Swagger Codegen

@wagyu298

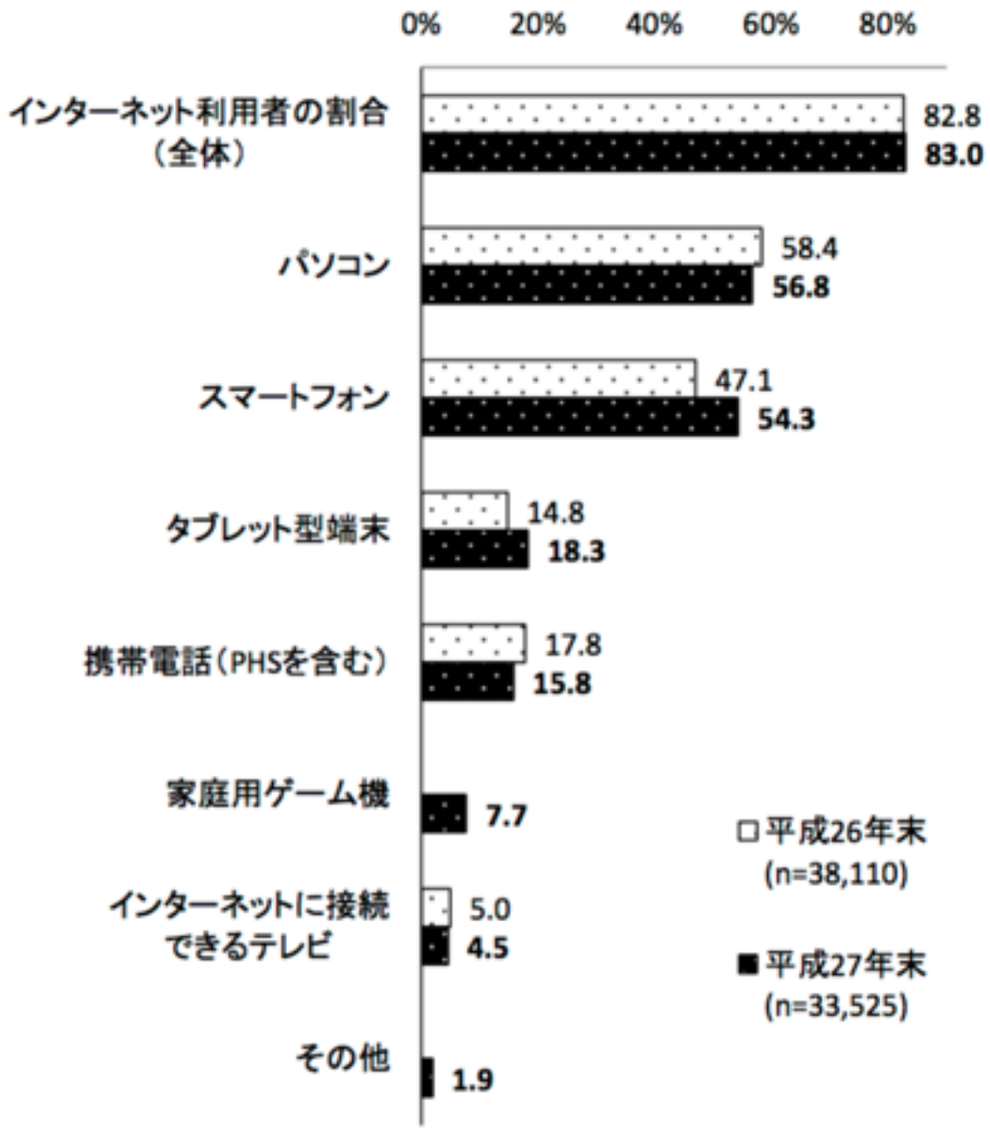
TL;DR

- Swagger → REST API の仕様を YAML で書くやつ
- Swagger Codegen → Swagger YAML からコードを生成するプログラム
  - Server Stub と Client API を生成できる
  - Server Stub は無いよりマシレベル、Client API はそのまま運用に使える
- Microservice とかアプリ向け

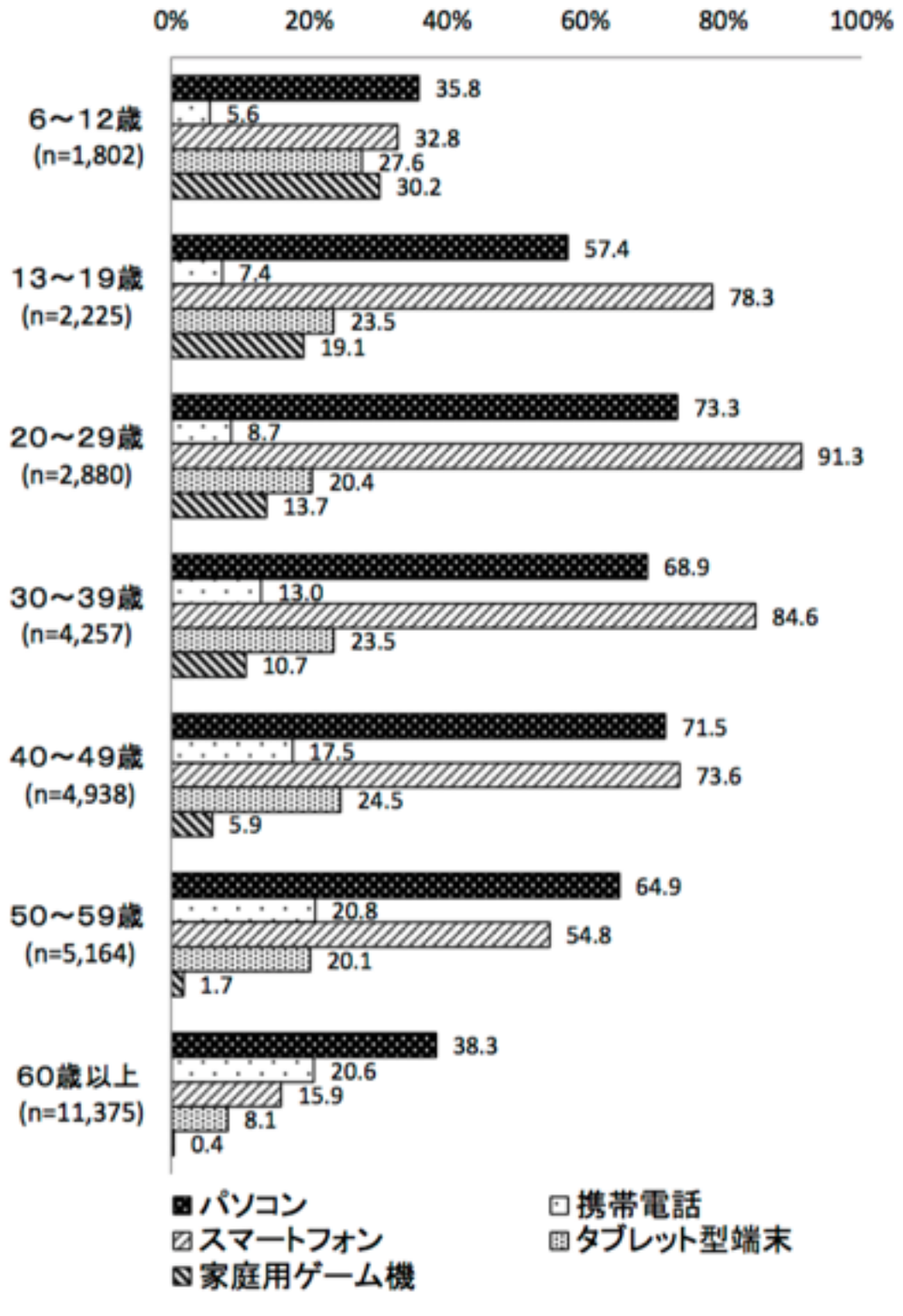
国内のスマホの使用状況について

総務省 平成 27 年通信利用動向調査の結果

図表1－5 インターネットの端末別利用状況（個人）

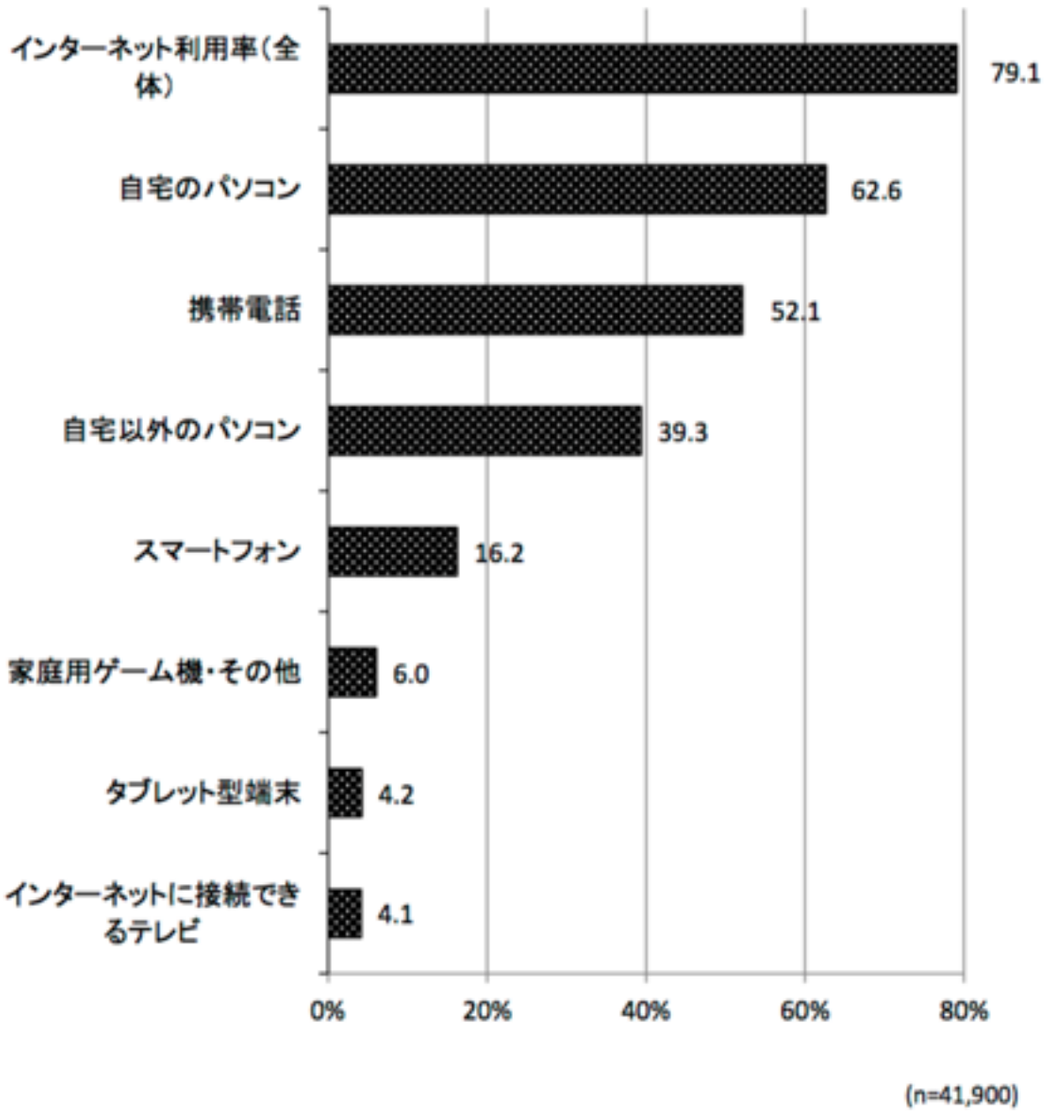


図表1－6 年齢階層別インターネット端末の利用状況（個人）（平成 27 年末）



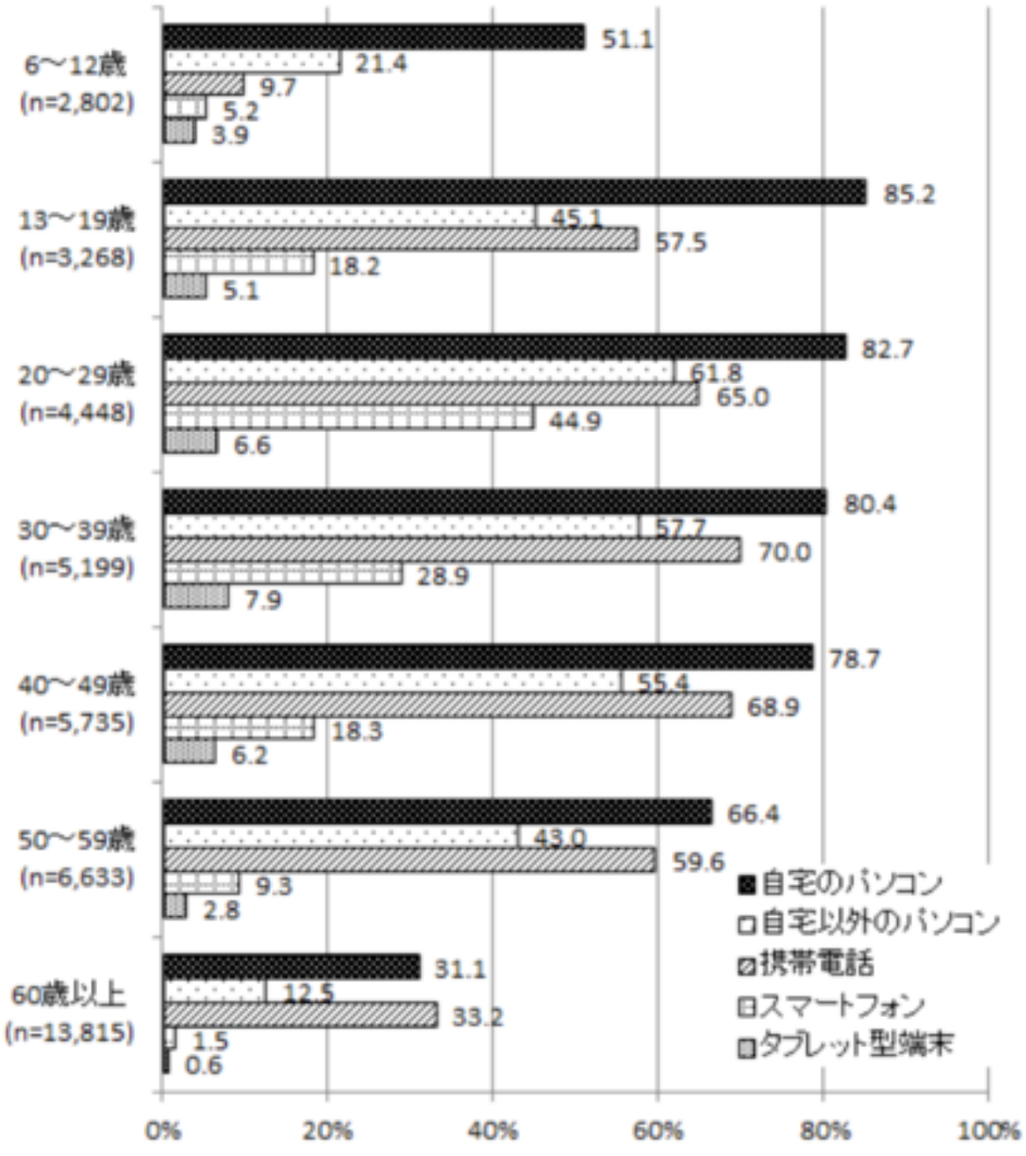
総務省 平成 23 年通信利用動向調査の結果

主要端末別インターネット利用率（個人）



(n=41,900)

主要端末別世代別インターネット利用率（個人）



パソコンとかオワコン

Swagger

- OpenAPI 2.0 のこと
  - http://swagger.io
  - https://www.openapis.org
  - サンプル http://petstore.swagger.io/v2/swagger.yaml
- REST API 定義ファイルを YAML で記述し、ネットワーク越しに仕様を伝搬させたりしたい感じのやつ

OpenID みたく
無かったことになりそうな
空気が過分にある

XML-RPC「Swagger がやられたようだな…」
SOAP「ククク…奴は四天王の中でも最弱…」
CORBA「人間ごときに負けるとはRPCの面汚しよ…」

こうならないと良いね！

便利な部分もあるので
めげずに使っていきたい

Swagger Codegen

- Swagger Specification ファイルからコードを生成するプログラム
- クライアント API とサーバスタブを出力できる
- 25+ のプログラミング言語に対応

android, aspnet5, aspnetcore, async-scala, bash, cwiki, csharp, cpprest, dart, elixir, flash, python-flask, go, groovy, java, jaxrs, jaxrs-cxf-client, jaxrs-cxf, jaxrs-resteasy, jaxrs-spec, jaxrs-cxf-cdi, inflector, **javascript**, javascript-closure-angular, jmeter, nancyfx, **nodejs-server**, **objc**, perl, php, python, qt5cpp, ruby, scala, scalatra, finch, silex-PHP, sinatra, rails5, slim, spring, dynamic-html, html, html2, swagger, swagger-yaml, swift, **swift3**, tizen, typescript-angular2, typescript-angular, typescript-node, typescript-fetch, akka-scala, CsharpDotNet2, clojure, haskell, lumen, go-server, erlang-server, undertow, msf4j, ze-ph

Microservice や
アプリ開発に使えるはず

Swagger Codegen で
ネイティブアプリ開発環境を
5分でセットアップする

要るもの1

- Swagger Codegen の最新版
  - swift3 サポートが master ブランチにしか無いのでビルドする必要がある／swift3 が要らなければ homebrew からのインストールで OK です
- http://swagger.io/swagger-codegen/
- http://swagger.io/docs/
- https://github.com/swagger-api/swagger-codegen

要るもの2

- Java7
- ビルドするなら mvn (maven)
  - brew install maven で入ります

要るもの3

- Open API で仕様を記述した YAML ファイル

- 今回は公式のサンプルを使います

YAML ファイルを取得

```
bash-3.2$ curl http://petstore.swagger.io/v2/swagger.yaml > swagger.yaml
```

Server Stub の作成

```
bash-3.2$ java -jar swagger-codegen-cli.jar generate -i swagger.yaml -
l nodejs-server -o nodejs
```

```
bash-3.2$ cd nodejs
bash-3.2$ npm install
bash-3.2$ node index.js
Your server is listening on port 8080 (http://localhost:8080)
Swagger-ui is available on http://localhost:8080/docs
```

http://localhost:8080/v2/pet/12 でアクセスできます

iOS Client API の作成

```
bash-3.2$ java -jar swagger-codegen-cli.jar generate -i swagger.yaml
-l swift3 -o PetShop -DprojectName=PetShop
```

CocoaPods ライブラリ PetShopAPI ができます

API 呼び出しは Alamofire です

(Android は volley です)

podspec 等が若干古いのでいじくります

```
# Uncomment this line to define a global platform for your project
# platform :ios, '9.0'

target 'SwaggerExam' do
  # Comment this line if you're not using Swift and don't want to use dynamic frameworks
  use_frameworks!

  # Pods for SwaggerExam
  pod 'PetShopAPI', :path => '../PetShopAPI'

  target 'SwaggerExamTests' do
    inherit! :search_paths
    # Pods for testing
  end

  post_install do |installer|
    installer.pods_project.targets.each do |target|
      target.build_configurations.each do |config|
        config.build_settings['SWIFT_VERSION'] = '3.0'
      end
    end
  end

end
```

（時間が無いので）API を呼び出してログ表示します

```
func application(_ application: UIApplication, didFinishLaunchingWithOptions
launchOptions: [UIApplicationLaunchOptionsKey: Any]?) -> Bool {
    // Override point for customization after application launch.
    PetShopAPI.basePath = "http://localhost:8080/v2"
    PetAPI.getPetById(petId: 12) { (pet, error) in
        if let error = error {
            print(error)
            return
        }
        guard let pet = pet else { return }
        print(pet.encodeToJSON())
    }
    return true
}
```

Info.plist に app transport security の設定が必要

Server Stub を golang に変えてみます

```
bash-3.2$ java -jar swagger-codegen-cli.jar
generate -i swagger.yaml -l go-server -o go-server
-DpackageName=petshop
```

node-server と異なりテスト用のデータが
帰ってこないのでエラーになります

終わり