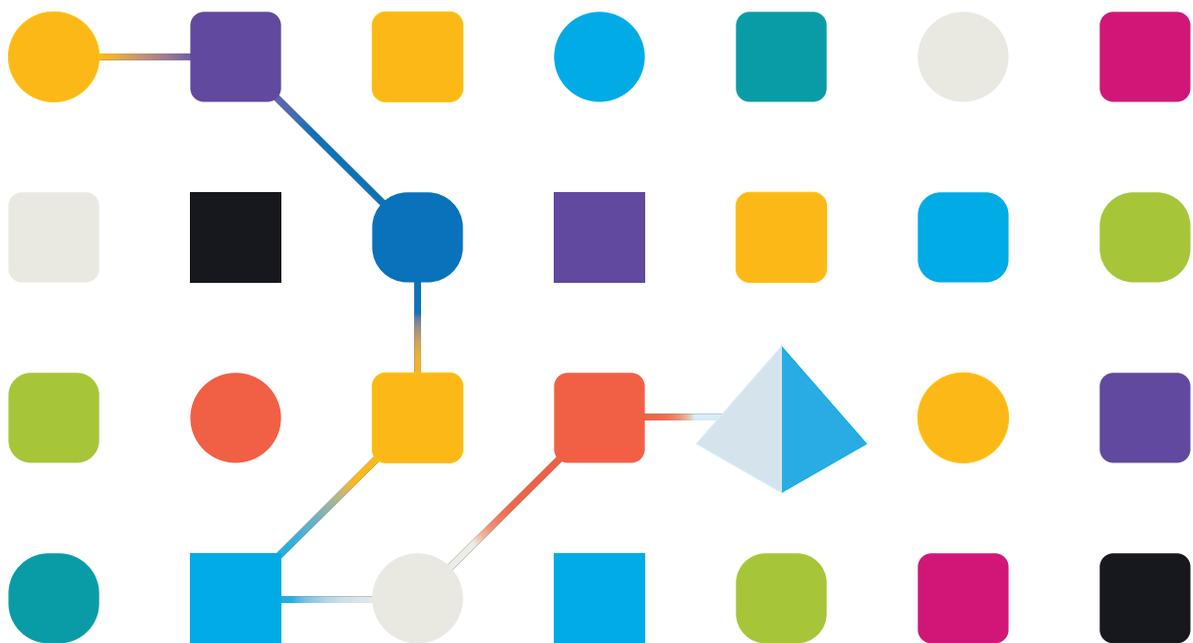


# blueprism<sup>®</sup>

## Interact Web API Service 4.2

### User Guide

Document Revision: 1.3



## Trademarks and Copyright

The information contained in this guide is the proprietary and confidential information of Blue Prism Cloud Limited and/or its affiliates and should not be disclosed to a third-party without the written consent of an authorized Blue Prism representative. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying without the written permission of Blue Prism Cloud Limited or its affiliates.

© **Blue Prism Cloud Limited 2001 – 2021**

“Blue Prism”, the “Blue Prism” logo and Prism device are either trademarks or registered trademarks of Blue Prism Limited and its affiliates. All Rights Reserved.

All other trademarks are hereby acknowledged and are used to the benefit of their respective owners. Blue Prism Cloud Limited and its affiliates are not responsible for the content of external websites referenced by this guide.

Blue Prism Limited, 2 Cinnamon Park, Crab Lane, Warrington, WA2 0XP, United Kingdom.  
Registered in England: Reg. No. 4260035. Tel: +44 370 879 3000. Web: [www.blueprism.com](http://www.blueprism.com)

# Contents

<b>Interact Web API Service</b> .....	<b>4</b>
Overview .....	4
<b>Submission</b> .....	<b>5</b>
<b>Communicating back to Interact</b> .....	<b>7</b>
Configure credentials for the Interact Web API service .....	10
<b>Actions</b> .....	<b>13</b>
Get Submission .....	13
Get Form Schema .....	16
Create Submission .....	20
Upload File .....	21
Edit Submission Field Value .....	23
Update Config .....	33
Update Status .....	35
Raise Submission .....	37
Get Users by Form Name .....	38
Change Field State to Optional .....	40
Change Field State to Mandatory .....	41
Change Field State to Read Only .....	42
Change Field State to Hidden .....	43
Change Page State to Visible .....	44
Change Page State to Hidden .....	45
Move Submission to History Tab .....	46
Move Submission to Approved Tab .....	47
Move Submission to Review Tab .....	48
Move Submission to Declined Tab .....	49
Move Submission to Draft Tab .....	50
Move Submission to Inbox Tab .....	51
Move Submission to Archived Tab .....	52
<b>Human/digital worker collaboration</b> .....	<b>53</b>
<b>Interact Web API Service object restrictions</b> .....	<b>55</b>

## Interact Web API Service

This user guide covers Blue Prism Interact Web Application Programming Interface (API) Service.

Blue Prism® Interact enables developers to create web-based forms that will allow a user to interact with automations, not just at the point of initialization but also during the process if a human is required to review progress, enabling a “human/digital worker collaboration” approach.

Information from Interact is added into a Blue Prism queue for processing by the digital worker. Blue Prism can then send updated information, dynamically, back to Interact. This way a user can review and act upon the processed data and participate in the automation process.

It is assumed as part of this guide that the user is familiar with Blue Prism digital workforce and has experience with components such as Hub and Blue Prism.

### Overview

As a digital workforce expands the addressable use cases of an enterprise, there is a further need for digital workers and humans to collaborate in the end-to-end execution of a business process. In addition, innovative ways of assigning work to a digital workforce are required to provide flexibility on who can interact with the resource. Blue Prism Interact provides new and existing Blue Prism users with a collaboration interface for an end user's interaction with their digital workforce within a business process. Initiate, verify, receive and authorize varied work related to your business processes. In addition, Blue Prism Interact reduces the skill criteria required by allowing users to create dynamic web interfaces either by using the no-code form designer or using a Blue Prism Visual Business Object (VBO) to create forms based on the business process requirements.

Blue Prism Interact functionality is split across two Blue Prism components; firstly, a developer utilizing Hub and the Forms plugin to create and publish Forms for an end user to utilize; secondly the end user interface where they can, by utilizing published Forms, interact with the digital workforce.

The Interact end user interface is accessed via a web browser, negating the need for any locally installed software. Interact can be accessed as an internal web application or published externally.

This user guide covers the gathering of information from a Blue Prism queue and how the information can be updated and returned after processing, back to the Interact user interface.

As well as detailing how to utilize the features of the Interact Web API Service we will also demonstrate the functionality by stepping through an example process illustrating how Forms can be updated dynamically.

## Submission

Submitting a Form is detailed in the [Interact user guide](#). All submissions are processed through the IADA parser and are added to the Blue Prism queue defined in the Form. This information can then be retrieved from the queue using the 'Get Next Item' action from the Blue Prism Internal Business Objects, Work Queues.

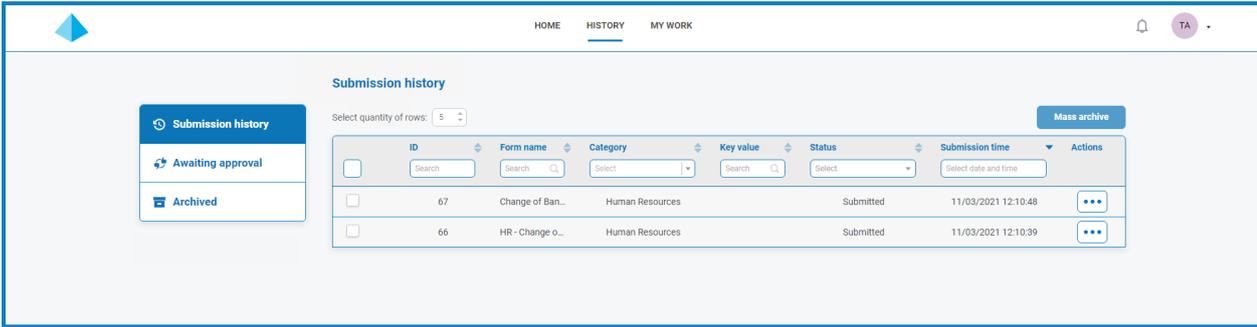
Using the 'Get Next Item' action retrieves a collection into your automation ready for processing. It is recommended that the collection is not defined with fields. This is so all the information can be pulled from Interact. Defining fields in the collection will cause the automation to fail if the Form is updated with either a new field or a field is deleted.

A typical collection is displayed below.

Fields	Initial Values	Current Values	Rows:	
first-name (Text)	last-name (Text)	date-change-request (Text)	name-of-new-bank (Text)	sort-code (Text)
Alfred	Jones	30/11/2020	MoneySafe Ltd	123456

It is important to note one data item that is automatically added to the collection, this is the '\_requestId' data item.

This data item is the submission ID of the Form submitted by the user. This ID is the number showing in the Submission History page as illustrated below.



ID	Form name	Category	Key value	Status	Submission time	Actions
67	Change of Ban...	Human Resources		Submitted	11/03/2021 12:10:48	...
66	HR - Change o...	Human Resources		Submitted	11/03/2021 12:10:39	...

This is the first key item of the linkage between Interact and Blue Prism. The second part of the linkage is the fields in the Form that contain the data items.

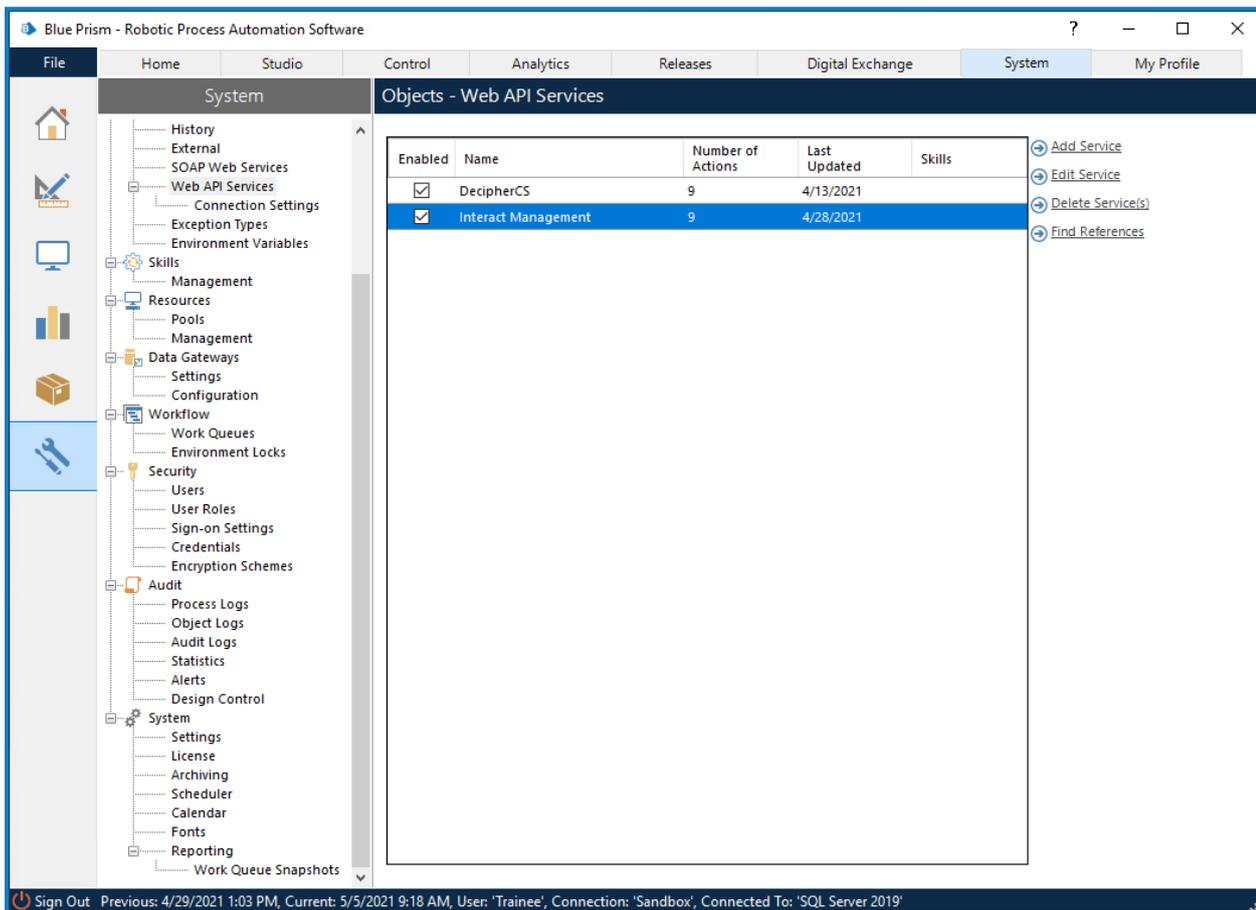
You can see in the collection displayed above, the Automation ID for the data items, for example first-name, last-name, date-of-employment, and so on. These are the 'names' that were defined for the fields when the Form was created in Hub. Hub will suggest an Automation ID for the field as you type the 'label', you can choose to use the one supplied or change it for something else to suit your particular need.

 Automation IDs in Form fields must be in lowercase and the use of special characters is not allowed. If a 'space' is entered, then the character is replaced by a 'hyphen' character. This is to aid readability.

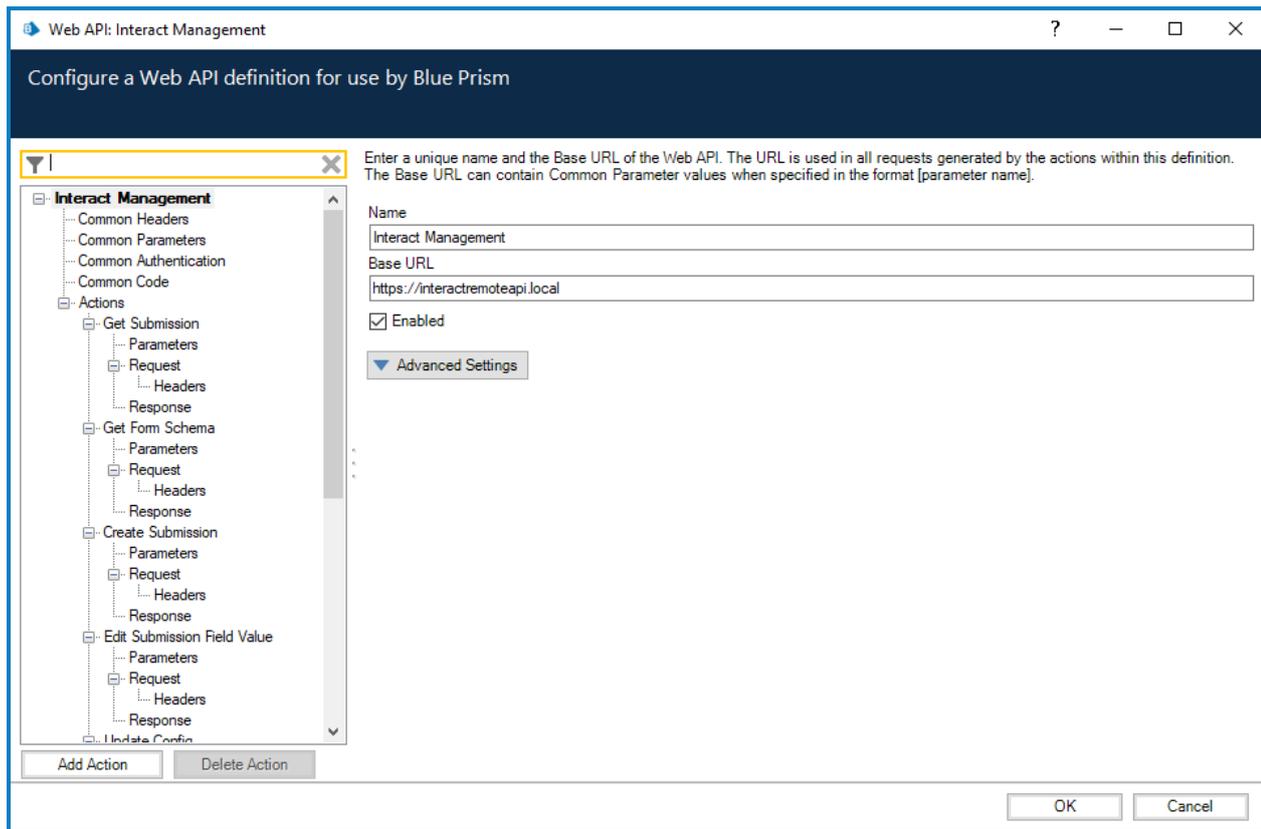
## Communicating back to Interact

The communication channel back to Interact from Blue Prism is performed by a Blue Prism Web Service. The web service is available as a release that can be imported into Blue Prism. Once imported it will need to be updated with the appropriate base URL and authorization codes to enable secure communication.

 If you upgrade Interact to a new release, you must ensure you import the latest release file into Blue Prism.



The image below shows the web API service.



In the main window of the web API service we can see the Base URL defined, this will need to be updated to reflect the configuration of your platform setup.

The Common Parameters section is left blank and the Common Authentication section is used. This requests a Bearer Token using stored credentials from the Credential store.

Web API: Interact Management

Configure a Web API definition for use by Blue Prism

This authentication type uses a client identifier and client secret stored in a credential to retrieve an OAuth 2.0 access token from an authorization server. The access token is then used to authenticate Web API requests.

Authentication Type  
OAuth 2.0 (Client Credentials)

Authorization URI  
https://ims.local/connect/token

Credential  
Interact Credentials  
 Expose to process  
Parameter Name  
OAuth 2 (Client Credentials) Authentication Credential Name

Scope  
interact-remote-api

Add Action Delete Action

OK Cancel

To set up the credentials a 'client\_secret' must to be configured. This is generated and displayed during the installation process and is stored in the Interact Remote API Secret Key file in C:\Program Files (x86)\Blue Prism\Interact Remote API secret key.txt. See the [Interact Installation guide](#) for more details.

## Configure credentials for the Interact Web API service

When configuring the credentials for the OAuth 2.0 (Client Credentials) the following four values must be configured in the Blue Prism System Settings ([Security > Credentials](#)):

- **Client ID** – Enter *InteractRemoteClient*
- **Client Secret** – Enter the code displayed and copied during installation or, if not copied, from the secret key file contents found in: C:\Program Files (x86)\Blue Prism\Interact Remote API secret key.txt
- **grant\_type** – Set the value to *client\_credentials*
- **scope** – Set the value to *interact-remote-api*

**Credential Details**

Name: Interact Credentials

Description: Credentials for the Interact Remote API

Type: OAuth 2.0 (Client Credentials)

**Application Credentials** | Access Rights

Use this credential type for OAuth 2.0 web authentication using client credentials.

Client ID: InteractRemoteClient

Expires:  2/10/2099

Client Secret: ●●●●●●

Marked as invalid

**Additional Properties**

Name	Value
grant_type	●●●●●●
scope	●●●●●●

OK Cancel

Remember to set up the Access Rights tab in the Credentials Details to ensure it can access the processes for the allowed users.

The web service has nine defined actions. These actions are available within the Utility - Interact API Blue Prism VBO along with a number of additional actions which map to the web service. The actions are:

Interact Web API	Blue Prism VBO: Utility - Interact API
Get Submission	Get Submission
Get Form Schema	Get Form Schema
Create Submission	Create Submission
Edit Submission Field Value	<p>Edit &lt;Type&gt; Field Value, where the type is:</p> <ul style="list-style-type: none"> <li>• <a href="#">Checkbox Single</a></li> <li>• <a href="#">Checkbox Multiple</a></li> <li>• <a href="#">Date</a></li> <li>• <a href="#">Dropdown Single</a></li> <li>• <a href="#">Dropdown Multiple</a></li> <li>• <a href="#">Radio</a></li> <li>• <a href="#">Number</a></li> <li>• <a href="#">Table</a></li> <li>• <a href="#">Text</a></li> <li>• <a href="#">Time</a></li> </ul>
Update Config	<p><a href="#">Update Config</a> – this is superseded by the individual Change actions below.</p> <p><a href="#">Change Field State to Optional</a></p> <p><a href="#">Change Field State to Mandatory</a></p> <p><a href="#">Change Field State to Read Only</a></p> <p><a href="#">Change Field State to Hidden</a></p> <p><a href="#">Change Page State to Visible</a></p> <p><a href="#">Change Page State to Hidden</a></p>
Update Status	<p><a href="#">Update Status</a> – this is superseded by the individual Move actions below.</p> <p><a href="#">Move Submission to History Tab</a></p> <p><a href="#">Move Submission to Approved Tab</a></p> <p><a href="#">Move Submission to Review Tab</a></p> <p><a href="#">Move Submission to Declined Tab</a></p> <p><a href="#">Move Submission to Draft Tab</a></p> <p><a href="#">Move Submission to Inbox Tab</a></p> <p><a href="#">Move Submission to Archived Tab</a></p>
Raise Submission	<a href="#">Raise Submission</a>
Get Users by Form Name	<a href="#">Get Users by Form Name</a>
Upload File	<a href="#">Upload File</a>

The actions are called from the Utility - Interact API VBO so, after the initial configuration of the URL and authorization codes, the Web Service does not need to be amended. Each of the actions within the Utility - Interact API VBO are explained in the next section.

## Actions

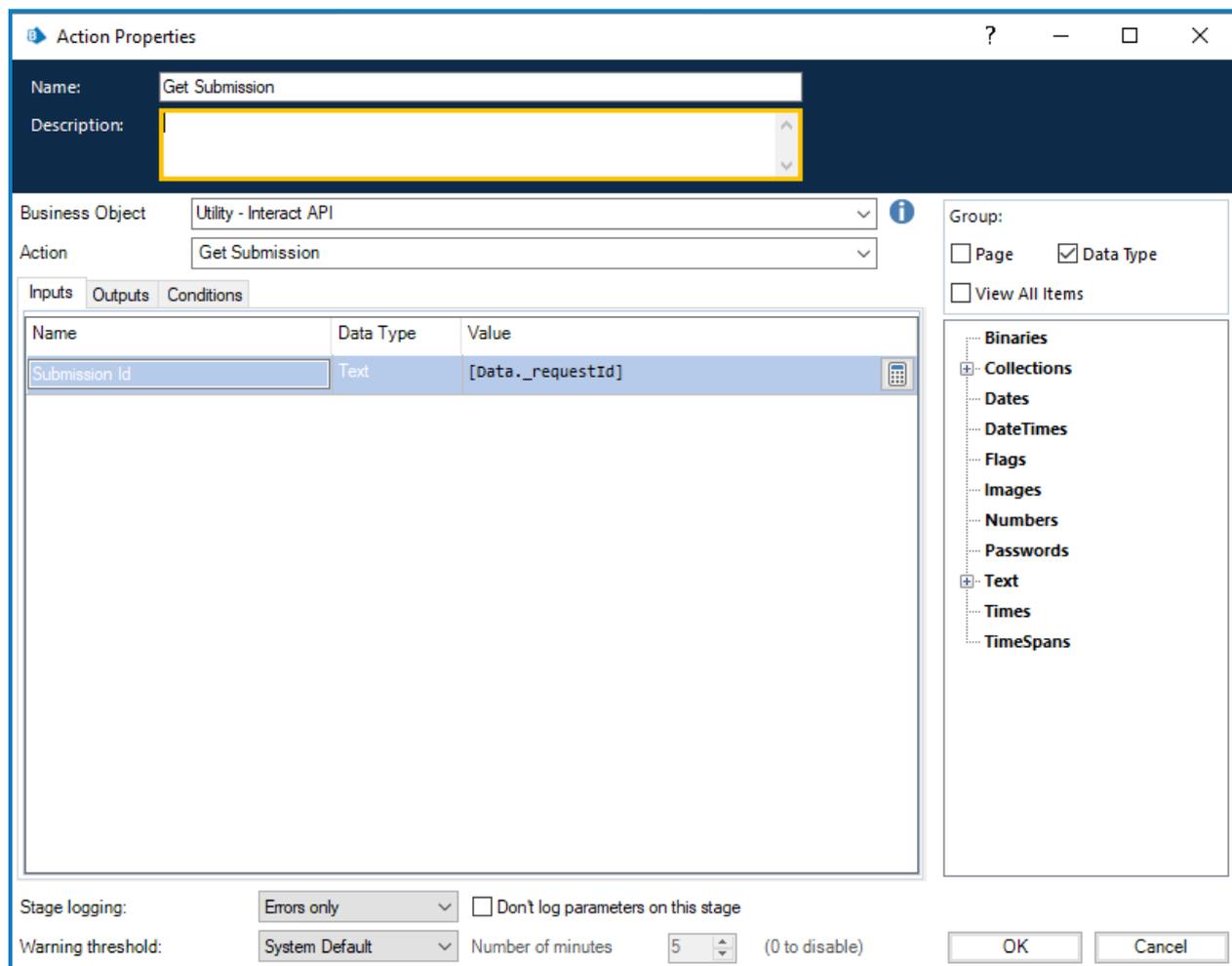
There are a number of actions within the Interact Web API Service, which are called using the Blue Prism Object (Utility - Interact API). Going through each of the actions in turn, the sections below explain how to use them in your automations.

 All of the web API service actions should be called from the Utility - Interact API VBO in Blue Prism, and not directly from the web service.

## Get Submission

The Get Submission Object action does a very similar action as the Get Next Item action when it retrieves the information from a queue.

To utilize the action, an action is added to your process and the Business Object selected, with the action set to **Get Submission**. There is only one input parameter which is the Submission ID as illustrated below.

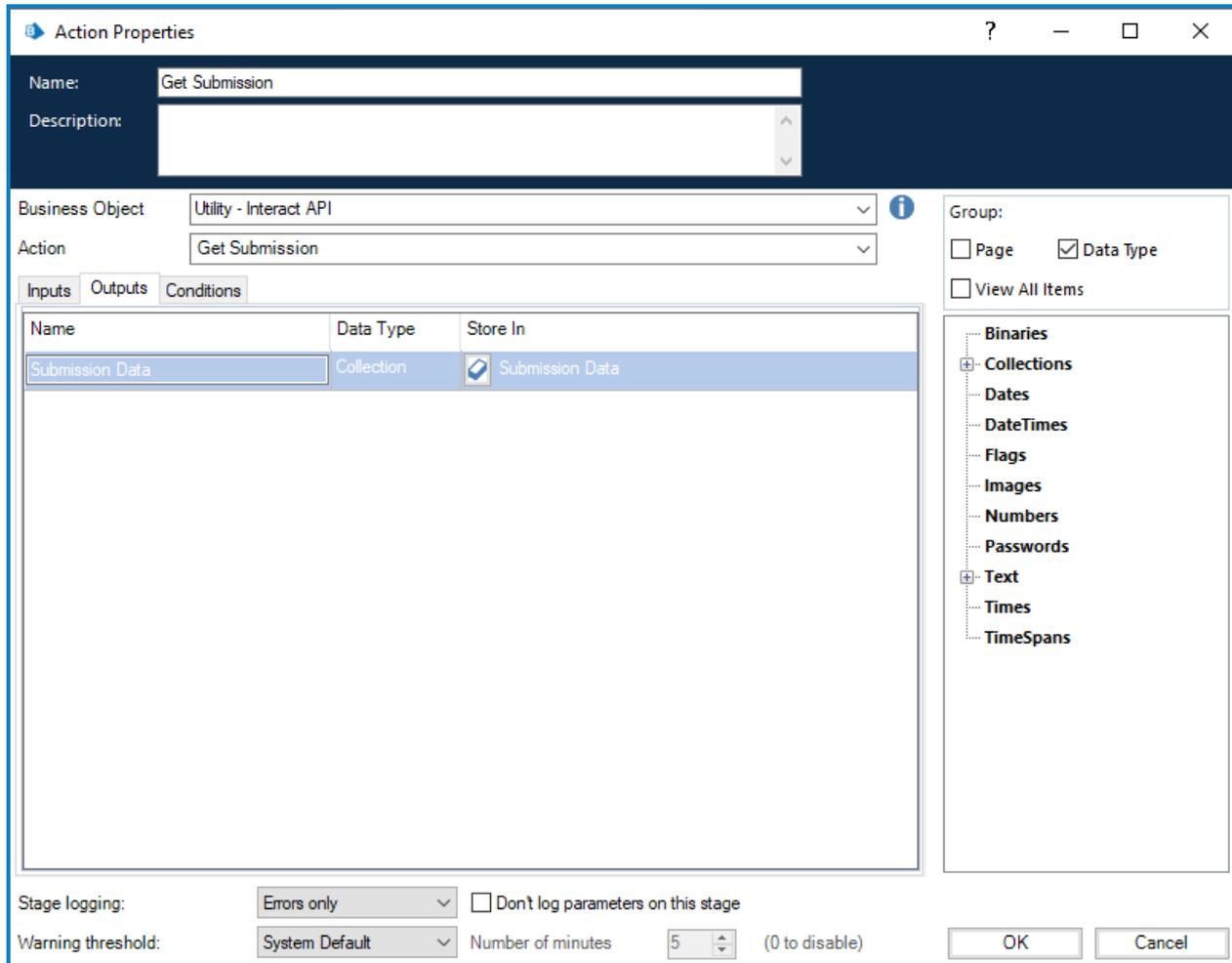


The screenshot shows the 'Action Properties' dialog box for the 'Get Submission' action. The 'Name' field is set to 'Get Submission'. The 'Business Object' is 'Utility - Interact API' and the 'Action' is 'Get Submission'. The 'Inputs' tab is active, showing a table with one input parameter:

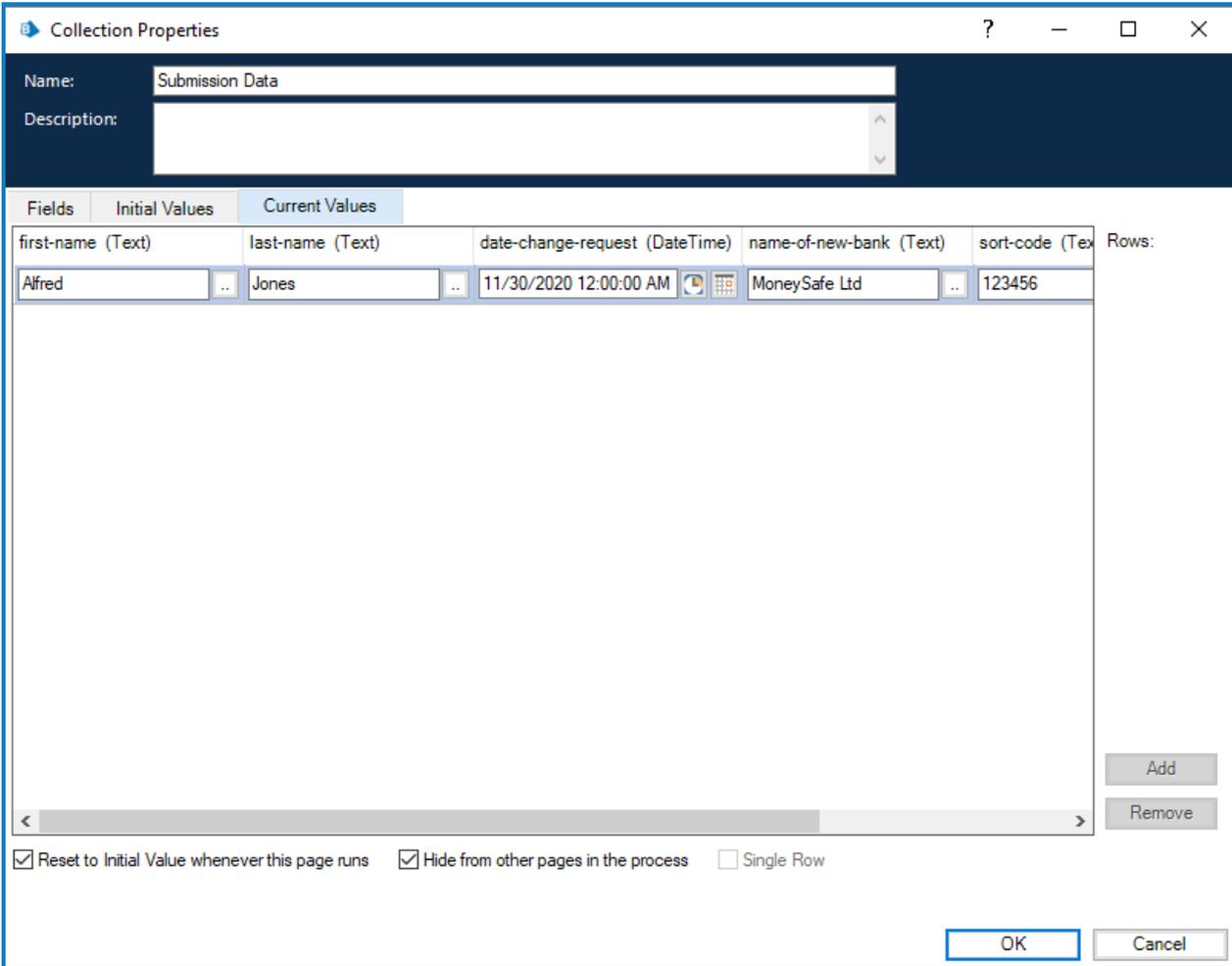
Name	Data Type	Value
Submission Id	Text	[Data._requestId]

The 'Outputs' and 'Conditions' tabs are also visible. On the right, the 'Group' section has 'Page' unchecked and 'Data Type' checked. Below that, a tree view shows various data types: Binaries, Collections, Dates, DateTimes, Flags, Images, Numbers, Passwords, Text (selected), Times, and TimeSpans. At the bottom, 'Stage logging' is set to 'Errors only' and 'Warning threshold' is set to 'System Default' with a 'Number of minutes' of 5.

The output parameters retrieves the contents of the Interact Form, using the submission ID into a collection.



One difference when the submission is retrieved using this methodology rather than the standard Get Next Item action is the formatting of Date elements. Using the Get Next Item technique the Date is returned in a Text format, whereas the Get Submission returns it as a DateTime format as below.



The screenshot shows a 'Collection Properties' dialog box with the following details:

- Name: Submission Data
- Description: (empty)
- Fields: first-name (Text), last-name (Text), date-change-request (DateTime), name-of-new-bank (Text), sort-code (Text)
- Current Values: Alfred, Jones, 11/30/2020 12:00:00 AM, MoneySafe Ltd, 123456
- Buttons: Add, Remove, OK, Cancel
- Options:  Reset to Initial Value whenever this page runs,  Hide from other pages in the process,  Single Row

Fields	Initial Values	Current Values
first-name (Text)		Alfred
last-name (Text)		Jones
date-change-request (DateTime)		11/30/2020 12:00:00 AM
name-of-new-bank (Text)		MoneySafe Ltd
sort-code (Text)		123456

For a complete list of restrictions please see the section 'Interact Web API Service Restrictions'.

 You cannot use the 'Get Submission' action unless you know the submission ID, as this is unique per submission it cannot be hardcoded into your automation. This must be retrieved using the 'Get Next Item' action.

## Get Form Schema

The Get Form Schema action gets a structure of an Interact Form in a collection allowing you to manipulate and then using the Create Submission action create an Interact Form either empty or partially created for a user to complete and submit.

There is only one input parameter which is the name of the Form within Interact that you are retrieving the structure for, as illustrated below.

**Action Properties**

Name:

Description:

Business Object:

Action:

Group:  Page  Data Type  View All Items

Inputs | Outputs | Conditions

Name	Data Type	Value
Form Name	Text	"Change of Bank Details"

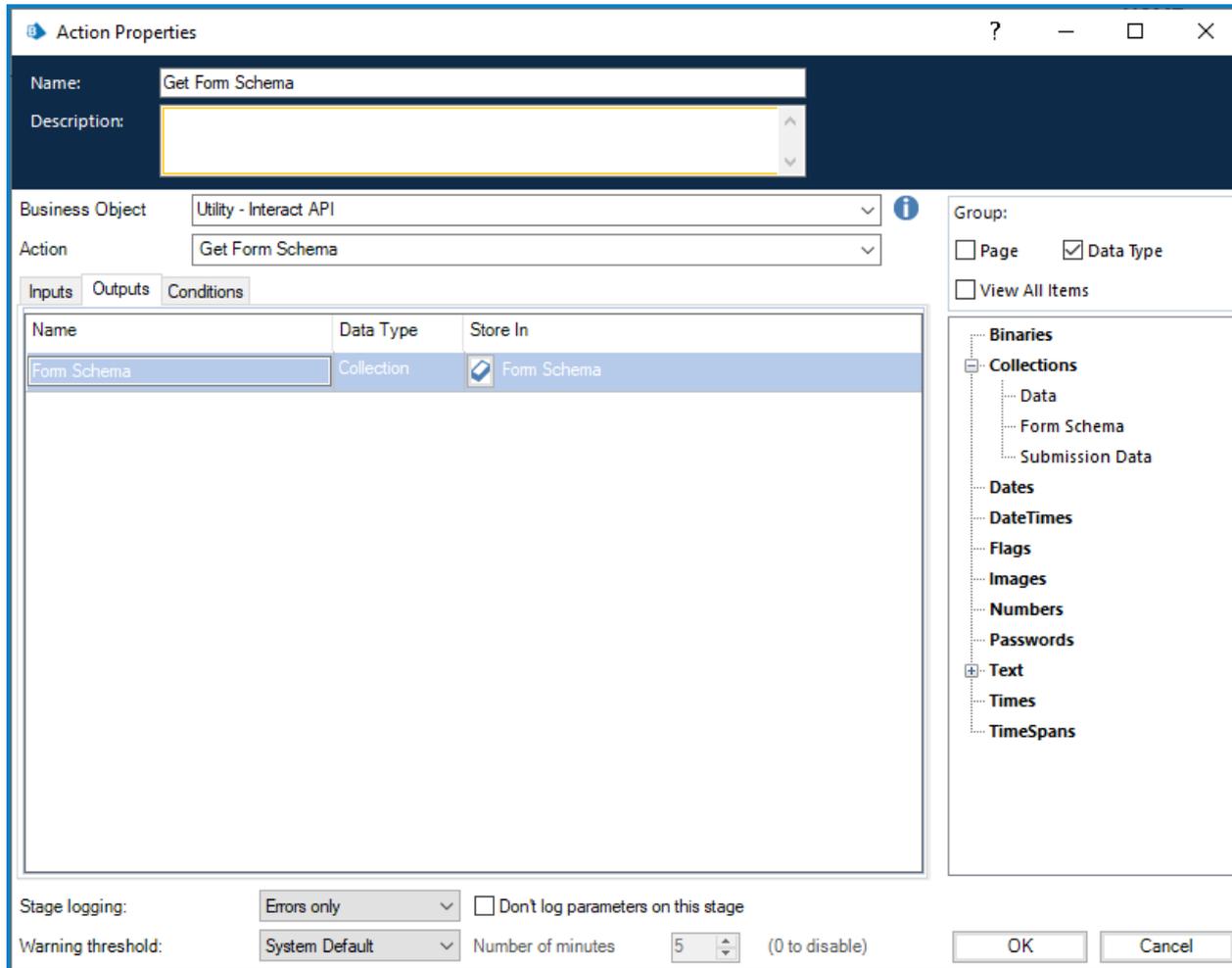
Binaries

- Collections
  - Data
  - Form Schema
  - Submission Data
- Dates
- DateTimes
- Flags
- Images
- Numbers
- Passwords
- Text**
- Times
- TimeSpans

Stage logging:   Don't log parameters on this stage

Warning threshold:  Number of minutes  (0 to disable)

The outputs parameters are similar to the 'Get Submission' action where the request status is returned along with a collection containing the fields.



The retrieved collection is a blank collection containing the structure of the Form.

Collection Properties

Name: Form Schema

Description:

Fields	Initial Values	Current Values
first-name (Text)	last-name (Text)	date-change-request (Text)
name-of-new-bank (Text)	sort-code (Text)	

Rows:

Reset to Initial Value whenever this page runs
  Hide from other pages in the process
  Single Row

OK Cancel

The structure of the fields presented in a similar format as the 'Get Next Item' format and therefore the Date element is requesting a Text data type to be submitted. The reason that Text type is used, is to support the multiple format Date element types within Interact, where the Date can be supplied in UK, US or International format.

The difference between the 'Get Next Item' and 'Get Form Schema' formats is the upload element field. The 'Get Next Item' action returns a link to where the file is stored, however, 'Get Form Schema' action returns a collection which includes not just a link but details of the file uploaded. This is also required when using the Upload File action.

The screenshot shows the 'Collection Properties' dialog box for a collection named 'Submission Data'. The 'Name' field is 'Submission Data' and the 'Description' field is empty. Below the description, there is a link to 'Submission Data' and the text 'attach-cv'. The main area of the dialog is a table with columns for 'name (Text)', 'size (Number)', 'extension (Text)', 'link (Text)', and 'type (Text)'. The table has one row with the following values: 'Invoice.pdf', '32808', 'pdf', 'https://file.local/Files/self', and 'application/pdf'. There are 'Add Row' and 'Remove Row' buttons on the right side of the table. At the bottom of the dialog, there are three checkboxes: 'Reset to Initial Value whenever this page runs' (checked), 'Hide from other pages in the process' (checked), and 'Single Row' (unchecked). 'OK' and 'Cancel' buttons are at the bottom right.

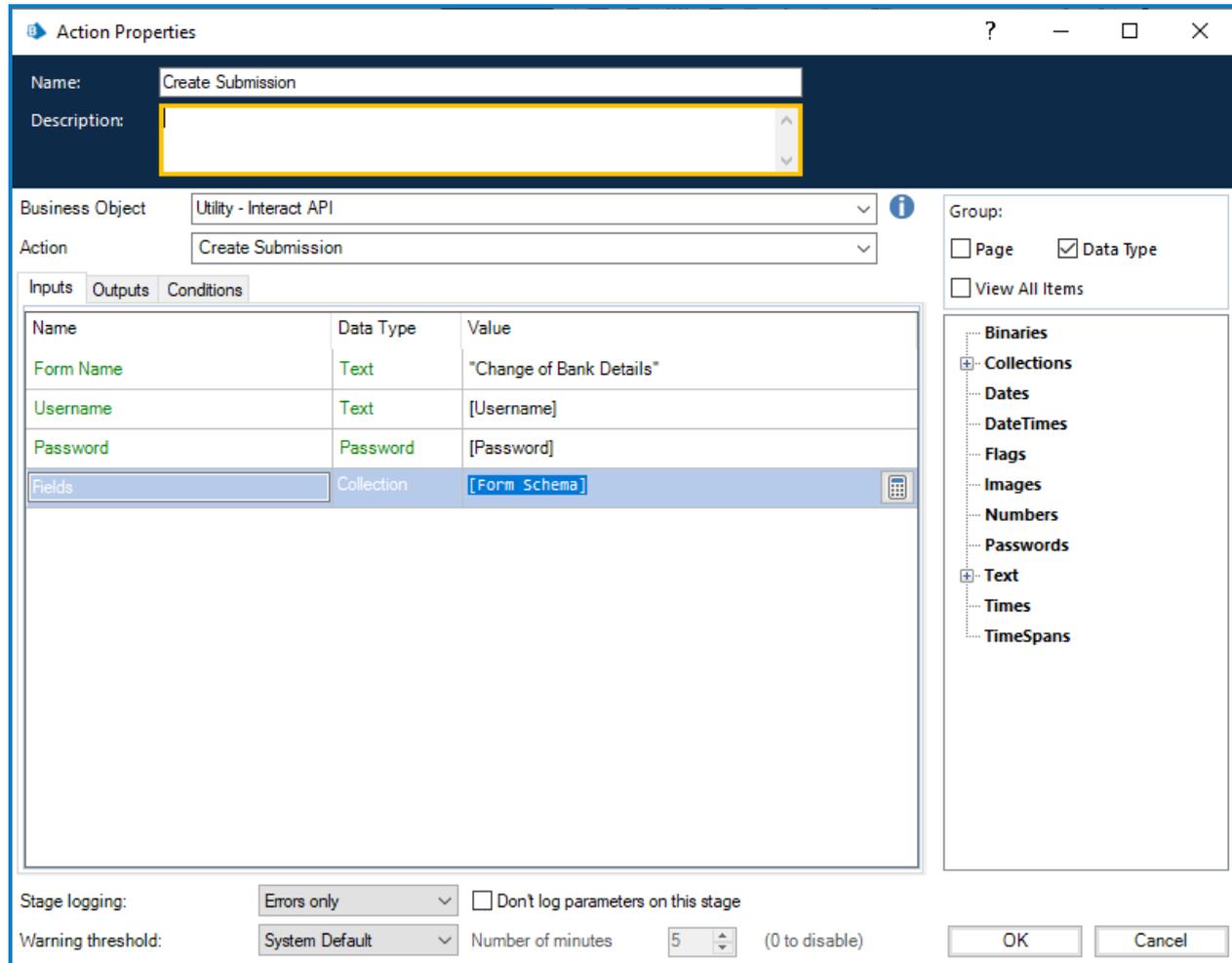
name (Text)	size (Number)	extension (Text)	link (Text)	type (Text)
Invoice.pdf	32808	pdf	https://file.local/Files/self	application/pdf

The format of the collection containing the details of the upload file must be set correctly, as illustrated above:

- **Name** – The name of the file including the extension.
- **Size** – The size of the file in bytes.
- **Extension** – The file extension, without the leading period or full stop.
- **Link** – The link to the uploaded file.
- **Type** – The type of file for example 'application/pdf' or 'image/png'.

## Create Submission

There are four parameters that need to be defined within the Create Submission action.



**Action Properties**

Name: Create Submission

Description:

Business Object: Utility - Interact API

Action: Create Submission

Group:  Page  Data Type  View All Items

Name	Data Type	Value
Form Name	Text	"Change of Bank Details"
Username	Text	[Username]
Password	Password	[Password]
Fields	Collection	[Form Schema]

Stage logging: Errors only  Don't log parameters on this stage

Warning threshold: System Default Number of minutes 5 (0 to disable)

OK Cancel

These four parameters are as follows:

- **Username** – This is the username of the Interact user that the submission will be created for.
- **Password** – The password for this Interact user.
- **Form Name** – The Form that will be created within Interact.
- **Fields** – The values that will be set within the created submission.

To assist in creating a new submission, [Get Form Schema on page 16](#) can be used to retrieve the basic structure of the Form. This could then be populated with new data items and then created as a new submission for a user.

 To create a submission, the fields and form schema must match that of the Form within Interact. Missing fields or an incorrect structure will end up creating a corrupt submission.

Once initiated the created submission will appear in the specified users Submission History as a submitted Form and will be added to the specified Blue Prism queue detailed in the Form construct.

There are no outputs parameters for the Create Submission action.

The user whose credentials are used to create the new submission must be allowed to access the Form. So, the Interact Role which contains the Form must have the user specified in the user list.

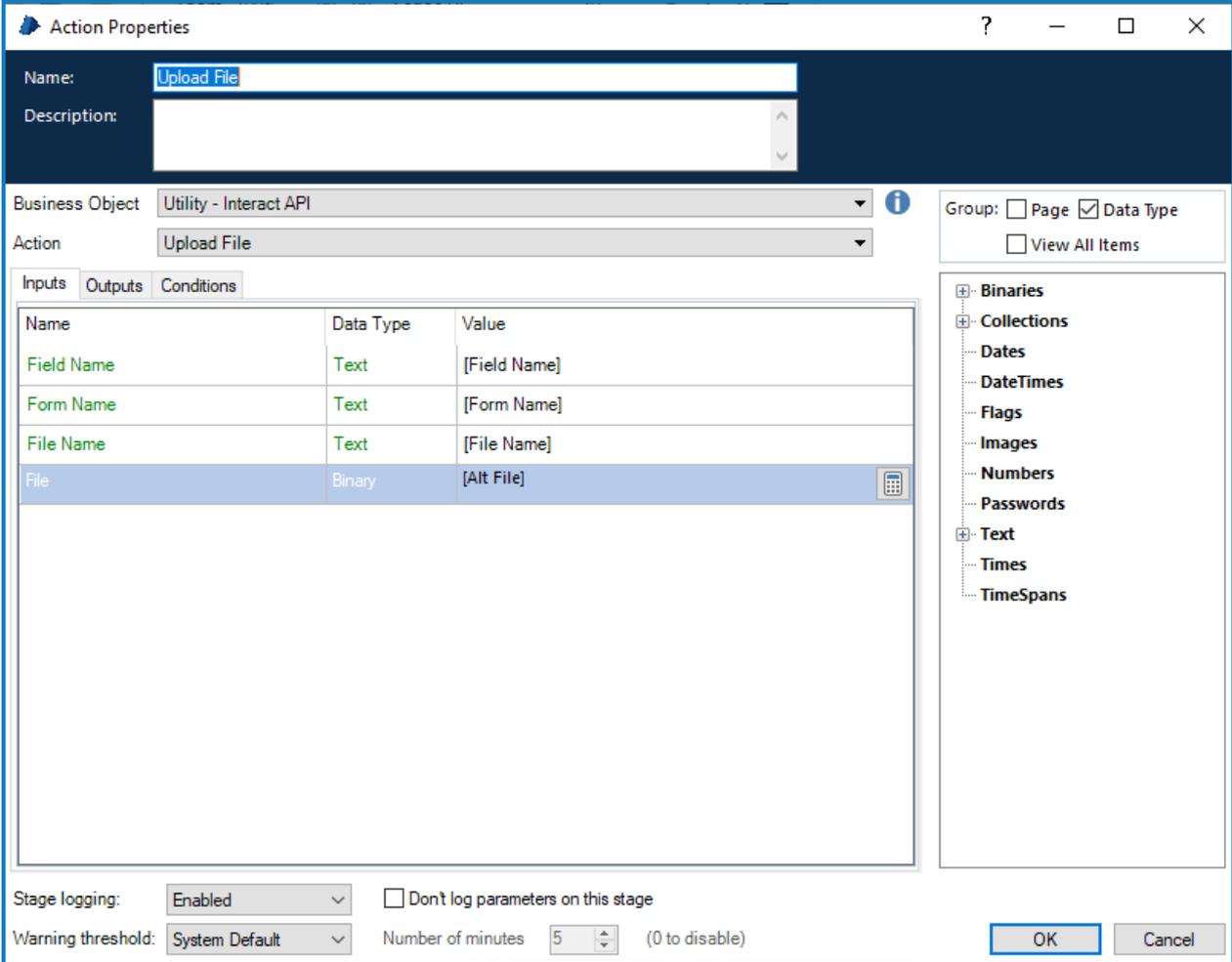
## Upload File

The Upload File action allows you to upload a file to local storage, for example, an image file or a document. It can be used in conjunction with [Create Submission on the previous page](#) or [Raise Submission on page 37](#).

The Upload file action only uploads one file at a time. When uploading multiple files to a form, you must run the upload file action for each file then store the output links separately.

There are four inputs for the action, as illustrated below:

- **Form Name** – The Form within Interact that the uploaded file will be associated with.
- **Field Name** – The automation ID of the field on the Form.
- **File Name** – The file that will be uploaded into the local storage. This must include the full name of the file including the extension.
- **File** – The contents of the file attached using a Data Item with the type set to Binary.



**Action Properties**

Name:

Description:

Business Object:

Action:

Group:  Page  Data Type  
 View All Items

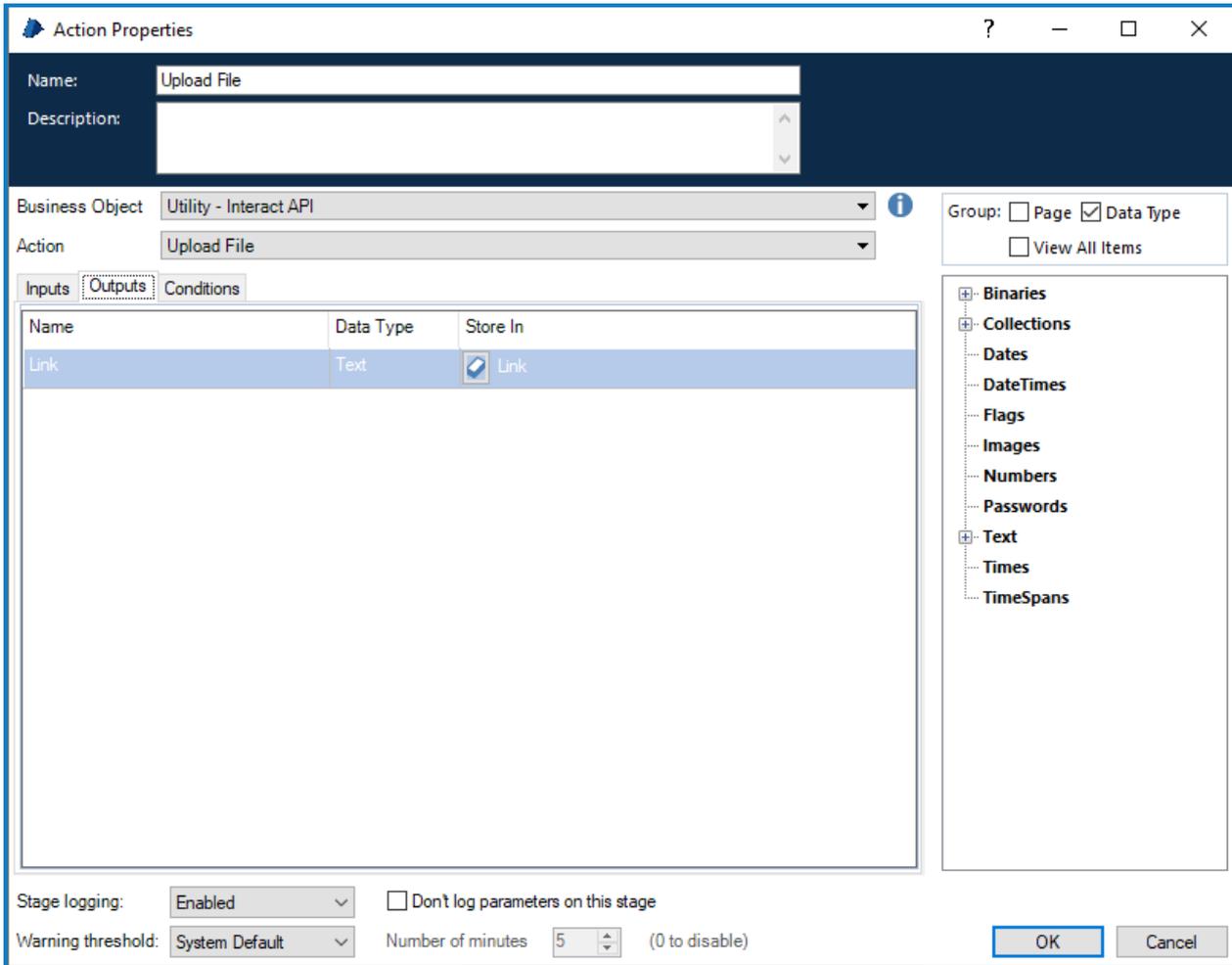
Name	Data Type	Value
Field Name	Text	[Field Name]
Form Name	Text	[Form Name]
File Name	Text	[File Name]
File	Binary	[Alt File]

Binaries  
Collections  
Dates  
DateTimes  
Flags  
Images  
Numbers  
Passwords  
Text  
Times  
TimeSpans

Stage logging:   Don't log parameters on this stage

Warning threshold:  Number of minutes  (0 to disable)

When uploading a file, you are provided with a link which can then be used in [Create Submission on the previous page](#) or [Raise Submission on page 37](#).



**Action Properties**

Name: Upload File

Description:

Business Object: Utility - Interact API

Action: Upload File

Group:  Page  Data Type  
 View All Items

Inputs Outputs Conditions

Name	Data Type	Store In
Link	Text	<input checked="" type="checkbox"/> Link

Binaries  
Collections  
Dates  
DateTimes  
Flags  
Images  
Numbers  
Passwords  
Text  
Times  
TimeSpans

Stage logging: Enabled  Don't log parameters on this stage

Warning threshold: System Default Number of minutes 5 (0 to disable)

OK Cancel

Once the file has been uploaded, you can attach the contents of the file to a [Create Submission on page 20](#) or [Raise Submission on page 37](#). To do this, you need to:

1. **Upload File** – Upload the file to the local storage.
2. **Get Form Schema** – Obtain the structure of the form where the file will be attached to.
3. **Create Submission or Raise Submission** – Which references the modified Form structure and attaches the uploaded file to the new submission.



When using Create Submission, the attached file can be downloaded and viewed from the Submission History tab of Interact. If Raise Submission is used, the user can only edit the link replacing the file, the user cannot view the attached file, unless it is an image, nor can they download the file.

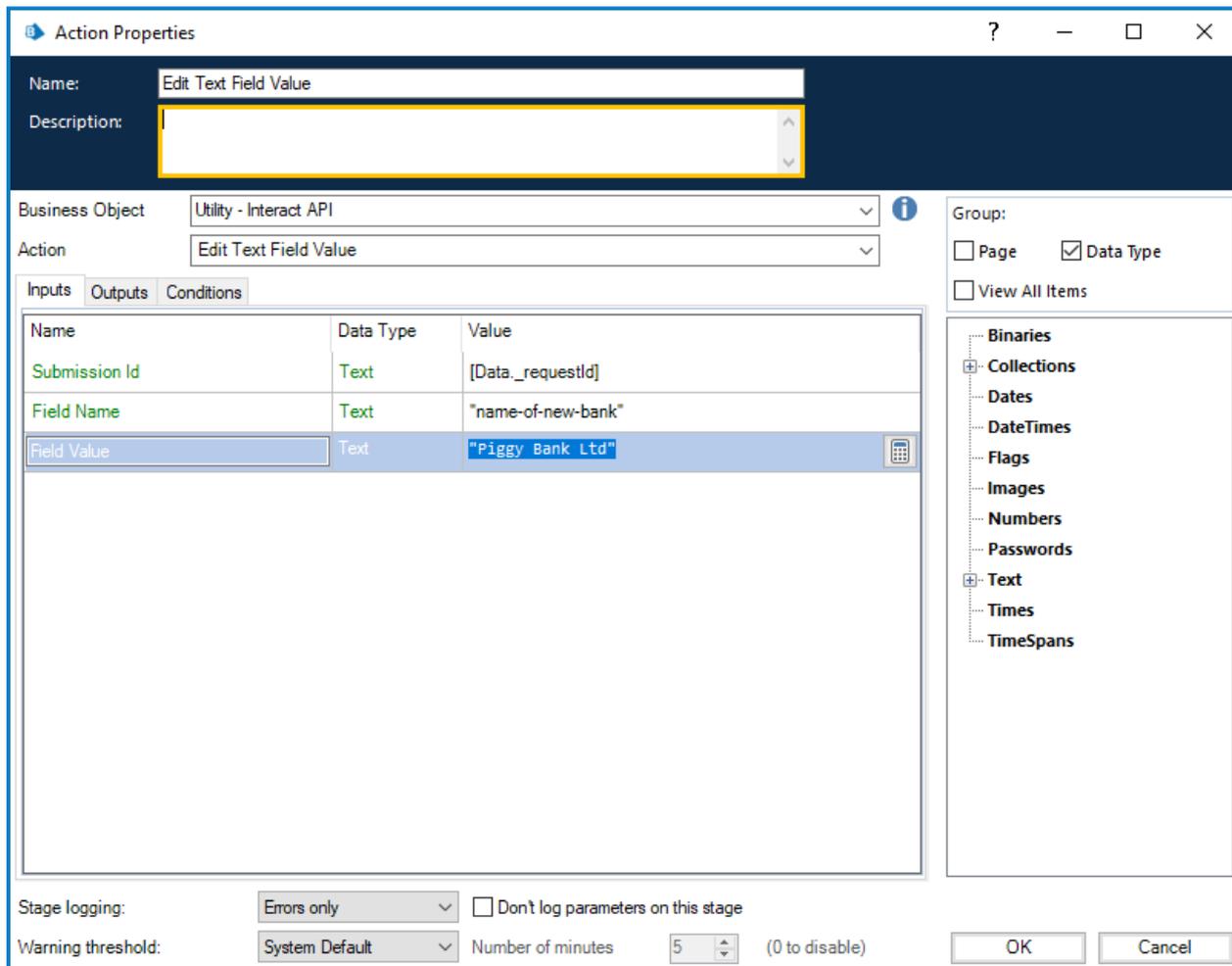
## Edit Submission Field Value

The Edit Submission Field Value action, like all the other actions, is called from the object. There are several actions within the object which allow you to update specific field elements, these actions link directly to the element type that is used in the Interact Form.

The Field Name in the sections below is the Automation ID that was created within the Form designer. Details for the different element types are described in the following sections.

### Text element

The Text element requires a simple Text field being passed to it.



## Number element

The Number element will accept either a number or a text being passed to it. Though the field type is a Text type you must supply a number using only the acceptable delimiters such as commas and decimal points. If text is sent back the Interact Form will display a message regarding invalid data.

**Action Properties**

Name:

Description:

Business Object:

Action:

Group:  Page  Data Type  View All Items

Inputs | Outputs | Conditions

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"salary"
Field Value	Text	"54,187.67"

- Binaries
- Collections
- Dates
- DateTimes
- Flags
- Images
- Numbers
- Passwords
- Text
- Times
- TimeSpans

Stage logging:   Don't log parameters on this stage

Warning threshold:  Number of minutes:  (0 to disable)

## Date element

The Date element uses Text format to edit values in an Interact Form, this is to support the three different formats within the Interact Form. The Form will allow you to use these formats for dates:

- DD/MM/YYYY – UK format;
- MM/DD/YYYY – US format;
- YYYY/MM/DD – International format.

If you have set the Interact Form to accept the International format for dates and you send the date back in either the UK or US format, the Interact Form will display an invalid data message.

**Action Properties**

Name: Edit Date Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Date Field Value

Group:  Page  Data Type  View All Items

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"date-of-birth"
Field Value	Text	"22/06/1972"

Stage logging: Errors only  Don't log parameters on this stage

Warning threshold: System Default Number of minutes: 5 (0 to disable)

OK Cancel

## Time element

The Time element uses Text format to edit values in an Interact Form. You can supply the Time in one of two formats these being:

- 03:00:00 PM; or
- 15:00:00.

Both will return 3 o'clock in the afternoon in Interact.

**Action Properties**

Name: Edit Time Field Value  
 Description: [Empty text area]

Business Object: Utility - Interact API  
 Action: Edit Time Field Value

Inputs | Outputs | Conditions

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"current-time"
Field Value	Text	"15:00:00"

Group:  
 Page  Data Type  
 View All Items

- Binaries
- Collections
- Dates
- DateTimes
- Flags
- Images
- Numbers
- Passwords
- Text
- Times
- TimeSpans

Stage logging: Errors only  Don't log parameters on this stage  
 Warning threshold: System Default Number of minutes: 5 (0 to disable)

OK Cancel

### Checkbox element (Single)

The Checkbox Edit Value is two different actions supporting use of both Single and Multiple Checkboxes. To update the values for a Single Checkbox, it is a simple text string that is submitted.

The screenshot shows the 'Action Properties' dialog box for the action 'Edit Checkbox Single Field Value'. The dialog is titled 'Action Properties' and has a dark blue header. Below the header, there are fields for 'Name' (containing 'Edit Checkbox Single Field Value') and 'Description' (empty). The 'Business Object' is set to 'Utility - Interact API' and the 'Action' is 'Edit Checkbox Single Field Value'. There are tabs for 'Inputs', 'Outputs', and 'Conditions', with 'Inputs' selected. A table lists the input parameters:

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"what-pets-do-you-have"
Field Value	Text	"dog"

Below the table, there are settings for 'Stage logging' (set to 'Errors only') and 'Warning threshold' (set to 'System Default'). There is also a checkbox for 'Don't log parameters on this stage' and a spinner for 'Number of minutes' (set to 5). On the right side, there is a 'Group' section with checkboxes for 'Page' (unchecked) and 'Data Type' (checked), and a 'View All Items' checkbox (unchecked). A tree view on the right shows a hierarchy of data types: Binaries, Collections, Dates, DateTimes, Flags, Images, Numbers, Passwords, Text (selected), Times, and TimeSpans. At the bottom right, there are 'OK' and 'Cancel' buttons.

### Checkbox element (Multiple)

For the Checkbox Edit Value (Multiple) again it is a simple text string that is submitted but using a comma separated string for the different values.

The screenshot shows the 'Action Properties' dialog box for the action 'Edit Checkbox Multiple Field Value'. The 'Name' field is filled with 'Edit Checkbox Multiple Field Value'. The 'Description' field is empty. The 'Business Object' is 'Utility - Interact API' and the 'Action' is 'Edit Checkbox Multiple Field Value'. The 'Inputs' tab is selected, showing a table with three rows: 'Submission Id' (Text, [Data\_requestId]), 'Field Name' (Text, "what-pets-do-you-have"), and 'Field Value' (Text, "dog, cat, rabbit"). The 'Field Value' row is highlighted. On the right, the 'Group' section has 'Page' unchecked and 'Data Type' checked. Below it is a tree view with categories like Binaries, Collections, Dates, DateTimes, Flags, Images, Numbers, Passwords, Text, Times, and TimeSpans. At the bottom, 'Stage logging' is set to 'Errors only' and 'Warning threshold' is 'System Default' with a 'Number of minutes' set to 5.

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"what-pets-do-you-have"
Field Value	Text	"dog, cat, rabbit"

## Dropdown element (Single)

The Dropdown element, like the Checkbox element, supports two actions for the single and multiple entries. To update the values for a Single Dropdown, it is a simple text string that is submitted.

**Action Properties**

Name: Edit Dropdown Single Field Value

Description: [Empty text box]

Business Object: Utility - Interact API

Action: Edit Dropdown Single Field Value

Group:  Page  Data Type  View All Items

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"what-pets-do-you-have"
Field Value	Text	"snake"

Binaries  
Collections  
Dates  
DateTimes  
Flags  
Images  
Numbers  
Passwords  
Text  
Times  
TimeSpans

Stage logging: Errors only  Don't log parameters on this stage

Warning threshold: System Default Number of minutes: 5 (0 to disable)

OK Cancel

## Dropdown element (Multiple)

For the Dropdown Edit Value (Multiple) again it is a simple text string that is submitted but using a comma separated string for the different values.

The screenshot shows the 'Action Properties' dialog box for the action 'Edit Dropdown Multiple Field Value'. The dialog is titled 'Action Properties' and has a dark blue header. Below the header, there are fields for 'Name' (containing 'Edit Dropdown Multiple Field Value') and 'Description' (an empty text area). Below these are dropdown menus for 'Business Object' (set to 'Utility - Interact API') and 'Action' (set to 'Edit Dropdown Multiple Field Value'). To the right of these dropdowns is an information icon. Below the dropdowns are tabs for 'Inputs', 'Outputs', and 'Conditions'. The 'Inputs' tab is active, showing a table with three rows: 'Submission Id' (Text, [Data\_requestId]), 'Field Name' (Text, "what-pets-do-you-have"), and 'Field Value' (Text, "fish, snake"). Below the table is a large empty text area. To the right of the table is a 'Group' section with checkboxes for 'Page', 'Data Type' (checked), and 'View All Items'. Below the 'Group' section is a tree view showing categories like 'Binaries', 'Collections', 'Dates', 'DateTimes', 'Flags', 'Images', 'Numbers', 'Passwords', 'Text', 'Times', and 'TimeSpans'. At the bottom of the dialog, there are settings for 'Stage logging' (set to 'Errors only') and 'Warning threshold' (set to 'System Default', with a 'Number of minutes' spinner set to 5). There are 'OK' and 'Cancel' buttons at the bottom right.

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"what-pets-do-you-have"
Field Value	Text	"fish, snake"

## Radio group element

The Radio group element is like a single Checkbox or Dropdown element. The edited value is a single text field which includes the required updated value.

**Action Properties**

Name:

Description:

Business Object:

Action:

Group:  Page  Data Type  View All Items

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"gender-selection"
Field Value	Text	"female"

Binaries  
Collections  
Dates  
DateTimes  
Flags  
Images  
Numbers  
Passwords  
Text  
Times  
TimeSpans

Stage logging:   Don't log parameters on this stage

Warning threshold:  Number of minutes  (0 to disable)

## Table element

The Table element uses a collection to enable you to update the values. The collection must contain the same number of columns as the form field within the targeted form. Automation IDs must be used for the field names of the collection. The number of rows can be increased or decreased to accommodate the data being updated.

**Action Properties**

Name: Edit Table Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Table Field Value

Group:  Page  Data Type  View All Items

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"sales-orders"
Field Value	Collection	[Sales Orders]

Stage logging: Errors only  Don't log parameters on this stage

Warning threshold: System Default Number of minutes: 5 (0 to disable)

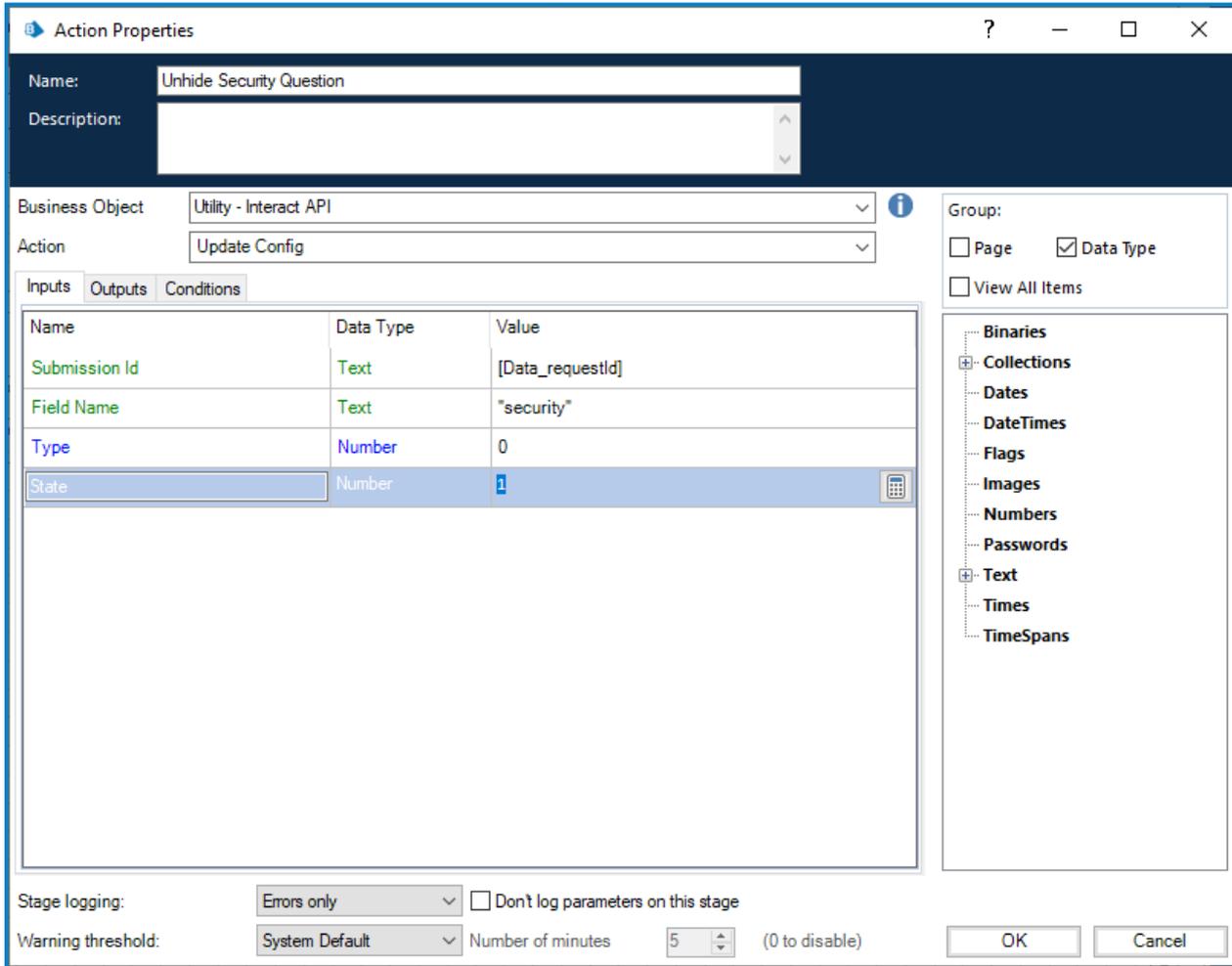
OK Cancel

The Edit Table action overwrites the whole table as opposed to appending rows to an existing table.

## Update Config

 The Update Config action has been replaced with individual actions and will be removed from the Blue Prism Object in a future release. Each of the state options now has an independent action. If you are using Update Config, you should change to use the appropriate action.

The Update Config action allows you to change the state of a field or a page within a submission. The inputs for the action are illustrated below. This example shows a field with the name of 'security' being set to a 'mandatory' state. The inputs are the Submission ID, the Automation ID of the field or the page name if you are change the state of a page, a Type and the State inputs.



**Action Properties**

Name: Unhide Security Question  
Description:

Business Object: Utility - Interact API  
Action: Update Config

Group:  Page  Data Type  
 View All Items

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"security"
Type	Number	0
State	Number	1

Stage logging: Errors only  Don't log parameters on this stage  
Warning threshold: System Default Number of minutes 5 (0 to disable)

OK Cancel

There are no outputs from the Update Config action.

The Type and the State inputs work together. For the Type there are two values that can be used:

- 0 – to modify the state of a field
- 2 – to modify the state of a page

If modifying the state of a field (Type set to '0'), then the State options are as follows:

- 0 – changes the field to an optional state. Please use the action [Change Field State to Optional on page 40](#).
- 1 – changes the field to a mandatory state. Please use the action [Change Field State to Mandatory on page 41](#).
- 2 – changes the field to a read-only state. Please use the action [Change Field State to Read Only on page 42](#).
- 3 – changes the field to a hidden state. Please use the action [Change Field State to Hidden on page 43](#).

If modifying the state of a page (Type set to '2'), then the State options are as follows:

- 0 – changes the page to a shown state. Please use the action [Change Page State to Visible on page 44](#).
- 1 – changes the page to a shown state. Please use the action [Change Page State to Visible on page 44](#).
- 2 – changes the page to a shown state. Please use the action [Change Page State to Visible on page 44](#).
- 3 – changes the page to a hidden state. Please use the action [Change Page State to Hidden on page 45](#).

The Update Config is typically used in a human/digital worker collaboration scenario where the Form is dynamically updated as the human interacts with the automation. This is detailed separately showing how these Actions can be used to update a Form with input from a user.

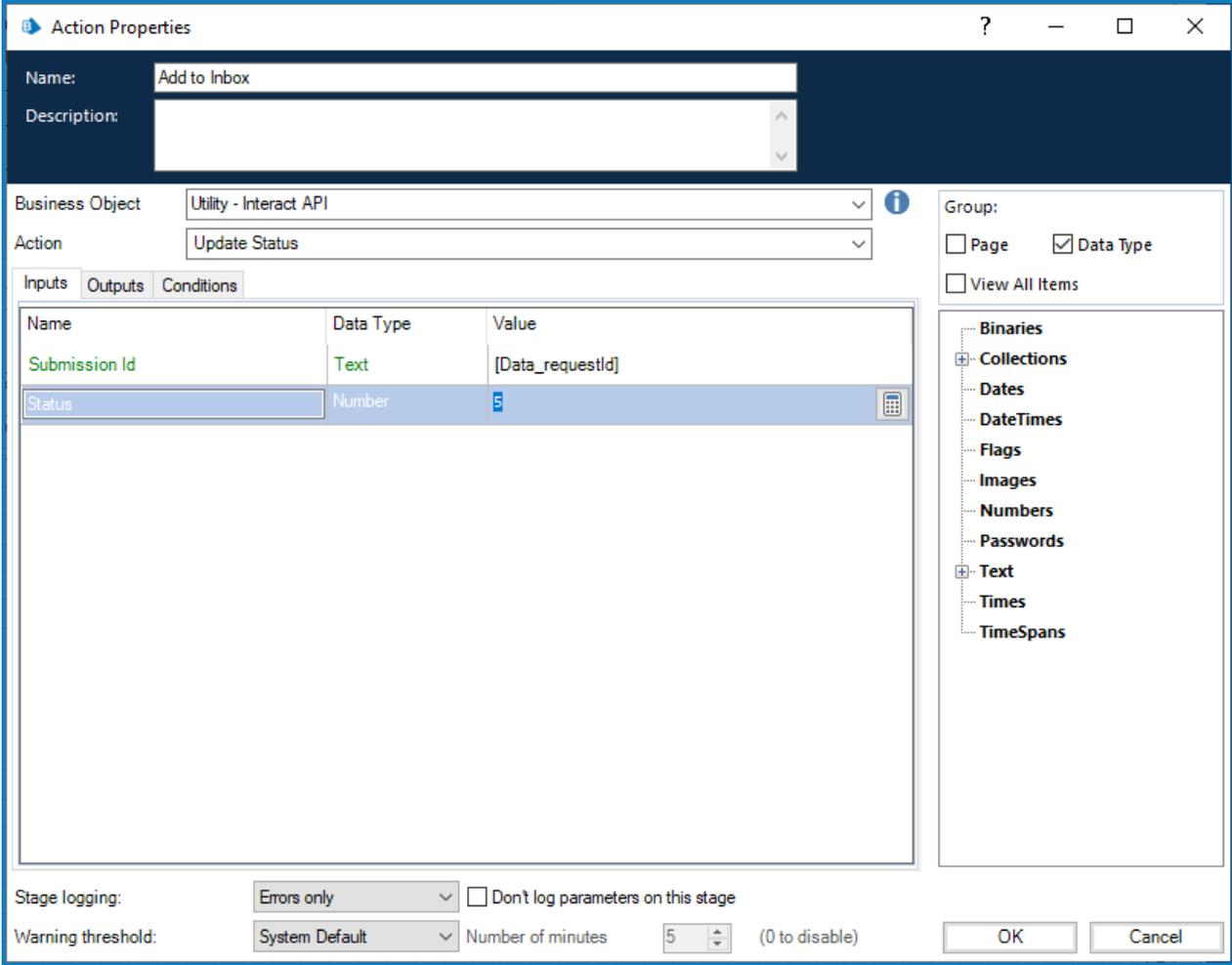
The Paragraph elements are blocks of text that cannot be changed in a Form once the Form has been deployed. However, it is possible using 'Get Form Schema' to obtain the element ID of the Paragraph field, then using Update Config, hide or unhide them as the update progresses.

## Update Status

 The Update Status action has been replaced with individual actions and will be removed from the Blue Prism Object in a future release. Each of the status options now has an independent action. If you are using Update Status, you should change to use the appropriate action.

The Update Status action, like the [Update Config on page 33](#) action, is typically used in a Human-in-the-Loop scenario. The action allows you to move the submission between different Interact tabs. This action allows you to move the submission to a different tab in Interact, and change the status of the submission in the **Submission history** tab.

The inputs for the action are illustrated below and are the Submission ID and the Status.



Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Status	Number	5

There are six statuses that can be applied to a submission, these are defined as follows:

- 0 – Sets the status to Submitted and moves the submission to the **Submission history** tab under **History**, if it is not already there. The Status filter can be used to only display submitted items. Please use the action [Move Submission to History Tab on page 46](#).
- 1 – Sets the status to Approved and moves the submission to the **Submission history** tab under **History**, if it is not already there. The Status filter can be used to only display approved items. Please use the action [Move Submission to Approved Tab on page 47](#).
- 2 – Moves the submission to the **Awaiting approval** tab under **History** for the submitter, and into the **Approvals** tab under **My Work** for the approver. Please use the action [Move Submission to Review Tab on page 48](#).

- 3 – Sets the status to Declined and moves the submission to the **Submission history** tab under **History**, if it is not already there. The Status filter can be used to only display declined items. Please use the action [Move Submission to Declined Tab on page 49](#).
- 4 – Moves the submission to the **Draft** tab under **My Work**. Please use the action [Move Submission to Draft Tab on page 50](#).
- 5 – Moves the submission to the **Inbox** tab under **My Work**. Please use the action [Move Submission to Inbox Tab on page 51](#).
- 6 – Moves the submission to the **Archived** tab under **History**. Please use the action [Move Submission to Archived Tab on page 52](#).

## Raise Submission

The Raise Submission action allows you to create a submission in a user's particular Inbox folder. The submission raised can be fully or partially completed, or even completely empty depending on your requirements.

The inputs for the action are the Form Name, the User ID (which can be obtained by running 'Get Users by Form Name') and the data to be loaded in the fields, as illustrated below. The data file loaded can be created by running the 'Get Form Schema' command.

Name	Data Type	Value
Form Name	Text	[Form Name]
User Id	Number	1
Fields	Collection	[Form Schema]

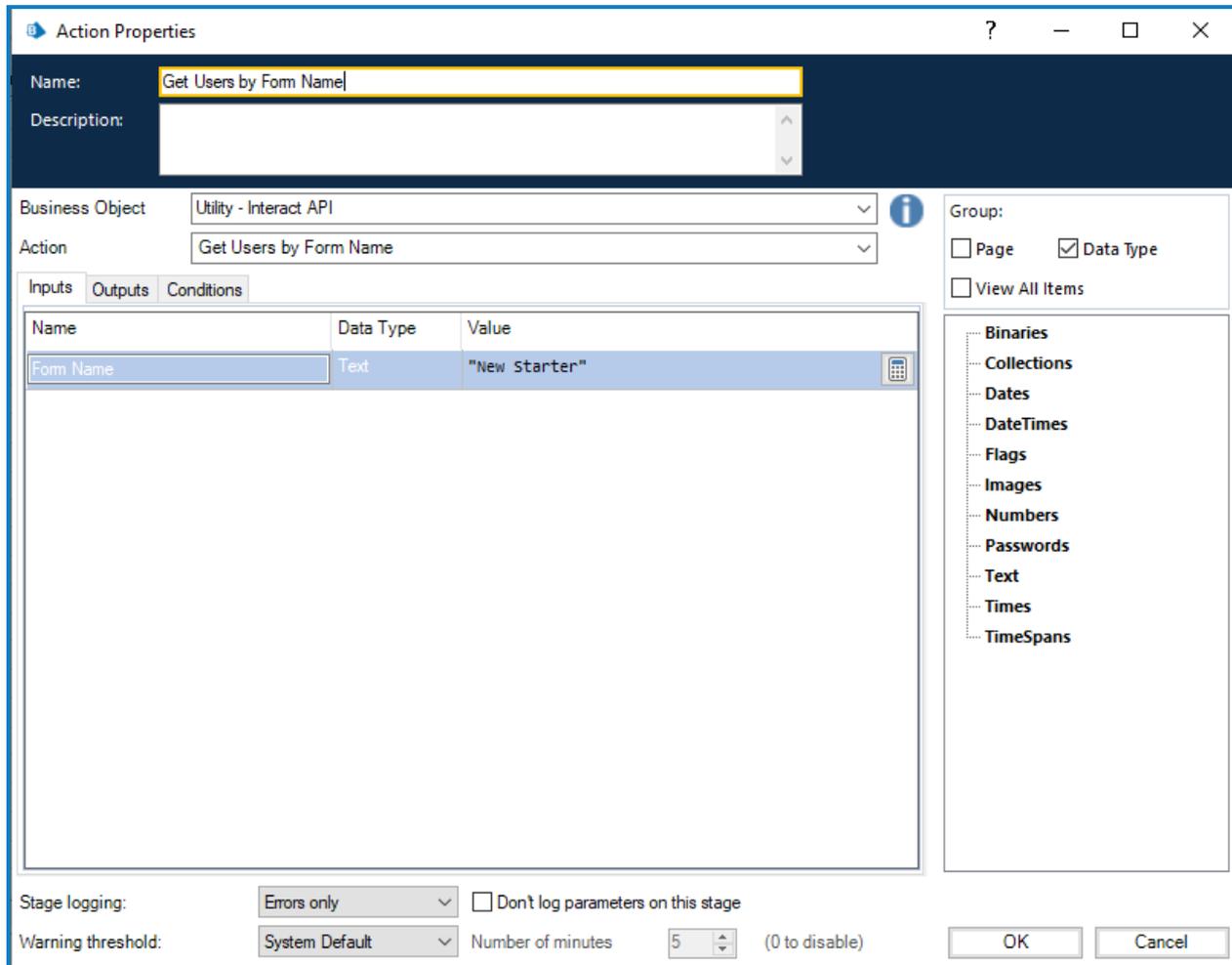
There are no outputs for the Raise Submission action.



When using Raise Submission with [Upload File on page 21](#), ensure you correctly format the collection containing the details of the file being attached. See [Get Form Schema on page 16](#) for more details about the format of the collection.

## Get Users by Form Name

The Get Users by Form Name allows you to collect a list of users associated with a particular Form Name. The input for the action is the Form Name, as illustrated below.



**Action Properties**

Name:

Description:

Business Object:

Action:

Inputs | Outputs | Conditions

Name	Data Type	Value
Form Name	Text	"New Starter"

Group:

Page  Data Type

View All Items

- Binaries
- Collections
- Dates
- DateTimes
- Flags
- Images
- Numbers
- Passwords
- Text
- Times
- TimeSpans

Stage logging:   Don't log parameters on this stage

Warning threshold:  Number of minutes  (0 to disable)

OK Cancel

The output is a collection that lists the users associated with that Form Name.

**Action Properties**

Name:

Description:

Business Object:

Action:

Group:  Page  Data Type  View All Items

Inputs Outputs **Conditions**

Name	Data Type	Store In
Users	Collection	<input checked="" type="checkbox"/> Users

Binaries  
 Collections  
 Dates  
 DateTimes  
 Flags  
 Images  
 Numbers  
 Passwords  
 Text  
 Times  
 TimeSpans

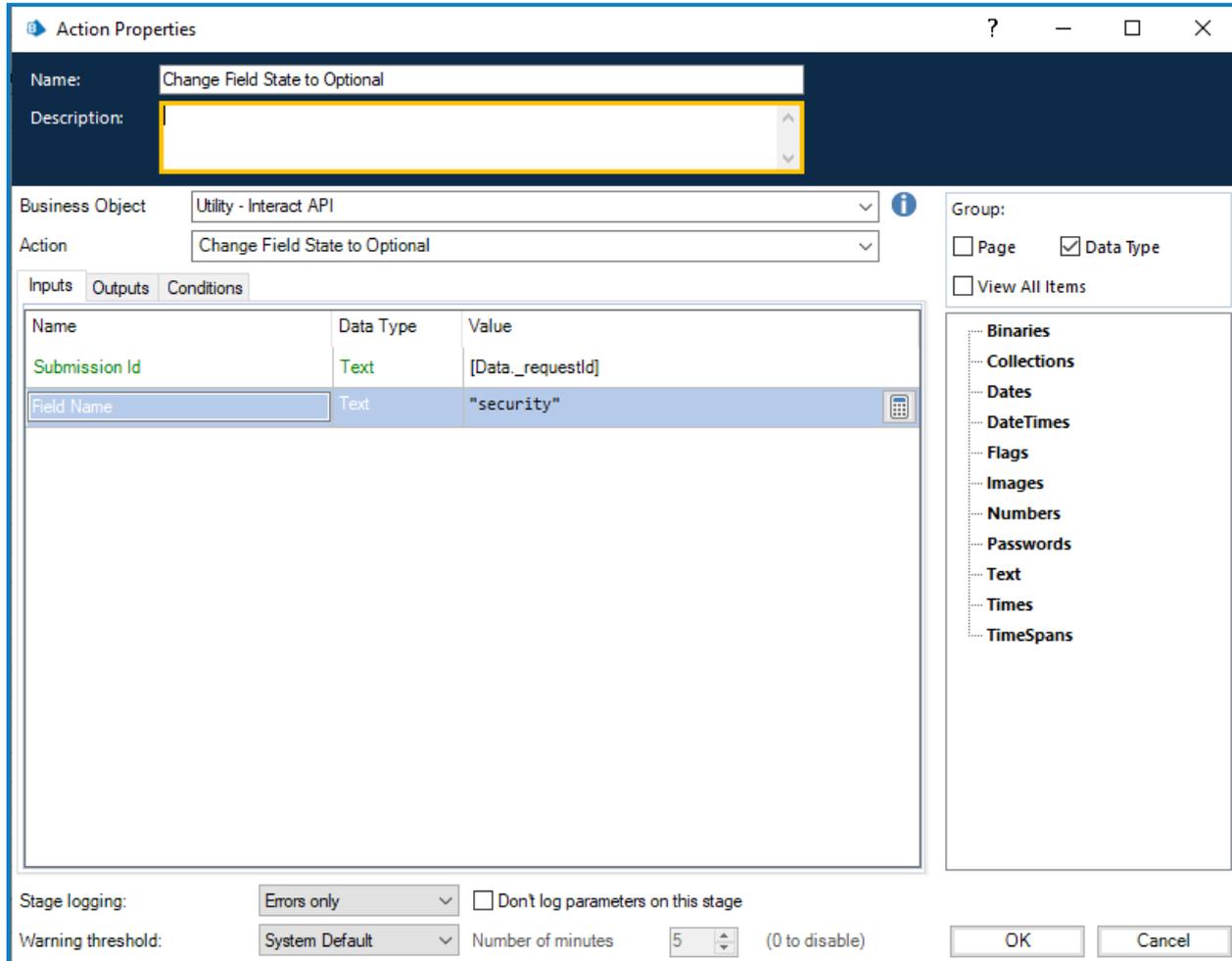
Stage logging:   Don't log parameters on this stage

Warning threshold:  Number of minutes  (0 to disable)

## Change Field State to Optional

The Change Field State to Optional allows you to change the state of a field within a submission, moving it to an Optional condition.

The inputs for the action are the Submission ID and the Field Name you are changing the state to optional, as illustrated below. This example shows a field with the name of 'security' being set to a 'optional' state.

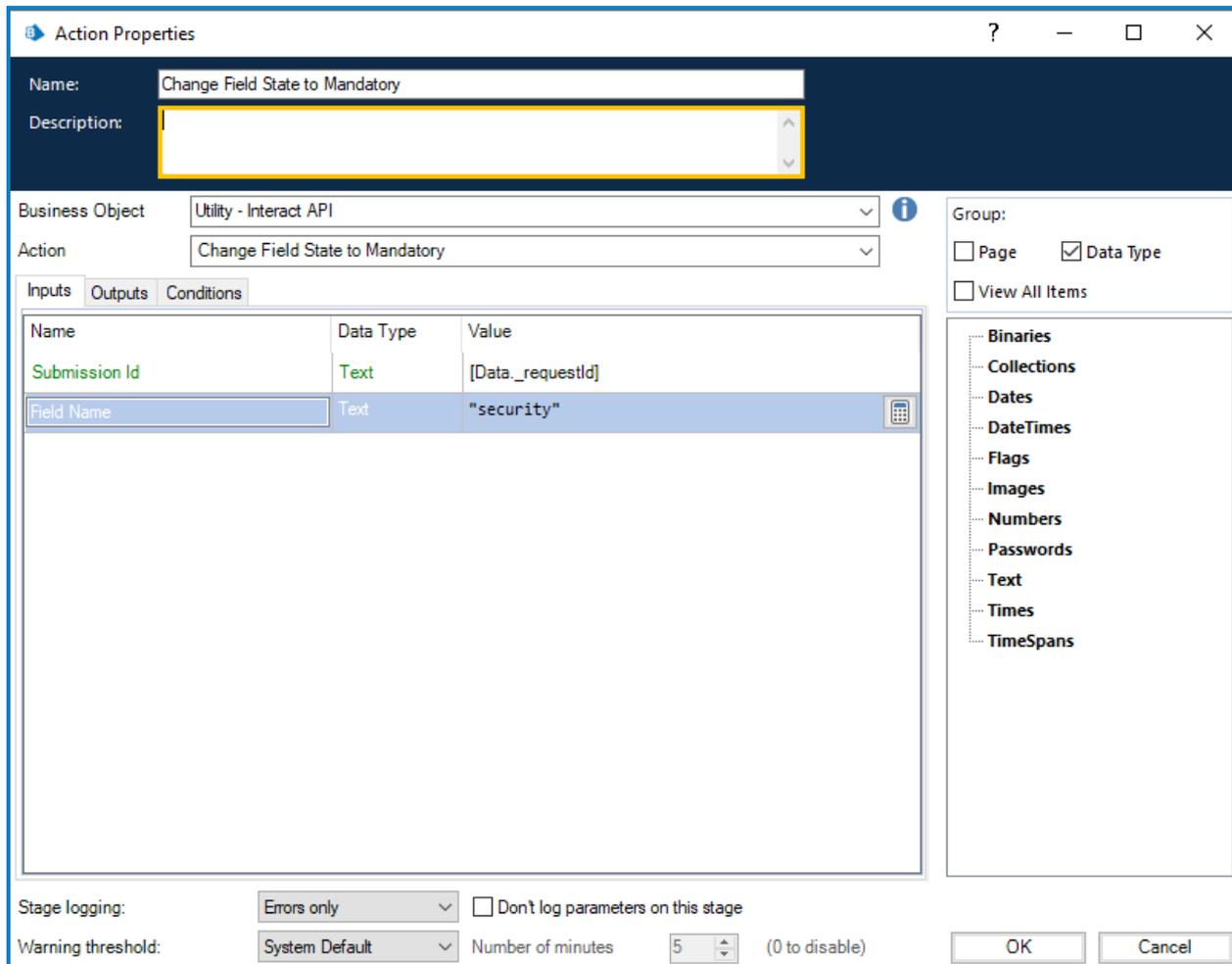


There are no outputs for the Change Field State to Optional.

## Change Field State to Mandatory

The Change Field State to Mandatory allows you to change the state of a field within a submission, moving it to a Mandatory condition.

The inputs for the action are the Submission ID and the Field Name you are changing the state to mandatory, as illustrated below. This example shows a field with the name of 'security' being set to a 'mandatory' state.



There are no outputs for the Change Field State to Mandatory.

## Change Field State to Read Only

The Change Field State to Read Only allows you to change the state of a field within a submission, moving it to a Read Only condition.

The inputs for the action are the Submission ID and the Field Name you are changing the state to read only, as illustrated below. This example shows a field with the name of 'security' being set to a 'read only' state.

The screenshot shows the 'Action Properties' dialog box for the 'Change Field State to Read Only' action. The dialog is titled 'Action Properties' and has a dark blue header. The 'Name' field is set to 'Change Field State to Read Only'. The 'Description' field is empty. The 'Business Object' is 'Utility - Interact API' and the 'Action' is 'Change Field State to Read Only'. The 'Inputs' tab is selected, showing a table with two input fields: 'Submission Id' (Text, [Data\_requestId]) and 'Field Name' (Text, "security"). The 'Outputs' and 'Conditions' tabs are also visible. On the right side, there are checkboxes for 'Page', 'Data Type', and 'View All Items'. Below these are several categories: Binaries, Collections, Dates, DateTimes, Flags, Images, Numbers, Passwords, Text, Times, and TimeSpans. At the bottom, there are settings for 'Stage logging' (Errors only) and 'Warning threshold' (System Default, 5 minutes). The 'OK' and 'Cancel' buttons are at the bottom right.

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"security"

There are no outputs for the Change Field State to Read Only.

## Change Field State to Hidden

The Change Field State to Hidden allows you to change the state of a field within a submission, moving it to a Hidden condition.

The inputs for the action are the Submission ID and the Field Name you are changing the state to hidden, as illustrated below. This example shows a field with the name of 'security' being set to a 'hidden' state.

The screenshot shows the 'Action Properties' dialog box for the 'Change Field State to Hidden' action. The dialog is titled 'Action Properties' and has a dark blue header. The 'Name' field is set to 'Change Field State to Hidden'. The 'Description' field is empty. The 'Business Object' is set to 'Utility - Interact API'. The 'Action' is set to 'Change Field State to Hidden'. The 'Group' is set to 'Data Type'. The 'Inputs' tab is selected, showing a table with two input fields: 'Submission Id' (Text, [Data\_requestId]) and 'Field Name' (Text, "security"). The 'Outputs' and 'Conditions' tabs are empty. The 'Stage logging' is set to 'Errors only' and 'Don't log parameters on this stage' is unchecked. The 'Warning threshold' is set to 'System Default' and 'Number of minutes' is set to 5. The 'OK' and 'Cancel' buttons are at the bottom right.

Name	Data Type	Value
Submission Id	Text	[Data_requestId]
Field Name	Text	"security"

There are no outputs for the Change Field State to Hidden.

## Change Page State to Visible

The Change Page State to Visible allows you to make a page become visible within a submission.

The inputs for the action are the Submission ID and the Page Name you are changing to become visible, as illustrated below. This example shows a page with the name of 'inputs' being set to a 'visible' state.

**Action Properties**

Name: Change Page State to Visible

Description:

Business Object: Utility - Interact API

Action: Change Page State to Visible

Group:  Page  Data Type  View All Items

Inputs Outputs Conditions

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Page Name	Text	"Inputs"

Binaries  
Collections  
Dates  
DateTimes  
Flags  
Images  
Numbers  
Passwords  
Text  
Times  
TimeSpans

Stage logging: Errors only  Don't log parameters on this stage

Warning threshold: System Default Number of minutes: 5 (0 to disable)

OK Cancel

There are no outputs for the Change Page State to Visible.

## Change Page State to Hidden

The Change Page State to Hidden allows you to make a page hidden from view within a submission.

The inputs for the action are the Submission ID and the Page Name you are changing to become hidden , as illustrated below. This example shows a page with the name of 'inputs' being set to a 'hidden' state.

**Action Properties**

Name: Change Page State to Hidden

Description:

Business Object: Utility - Interact API

Action: Change Page State to Hidden

Group:  Page  Data Type  View All Items

Inputs Outputs Conditions

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Page Name	Text	"Inputs"

Binaries  
Collections  
Dates  
DateTimes  
Flags  
Images  
Numbers  
Passwords  
Text  
Times  
TimeSpans

Stage logging: Errors only  Don't log parameters on this stage

Warning threshold: System Default Number of minutes: 5 (0 to disable)

OK Cancel

There are no outputs for the Change Page State to Hidden.

## Move Submission to History Tab

The Move Submission to History Tab allows you to move a submission from its current folder to the **Submission history** folder under **History**. This action also updates the status of the submission to Submitted. The Status filter in Interact can be used to display just the submissions with this status.

The input for the action is the Submission ID of the submission being moved, as illustrated below.

The screenshot shows the 'Action Properties' dialog box for the 'Move Submission to Submitted History Tab' action. The dialog is titled 'Action Properties' and has a dark blue header. The 'Name' field is set to 'Move Submission to Submitted History Tab'. The 'Description' field is empty. The 'Business Object' is 'Utility - Interact API'. The 'Action' is 'Move Submission to Submitted History Tab'. The 'Group' is 'Data Type'. The 'Inputs' tab is selected, showing a table with one input: 'Submission Id' of type 'Text' with value '[Data.\_requestId]'. The 'Outputs' and 'Conditions' tabs are empty. The 'Stage logging' is set to 'Errors only' and 'Don't log parameters on this stage' is unchecked. The 'Warning threshold' is 'System Default' and 'Number of minutes' is '5'.

Name	Data Type	Value
Submission Id	Text	[Data._requestId]

There are no outputs for the Move Submission to History Tab.

## Move Submission to Approved Tab

The Move Submission to Approved Tab allows you to move a submission from its current folder to the **Submission history** folder under **History**. This action also updates the status of the submission to Approved. The Status filter in Interact can be used to display just the submissions with this status.

The input for the action is the Submission ID of the submission being moved, as illustrated below.

The screenshot shows the 'Action Properties' dialog box for the 'Move Submission to Approved Tab' action. The dialog is titled 'Action Properties' and has a dark blue header. The 'Name' field is set to 'Move Submission to Approved Tab'. The 'Description' field is empty. The 'Business Object' is 'Utility - Interact API'. The 'Action' is 'Move Submission to Approved Tab'. The 'Group' is 'Data Type'. The 'Inputs' tab is selected, showing a table with one input: 'Submission Id' of type 'Text' with value '[Data.\_requestId]'. The 'Outputs' and 'Conditions' tabs are empty. The 'Stage logging' is set to 'Errors only' and 'Don't log parameters on this stage' is unchecked. The 'Warning threshold' is 'System Default' and 'Number of minutes' is '5'.

Name	Data Type	Value
Submission Id	Text	[Data._requestId]

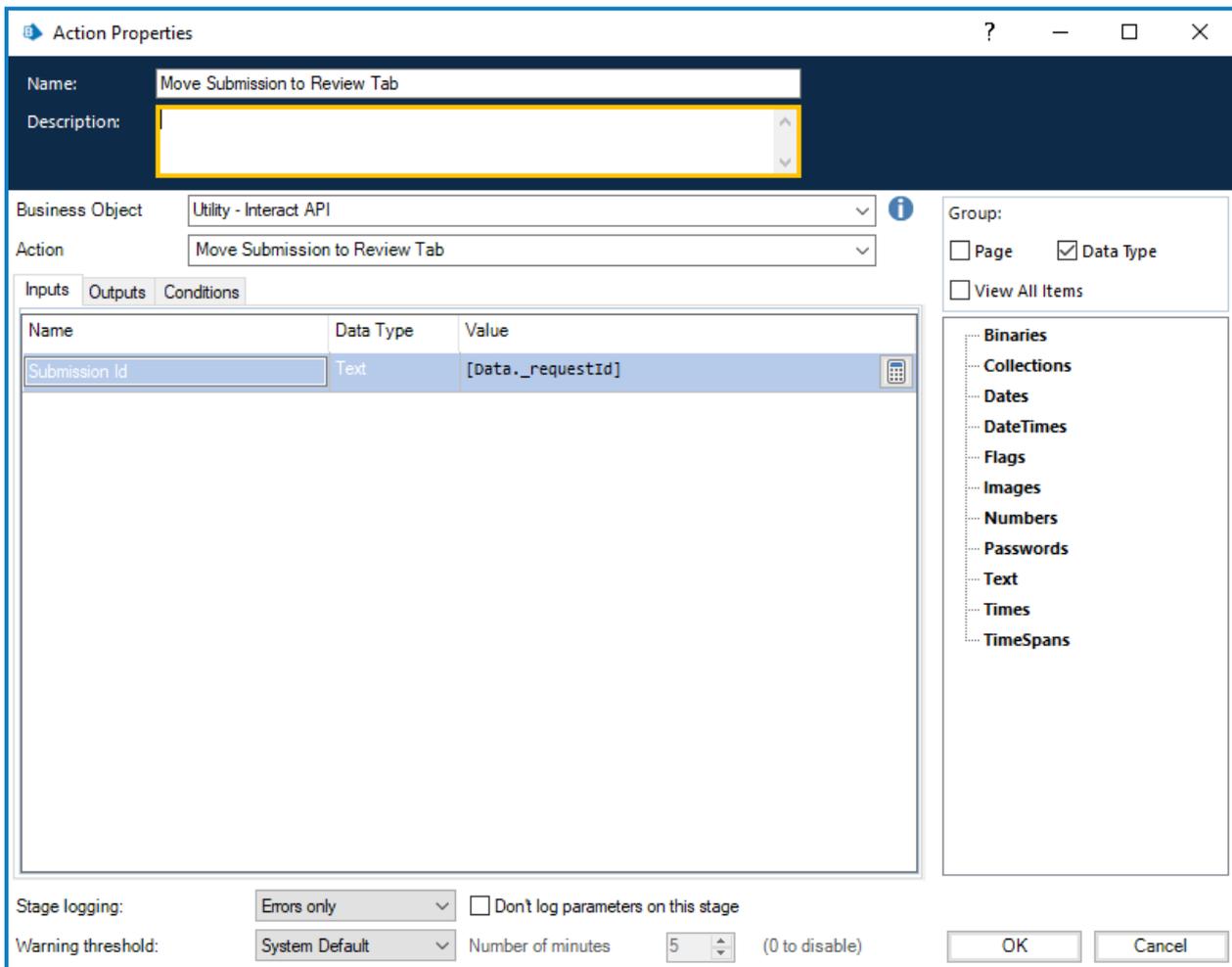
There are no outputs for the Move Submission to Approved.

## Move Submission to Review Tab

The Move Submission to Review Tab allows you to move the submission to the **Awaiting approval** folder under **History** for the submitter, and into the **Approvals** folder under **My Work** for the approver.

 This requires a form to be configured with either **Single Approver** or **Any Approver**. If not, the form will move to the **Awaiting approval** folder but will not move to the **My Work** tab of an approver.

The input for the action is the Submission ID of the submission being moved, as illustrated below.



The screenshot shows the 'Action Properties' dialog box for the 'Move Submission to Review Tab' action. The dialog is titled 'Action Properties' and has a dark blue header. The 'Name' field is set to 'Move Submission to Review Tab'. The 'Description' field is empty. The 'Business Object' is 'Utility - Interact API'. The 'Action' is 'Move Submission to Review Tab'. The 'Group' section has 'Page' unchecked and 'Data Type' checked. The 'View All Items' checkbox is also unchecked. The 'Inputs' tab is selected, showing a table with one input: 'Submission Id' of type 'Text' with value '[Data.\_requestId]'. The 'Outputs' and 'Conditions' tabs are empty. The 'Stage logging' section has 'Errors only' selected and 'Don't log parameters on this stage' unchecked. The 'Warning threshold' is 'System Default' and the 'Number of minutes' is set to 5 (0 to disable). The 'OK' and 'Cancel' buttons are at the bottom right.

Name	Data Type	Value
Submission Id	Text	[Data._requestId]

There are no outputs for the Move Submission to Review Tab.

## Move Submission to Declined Tab

The Move Submission to Declined Tab allows you to move a submission from its current folder to the **Submission history** folder under **History**. This action also updates the status of the submission to Declined. The Status filter in Interact can be used to display just the submissions with this status.

The input for the action is the Submission ID of the submission being moved, as illustrated below.

The screenshot shows the 'Action Properties' dialog box for the 'Move Submission to Declined Tab' action. The dialog is titled 'Action Properties' and has a dark blue header. Below the header, there are fields for 'Name' (Move Submission to Declined Tab) and 'Description' (empty). The 'Business Object' is set to 'Utility - Interact API' and the 'Action' is 'Move Submission to Declined Tab'. The 'Group' section has 'Page' unchecked and 'Data Type' checked. The 'View All Items' checkbox is also unchecked. The 'Inputs' tab is selected, showing a table with one input: 'Submission Id' of type 'Text' with value '[Data.\_requestId]'. The 'Outputs' and 'Conditions' tabs are empty. The 'Stage logging' section has 'Errors only' selected and 'Don't log parameters on this stage' unchecked. The 'Warning threshold' is set to 'System Default' with a value of 5 minutes.

Name	Data Type	Value
Submission Id	Text	[Data._requestId]

There are no outputs for the Move Submission to Declined Tab.

## Move Submission to Draft Tab

The Move Submission to Draft Tab allows you to move a submission from its current folder to the **Draft** folder under **My Work**.

The input for the action is the Submission ID of the submission being moved, as illustrated below.

The screenshot shows the 'Action Properties' dialog box for the 'Move Submission to Draft Tab' action. The dialog is titled 'Action Properties' and has a dark blue header. The 'Name' field is set to 'Move Submission to Draft Tab'. The 'Description' field is empty. The 'Business Object' is 'Utility - Interact API' and the 'Action' is 'Move Submission to Draft Tab'. The 'Inputs' tab is selected, showing a table with one input: 'Submission Id' of type 'Text' with the value '[Data.\_requestId]'. The 'Outputs' and 'Conditions' tabs are empty. On the right, the 'Group' section has 'Page' unchecked and 'Data Type' checked. Below this is a list of data types: Binaries, Collections, Dates, DateTimes, Flags, Images, Numbers, Passwords, Text, Times, and TimeSpans. At the bottom, 'Stage logging' is set to 'Errors only' and 'Warning threshold' is set to 'System Default' with a 'Number of minutes' of 5. 'OK' and 'Cancel' buttons are at the bottom right.

Name	Data Type	Value
Submission Id	Text	[Data._requestId]

There are no outputs for the Move Submission to Draft Tab.

## Move Submission to Inbox Tab

The Move Submission to Inbox Tab allows you to move a submission from its current folder to the **Inbox** folder under **My Work**.

The input for the action is the Submission ID of the submission being moved, as illustrated below.

The screenshot shows the 'Action Properties' dialog box for the 'Move Submission to Inbox Tab' action. The dialog is titled 'Action Properties' and has a dark blue header. The 'Name' field is set to 'Move Submission to Inbox Tab'. The 'Description' field is empty. The 'Business Object' is 'Utility - Interact API' and the 'Action' is 'Move Submission to Inbox Tab'. The 'Group' section has 'Page' unchecked and 'Data Type' checked. The 'View All Items' checkbox is also unchecked. The 'Inputs' tab is selected, showing a table with one input: 'Submission Id' of type 'Text' with the value '[Data.\_requestId]'. The 'Outputs' and 'Conditions' tabs are empty. The 'Stage logging' section has 'Errors only' selected and 'Don't log parameters on this stage' unchecked. The 'Warning threshold' is set to 'System Default' with a value of 5 minutes. The 'OK' and 'Cancel' buttons are at the bottom right.

Name	Data Type	Value
Submission Id	Text	[Data._requestId]

There are no outputs for the Move Submission to Inbox Tab.

## Move Submission to Archived Tab

The Move Submission to Archived Tab allows you to move a submission from its current folder to the Archived folder under **History**.

The input for the action is the Submission ID of the submission being moved, as illustrated below.

The screenshot shows the 'Action Properties' dialog box for the 'Move Submission to Archived Tab' action. The dialog is titled 'Action Properties' and has a dark blue header. The 'Name' field is set to 'Move Submission to Archived Tab'. The 'Description' field is empty. The 'Business Object' is 'Utility - Interact API' and the 'Action' is 'Move Submission to Archived Tab'. The 'Inputs' tab is selected, showing a table with one input: 'Submission Id' of type 'Text' with the value '[Data.\_requestId]'. The 'Outputs' and 'Conditions' tabs are empty. The 'Group' section has 'Page' unchecked and 'Data Type' checked. The 'View All Items' checkbox is also unchecked. A list of data types is shown on the right: Binaries, Collections, Dates, DateTimes, Flags, Images, Numbers, Passwords, Text, Times, and TimeSpans. At the bottom, 'Stage logging' is set to 'Errors only' and 'Don't log parameters on this stage' is unchecked. The 'Warning threshold' is 'System Default' and the 'Number of minutes' is set to 5.

Name	Data Type	Value
Submission Id	Text	[Data._requestId]

There are no outputs for the Move Submission to Archived Tab.

## Human/digital worker collaboration

To best show the way that human/digital worker collaboration operates let us consider a simple scenario. This scenario is a simple change of address request from a user.

The steps on a high-level are detailed below:

- The user will initiate the request through Interact;
- The digital worker will review the request and move the item to the user's inbox;
- The user will then review and update the request, providing their response to a security question, and submit the request;

 At this point the user can leave the Form open in the inbox, see the Form being updated dynamically by the digital worker and provide additional information when requested.

- The digital worker processes the information provided and then requests additional information from the user, the new address;
- The user provides the information and the digital worker updates the record and then completes the task.

The Form created in Interact is displayed below.

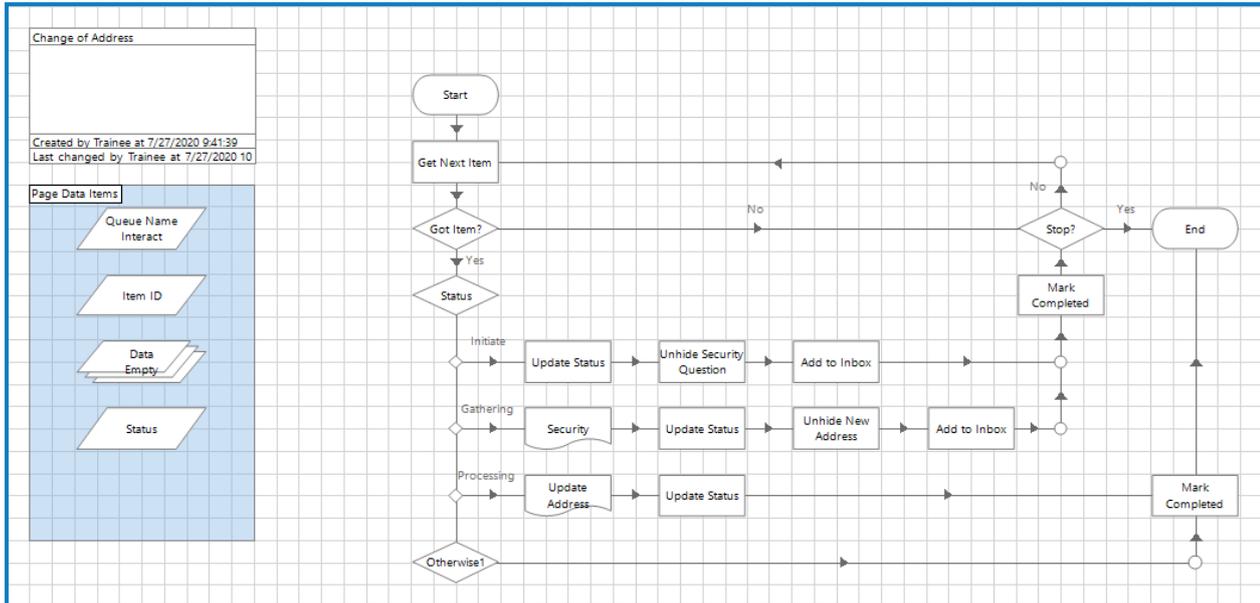
The Form captures initialing the First and Last Names, as well as the current address. The other fields, as you can see in the Play Test Area, are hidden. These are the security question and the new address fields.



The screenshot shows the Interact form editor interface for a form titled "HR - Change of Address". The form is displayed in a vertical flow layout. The top section is labeled "Initial Details" with the subtitle "Initiating request for Change of Address". Below this are three sections: "Instructions" (Read only), "Status" (Read only), and "First Name" (Mandatory). The "First Name" field is currently empty and has a "Mandatory" label. The form is surrounded by a blue border, and the top right corner contains navigation icons for "Editor", "Play test", "Create page", "Collapse all", "Expand all", "Pages", "Fields", "Rules", and "Save".

The automation associated with the process is illustrated below. This is a process that loops through and allows to the user to work dynamically with the digital worker.

The process operates dependent on the value of the 'Status' flag. This is a read-only radio group across the top of the Form. As the process progresses through the different stages this flag is updated and the Status changes in the Form.



## Interact Web API Service object restrictions

The following restrictions apply to the Interact Web API Service object.

The table below lists the restrictions.

Function	Element Type	Applicable Restrictions
Get Submission	Number	Number elements are delivered as 'Text' format when using the Get Submission Web API Service. Cast to Number once inside Blue Prism
Get Submission	Date	Retrieving a Date element using the Get Submission Web API Service, returns the Date in 'DateTime' format. It is recommended that DateTime values are converted to Text once inside Blue Prism. The use of Text is recommended to support the different Date formats within Interact.
IADA	Number	Number elements are delivered as 'Text' format into a Blue Prism queue. Cast to Number once inside Blue Prism
Create Submission	Number	When creating a new submission using the Web API Service the submission will fail if a Number element is left blank. Submit Number element fields with a number set
Create Submission	Radio Group	When creating a new submission using the Web API Service the submission will fail if a Radio Group element is left blank. Submit Radio Group element fields with a value set
Create Submission	Upload	When creating a new submission using the Web API Service you cannot send anything back when using the Upload element. This is the expected functionality.
Edit Submission	Upload	When editing a submission using the Web API Service you cannot send anything back when using the Upload element. This is the expected functionality.
Interact Form	Priority and SLA	Though the Priority and SLA can be set in the Interact Form creator, they have no significance in this release as IADA 'Get Next Prioritized Item' (GNPI) function is not available in an on-premise configuration. Expected functionality for this release.
Interact Forms	Rules	Rules are not enforced when updating Interact Forms from Blue Prism. For example, a rule that reveals a hidden field when a flag is set, will not be initiated if the flag is set correctly when updated. If the field is not read-only a user can click in the field and then click elsewhere in the Form to see the Rule applied.   You can use Update Config to replace some rules.
Interact Forms	Verification	Regex and other verification such as character string lengths are not enforced when updating Interact Forms from Blue Prism. If the field is not read-only a user can click in the field and then click elsewhere in the Form to see the Rule applied. No workaround available.