Tegaki コネクタ

ユーザーガイド

版 1.0

目次

[概要 1](#_Toc43909643)

[なぜ、Tegakiコネクタが必要か 1](#_Toc43909644)

[Tegakiコネクタのメリット 1](#_Toc43909645)

[Tegakiコネクタを構成する 1](#_Toc43909646)

[前提条件 1](#_Toc43909647)

[構成手順 2](#_Toc43909648)

[稼働確認 2](#_Toc43909649)

[オブジェクトの入力 2](#_Toc43909650)

[オブジェクトの出力 3](#_Toc43909651)

[全体の動き 3](#_Toc43909652)

[拡張について 3](#_Toc43909653)

[拡張の考え方 3](#_Toc43909654)

[拡張のポイント 4](#_Toc43909655)

# 概要

## なぜ、Tegakiコネクタが必要か

AI-OCRエンジンとして名高いCogent Labs社のTegakiをBlue Prismから利用するには、TegakiのUIまたはWeb APIを操作する必要がある。いずれを用いるにせよ、前処理（ファイル読込など）や後処理（読み取り結果の構造化など）は必要となるため、Blue Prismのオブジェクト（VBO）としてTegakiコネクタを開発した。

## Tegakiコネクタのメリット

Blue Prismユーザーは、このアセットを活用し、またユーザー固有の要件に基づいて拡張をすることで、以下のようなメリットを享受できる。

1. Blue PrismとTegakiを連携させる際に、連携に必要な前処理、後処理にかかる開発工数・時間を削減できる
2. Blue PrismとTegakiを連携させる際のユースケースに応じて、開発済みの部品を拡張できる
3. Blue PrismからREST APIを呼ぶ際に必要な、基本的な処理を学習できる

# Tegakiコネクタを構成する

## 前提条件

1. Blue Prism 6.8で構成されたエンタープライズRPA環境[[1]](#footnote-1)
2. Tegaki環境およびWeb APIでアクセスするためのAPIキー[[2]](#footnote-2)
3. 以下のオブジェクト（VBO）。すべてBlue Prism DX（<https://digitalexchange.blueprism.com/dx/search>）から入手可能
   1. Base64Encoder
   2. Utility – General
   3. Utility – JSON
   4. Utility – Blue Prism Process Info
   5. Utility – File Management

## 構成手順

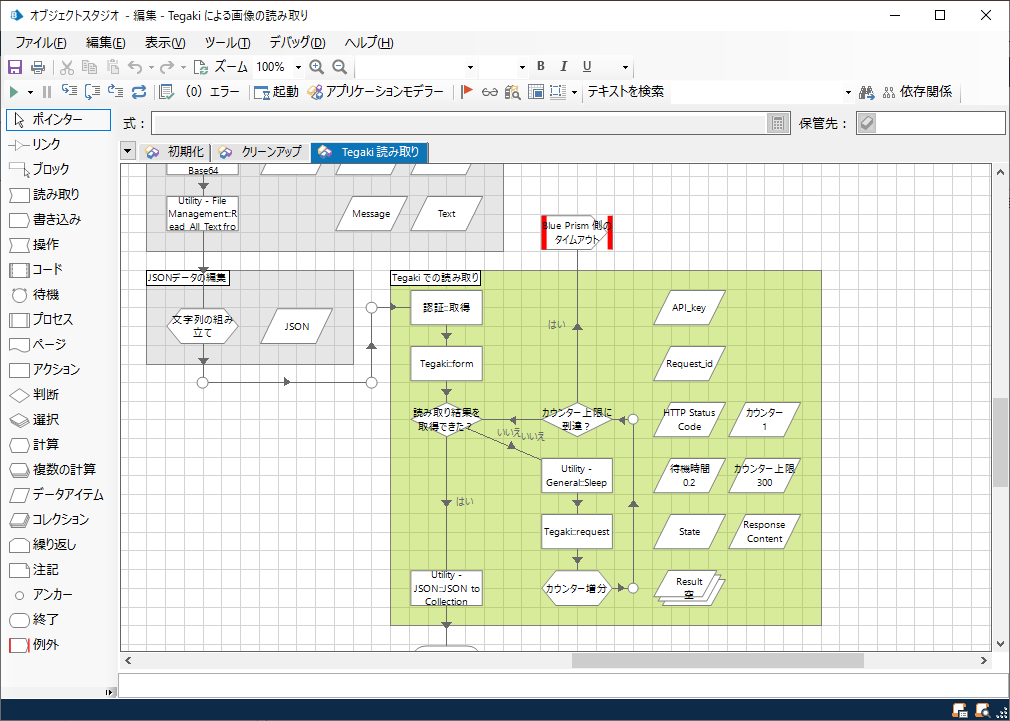
### bprelease のインポート

Tegaki コネクタ v1.0.bpreleaseをBlue Prismにインポートする。

### 認証情報の設定

認証情報「Tegaki API key」のトークンにTegaki の api key の値を設定し、適切な権限付与を行う。

### チューニング



データアイテム「待機時間」と「カウンター上限」を使って、画像アップロード後のTegakiの処理完了を待っている。「待機時間」はTegakiに処理完了を問い合わせる間隔で、カウンター上限は問合せ回数の上限を意味している。環境に合わせて、これらの値を調整できる。

# 稼働確認

## オブジェクトの入力

Tegaki コネクタの入力は２項目ある。

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **項目名** | **型** | **概要** |
| 1 | 画像イメージのファイルパス | テキスト | 画像イメージのファイルパス（フルパス） |
| 2 | テンプレート JSON のファイルパス | テキスト | テンプレート JSON のファイルパス（フルパス） |

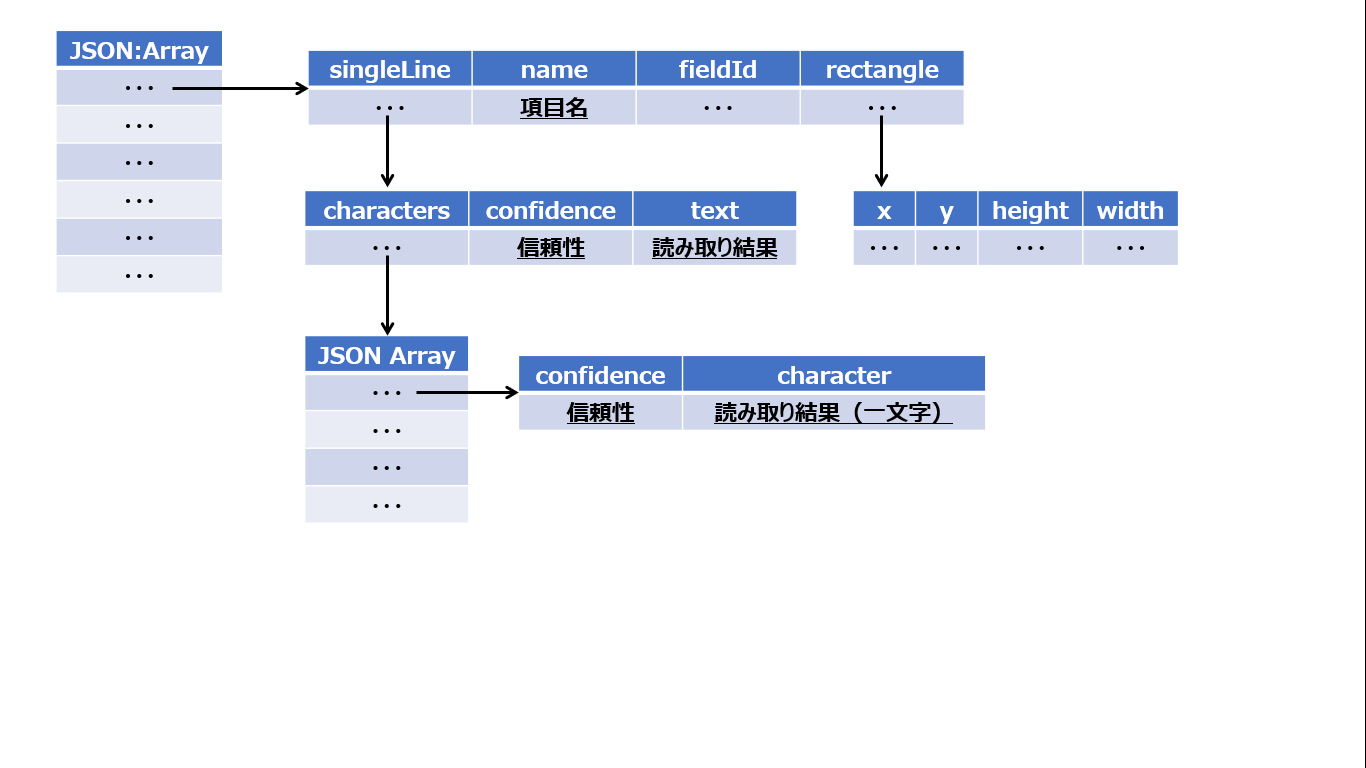
適当なプロセスを作成し、項目に適切な値を設定する。

## オブジェクトの出力

Tegaki コネクタの出力は２項目ある。

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **項目名** | **型** | **概要** |
| 1 | 読み取り結果（コレクション） | コレクション | 読み取り結果（構造体） |
| 2 | 読み取り結果（テキスト） | テキスト | 読み取り結果（JSON形式のテキスト） |

コレクションの構造体は以下のようになっている。



## 全体の動き

全体の動きは、デモ動画を参照。

<https://www.youtube.com/watch?v=o_KJZxu5ACY>

# 拡張について

## 拡張の考え方

大きく分けて２つの方向性が考えられる。

1. Tegaki コネクタそのものの精度向上
2. Tegaki コネクタへの機能追加

## 拡張のポイント

主な拡張のポイントは以下である。

1. request アクションの結果の State に応じた処理の追加
2. より多くの API への対応

以上

1. 6.8で動作確認をしているが、基本的な構造は6.4以降の機能で構成されているため、6.4以降であれば多少の修正で動くことが見込める [↑](#footnote-ref-1)
2. TegakiのAPIはv2を前提としている [↑](#footnote-ref-2)