



目次

- 1. 改訂情報
- 2. はじめに
 - 2.1. 本書の目的
 - 2.2. 対象読者
 - 2.3. 本書の構成
- 3. IM-RPAとは
 - 3.1. 概要
 - 3.2. WinActor
 - 3.3. BizRobo!
 - 3.4. UiPath
- 4. WinActor連携
 - 4.1. セットアップ
 - 4.1.1. IM-Juggling プロジェクトの編集
 - 4.1.2. ライセンスコードの登録
 - 4.1.3. エージェントのダウンロード
 - 4.1.4. エージェントの起動
 - 4.2. 設定ファイル
 - 4.2.1. 概要
 - 4.2.2. リファレンス
 - 4.3. IM-LogicDesigner タスク説明
 - 4.3.1. エージェント呼び出し
- 5. BizRobo!連携
 - 5.1. セットアップ
 - 5.1.1. IM-Juggling プロジェクトの編集
 - 5.1.2. ライセンスコードの登録
 - 5.2. 設定ファイル
 - 5.2.1. 概要
 - 5.2.2. リファレンス
 - 5.3. IM-LogicDesigner タスク説明
 - 5.3.1. ロボットの配置
 - 5.3.2. ロボットの削除
 - 5.3.3. タイプの配置
 - 5.3.4. タイプの削除
 - 5.3.5. スニペットの配置
 - 5.3.6. スニペットの削除
 - 5.3.7. リソースの配置
 - 5.3.8. リソースの削除
 - 5.3.9. ライブラリの展開
 - 5.4. IM-LogicDesigner ユーザ定義説明
 - 5.4.1. BizRobo!定義の新規作成
- 6. UiPath連携
 - 6.1. セットアップ
 - 6.1.1. IM-Juggling プロジェクトの編集
 - 6.1.2. ライセンスコードの登録
 - 6.2. 設定ファイル
 - 6.2.1. 概要
 - 6.2.2. リファレンス
 - 6.3. IM-LogicDesigner タスク説明
 - 6.3.1. ジョブステータス取得
 - 6.4. IM-LogicDesigner ユーザ定義説明
 - 6.4.1. UiPath定義の新規作成
- 7. 著作権および特記事項

改訂情報

変更年月日	変更内容
-------	------

2020-09-01	初版
------------	----

はじめに

本書の目的

本書は、IM-RPA連携を利用するユーザのみなさまの支援を目的としたドキュメントです。

対象読者

本書では以下のユーザを対象としています。

- WinActor連携を利用し、シナリオをリモートから実行したい
- BizRobo!連携を利用し、BizRobo!のサービスを呼び出したい
- UiPath連携を利用し、UiPathをリモートから実行したい

本書の構成

本書は以下のように構成されています。

- [IM-RPAとは](#)
本書、および、IM-RPA連携の概要について説明します。
- [WinActor連携](#)
WinActor連携のセットアップ手順、およびIM-LogicDesignerタスクの仕様を説明します。
- [BizRobo!連携](#)
BizRobo!連携のセットアップ手順、およびIM-LogicDesignerタスクの仕様を説明します。
- [UiPath連携](#)
UiPath連携のセットアップ手順、およびIM-LogicDesignerタスクの仕様を説明します。
- [著作権および特記事項](#)
著作権および特記事項について記載します。

IM-RPAとは

項目

- [概要](#)
- [WinActor](#)
- [BizRobo!](#)
- [UiPath](#)

概要

IM-RPAは、WinActor, BizRobo!, UiPathとの接続を容易に行うための IM-LogicDesignerタスクを提供します。
 IM-LogicDesignerを介す事により、IM-Workflowや、IM-BPMといった機能との連携を容易にします。
 IM-RPAを利用することにより、業務プロセス中に任意のロボットを組み込み業務の自動化を推進します。

WinActor

WinActorは、NTTグループにより開発・利用されてきた長い歴史と豊富な導入実績に裏打ちされた機能を備えた純国産「RPA」ソリューションです。
 WinActor連携タスクでは、複数の WinActorクライアントをグループ化し、シナリオの実行を行うことが可能です。

BizRobo!

BizRobo!は、ホワイトカラーの生産性を革新する、ソフトウェアロボット（Digital Labor）の導入・運用を支援するデジタルレイバープラットフォームです。
 「ロボット」と「IT」によって、ホワイトカラーをルーティンワークから解放し、企業を始め社会全体の生産性向上を図り、未来の働き方を変えていきます。
 BizRobo!連携タスクでは BizRobo!に対して、シナリオのアップロード、シナリオの実行等を行うことが可能です。
 これにより、IM-Workflowにより承認されたシナリオを BizRobo!へアップロードするといった使い方により、ロボットの統制を行うことも可能です。

UiPath

UiPathは、気軽にRPAロボットを作成・実行できるUiPath Studioと、作成したロボットを組織全体で管理・運用するUiPath Orchestratorを備えるRPA製品です。
 UiPath連携タスクでは UiPath Orchestratorに対して、ジョブの実行等を行うことが可能です。
 UiPath Orchestratorを介することにより、ロボット実行の非同期化が可能となり、これにより大規模なロボット運用をより柔軟に行うことが可能です。

WinActor連携

ここでは、WinActor連携のセットアップ手順、およびIM-LogicDesignerタスクの仕様を説明します。

セットアップ

項目

- IM-Juggling プロジェクトの編集
- ライセンスコードの登録
- エージェントのダウンロード
- エージェントの起動

IM-Juggling プロジェクトの編集

WinActor連携は、IM-RPAモジュールを使用します。

以下の手順で設定を行ってください。

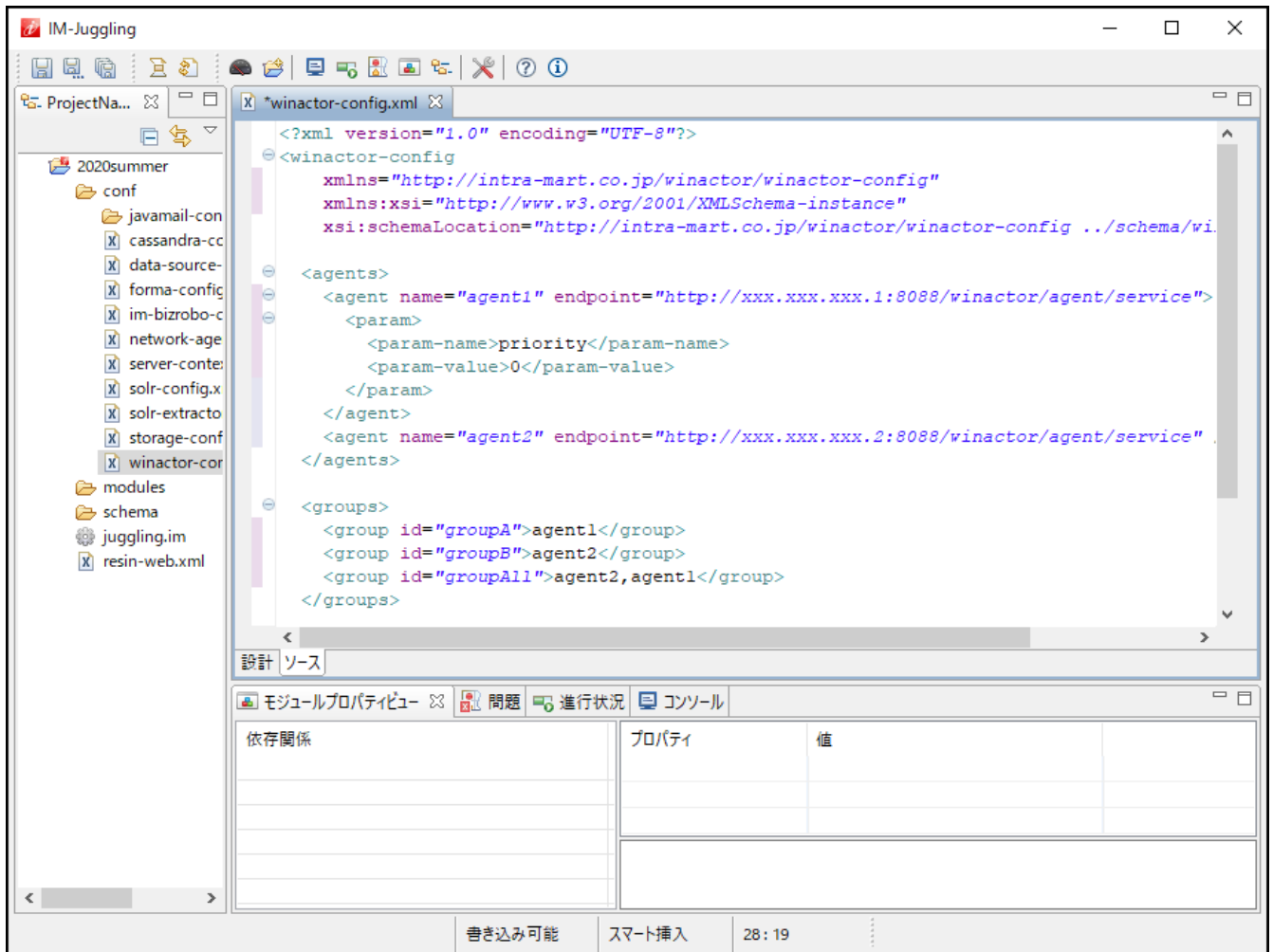
1. IM-Jugglingにて、プロジェクトにIM-RPAモジュールを追加してください。
以下のドキュメントを参照してください。
 - 「[intra-mart Accel Platform セットアップガイド](#)」 - 「[プロジェクトの作成とモジュールの選択](#)」



注意

IM-RPAモジュールのご利用には、エンタープライズ版の構成、およびライセンスが必要です。

2. 「設定ファイルが存在しません」という赤字のメッセージをクリックします。
表示されるダイアログにて「OK」をクリックし、このプロジェクトに「WinActorクライアント設定ファイル」(winactor-config.xml)を追加します。
3. 「ProjectNavigator」内の「< (プロジェクト名) /conf/winactor-config.xml> ファイル」をダブルクリックで開き、「ソース」を選択します。



4. <winactor-config>/<agents>/<agent> タグのendpoint属性に対して、接続先エージェントのエンドポイントを記述します。エージェントが複数存在する場合は、<agent>タグを繰り返し設定してください。エンドポイントは以下の形のURLです。ポート番号は標準では8088を利用します。

http://<エージェントのアドレス>:8088/winactor/agent/service

5. <winactor-config>/<retry>/<max-count> および <wait-seconds> タグに、エージェントへの接続試行回数と待ち時間を設定します。
6. 「WinActorクライアント設定ファイル」を保存します。
7. IM-JugglingでWARファイルを出力し、アプリケーションサーバへデプロイを行ってください。

i コラム

エージェントを選出するエージェントセレクトに対して、各エージェント別にパラメータを与えることができます。ただし、エージェントセレクトの標準実装では、このパラメータは利用していません。エージェントセレクトを独自実装で差し替えた場合、必要に応じてパラメータを名前と値の組み合わせで記述可能です。<agent>タグ内に<param>タグ、および<param-name>、<param-value>タグの組み合わせで設定してください。

ライセンスコードの登録

IM-RPAを利用するためには、テナント環境セットアップ後、製品ライセンスコードの登録が必要です。テナント環境セットアップについては、「[intra-mart Accel Platform セットアップガイド](#)」-「テナント環境セットアップ」を参照してください。

1. IM-RPAのライセンス登録を行います。
詳細は「[ライセンスの登録](#)」を参照してください。

エージェントのダウンロード

WinActorエージェントは、Product File Download (<https://product.intra-mart.jp/srcdownload/>)よりダウンロードしてください。

ダウンロードには、IM-RPAのライセンスキーが必要です。

RPASDという文字列から始まるライセンスキーを入力することでWinActorエージェントのダウンロードリンクが表示されます。

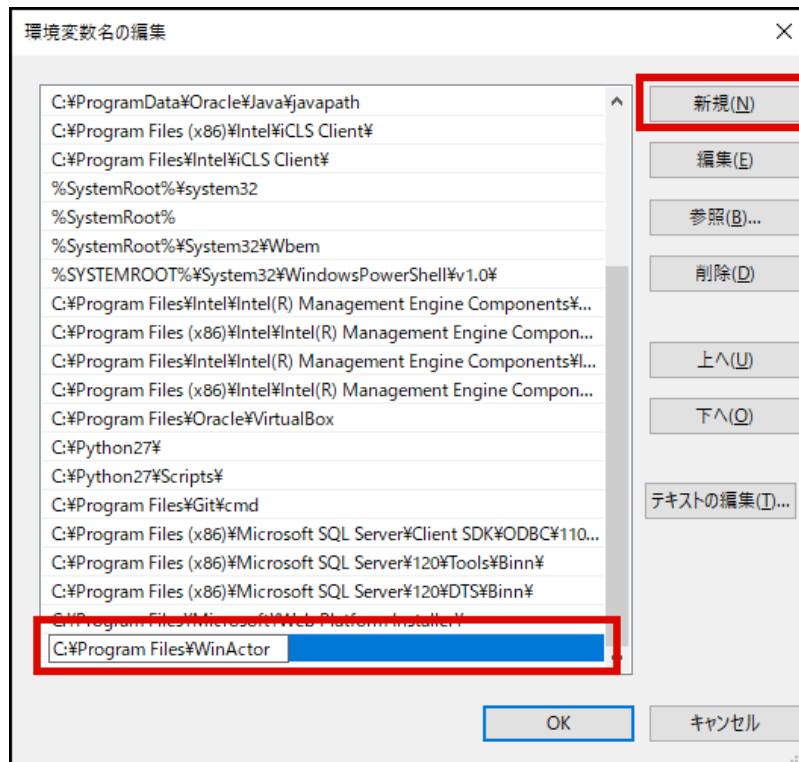
エージェントの起動

WinActorエージェントは、実行可能jar形式で同梱しています。

WinActorがインストールされている各端末にて、以下の手順を実施し、エージェントを起動してください。

1. Java 8 JDKをインストールします。
2. WinActorの実行ファイルが存在しているフォルダのパスを、システム環境変数 **PATH** に設定します。

例) WinActorを C:\Program Files\WinActor にインストールした場合
システム環境変数 **PATH** の編集画面にて、上記パスを新規追加します。



3. 「im_winactor_agent-8.x.x.jar」を任意のフォルダに配置します。
4. jarファイルを配置したフォルダにて、以下のコマンドを実行し、エージェントを起動します。

```
java -jar im_winactor_agent-8.x.x.jar
```

5. エージェントが起動すると、リクエスト受付の待機状態が始まります。
アプリケーションサーバからのリクエストを受信すると、WinActorが自動的に起動され、シナリオが実行されます。

i コラム

エージェントの起動時に、以下の表のシステムプロパティをJavaの引数として `-Dxxx=xxx` 形式で指定可能です。
次のコマンドのように、引数は `-jar` よりも前に指定してください。

```
java -Dserver.port=18088 -jar im_winactor_agent-8.x.x.jar
```

プロパティ名	デフォルト値	意味
server.port	8088	エージェントのポート番号
process.timeout.default	1800	シナリオ実行のタイムアウト時間のデフォルト値 (秒)
path.exec.winactor	WinActor7.exe	WinActorの実行ファイル名
logging.file	winactor_agent.log	出力ログファイル名
logging.level.jp.co.intra_mart.foundation.winactor.AgentRestController	INFO	出力ログレベル
spring.servlet.multipart.max-file-size	200MB	シナリオファイル/データ一覧ファイルの許容最大サイズ
spring.servlet.multipart.max-request-size	200MB	シナリオファイルなどを含む、エージェントへの通信の許容最大サイズ

設定ファイル

項目

- 概要
- リファレンス
 - エージェント設定
 - エージェント個別設定
 - パラメータ設定
 - パラメータ名前設定
 - パラメータ値設定
 - グループ設定
 - グループ個別設定
 - リトライ設定
 - リトライ最大回数設定
 - リトライ待ち秒数設定

概要

WinActor連携 に関する設定です。

モジュール	WinActor連携
フォーマットファイル(xsd)	WEB-INF/schema/winactor-config.xsd
設定場所	WEB-INF/conf/winactor-config.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<winactor-config
  xmlns="http://intra-mart.co.jp/winactor/winactor-config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://intra-mart.co.jp/winactor/winactor-config ../schema/winactor-config.xsd">

  <agents>
    <agent name="agent1" endpoint="http://xxx.xxx.xxx.1:8088/winactor/agent/service">
      <param>
        <param-name>priority</param-name>
        <param-value>0</param-value>
      </param>
    </agent>
    <agent name="agent2" endpoint="http://xxx.xxx.xxx.2:8088/winactor/agent/service" />
  </agents>

  <groups>
    <group id="groupA">agent1</group>
    <group id="groupB">agent2</group>
    <group id="groupAll">agent2,agent1</group>
  </groups>

  <retry>
    <max-count>1</max-count>
    <wait-seconds>5</wait-seconds>
  </retry>

</winactor-config>
```

リファレンス

エージェント設定

タグ名 agents

接続先エージェント全てに関する設定を定義します。

【設定項目】

```
<winactor-config>
  <agents>
    :
  </agents>
</winactor-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	agents タグを親とするタグ
単位・型	なし
省略時のデフォルト値	なし
親タグ	winactor-config

エージェント個別設定

タグ名 agent

接続先エージェント一つに関する設定を定義します。

【設定項目】

```
<winactor-config>
<agents>
  <agent name="agent1" endpoint="http://xxx.xxx.xxx.1:8088/winactor/agent/service">
  :
  </agent>
  <agent name="agent2" endpoint="http://xxx.xxx.xxx.2:8088/winactor/agent/service" />
</agents>
</winactor-config>
```

必須項目	○
複数設定	○
設定値・設定する内容	agent タグを親とするタグ
単位・型	なし
省略時のデフォルト値	なし
親タグ	agents

【属性】

属性名	説明	必須	デフォルト値
name	エージェント名	○	なし
endpoint	接続先エンドポイントのURL	○	なし

パラメータ設定

タグ名 param

このエージェントに関するパラメータを一つ設定します。

【設定項目】

```
<winactor-config>
<agents>
  <agent name="agent1" endpoint="http://xxx.xxx.xxx.1:8088/winactor/agent/service">
    <param>
    :
    </param>
  </agent>
</agents>
</winactor-config>
```

必須項目	×
複数設定	○
設定値・設定する内容	param タグを親とするタグ
単位・型	なし
省略時のデフォルト値	なし
親タグ	agent

パラメータ名前設定

タグ名 param-name

このパラメータの名前を設定します。

【設定項目】

```
<winactor-config>
  <agents>
    <agent name="agent1" endpoint="http://xxx.xxx.xxx.1:8088/winactor/agent/service">
      <param>
        <param-name>priority</param-name>
      :
    </param>
  </agent>
</agents>
</winactor-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	このパラメータの名前
単位・型	文字列
省略時のデフォルト値	なし
親タグ	param

パラメータ値設定

タグ名 param-value

このパラメータの値を設定します。

【設定項目】

```
<winactor-config>
  <agents>
    <agent name="agent1" endpoint="http://xxx.xxx.xxx.1:8088/winactor/agent/service">
      <param>
        :
        <param-value>0</param-value>
      </param>
    </agent>
  </agents>
</winactor-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	このパラメータの値
単位・型	文字列
省略時のデフォルト値	なし
親タグ	param

グループ設定

タグ名 groups

エージェントのグループ分けを設定します。

【設定項目】

```
<winactor-config>
  <groups>
    :
  </groups>
</winactor-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	groups タグを親とするタグ
単位・型	なし
省略時のデフォルト値	なし
親タグ	winactor-config

グループ個別設定

タグ名 group

グループの個別設定をします。

【設定項目】

```
<winactor-config>
  <groups>
    <group id="groupA">agent1</group>
    <group id="groupB">agent2</group>
    <group id="groupAll">agent2,agent1</group>
    :
  </groups>
</winactor-config>
```

必須項目	○
複数設定	○
設定値・設定する内容	対象のエージェント名（agentで指定した名前） 複数指定する場合はカンマ区切りで指定。
単位・型	文字列
省略時のデフォルト値	なし
親タグ	groups

【属性】

属性名	説明	必須	デフォルト値
id	グループID	○	なし

リトライ設定

タグ名 retry

エージェントへの接続の際、設定したエージェント全てが実行中であったときのリトライに関する設定を定義します。

【設定項目】

```
<winactor-config>
  <retry>
  :
  </retry>
</winactor-config>
```

必須項目

複数設定

設定値・設定する内容 retry タグを親とするタグ

単位・型 なし

省略時のデフォルト値 なし

親タグ winactor-config

リトライ最大回数設定

タグ名 max-count

リトライの最大回数を設定します。

0を設定した場合、リトライを行いません。

【設定項目】

```
<winactor-config>
  <retry>
    <max-count>3</max-count>
  :
  </retry>
</winactor-config>
```

必須項目

複数設定

設定値・設定する内容 リトライの最大回数

単位・型 整数値 (0-)

省略時のデフォルト値 なし

親タグ retry

リトライ待ち秒数設定

タグ名 wait-seconds

次にリトライするまでの待ち時間の秒数を設定します。

【設定項目】

```
<winactor-config>
  <retry>
  :
    <wait-seconds>10</wait-seconds>
  </retry>
</winactor-config>
```

必須項目

複数設定	×
設定値・設定する内容	リトライするまでの待ち時間の秒数
単位・型	整数値 (0-)
省略時のデフォルト値	なし
親タグ	retry

IM-LogicDesigner タスク説明

項目

- エージェント呼び出し
 - 入力値
 - 出力値

エージェント呼び出し

WinActorエージェントを呼び出し、シナリオを実行するタスクです。

エージェントセレクタの標準実装では、エージェントは登録されている中からランダムな順序で選択されます。

選択されたエージェントが現在シナリオ実行中であるならば、次のエージェントが選択されます。

すべてのエージェントがシナリオ実行中の場合は、設定ファイルにて指定した時間だけ待った後、最初からリトライを行います。

リトライ回数が設定された最大回数を超えた場合、エラーが発生します。

シナリオの実行結果として、更新されたデータ一覧ファイルや、特定のパスのファイルを取得することが可能です。

コラム

ロジックフロー定義編集画面にて、WinActorのカテゴリおよびタスクのアイコンが表示されない場合、アイコンのリカバリを実行することで表示されるようになる場合があります。

詳細は、「[IM-LogicDesigner ユーザ操作ガイド](#)」 - 「[ユーザアイコンをリカバリする](#)」を参照してください。

入力値

```
im_winactorCallAgent <object>
├─ dataFile <storage>
├─ groupId <string>
├─ resultFilePath <string>
├─ scenario <storage> *
├─ timeout <integer>
└─ vmargs <string>
```

項目名	必須/任意	型	配列/リスト	説明
im_winactorCallAgent	任意	object	なし	-
dataFile	任意	storage	なし	シナリオにて利用するデータ一覧ファイルを指定してください。
groupId	任意	string	なし	実行するロボットのグループIDを指定してください。
resultFilePath	任意	string	なし	シナリオによって出力された結果ファイルを取得したい場合に指定します。 エージェントのファイルシステム上の絶対パスを指定してください。
scenario	必須	storage	なし	実行するシナリオファイルを指定してください。
timeout	任意	integer	なし	シナリオ実行のタイムアウト時間を指定します。 (単位: 秒) エージェントとの通信のタイムアウト時間ではありません。

項目名	必須/任意	型	配列/リスト	説明
vmargs	任意	string	なし	WinActorに与えるVM引数 (-VM) の内容を指定します。 詳細については、WinActorのドキュメントに含まれる 「WinActor簡易マニュアル」を参照してください。

! 注意

エージェントを呼び出してシナリオを実行している間、アプリケーションサーバとエージェントとの間の通信のセッションは維持され続けている必要があります。
適切に `timeout` を設定し、通信のタイムアウトが発生する前にエージェントがレスポンスを返すようにしてください。

出力値

```
im_winactorCallAgent <object>
├─ dataFile <binary>
├─ resultFile <binary>
├─ stderrLog <binary>
└─ stdoutLog <binary>
```

項目名	型	配列/リスト	説明
im_winactorCallAgent	object	なし	-
dataFile	binary	なし	シナリオによって更新されたデータ一覧ファイル
resultFile	binary	なし	入力値 <code>resultFilePath</code> によって指定された結果ファイル
stderrLog	binary	なし	WinActorが標準エラーに出力した内容 (通常、何も出力されません)
stdoutLog	binary	なし	WinActorが標準出力に出力した内容 (通常、何も出力されません)

BizRobo!連携

ここでは、BizRobo!連携のセットアップ手順、およびIM-LogicDesignerタスクの仕様を説明します。

セットアップ

項目

- IM-Juggling プロジェクトの編集
- ライセンスコードの登録

IM-Juggling プロジェクトの編集

BizRobo!連携は、IM-RPAモジュールを使用します。
以下の手順で設定を行ってください。

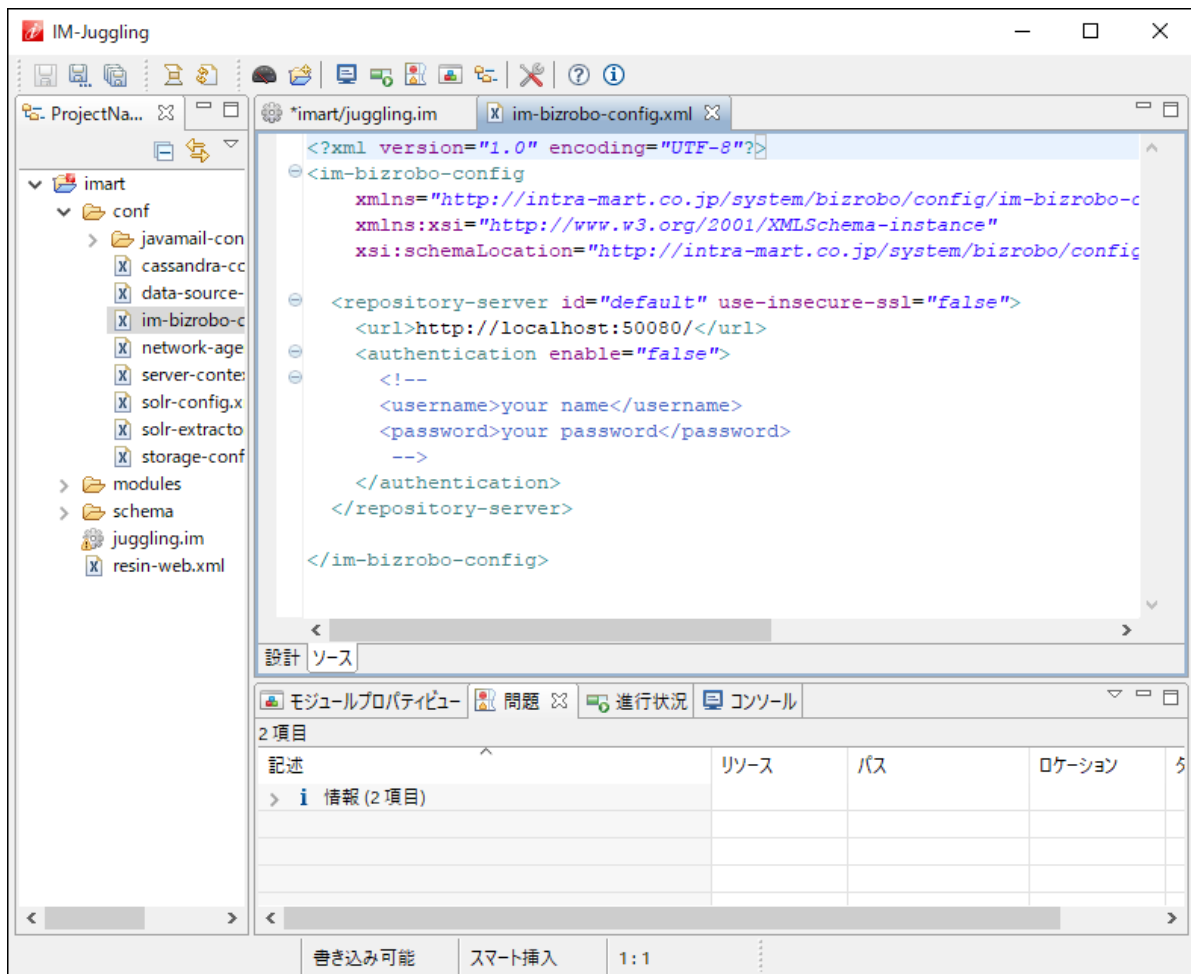
1. IM-Jugglingにて、プロジェクトにIM-RPAモジュールを追加してください。
以下のドキュメントを参照してください。
 - 「[intra-mart Accel Platform セットアップガイド](#)」 - 「[プロジェクトの作成とモジュールの選択](#)」



注意

IM-RPAモジュールのご利用には、エンタープライズ版の構成、およびライセンスが必要です。

2. 「設定ファイルが存在しません」という赤字のメッセージをクリックします。
表示されるダイアログにて「OK」をクリックし、このプロジェクトに「BizRobo!クライアント設定ファイル」(im-bizrobo-config.xml)を追加します。
3. 「ProjectNavigator」内の「< (プロジェクト名) /conf/im-bizrobo-config.xml> ファイル」をダブルクリックで開き、「ソース」を選択します。



4. [設定ファイルの説明](#)に従って、設定を記述してください。

5. 「BizRobo!クライアント設定ファイル」を保存します。
6. IM-JugglingでWARファイルを出力し、アプリケーションサーバへデプロイを行ってください。

ライセンスコードの登録

IM-RPAを利用するためには、テナント環境セットアップ後、製品ライセンスコードの登録が必要です。テナント環境セットアップについては、「[intra-mart Accel Platform セットアップガイド](#)」 - 「[テナント環境セットアップ](#)」を参照してください。

1. IM-RPAのライセンス登録を行います。
詳細は「[ライセンスの登録](#)」を参照してください。

設定ファイル

項目

- [概要](#)
- [リファレンス](#)
 - [テナント別設定](#)
 - [URL設定](#)
 - [認証設定](#)
 - [ユーザ名設定](#)
 - [パスワード設定](#)

概要

BizRobo!連携 に関する設定です。

モジュール	BizRobo!連携
フォーマットファイル(xsd)	WEB-INF/schema/im-bizrobo-config.xsd
設定場所	WEB-INF/conf/im-bizrobo-config.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<im-bizrobo-config
  xmlns="http://intra-mart.co.jp/system/bizrobo/config/im-bizrobo-config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://intra-mart.co.jp/system/bizrobo/config/im-bizrobo-config ../schema/im-bizrobo-config.xsd ">

  <repository-server id="default" use-insecure-ssl="false">
    <url>http://localhost:50080/</url>
    <authentication enable="false">
      <!--
      <username>your name</username>
      <password>your password</password>
      -->
    </authentication>
  </repository-server>

</im-bizrobo-config>
```

リファレンス

テナント別設定

タグ名 repository-server

テナント毎の BizRobo!接続に関する設定を定義します。

【設定項目】

```
<im-bizrobo-config>
  <repository-server id="default" use-insecure-ssl="false">
    :
  </server>
</im-bizrobo-config>
```

必須項目	×
複数設定	○
設定値・設定する内容	repository-server タグを親とするタグ
単位・型	なし
省略時のデフォルト値	なし
親タグ	im-bizrobo-config

【属性】

属性名	説明	必須	デフォルト値
id	テナントID このタグの設定の対象となるテナントIDを指定してください。	×	なし
use-insecure-ssl	セキュアでないSSLを利用するか否か BizRobo! Management Consoleへの接続にhttpsプロトコルを利用する場合は <code>true</code> を設定してください。	×	なし

URL設定

タグ名 `url`

BizRobo! Management ConsoleのURLを設定します。

【設定項目】

```
<im-bizrobo-config>
  <repository-server id="default" use-insecure-ssl="false">
    <url>http://localhost:50080/</url>
    :
  </server>
</im-bizrobo-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	BizRobo! Management ConsoleへのURL
単位・型	URL
省略時のデフォルト値	なし
親タグ	repository-server

認証設定

タグ名 `authentication`

BizRobo! Management Consoleへの接続時の認証設定です。

【設定項目】

```
<im-bizrobo-config>
  <repository-server id="default" use-insecure-ssl="false">
    :
    <authentication enable="true">
      :
      </authentication>
    </server>
  </im-bizrobo-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	authentication タグを親とするタグ
単位・型	なし
省略時のデフォルト値	なし
親タグ	repository-server

【属性】

属性名	説明	必須	デフォルト値
enable	認証を必要とするか否か BizRobo! Management Consoleへの接続に認証が必要な場合 true を設定してください。	×	なし

ユーザ名設定

タグ名 username

BizRobo! Management Consoleへの接続時のユーザ名設定です。

【設定項目】

```
<im-bizrobo-config>
  <repository-server id="default" use-insecure-ssl="false">
    :
    <authentication enable="true">
      <username>your name</username>
      :
    </authentication>
  </server>
</im-bizrobo-config>
```

必須項目	×
複数設定	×
設定値・設定する内容	BizRobo! Management Consoleへ接続する際のユーザ名
単位・型	文字列
省略時のデフォルト値	なし
親タグ	authentication

パスワード設定

タグ名 password

BizRobo! Management Consoleへの接続時のパスワード設定です。

【設定項目】

```
<im-bizrobo-config>
  <repository-server id="default" use-insecure-ssl="false">
    :
    <authentication enable="true">
      :
      <password>your password</password>
    </authentication>
  </server>
</im-bizrobo-config>
```

必須項目	×
複数設定	×
設定値・設定する内容	BizRobo! Management Consoleへ接続する際のパスワード
単位・型	文字列
省略時のデフォルト値	なし
親タグ	authentication

IM-LogicDesigner タスク説明

項目

- ロボットの配置
- ロボットの削除
- タイプの配置
- タイプの削除
- スニペットの配置
- スニペットの削除
- リソースの配置
- リソースの削除
- ライブラリの展開

ロボットの配置

リポジトリにロボットを配置します。

入力値

```
im_bizRoboDeployRobot <object>
  └─ fileData <binary> *
  └─ fileName <string> *
  └─ projectName <string> *
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeployRobot	任意	any	なし	-
fileData	必須	binary	なし	ロボットのデータストリーム
fileName	必須	string	なし	ロボットのファイル名
projectName	必須	string	なし	配置先のプロジェクト名

出力値

```
im_bizRoboDeployRobot <any>
```

項目名	型	配列/リスト	説明
im_bizRoboDeployRobot	any	なし	出力値として利用可能な値はありません。

ロボットの削除

リポジトリからロボットを削除します。

入力値

```
im_bizRoboDeleteRobot <object>
├─ fileName <string> *
├─ projectName <string> *
└─ silent <boolean>
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeleteRobot	任意	any	なし	-
fileName	必須	string	なし	ロボットのファイル名
projectName	必須	string	なし	配置先のプロジェクト名
silent	任意	boolean	なし	エラーを無視するか <code>false</code> を指定するとロボットが存在しない場合にエラーが発生します。

出力値

```
im_bizRoboDeleteRobot <any>
```

項目名	型	配列/リスト	説明
im_bizRoboDeleteRobot	any	なし	出力値として利用可能な値はありません。

タイプの配置

リポジトリにタイプを配置します。

入力値

```
im_bizRoboDeployType <object>
├─ fileData <binary> *
├─ fileName <string> *
└─ projectName <string> *
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeployType	任意	any	なし	-
fileData	必須	binary	なし	タイプのデータストリーム
fileName	必須	string	なし	タイプのファイル名
projectName	必須	string	なし	配置先のプロジェクト名

出力値

```
im_bizRoboDeployType <any>
```

項目名	型	配列/リスト	説明
-----	---	--------	----

項目名	型	配列/リスト	説明
im_bizRoboDeployType	any	なし	出力値として利用可能な値はありません。

タイプの削除

リポジトリからタイプを削除します。

入力値

```
im_bizRoboDeleteType <object>
├─ fileName <string> *
├─ projectName <string> *
└─ silent <boolean>
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeleteType	任意	any	なし	-
fileName	必須	string	なし	タイプのファイル名
projectName	必須	string	なし	配置先のプロジェクト名
silent	任意	boolean	なし	エラーを無視するか false を指定するとタイプが存在しない場合にエラーが発生します。

出力値

```
im_bizRoboDeleteType <any>
```

項目名	型	配列/リスト	説明
im_bizRoboDeleteType	any	なし	出力値として利用可能な値はありません。

スニペットの配置

リポジトリにスニペットを配置します。

入力値

```
im_bizRoboDeploySnippet <object>
├─ fileData <binary> *
├─ fileName <string> *
└─ projectName <string> *
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeploySnippet	任意	any	なし	-
fileData	必須	binary	なし	スニペットのデータストリーム
fileName	必須	string	なし	スニペットのファイル名
projectName	必須	string	なし	配置先のプロジェクト名

出力値

```
im_bizRoboDeploySnippet <any>
```

項目名	型	配列/リスト	説明
im_bizRoboDeploySnippet	any	なし	出力値として利用可能な値はありません。

スニペットの削除

リポジトリからスニペットを削除します。

入力値

```
im_bizRoboDeleteSnippet <object>
├─ fileName <string> *
├─ projectName <string> *
└─ silent <boolean>
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeleteSnippet	任意	any	なし	-
fileName	必須	string	なし	スニペットのファイル名
projectName	必須	string	なし	配置先のプロジェクト名
silent	任意	boolean	なし	エラーを無視するか <code>false</code> を指定するとスニペットが存在しない場合にエラーが発生します。

出力値

```
im_bizRoboDeleteSnippet <any>
```

項目名	型	配列/リスト	説明
im_bizRoboDeleteSnippet	any	なし	出力値として利用可能な値はありません。

リソースの配置

リポジトリにリソースを配置します。

入力値

```
im_bizRoboDeployResource <object>
├─ fileData <binary> *
├─ fileName <string> *
└─ projectName <string> *
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeployResource	任意	any	なし	-
fileData	必須	binary	なし	リソースのデータストリーム
fileName	必須	string	なし	リソースのファイル名
projectName	必須	string	なし	配置先のプロジェクト名

出力値

```
im_bizRoboDeployResource <any>
```

項目名	型	配列/リスト	説明
-----	---	--------	----

項目名	型	配列/リスト	説明
im_bizRoboDeployResource	any	なし	出力値として利用可能な値はありません。

リソースの削除

リポジトリからリソースを削除します。

入力値

```
im_bizRoboDeleteResource <object>
├─ fileName <string> *
├─ projectName <string> *
└─ silent <boolean>
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeleteResource	任意	any	なし	-
fileName	必須	string	なし	リソースのファイル名
projectName	必須	string	なし	配置先のプロジェクト名
silent	任意	boolean	なし	エラーを無視するか <code>false</code> を指定するとリソースが存在しない場合にエラーが発生します。

出力値

```
im_bizRoboDeleteResource <any>
```

項目名	型	配列/リスト	説明
im_bizRoboDeleteResource	any	なし	出力値として利用可能な値はありません。

ライブラリの展開

リポジトリにライブラリを展開します。

入力値

```
im_bizRoboDeployLibrary <object>
├─ library <binary> *
└─ projectName <string> *
```

項目名	必須/任意	型	配列/リスト	説明
im_bizRoboDeployLibrary	任意	any	なし	-
library	必須	binary	なし	ライブラリのデータストリーム
projectName	必須	string	なし	配置先のプロジェクト名

出力値

```
im_bizRoboDeployLibrary <any>
```

項目名	型	配列/リスト	説明
im_bizRoboDeployLibrary	any	なし	出力値として利用可能な値はありません。

IM-LogicDesigner ユーザ定義説明

項目

- BizRobo!定義の新規作成

BizRobo!定義の新規作成

IM-LogicDesignerのユーザ定義として、BizRobo!ロボット実行タスクが作成できます。
以下の手順で、IM-LogicDesignerにユーザ定義を追加してください。

1. ユーザ定義一覧より、BizRobo!定義を新規作成します。
詳細な手順は、「IM-LogicDesigner ユーザ操作ガイド」 - 「ユーザ定義を新規登録する」を参照してください。
2. 共通設定を定義します。
「入力値」および「返却値」については、次項のプロジェクトおよびロボットを選択することで自動設定します。
3. BizRobo! 定義の詳細情報を定義します。

<画面項目>

項目	説明
プロジェクト	対象のプロジェクトを選択します。
ロボット	実行対象のロボットを選択します。

注意

ユーザ定義の新規作成時に次のようなエラーが表示された場合、BizRobo! Management Consoleへの接続に失敗しています。
接続ファイルの記述内容やネットワーク環境に問題がないかご確認ください。



UiPath連携

ここでは、UiPath連携のセットアップ手順、およびIM-LogicDesignerタスクの仕様を説明します。

セットアップ

項目

- IM-Juggling プロジェクトの編集
- ライセンスコードの登録

IM-Juggling プロジェクトの編集

UiPath連携は、IM-RPAモジュールを使用します。
以下の手順で設定を行ってください。

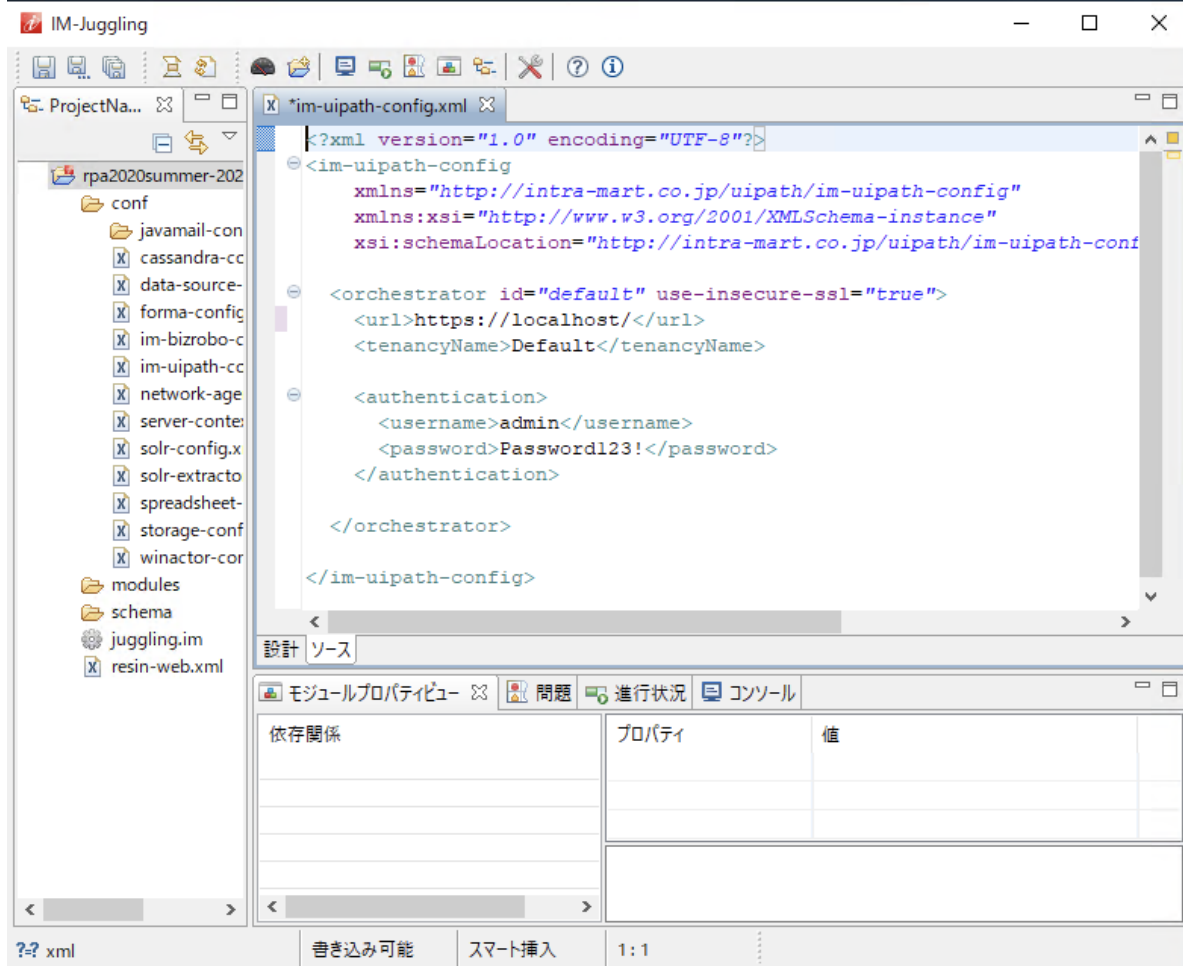
1. IM-Jugglingにて、プロジェクトにIM-RPAモジュールを追加してください。
以下のドキュメントを参照してください。
 - 「[intra-mart Accel Platform セットアップガイド](#)」 - 「[プロジェクトの作成とモジュールの選択](#)」



注意

IM-RPAモジュールのご利用には、エンタープライズ版の構成、およびライセンスが必要です。

2. 「設定ファイルが存在しません」という赤字のメッセージをクリックします。
表示されるダイアログにて「OK」をクリックし、このプロジェクトに「UiPathクライアント設定ファイル」(im-uipath-config.xml)を追加します。
3. 「ProjectNavigator」内の「<(プロジェクト名)/conf/im-uipath-config.xml> ファイル」をダブルクリックで開き、「ソース」を選択します。



4. [設定ファイルの説明](#)に従って、設定を記述してください。

5. 「UiPathクライアント設定ファイル」を保存します。
6. IM-JugglingでWARファイルを出力し、アプリケーションサーバへデプロイを行ってください。

ライセンスコードの登録

IM-RPAを利用するためには、テナント環境セットアップ後、製品ライセンスコードの登録が必要です。テナント環境セットアップについては、「[intra-mart Accel Platform セットアップガイド](#)」-「[テナント環境セットアップ](#)」を参照してください。

1. IM-RPAのライセンス登録を行います。
詳細は「[ライセンスの登録](#)」を参照してください。

設定ファイル

項目

- 概要
- リファレンス
 - テナント別設定
 - URL設定
 - テナント名設定
 - 認証設定
 - ユーザ名設定
 - パスワード設定

概要

UiPath連携 に関する設定です。

モジュール	UiPath連携
フォーマットファイル(xsd)	WEB-INF/schema/im-uipath-config.xsd
設定場所	WEB-INF/conf/im-uipath-config.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<im-uipath-config
  xmlns="http://intra-mart.co.jp/uipath/im-uipath-config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://intra-mart.co.jp/uipath/im-uipath-config ../schema/im-uipath-config.xsd ">

  <orchestrator id="default" use-insecure-ssl="true">
    <url>https://xxx.xxx.xxx.xxx/</url>
    <tenancyName>Default</tenancyName>

    <authentication>
      <username>username</username>
      <password>password</password>
    </authentication>
  </orchestrator>
</im-uipath-config>
```

リファレンス

テナント別設定

タグ名 orchestrator

テナント毎の UiPath Orchestrator接続に関する設定を定義します。

【設定項目】

```
<im-uipath-config>
  <orchestrator id="default" use-insecure-ssl="true">
    :
  </orchestrator>
</im-uipath-config>
```

必須項目	×
複数設定	○
設定値・設定する内容	orchestrator タグを親とするタグ
単位・型	なし
省略時のデフォルト値	なし
親タグ	im-uipath-config

【属性】

属性名	説明	必須	デフォルト値
id	テナントID このタグの設定の対象となるテナントIDを指定してください。	×	なし
use-insecure-ssl	セキュアでないSSLを利用するか否か UiPath Orchestratorへの接続にhttpsプロトコルを利用する場合は true を設定してください。	×	なし

URL設定

タグ名 url

UiPath OrchestratorのURLを設定します。

【設定項目】

```
<im-uipath-config>
  <orchestrator id="default" use-insecure-ssl="true">
    <url>http://xxx.xxx.xxx.10/</url>
    :
  </orchestrator>
</im-uipath-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	UiPath OrchestratorへのURL
単位・型	URL
省略時のデフォルト値	なし
親タグ	orchestrator

テナント名設定

タグ名 tenancyName

UiPath Orchestratorのテナント名を設定します。

【設定項目】

```
<im-uipath-config>
  <orchestrator id="default" use-insecure-ssl="true">
    <tenancyName>Default</tenancyName>
    :
  </orchestrator>
</im-uipath-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	UiPath Orchestrator上で管理するテナント名
単位・型	文字列
省略時のデフォルト値	なし
親タグ	orchestrator

認証設定

タグ名 authentication

UiPath Orchestratorへの接続時の認証設定です。

【設定項目】

```
<im-uipath-config>
  <orchestrator id="default" use-insecure-ssl="true">
    :
    <authentication>
    :
  </authentication>
</orchestrator>
</im-uipath-config>
```

必須項目	○
複数設定	×
設定値・設定する内容	authentication タグを親とするタグ
単位・型	なし
省略時のデフォルト値	なし
親タグ	orchestrator

ユーザ名設定

タグ名 username

UiPath Orchestratorへの接続時のユーザ名設定です。

【設定項目】

```
<im-uipath-config>
  <orchestrator id="default" use-insecure-ssl="true">
    :
    <authentication>
      <username>username</username>
    :
  </authentication>
</orchestrator>
</im-uipath-config>
```

必須項目	×
複数設定	×
設定値・設定する内容	UiPath Orchestratorへ接続する際のユーザ名
単位・型	文字列
省略時のデフォルト値	なし
親タグ	authentication

パスワード設定

タグ名 password

UiPath Orchestratorへの接続時のパスワード設定です。

【設定項目】

```
<im-uipath-config>
  <orchestrator id="default" use-insecure-ssl="true">
    :
    <authentication>
      :
      <password>your password</password>
    </authentication>
  </orchestrator>
</im-uipath-config>
```

必須項目	×
複数設定	×
設定値・設定する内容	UiPath Orchestratorへ接続する際のパスワード
単位・型	文字列
省略時のデフォルト値	なし
親タグ	authentication

IM-LogicDesigner タスク説明

項目

- ジョブステータス取得
 - 入力値
 - 出力値

ジョブステータス取得

UiPathジョブステータスを取得するタスクです。

コラム

ロジックフロー定義編集画面にて、UiPathのカテゴリおよびタスクのアイコンが表示されない場合、アイコンのリカバリを実行することで表示されるようになる場合があります。

詳細は、「IM-LogicDesigner ユーザ操作ガイド」 - 「ユーザアイコンをリカバリする」を参照してください。

入力値

```
im_uipathGetJobStatus <object>
└─ id <integer>
```

項目名	必須/任意	型	配列/リスト	説明
im_uipathGetJobStatus	任意	object	なし	-
id	必須	integer	なし	ジョブID

出力値

```
im_uipathGetJobStatus <object>
└─ batchExecutionKey <string>
└─ creationTime <string>
└─ endTime <string>
└─ hostMachineName <string>
└─ id <integer>
└─ info <string>
└─ inputArguments <map>
└─ key <string>
└─ outputArguments <map>
└─ releaseName <string>
└─ source <string>
└─ sourceType <string>
└─ startTime <string>
└─ startingScheduleId <integer>
└─ state <string>
└─ type <string>
```

項目名	型	配列/リスト	説明
im_uipathGetJobStatus	object	なし	-
batchExecutionKey	string	なし	バッチの実行ID
creationTime	string	なし	ジョブが作成された日時
endTime	string	なし	ジョブの完了日時
hostMachineName	string	なし	プロセスが実行されたマシン名
id	string	なし	ジョブID
info	string	なし	ジョブの実行状態に関する補足説明
inputArguments	map	なし	ジョブ実行時の入力値
key	string	なし	ジョブの実行ごとに一意となる文字列
outputArguments	string	なし	ジョブ実行結果の出力値
releaseName	string	なし	プロセス名
source	string	なし	ジョブのソースコード
sourceType	string	なし	ジョブのソースタイプ
startTime	string	なし	ジョブの開始日時
startingScheduleId	integer	なし	スケジュールID
state	string	なし	ステータス
type	string	なし	タイプ

コラム

ジョブのステータスについてはUiPath Orchestratorのガイドを参照してください。
[「UiPath Orchestrator ジョブステータス」](#)を参照してください。

IM-LogicDesigner ユーザ定義説明

項目

- UiPath定義の新規作成

UiPath定義の新規作成

IM-LogicDesignerのユーザ定義として、UiPathプロセス実行タスクが作成できます。
以下の手順で、IM-LogicDesignerにユーザ定義を追加してください。

1. ユーザ定義一覧より、UiPath定義を新規作成します。
詳細な手順は、「IM-LogicDesigner ユーザ操作ガイド」 - 「ユーザ定義を新規登録する」を参照してください。
2. 共通設定を定義します。
「入力値」および「返却値」については、次項のプロセスを選択することで自動設定します。
3. UiPath 定義の詳細情報を定義します。

UiPath定義

プロセス*	<input type="text" value="▼"/>
非同期実行	<input type="checkbox"/>

<画面項目>

項目	説明
プロセス	対象のプロセスを選択します。
非同期実行	プロセス実行を非同期で実行する場合に選択します。

コラム

利用可能なデータの型は以下の5種類です。（括弧内はIM-LogicDesignerでの取り扱いです。）

多次元配列の取り扱いは未対応です。

String(String)

Int32(Integer)

Boolean(Boolean)

DateTime(IMDateTim)

Double(Double)

著作権および特記事項

intra-mart は株式会社エヌ・ティ・ティ・データイントラマートの登録商標です。

Oracle と Java は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。

WinActor はエヌ・ティ・ティ・アドバンステクノロジー株式会社の登録商標です。

BizRobo! および Basic Robo は、RPAテクノロジーズ株式会社の登録商標です。

UiPath および UiPath Orchestrator は、UiPath株式会社の登録商標です。

文中の社名、商品名等は各社の商標または登録商標である場合があります。

他の会社名、製品名およびサービス名などはそれぞれ各社の商標または登録商標です。

本製品を使用する場合は、本製品に含まれる各ソフトウェアのライセンスについても同意したものとします。

以上