`` で使いたいプラグインを指定すると自動的に実行される

Examples

- SE-0303では、以下の例が挙げられている
- SwiftGenでソースコードを生成する
- SwiftProtobufでソースコードを生成する

ビルドツールプラグインを
試してみる

ビルドツールプラグインを試してみる

- [juozasvalancius/ExampleSPMProjectWithSwiftLint](#)
- SwiftLintプラグインのサンプル
- この方は、SwiftPMプラグイン対応のためにSwiftLintにもコミットしている

Xcodeプロジェクトに適用してみた

- [usami-k/XcodeSwiftPMSample](#)

XcodeプロジェクトとSwiftPM

- XcodeプロジェクトとSwiftPMを組み合わせて使う
- 最近、見かけることが増えてきたプロジェクト構成

プロジェクト構成

- usami-k/XcodeSwiftPMSample
- アプリのXcodeプロジェクト（``Hello.xcodeproj``）には、必要最小限のものだけ入れる。
- アプリの実装コードは、Swiftパッケージ（``AppFeature``、``Core``）に入れる。
- ``Hello.xcodeproj`` でアプリに ``AppFeature`` パッケージをリンクする。

ビルド時の処理の定義を改善する

- ビルド時に何か処理を行いたいときはXcodeのビルドスクリプト機能を使っていた
- SwiftPMにビルドスクリプト機能がなかったため

ビルドツールプラグインの活用

- SwiftPMのビルドツールプラグインを使う
- Xcodeプロジェクトを編集せず、SwiftPMの機能でビルド時の処理を定義できる

ビルドツールプラグインの導入

- ``Package.swift`` の記述
- ``SwiftLintXcode`` がビルドツールプラグイン

```
targets: [  
  .binaryTarget(  
    name: "SwiftLintBinary",  
    url: "https://github.com/juozasvalancius/SwiftLint/releases/download/spm-accommodation/SwiftLintBinary-macos.ar  
    checksum: "cdc36c26225fba80efc3ac2e67c2e3c3f54937145869ea5dbcaa234e57fc3724"  
  ),  
  .plugin(  
    name: "SwiftLintXcode",  
    capability: .buildTool(),  
    dependencies: ["SwiftLintBinary"]  
  ),  
  .target(  
    name: "AppFeature",  
    dependencies: [  
      "Core"  
    ],  
    plugins: ["SwiftLintXcode"]),  
]
```

ビルド時の処理の記述

- `Plugins` フォルダ内の Swift コードで記述する

```
import PackagePlugin

@main
struct SwiftLintPlugins: BuildToolPlugin {
    func createBuildCommands(context: PluginContext, target: Target) async throws -> [Command] {
        return [
            .buildCommand(
                displayName: "Linting \(target.name)",
                executable: try context.tool(named: "swiftlint").path,
                arguments: [
                    "lint",
                    "--in-process-sourcekit",
                    "--path",
                    target.directory.string
                ],
                environment: [:]
            )
        ]
    }
}
```

まとめ

- Xcodeのビルドスクリプト機能をSwiftPMのビルドツールプラグインで置き換えが可能になった
- Xcodeプロジェクトへの依存を減らし、SwiftPMを活用することができる