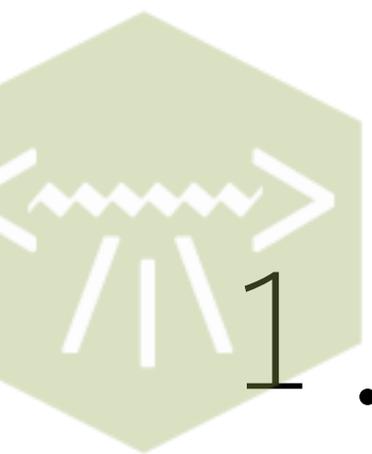




Webサービスを活用してAlteryxをパワーアップ ダウンロードツールでRestAPIを叩く

KCCSモバイルエンジニアリング株式会社
データソリューション部 梶谷

2020/09/29

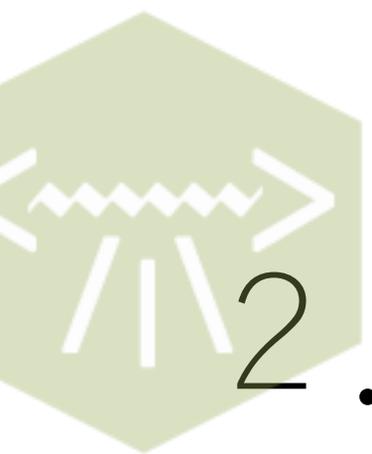


1.

自己紹介

自己紹介は配布資料では割愛します





2.

RestAPIとは？

Rest APIとは

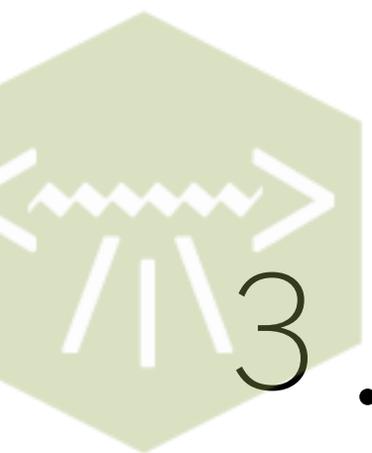
- WEBシステムを外部から利用する際に一般的に使われるのが RestAPI (なんちゃってRest含む)
- URLをパラメータ付きで投げると、WEBシステムに何かをやらせて、JSONやXML等で結果を取得できる
- C#、Java、PHP、Python、Rubyなどのプログラミング言語から利用するのが一般的 (RestAPIを呼び出すためのサンプルソースもたいてい公開されています)

Rest APIでできること

- WEBサービスから情報を取得する
- WEBサービスで実行してその結果を取得する
- WEBサービスに対してアクションを行う

例：

- ストリート画像の取得（Google StreetViewStatic API）
- 翻訳する（Google Translation API）
- メールを送る（SendGrid MailSend API）
- TableauServerのユーザの管理をする（Tableau Rest API）



3.

AlteryxでRestAPIを叩く

Alteryxでどうやるの？

- ダウンロード（Download）ツールを利用
- データを取得するためのパラメータを生成する必要あり
- 取得したデータは、JSONパース、XMLパースツールで加工
（正規表現ツールも役に立ちます）



引っかかるポイント

- ダウンロードツール自体の使い方
 - ヘッダ？ペイロード？
- 認証系
 - APIキー？アクセストークン？
- JSON？XML？



認証系

- 実はこれがすんなりいくとできたも同然です

大別すると2タイプくらいになります。

めちゃくちゃ簡単なタイプ

- 別途取得したAPIキーをURLに含めるタイプ
- IDとPass (+ α) からアクセストークンを取得し、それを使い回すタイプ

場合によってはハマるタイプ

JSON、XML

- JSON (JavaScript Object Notation) の略。テキストベースのデータ交換用フォーマット。様々なプログラミング言語で利用でき、WEBの世界で良く用いられる
- XML (Extensible Markup Language) 。マークアップ言語でありタグを使ってデータ構造を記述する。

【JSONのサンプル】

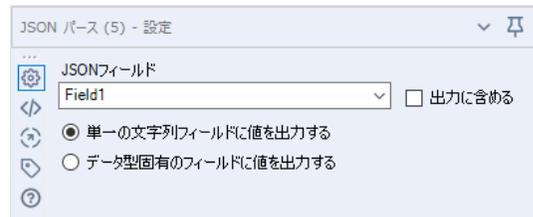
```
{  
  "issue": {  
    "project_id": 1,  
    "subject": "Example",  
    "priority_id": 4  
  }  
}
```

【XMLのサンプル】

```
<?xml version="1.0"?>  
<issue>  
  <project_id>1</project_id>  
  <subject>Example</subject>  
  <priority_id>4</priority_id>  
</issue>
```

JSON、XMLをAlteryxでパースしてみる

```
{  
  "issue": {  
    "project_id": 1,  
    "subject": "Example",  
    "priority_id": 4  
  }  
}
```



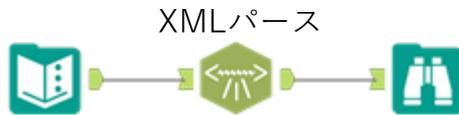
設定が簡単！

縦持ちなので、どんな場合でもコンパクトに扱える



レコード	JSON_Name	JSON_ValueString
1	issue.project_id	1
2	issue.subject	Example
3	issue.priority_id	4

```
<?xml version="1.0"?>  
<issue>  
  <project_id>1</project_id>  
  <subject>Example</subject>  
  <priority_id>4</priority_id>  
</issue>
```



レコード	Field1	issue	project_id	subject	priority_id
1	<?xml version="1.0"?>		1	Example	4



データによって設定を色々変えないといけない

横持ちなので、データによってはフィールド数がものすごいことになります

ダウンロードツールの使い方

- みんな大好きダウンロードツール



- 開発者カテゴリなので使ったことない人も多いかと思います

ダウンロードツール の設定項目

- **基本設定**

- 接続先URL、受け取ったデータの受け取り方

- **ヘッダー**

- HTTPヘッダ（Content-Typeや認証情報を含めることが多い）

- **ペイロード**

- 接続の仕方（POSTやGET）と送信データを設定

- **接続**

- FTP接続する際のIDとPassを設定（RestAPI利用時は不使用）

ダウンロードツール — 基本設定

ダウンロード (15) - 設定

基本設定 | ハッダー | パイロード | 接続

URL
フィールド

URL

URLテキストをエンコードする

出力

フィールドへ

文字列

データを次の形式でエンコード: Unicode UTF-8

BLOB

ファイルへ

一時ファイル

フィールドからのファイル名

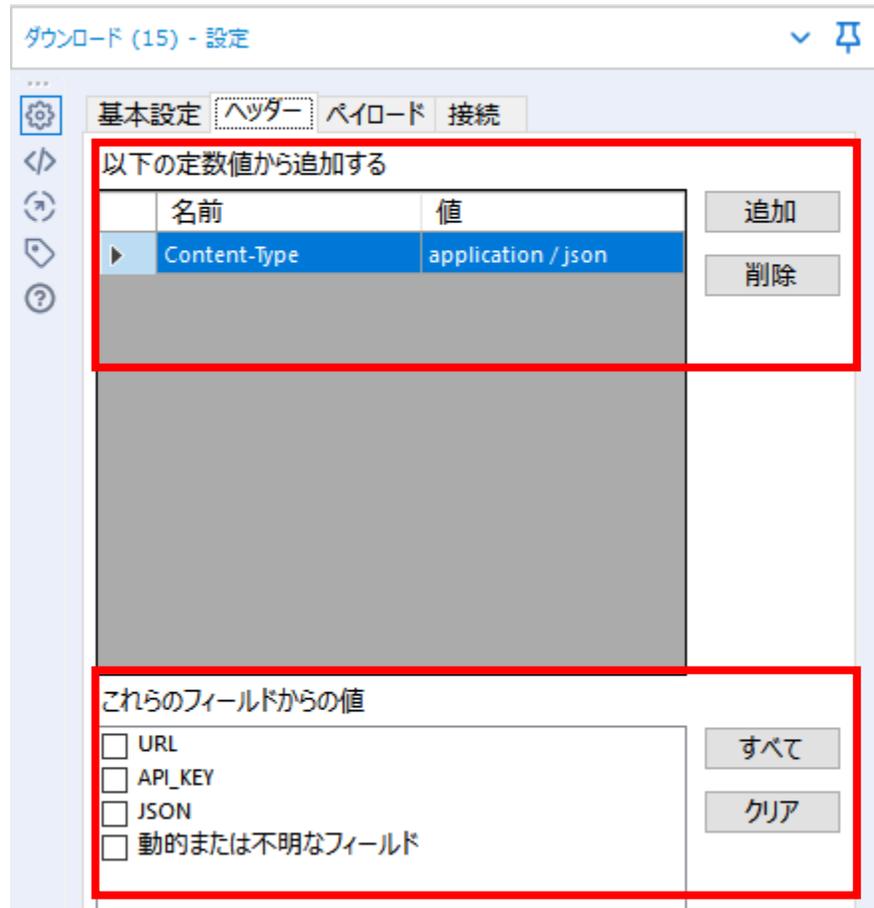
URL

URL：接続先URLが格納されているフィールドを指定

出力：WEBから得られたデータをどう取り扱うか

- **フィールドへ**：WF内で加工する場合はこちら
 - 文字列：テキストデータはこちら（JSON等）
 - BLOB：画像などのバイナリファイルはこちら
- **ファイルへ**：そのままファイルとして保存するならこちら

ダウンロードツール — ヘッダー



ヘッダーとして送るものを設定

不変なものは「追加」ボタンから設定するのが楽です。
例えば“Content-Type: application/json”をヘッダに設定せよ、とあった場合は
名前： Content-Type
値： application/json
と記載すればOK。

フィールドに格納されているデータを使う場合はこちらの該当する項目にチェックを入れます。
アクセストークンなどは変化するのでこちらになると思います。
例えば、上記のような場合は、「フィールド名」を「Content-Type」とし、中身の値を「application/json」として「Content-Type」フィールドにチェックを入れればOK。

ダウンロードツール — ペイロード

ダウンロード (15) - 設定

基本設定 ヘッダー **ペイロード** 接続

HTTP アクション POST

作成クエリ文字列/本文 **送信データ本体を選択**

以下の定数値から

名前	値
----	---

追加
削除

これらのフィールドからの値

- URL
- API_KEY
- JSON
- 動的または不明なフィールド

すべて
クリア

コンテンツタイプ application/x-www-form-urlencoded

フィールドからクエリ文字列/本文を取得する

JSON

クエリ文字列/本文で次を使用します

HTTPアクション：APIで指定されているアクションにします。

本体で送る送信データを選択

送信データ本体は3つから選択します。

「作成クエリ文字列/本文」「フィールドからクエリ文字列/本文を取得する」「クエリ文字列/本文で次を使用します」

パラメータなどをJSONで送信するような場合は、「フィールドからクエリ文字列/本文を取得する」でJSONが格納されているフィールドを指定します。

問い合わせパラメータを「作成クエリ文字列/本文」で指定するケースも有り（Pardot等）。



4. 具体例

具体例紹介

- Google Translation Basic API（認証方式：APIキーをURLに含めるタイプ）
- SendGrid Mail Send V3（認証方式：APIキーをヘッダに含めるタイプ）
- Tableau RestAPI（認証方式：アクセストークンを取得するタイプ）

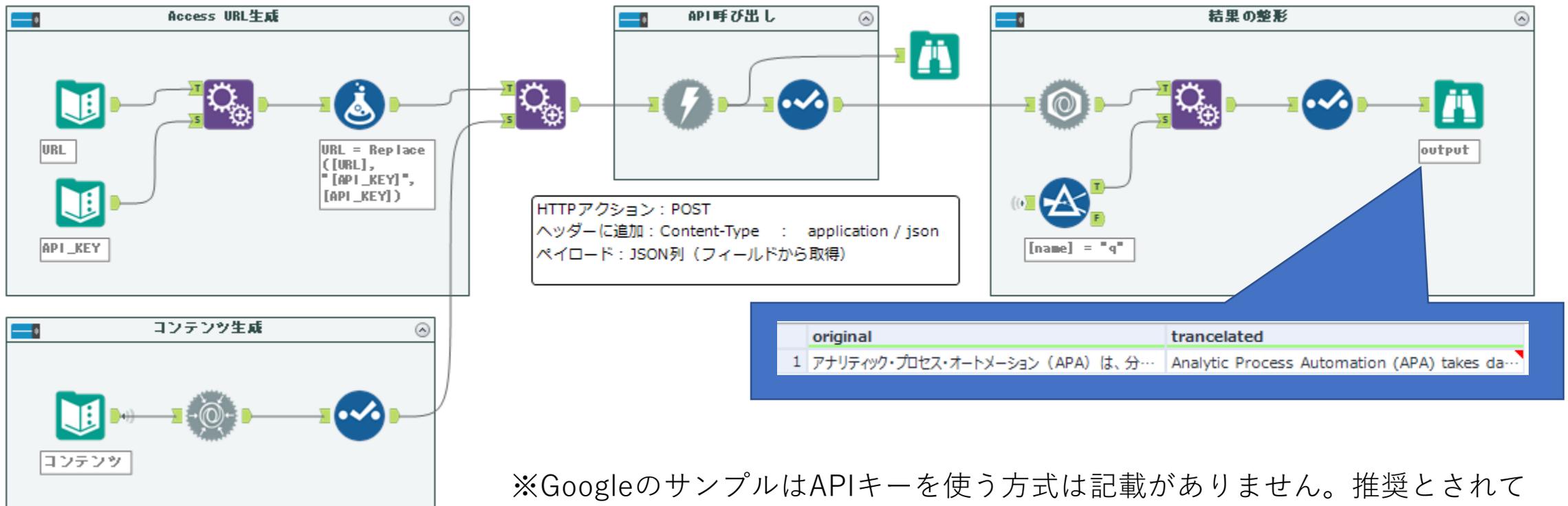
ここでは、認証方式別に3タイプ紹介します。

具体例 1 : Google TranslationBasic API

文章の翻訳を行うAPIです

URL	https://translation.googleapis.com/language/translate/v2?key=[API_KEY] ※APIキーはGoogleのダッシュボードから事前に発行															
ヘッダ	「Content-Type : application / json」を指定															
HTTPアクション	「POST」															
各種パラメータ	JSON形式でフィールドを作成し、ペイロードで指定。 <table border="1"><thead><tr><th></th><th>name</th><th>value</th></tr></thead><tbody><tr><td>1</td><td>q</td><td>アナリティック・プロセス・オートメーション(APA)は、分析、ビジネスインテリジ</td></tr><tr><td>2</td><td>source</td><td>ja</td></tr><tr><td>3</td><td>target</td><td>en</td></tr><tr><td>4</td><td>format</td><td>text</td></tr></tbody></table> <p>※qとtargetだけあれば動作します</p> <p>q : 翻訳する元のテキスト source : 翻訳元の言語 target : 翻訳先の言語 format : 翻訳元のテキストの形式 (htmlもしくはtextを指定)</p>		name	value	1	q	アナリティック・プロセス・オートメーション(APA)は、分析、ビジネスインテリジ	2	source	ja	3	target	en	4	format	text
	name	value														
1	q	アナリティック・プロセス・オートメーション(APA)は、分析、ビジネスインテリジ														
2	source	ja														
3	target	en														
4	format	text														

具体例 1 : Google TranslationBasic API



※GoogleのサンプルはAPIキーを使う方式は記載がありません。推奨とされているのはCloud SDKを使って事前にアクセストークンを取得する方法です。

<https://cloud.google.com/translate/docs/setup?hl=ja>

JSON作成

複雑なJSONの場合は結構面倒です
(今回はめちゃくちゃ簡単なタイプ)

レコード	name	value
1	q	アナリティック・プロセス・オートメーション (APA) は、分...
2	source	ja
3	target	en
4	format	text

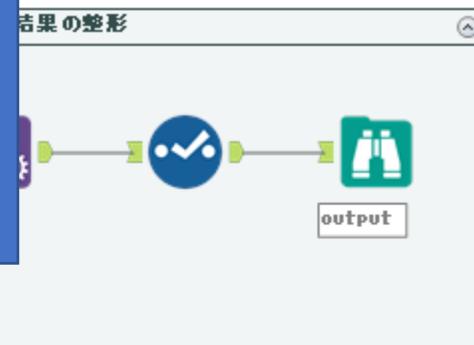
JSONビルド (25) - 設定

- 名前フィールド: name
- 文字列値フィールド: value
- Int値フィールド: [なし]
- float値フィールド: [なし]
- ブール値フィールド: [なし]

グループ化フィールド(任意)

- name
- value

すべて
クリア



送信するJSONの中身

```
{  
  "q": "アナリティック・プロセス・オートメーション (APA) は、分析、ビジネスインテリ  
ジェンス、データサイエンス、および機械学習ツールなど、かつては別々の市場として存在していたデー  
タおよび分析ソフトウェアが、次なるステージへと進化し、誕生しました。APAは、自動化とデジタルト  
ランスフォーメーションの鍵となる3つの柱を統合することで、データと分析の民主化、ビジネスプロセ  
スの自動化、および人材のスキルアップを実現し、クイック・ウィンと変革的な成果をもたらします。",  
  "source": "ja",  
  "target": "en",  
  "format": "text"  
}
```

翻訳するコンテンツ

Source : 変換元言語
Target : 変換先言語
Format : TEXT もしくは HTML

ダウンロードツールの設定

ダウンロード (15) - 設定

基本設定 ハッダー ペイロード 接続

URL
フィールド

URL

URLテキストをエンコードする

出力
フィールドへ

文字列

データを次の形式でエンコード: Unicode UTF-8

BLOB

ファイルへ

一時ファイル

フィールドからのファイル名

URL

ダウンロード (15) - 設定

基本設定 ハッダー ペイロード 接続

以下の定数値から追加する

名前	値
Content-Type	application / json

追加
削除

これらのフィールドからの値

URL
 API_KEY
 JSON
 動的または不明なフィールド

すべて
クリア

ダウンロード (15) - 設定

基本設定 ハッダー ペイロード 接続

HTTP アクション POST

作成クエリ文字列/本文

以下の定数値から

名前	値
----	---

追加
削除

これらのフィールドからの値

URL
 API_KEY
 JSON
 動的または不明なフィールド

すべて
クリア

コンテンツタイプ application/x-www-form

フィールドからクエリ文字列/本文を取得する

JSON

クエリ文字列/本文で次を使用します

翻訳結果！

【原文】

アナリティック・プロセス・オートメーション（APA）は、分析、ビジネスインテリジェンス、データサイエンス、および機械学習ツールなど、かつては別々の市場として存在していたデータおよび分析ソフトウェアが、次なるステージへと進化し、誕生しました。APAは、自動化とデジタルトランスフォーメーションの鍵となる3つの柱を統合することで、データと分析の民主化、ビジネスプロセスの自動化、および人材のスキルアップを実現し、クイック・ウィンと変革的な成果をもたらします。

【Google Translation APIで英語に翻訳】

Analytic Process Automation (APA) takes data and analytics software that once existed as separate markets, such as analytics, business intelligence, data science, and machine learning tools, to the next stage. Did. By integrating the three key pillars of automation and digital transformation, APA enables democratization of data and analytics, automation of business processes, and upskilling of talent, with quick wins and transformative results. Bring

具体例 1 : Google TranslationBasic API

【まとめ】

- APIキーをアクセスするURLに含めてアクセスするタイプ
- このタイプはあっさり構築可能です（パース作業の方が苦勞するかも・・・）
- 同様のタイプ
 - Google Maps API（Geocoding API、StreetViewStaticAPI、MapStaticAPI、DirectionsAPI等）
 - Google Natural Language API
 - Redmine
- Google Translation Basic APIは翻訳機能が使えるようになりますが、なかなかAlteryx内でどのような使い方をするかユースケースが難しいかもしれません。ただ、構築方法は他のGoogleのAPIも同じような感覚でできるので、参考にしてみてください

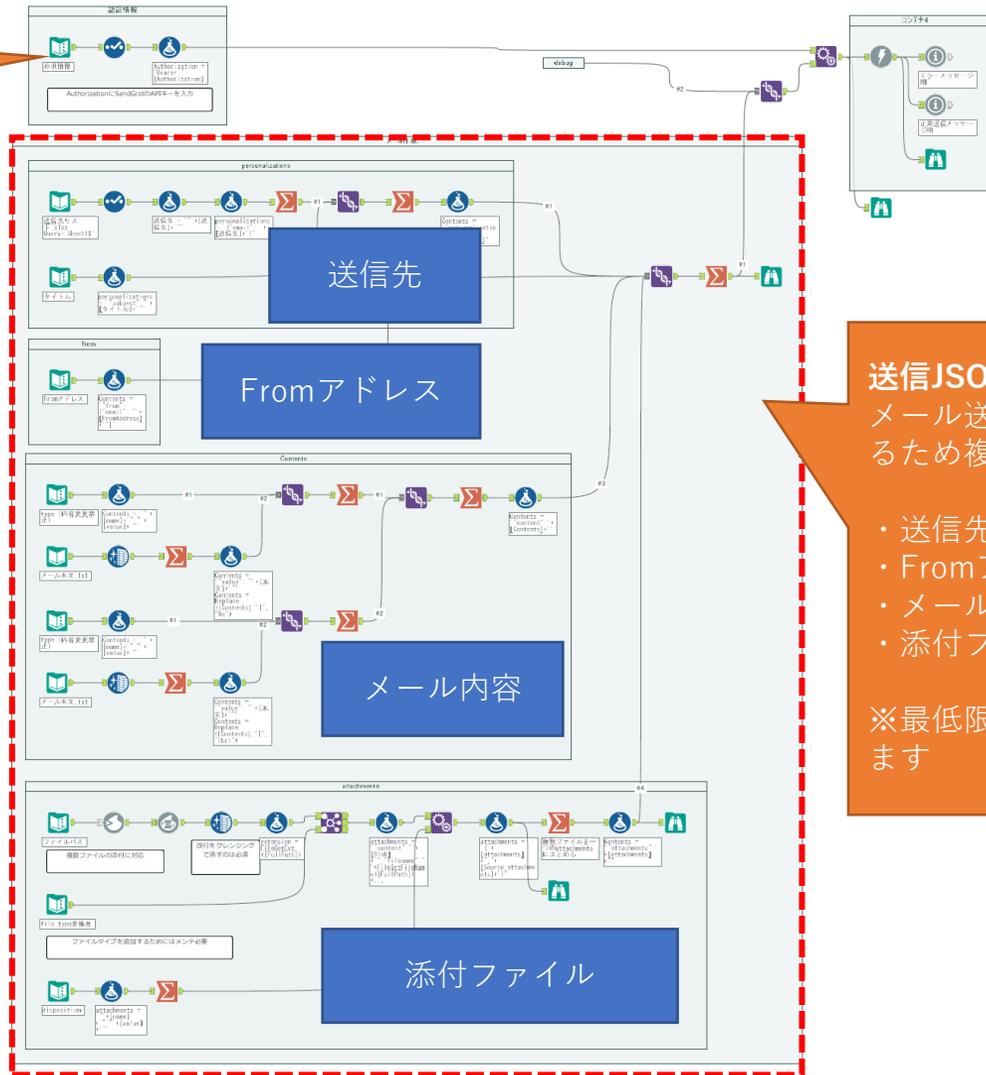
具体例 2 : Mail Send V3 (SendGrid)

メールを送信するAPIです

URL	https://api.sendgrid.com/v3/mail/send
ヘッダ	「Content-Type : application / json」 「Authorization : Bearer [APIキー]」
HTTPアクション	「POST」
各種パラメータ	JSON形式でフィールドを作成し、ペイロードで指定。 ※大量のパラメータがあるため詳細は割愛

具体例 2 : Mail Send V3 (SendGrid)

アクセスURL部分はシンプル



送信JSON :
メール送信する際に多くの情報を必要とするため複雑になっている

- ・ 送信先 (複数)
- ・ Fromアドレス
- ・ メール内容
- ・ 添付ファイル

※最低限の機能にしぼればシンプルに作れます

ダウンロードツールの設定

ダウンロード (2) - 設定

基本設定 | ヘッダー | パイロード | 接続

URL
フィールド
BaseURL

URLテキストをエンコードする

出力
フィールドへ
 文字列
データを次の形式でエンコード: Unicode

BLOB

ファイルへ
 一時ファイル
 フィールドからのファイル名
BaseURL

ダウンロード (2) - 設定

基本設定 | ヘッダー | パイロード | 接続

以下の定数値から追加する

名前	値	追加
Content-Type	application/json	追加

削除

これらのフィールドからの値

BaseURL
 Authorization
 JSON
 動的または不明なフィールド

すべて
クリア

ポイント:
フィールド名は指定された名前
を使うこと

ダウンロード (2) - 設定

基本設定 | ヘッダー | パイロード | 接続

HTTP アクション POST

作成クエリ文字列/本文

以下の定数値から

名前	値	追加
----	---	----

削除

これらのフィールドからの値

BaseURL
 Authorization
 JSON
 動的または不明なフィールド

すべて
クリア

コンテンツタイプ application/x-www-form-urlencoded

フィールドからクエリ文字列/本文を取得する
JSON

クエリ文字列/本文で次を使用します

ヘッダの指定について

RestAPIを利用する際、プログラミング言語等のサンプルが公開されているケースがほとんどですが、Alteryxと比較してみます。

例：cURLを使用する場合

```
curl -X "POST" "https://api.sendgrid.com/v3/mail/send" -H "Authorization: Bearer YOUR_API_KEY" -H "Content-Type: application/json" -d "[YOUR DATA HERE]"
```

ヘッダに「Authorization: Bearer YOUR_API_KEY」を指定せよ、という意味
※APIキーは別途取得

Alteryxでは、「:」の左側をフィールド名、右側をレコードの内容としてダウンロードツールのヘッダに指定します。

例：

Authorization
Bearer SG.0AMqsZrRSb21I4koWKp0cA.W5...

具体例 2 : Mail Send V3 (SendGrid)

【まとめ】

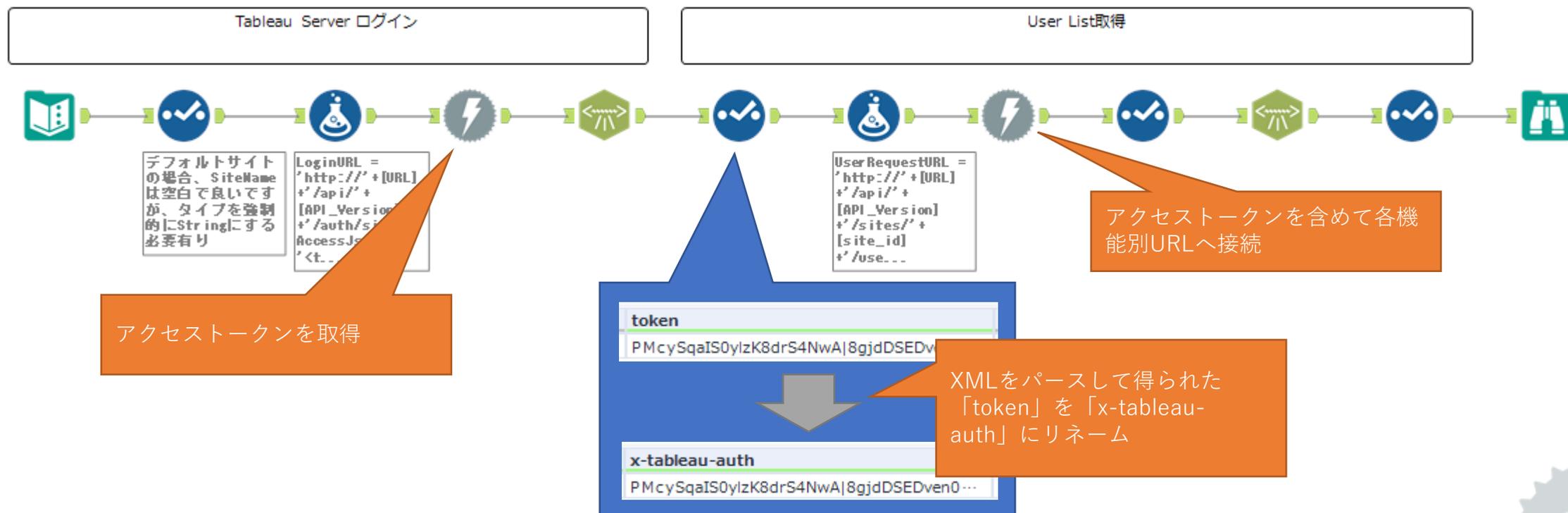
- APIキーをヘッダに含めてアクセスするタイプ
- このタイプもあっさり構築可能です
- 同様のタイプ
 - Google Storage API
- Alteryxのメールを送る機能が使いにくいような場合はSendGridを利用してみてもいいかでしょうか？

具体例 3 : REST API (Tableau)

URL	<p>ログイン : <code>http://[URL]/api/[API_Version]/auth/signin</code></p> <p>ユーザーリスト取得 : <code>http://[URL]/api/[API_Version]/sites/[site_id]/users?pageSize=1000</code></p> <p>※行いたいアクションに応じてURLが変わります</p> <p>※[URL]の部分はみなさんが設置しているTableauServerのURLです。Onlineの場合はアクセスしている時のURLから持ってきます (例 : 「us-west-2b.online.tableau.com」 など)</p>
ヘッダ	<p>トークン取得まで「なし」</p> <p>トークン取得後「x-tableau-auth : [取得トークン]」</p>
HTTPアクション	トークン取得まで「POST」、ユーザーリスト取得「GET」等
各種パラメータ	<p>トークン取得まで「XML形式でフィールドを作成し、ペイロードで指定」、リスト取得の場合はURLに対してパラメータを指定するのみ</p> <p>認証時のパラメータ : <tsRequest><credentials name=[ユーザーID] password=[パスワード]><site contentUrl="[サイト名]" /></credentials></tsRequest></p> <p>※Siteがデフォルトサイトの場合は[サイト名]は空白。</p>

サンプルワークフロー

必ず最初にユーザーIDとパスワードを使ってTableau Serverにログインし、アクセストークンを取得する必要があります。その後、やりたいことに応じて指定されたURLに、アクセストークンをヘッダにセットしアクセスします。



ダウンロードツールの設定（認証時）

ダウンロード (2) - 設定

基本設定 ヘッダー パイロード 接続

URL
フィールド

LoginURL

URLテキストをエンコードする

出力
フィールドへ

文字列

データ

BLOB

ファイルへ

一時ファイル

フィールドからのファイル名

URL

http://[URL]/api/[API_Version]
/auth/signin

※ヘッダは設定不要

ダウンロード (2) - 設定

基本設定 ヘッダー **パイロード** 接続

HTTP アクション POST

作成クエリ文字列/本文

以下の定数値から

名前	値
----	---

追加
削除

これらのフィールドからの値

URL
 SiteName
 UserName
 UserPass
 API_Version
 LoginURL
 AccessJson

コンテンツタイプ application/x-www-form-urlencoded

フィールドからクエリ文字列/本文を取得する

AccessJson

認証時のパラメータ：
<tsRequest><credentials
name=[ユーザーID]
password=[パスワード]><site
contentUrl=""
></credentials></tsRequest
>

※フィールド名は「AccessJson」と記載ありますが、中身はXML形式です

ダウンロードツールの設定（ユーザーリスト取得時）

ダウンロード (2) - 設定

基本設定 | ハッダー | パイロード | 接続

URL
フィールド
LoginURL

URLテキストをエンコードする

出力
フィールドへ
 文字列
データを次の形式でエンコード icode UTF-8

BLOB

ファイルへ
 一時ファイル
 フィールド

URL

http://[URL]/api/[API_Version]
/sites/[site_id]/users?pageSi
ze=1000

このうち、[site_id]
「API_Version」は認証時に取
得するデータを利用します

ダウンロード (8) - 設定

基本設定 | ハッダー | パイロード | 接続

以下の定数値から追加する

名前	値
----	---

これらのフィールドからの値

URL
 API_Version
 site_id
 x-tableau-auth
 SiteName
 UserName
 UserPass

ダウンロード (8) - 設定

基本設定 | ハッダー | **パイロード** | 接続

HTTPアクション GET (またはFTP)

作成クエリ文字列/本文

以下の定数値から

名前	値
----	---

これらのフィールドからの値

URL
 API_Version
 site_id
 x-tableau-auth
 SiteName
 UserName
 UserPass

コンテンツタイプ application/x-www-form-urlencoded

フィールドからクエリ文字列/本文を取得する
URL

クエリ文字列/本文で次を使用します

具体例 3 : REST API (Tableau)

【まとめ】

- ログインIDなどからアクセストークンを取得して、そのアクセストークンでアクセスするタイプ
- ログインID、パスワードを用いて一度認証を通し、得られたトークンをRestAPIでアクセスするたびに含める必要があるため、若干手間が増えます。よって、必ず以下のようなWFになります。



- 同様のタイプ
 - Pardot
 - LINE
- TableauのRestAPIは最初の認証を通すところのURLがわかりにくくハマるポイントです
- ユーザーの一括ライセンス割当やアクセス状況の確認などできてTableau Serverの管理が楽になります

その他使ったことがあるもの

- **LINE Works API**

- セキュリティがガチガチなのかJWTを利用したアクセストークンの取得が必要になり、Alteryxで実装しようとする、JWTで利用するSHA-256の暗号化アルゴリズムを利用するにはRかPythonツールに頼る必要があるため、難易度が非常に高いです（通常のLINEのAPIはそこまで難しくなさそう）。 ※結局その部分だけPythonツールで実装しましたが、生成するだけなら<https://jwt.io/> というサービスもあるようです

- **Google Maps API**

- 細かいパラメータはURLに含めます（ヘッダもペイロードも不要なものもあります）
- 出力データがバイナリ形式（画像）のものがあるため、受信フィールドはBLOBタイプにする必要があります（Map Static、StreetViewStatic）
- DirectionsAPIはルートの詳細が取れますが、デコードは一定のアルゴリズムで行う必要がありWF難度は高いです。簡易ルートなら簡単に扱えます。

- **Google Storage API**

- 一部機能はファイルのセキュリティレベルに応じてアクセストークンを取得する必要があるものがあります

- **Redmine**

- 素直

APIキーって漏れても大丈夫？

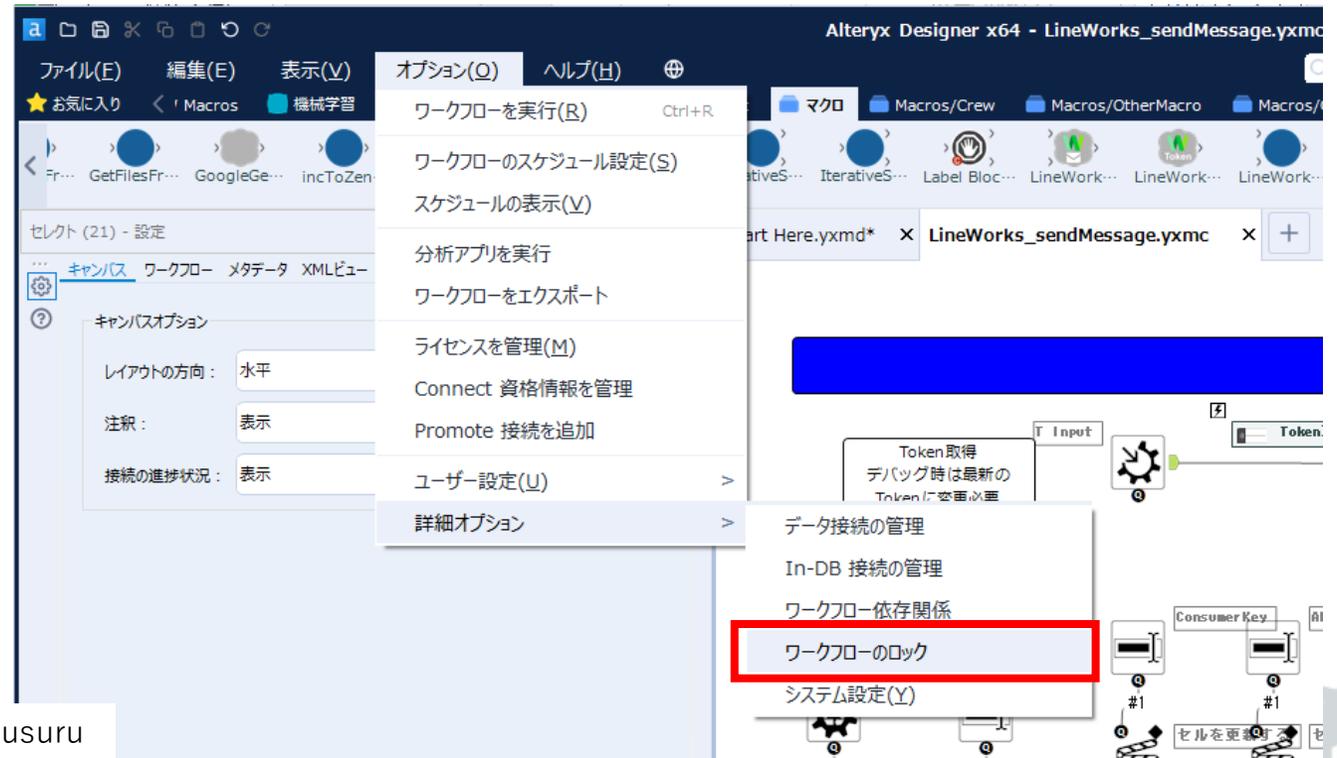
ダメです。外部に漏れると無限に使われる可能性がありますし、不正アクセスに繋がる可能性があります。

幸い、Alteryxにはワークフローの暗号化機能があります。

社内展開の際は、マクロに埋め込んで暗号化してしまうのが良いかと思えます。

※誤って上書きとかしてしまうと二度とWFが編集できなくなるのでご注意ください

<https://help.alteryx.com/ja/designer/wakufuroworotsukusuru>



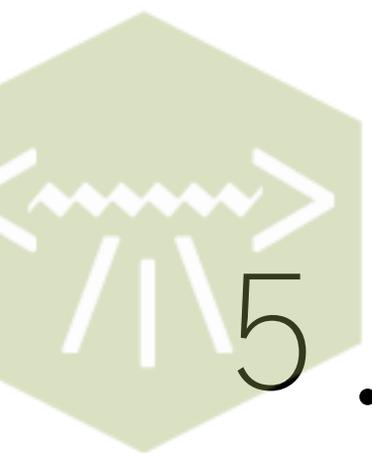
まとめ

【RestAPIを使う利点】

- Alteryxの機能以外が使えるようになるため、できることが無限大（！？）
- 利用しているWEBサービスから色々なデータを取得することで新たな分析が可能に！
- オープンデータや外部システムが開放しているデータを使えるため新たな角度の分析が可能に！

【今回お話しした内容】

- RestAPI利用に必須となるダウンロードツールの設定方法
- 認証系で、APIキーをただ含めて送るだけのものと、アクセストークンを取得しないといけないタイプの2つがある
- 受信データのJSONとXMLのパーズについて



5.

Appendix

やってみましょう

~Weekly Challengeへのいざない~

- Weekly Challenge #7 (英語版) がRestAPIを叩いて取得したデータをパースするお題です Advanced Data Parsing

インプット

コード	URL
1	https://www.quandl.com/api/v3/datasets/UTOR/TOR_USA.json?api_key=5aMivNdsRkZNB-afkjse

アウトプット

コード	Date	Arrivals - Thousands (TF)	Tourism expenditure in the country - US...	Tourism expenditure in the country - US...	Tourism expenditure in the country - US...
1	2017-12-31	76941	251361	210748	40613
2	2016-12-31	76407	246172	206901	39271
3	2015-12-31	77774	249183	206936	42247
4	2014-12-31	75022	235990	191919	44071
5	2013-12-31	69995	218496	177483	41013
6	2012-12-31	66657	200997	161633	39364
7	2011-12-31	62821	187629	150866	36763
8	2010-12-31	60010	167996	137010	30986
9	2009-12-31	55103	146002	119900	26102
10	2008-12-31	58007	164721	133765	30956
11	2007-12-31	56135	144223	119037	25186
12	2006-12-31	50977	126778	105141	21637
13	2005-12-31	49206	122077	101469	20608

宣伝？

Shape Fileで日本語のフィールド名を使いたい人！

(ご協力下さい！)

- AlteryxだとShapeFile（空間ファイル）の項目名に日本語が使えません
- 他のGISソフトだと使えます（フリーのソフトでさえ！）
- Communityの「New Idea」に使いたい！と投稿したので、反応（イイネやコメント）いただけると、実装されるかもしれません

Allow 2byte character field name at Shape File

<https://community.alteryx.com/t5/Alteryx-Designer-Ideas/Allow-2-byte-character-field-name-at-Shape-File/idi-p/617109>

宣伝その2

ブログ書いてます！

Alteryxの小技をご紹介する「**KCME TechBlog**」

<https://newssdx.kcme.jp/category/analytics/alteryx/>



検索

今回の関連URL

【AlteryxTips】AlteryxからTableauServerのRestAPIを叩いてみる

https://newssdx.kcme.jp/alteryx-tips-tableau_restapi/

【AlteryxTips】Google Maps APIを使ったストリートマップ取得の方法（Street View Static API）

https://newssdx.kcme.jp/alteryxtips-google_maps_api_streetviewstaticapi/

【AlteryxTips】Google Maps APIを使ったジオコーディングの方法（Geocoding API）

https://newssdx.kcme.jp/alteryxtips-google_maps_api_geocodingapi/