

# ヤフー広告API ハンズオン

- 技術的仕様
- OpenAPI
  - 導入理由、利用方法
- ハンズオン
- 移行方法
- トラブルシューティング・TIPS
- FAQ

# 技術的仕様

Transport	REST
Encoding	JSON
認証	OAuth2

# OpenAPI

- REST APIを記述するためのフォーマット及び周辺ツール群
  - What Is OpenAPI?  
<https://swagger.io/docs/specification/about/>
  - OpenAPI (Swagger) 超入門  
[https://qiita.com/teinen\\_qiita/items/e440ca7b1b52ec918f1b](https://qiita.com/teinen_qiita/items/e440ca7b1b52ec918f1b)

- yamlからAPIリファレンス生成
- yamlからリクエスト、レスポンスオブジェクト生成

## yamlファイル

```
openapi: 3.0.2
info:
  title: 'Yahoo!広告 ディスプレイ広告 API リファレンス β版 / Yahoo!
    Version'
  version: 'v0'
  description: |
    <div lang="ja">Yahoo!広告 ディスプレイ広告 APIのWebサービスについ
    「Try it out」のご利用には、事前にアプリケーションの登録が必要です。
    https://yahoojp-marketing.github.io/ads-display-api-docume
    </div>
    <div lang="en">Display Ads API Web Services supported in Y
    When you use "Try it out", you need to register your appli
    As one of redirect URI for application, you need to set "h
    </div>

components:
  securitySchemes:
    oAuth:
      type: oauth2
```

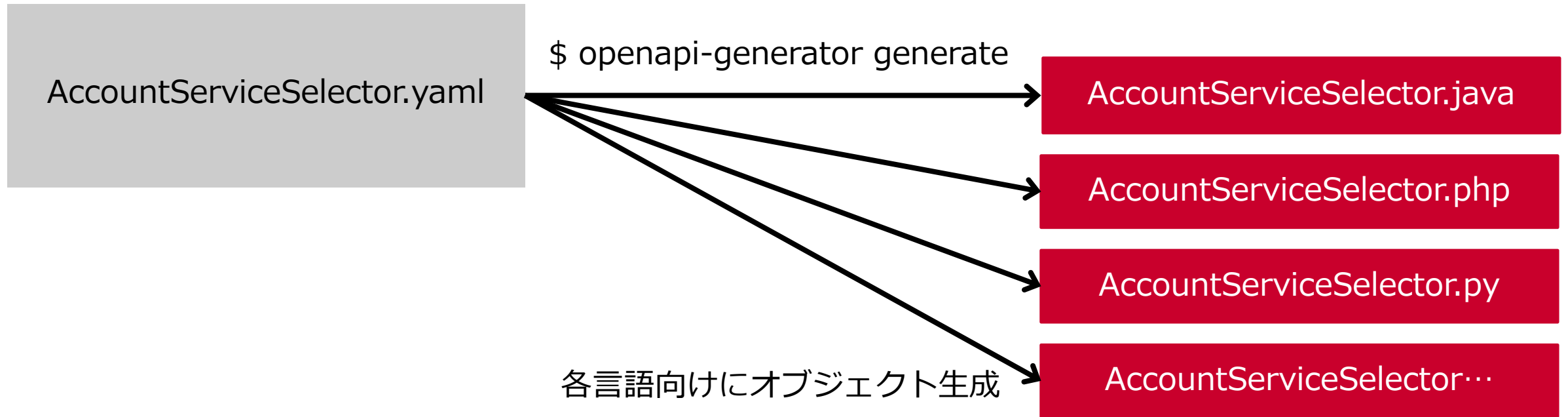
生成

## APIリファレンス

The screenshot shows the Yahoo! Japan API Reference page. The header includes the Yahoo! Japan logo and the text 'APIリファレンス β版'. Below the header, there are two tabs: '検索広告APIリファレンス v0' and 'ディスプレイ広告APIリファレンス v0'. The left sidebar lists various API services, including AccountAdProductService, AccountService, AdGroupAdService, AdGroupService, AdGroupTargetService, CampaignService, ConversionTrackerService, DictionaryService, MediaService, PlacementUrlIdeaService, PlacementUrlListService, ReportDefinitionService, RetargetingListService, RetargetingTagService, SearchKeywordIdeaService, SearchKeywordListService, StatsService, and VideoService. The main content area displays the 'Yahoo!広告 ディスプレイ広告 API リファレンス β版' page, which includes a link to the design file, a description of the API, and a section for 'Servers' with the URL 'https://ads-display.yahooapis.jp/api/v0'. The 'AccountAdProductService' section is highlighted, showing an 'account' button and a description of the service.

<https://github.com/yahoojp-marketing/ads-display-api-documents/blob/master/design/v0/Route.yaml>  
<https://yahoojp-marketing.github.io/ads-display-api-documents/>





# java向けオブジェクト作成例

\$ git clone <https://github.com/yahoojp-marketing/ads-display-api-documents.git>

\$ openapi-generator generate -i ./ads-display-api-documents/design/v0/Route.yaml -g java -o /path/to/obj

# openapi-generatorの-gオプションはphpやpython等各言語が指定可能

<https://github.com/OpenAPITools/openapi-generator>

- Yahoo!広告 APIでは下記を採用
  - OpenAPI 3
  - スキーマファースト  
(yamlファイルからコード生成するパターン)

# ハンズオン

- Yahoo!広告 ディスプレイ広告の

- 参照系/更新系
- アップロード/ダウンロード

※ Yahoo!広告 APIではアップロード/ダウンロードに仕様変更があります(Yahoo!プロモーション広告APIと比較して)

- git clone or リポジトリダウンロード  
<https://github.com/yahoojp-marketing/yahoo-ads-api-developers-workshop>
- アクセストークン再発行(Markdownにもあります)  
[https://biz-oauth.yahoo.co.jp/oauth/v1/token  
?grant\\_type=refresh\\_token  
&client\\_id=YOUR\\_CLIENT\\_ID  
&client\\_secret=YOUR\\_CLIENT\\_SECRET  
&refresh\\_token=REFRESH\\_TOKEN](https://biz-oauth.yahoo.co.jp/oauth/v1/token?grant_type=refresh_token&client_id=YOUR_CLIENT_ID&client_secret=YOUR_CLIENT_SECRET&refresh_token=REFRESH_TOKEN)

# 移行方法

※アプリケーションの作りによってケースバイケースですが…  
前提

- 広告データはYahoo!広告 API、 Yahoo!プロモーション広告API 双方で操作可能(作成済みレポートは除く)
- QPSはYahoo!広告 API、 Yahoo!プロモーション広告APIそれぞれ別でカウント



- Service毎に徐々に移行を推奨

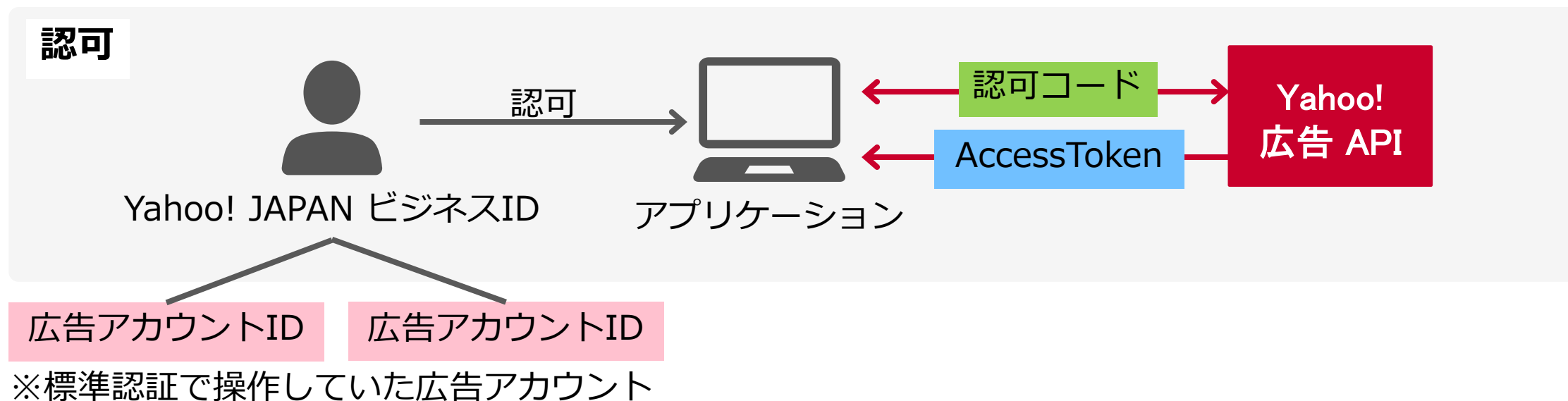
AAAService(Yahoo!プロモーション広告API)		AAAService(Yahoo!プロモーション広告API)
BBBService(Yahoo!プロモーション広告API)	→	BBBService(Yahoo!プロモーション広告API)
CCCService(Yahoo!プロモーション広告API)		CCCService(Yahoo!広告 API)

- 同一エンティティを新旧APIで更新することは非推奨

- 例 同じcampaignをYahoo!プロモーション広告APIとYahoo!広告 API双方からset
- 不具合があった場合の原因の切り分けが困難になるため



- ・ 標準認証で操作していた広告アカウントを操作できる  
ビジネスIDで認可し、認可コード、リフレッシュトークン、  
アクセストークンを発行



- ・ ツール利用者のビジネスIDで認可をしてもらうためのOAuth認可コードフローに対応した処理の新規開発



からYahoo!広告に対して以下のアクセスを許可しますか？

---

許可するYahoo! JAPANビジネスID

許可する権限 Yahoo!広告

Yahoo!広告を運用する権限です。

# トラブルシューティング・TIPS

- レスポンスで404 Not Found時の対処

正しくない例

/CampaignService/get # /api/v0が無い

/api/CampaignService/get # /v0が無い

/api/V0/CampaignService/get # Vが大文字

正しい例

/api/v0/CampaignService/get

## • レスポンスでBad request時の対処

例

```
< HTTP/1.1 400 Bad Request
< Content-Type: application/json;charset=utf-8
< Date: Tue, 24 Dec 2019 03:03:10 GMT
< X-Vcap-Request-Id: 67f3d5ab-7758-410a-4392-93e68f3fdd90
< X-Yahoo-Ads-Rid: 676b9f4a983c92fa1faef363812adea8
< Content-Length: 107
```

```
{"rid":"676b9f4a983c92fa1faef363812adea8","errors":[{"code":"0005","message":"Bad request.","details":[]}]}
```

- リクエストボディのjsonが正しくないor型不一致(配列フィールドに単発な値を入れている等)  
リクエストボディをjq等のjsonパーサで確認
- APIリファレンスのExample Valueでリクエストボディのjsonのサンプルを確認

## • パラメータを一つも送らない場合のリクエストボディ例

例

```
curl -H "Authorization: Bearer ${ACCESS_TOKEN}" ¥  
-H 'Content-Type: application/json' ¥  
https://ads-display.yahooapis.jp/api/v0/AccountService/get -d ¥  
{  
}
```

- '{ }'(波括弧)だけを送る
- ただし一部のServiceの一部のアクションは  
リクエストボディ自体不要(GETメソッド&ボディなしでコール)

GET

/DictionaryService/getThirdPartyTrackingDomain

GET

/DictionaryService/getMediaAdFormat

- 良い問い合わせ例

- 事象
- 事象発生日時
- リクエストヘッダー
- リクエストボディ
- レスポンスヘッダー
- レスポンスボディ

※レスポンスボディ or レスポンスヘッダーの  
**rid(X-Yahoo-Ads-Rid)**も問い合わせ時の情報に含めてください

# FAQ



EOF