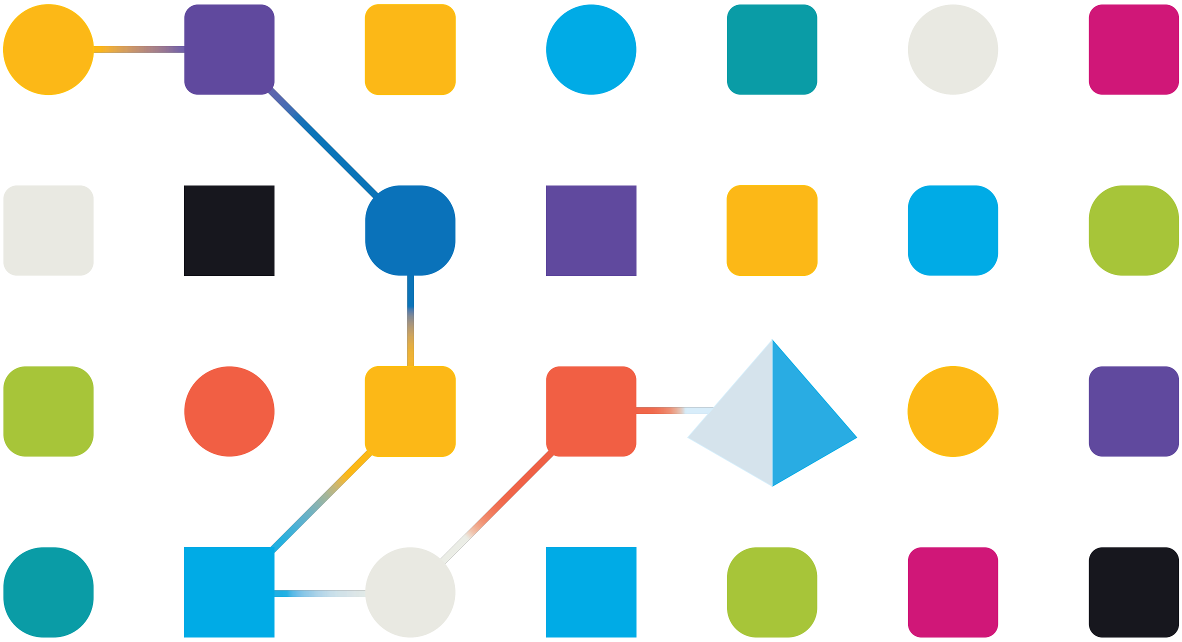
Interact Web API Service 5.0

User Guide

Document Revision: 3.0



Trademarks and Copyright

The information contained in this document is the proprietary and confidential information of Blue Prism Limited 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 Limited.

Blue Prism is not responsible for the content of external websites referenced by this document.

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

Contents

Interact Web API Service

This user guide covers SS&C | Blue Prism® Interact Web Application Programming Interface (API) Service.

SS&C | 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 the Blue Prism digital workforce and has experience with related components such as SS&C | Blue Prism® 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 end users to interact with their digital workforce within a business process. Initiate, verify, receive and authorize varied work related to your business processes. 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:

- The Interact plugin within SS&C | Blue Prism® Hub – Enables a developer utilizing Hub and the Forms plugin to create and publish Forms for an end user to use.
- The Interact web-based application – Provides the user interface where end users 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 back to the Interact user interface after processing.

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.

Version

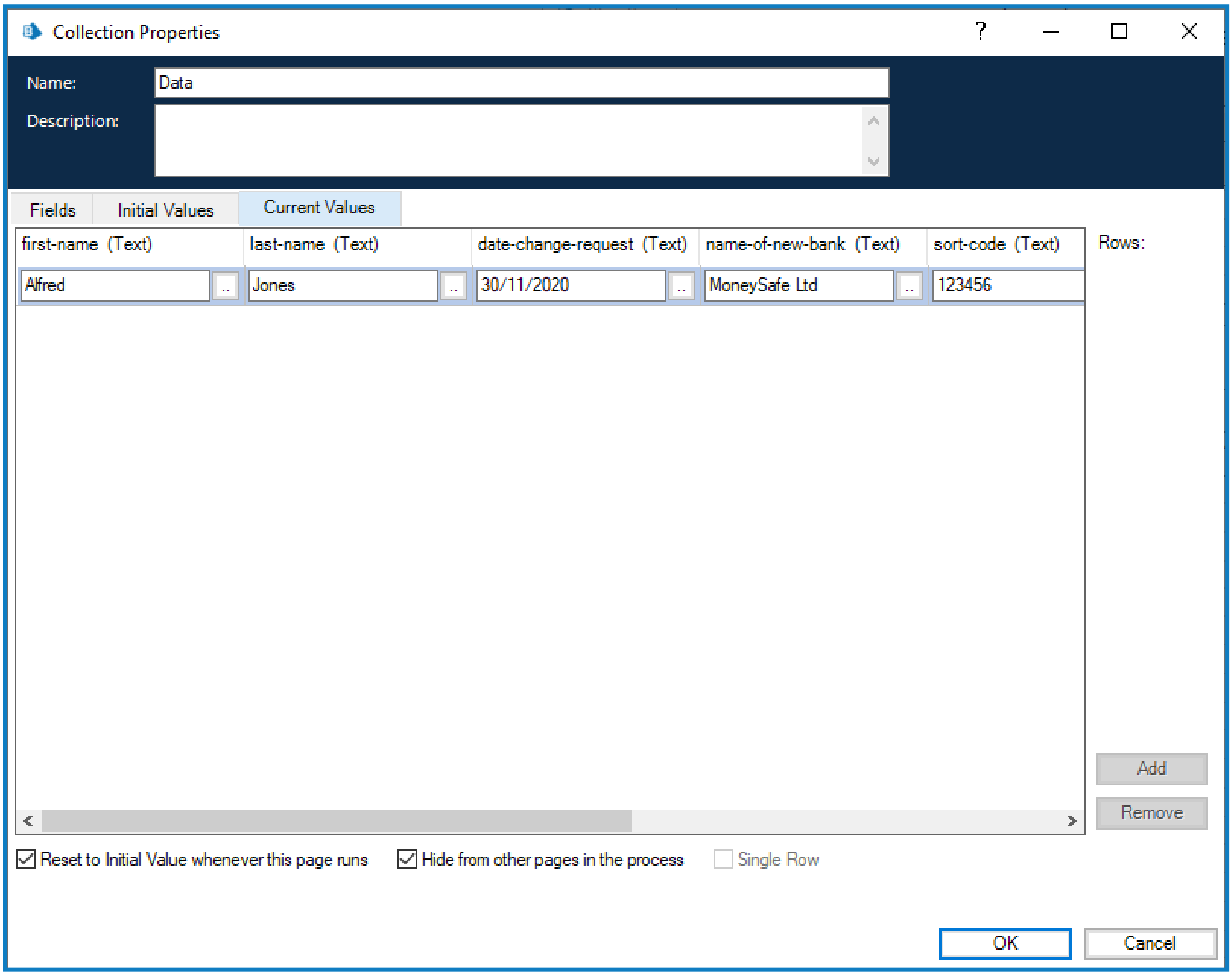
Interact 5.0 uses version 2 of the Interact Remote API, which was originally introduced in Interact 4.6.

Submission

Submitting a Form is detailed in the Interact user guide. All submissions are processed and 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.



Name: Data

Description:

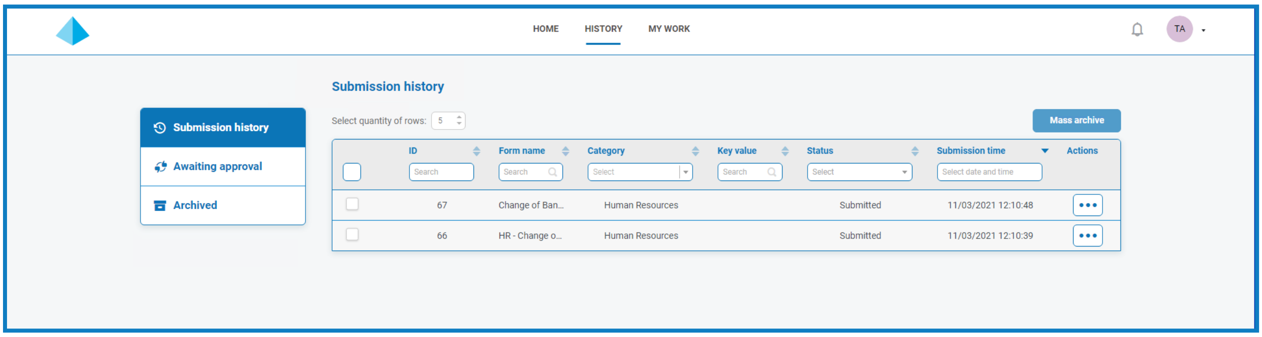
first-name (Text)	last-name (Text)	date-change-request (Text)	name-of-new-bank (Text)	sort-code (Text)
Alfred	Jones	30/11/2020	MoneySafe Ltd	123456

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

It is important to note that two pieces of information are added to the collection. These appear as fields at the end of the Current Values tab in the Collection Properties:

- **_requestId** data item – This is the Submission ID of the form submitted by the user.
- **_submitterDetails** collection – This displays as a link with more information relating to the user who submitted the form.

The _requestId data item (the Submission ID) is the number shown in the Submission History page as illustrated below.



Submission history

Select quantity of rows: 5

□	ID	Form name	Category	Key value	Status	Submission time	Actions
	Search	Search	Select	Search	Select	Select date and time	
□	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.

Configure Blue Prism to use Interact

Blue Prism and Interact communicate through the Blue Prism Interact Remote API. To use this API, the Interact API Service release file should be imported into Blue Prism, this includes a Web API Service and VBO. 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 import the latest release file into Blue Prism. The Interact API Service release file is available on the Blue Prism Portal.

To configure Blue Prism to use Interact, you need to:

1. Set up a service account in Hub and generate a secret key.
2. Import the Interact API Service VBO into Blue Prism.
3. Set up the credentials for the Interact Web API service account in Blue Prism.
4. Configure the Interact API Service to enable Blue Prism to communicate with Interact.

Set up a service account

To set up the Interact Remote API credentials in Blue Prism, a secret key is required. This is generated from the associated service account in Hub for use with the Interact Remote API. If you lose the key, you can regenerate another key from the service account. For more information, see Service accounts.

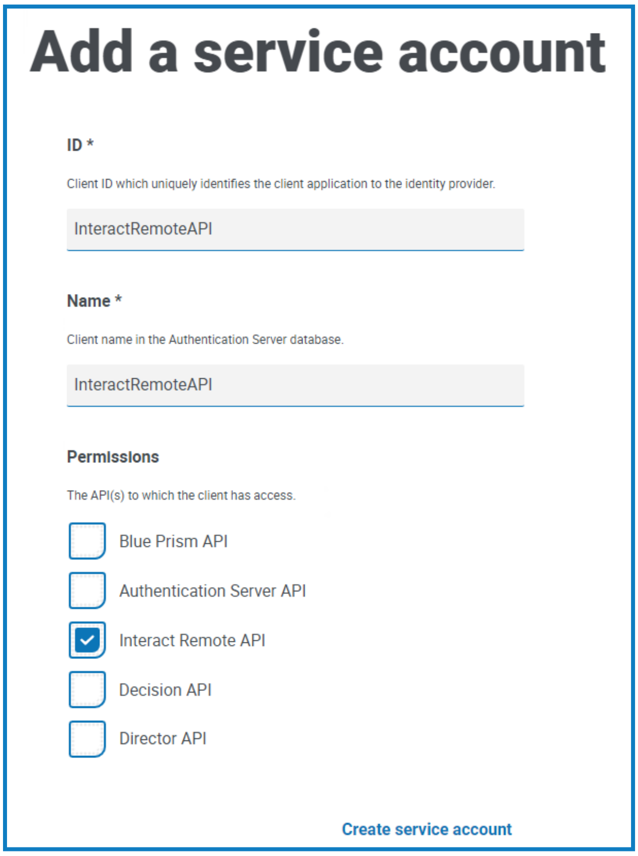
> Service accounts can only be created by a Hub administrator.

If you do not have a service account set up for the Interact Remote API, you can create a service account:

1. In Blue Prism Hub, on the Service accounts page, click **Add account**.
2. Enter a unique ID and a friendly name, for example, *InteractRemoteAPI*.

   > Do not use *InteractRemoteClient*. This name is allocated internally in the system.

3. Under **Permissions**, select **Interact Remote API**.



Add a service account

ID *

Client ID which uniquely identifies the client application to the identity provider.

InteractRemoteAPI

Name *

Client name in the Authentication Server database.

InteractRemoteAPI

Permissions

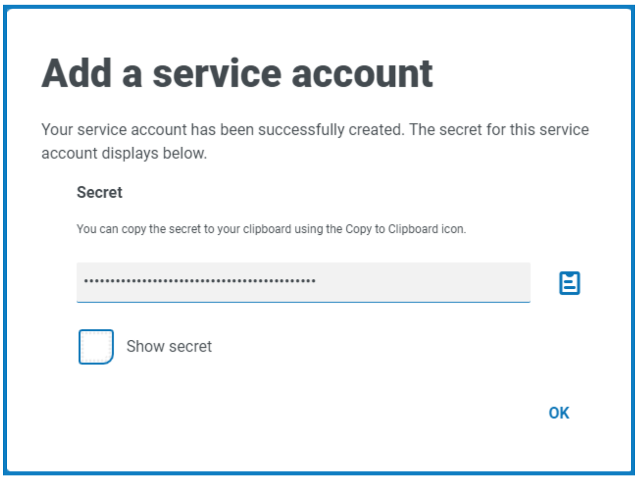
The API(s) to which the client has access.

□ Blue Prism API
□ Authentication Server API
☑ Interact Remote API
□ Decision API
□ Director API

4. Click **Create service account**.

   The Add a service account dialog displays with a generated secret key. You will need to enter this key into the Blue Prism interactive client when configuring the associated credential.

5. Copy the generated secret key to your clipboard ready to paste into the Blue Prism interactive client.



Add a service account

Your service account has been successfully created. The secret for this service account displays below.

Secret

You can copy the secret to your clipboard using the Copy to Clipboard icon.

••

□ Show secret

6. Click **OK** to close the dialog.

   The Service accounts page displays with the newly created account shown.

Import the VBO

1. Download the Interact API Service release file from the Blue Prism Portal.
2. In Blue Prism, select **File** and click **Import** > **Release / Skill** and follow the prompts to import the release file into Blue Prism. For more information, see Import a file.

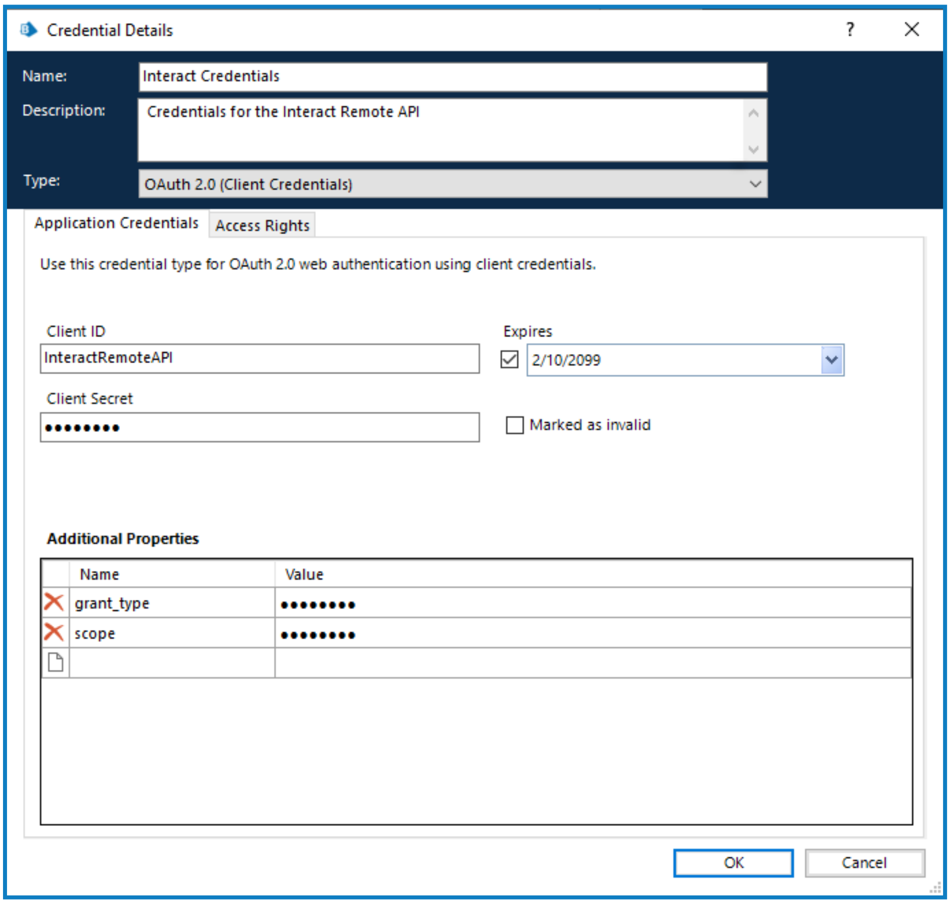
Set up credentials in Blue Prism

1. Log into the Blue Prism interactive client, select **System** and then click **Security** > **Credentials**. See Security > Credentials for additional information.
2. Click **New**.

   The Credential Details dialog displays.
3. On the Application Credentials tab of the Credential Details dialog:
   a. Enter a name.
   b. Change the **Type** to **OAuth 2.0 (Client Credentials)**.
   c. In **Client ID**, enter the ID that you used to create the service account above in Configure Blue Prism to use Interact on page 7, for example, *InteractRemoteAPI*.

   > The client ID is case sensitive. You must ensure you enter the client ID in the same case as defined in the service account.

   d. In **Client Secret**, enter the secret key that was generated for the service account.



Name: Interact Credentials

Description: Credentials for the Interact Remote API

Type: OAuth 2.0 (Client Credentials)

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

Client ID: InteractRemoteAPI

Expires: ☑ 2/10/2099

Client Secret: ••••••••

□ Marked as invalid

Additional Properties

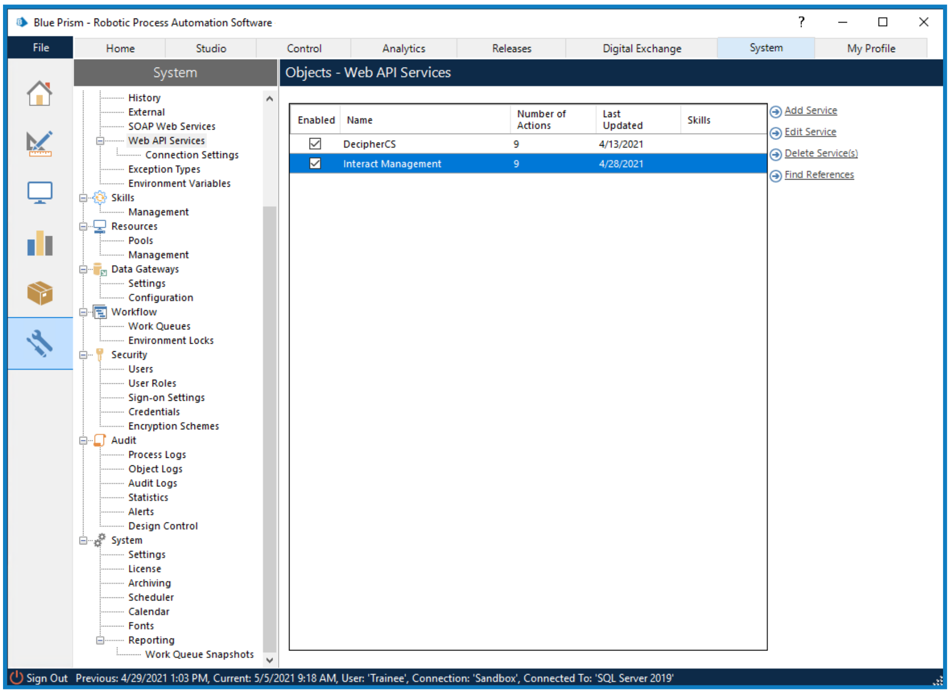
Name	Value
grant_type	••••••••
scope	••••••••

4. On the Access Rights tab of the Credential Details dialog, set up the required access permissions.
5. Click **OK**.

Configure the Interact API Service

1. In Blue Prism, select **System** and then click **Objects** > **Web API Services**.

   The Objects - Web API Services screen displays. For example:

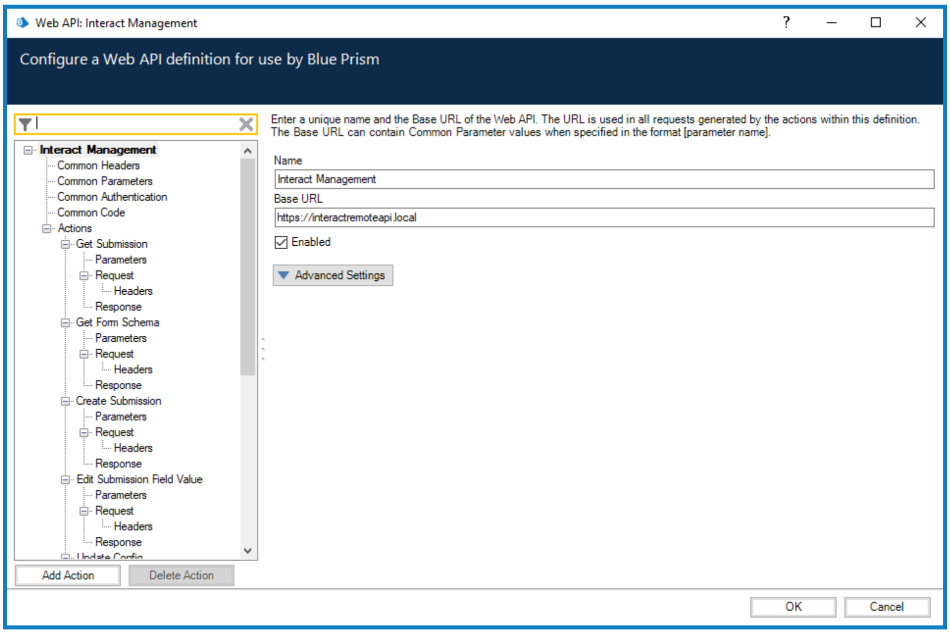


Enabled	Name	Number of Actions	Last Updated	Skills
☑	DecipherCS	9	4/13/2021	
☑	Interact Management	9	4/28/2021	

Sign Out Previous: 4/29/2021 1:03 PM, Current: 5/5/2021 9:18 AM, User: 'Trainee', Connection: 'Sandbox', Connected To: 'SQL Server 2019'

2. Select **Interact Management** and click **Edit Service**.

   The Web API: Interact Management screen displays.



Configure a Web API definition for use by Blue Prism

Enter a unique name and the Base URL of the Web API. The URL is used in all requests generated by the actions within this definition. The Base URL can contain Common Parameter values when specified in the format [parameter name].

Name: Interact Management

Base URL: https://interactremoteapi.local

☑ Enabled

3. On the Web API: Interact Management opening screen, in **Base URL**, enter the URL for your organization's Interact API service. This was defined during the installation of Interact.

4. Select **Common Authentication** in the navigation tree, then complete the following:
   a. Ensure that **Authentication Type** is set to **OAuth 2.0 (Client Credentials)**
   b. In **Authorization URI**, enter the Authentication Server URL in the format:

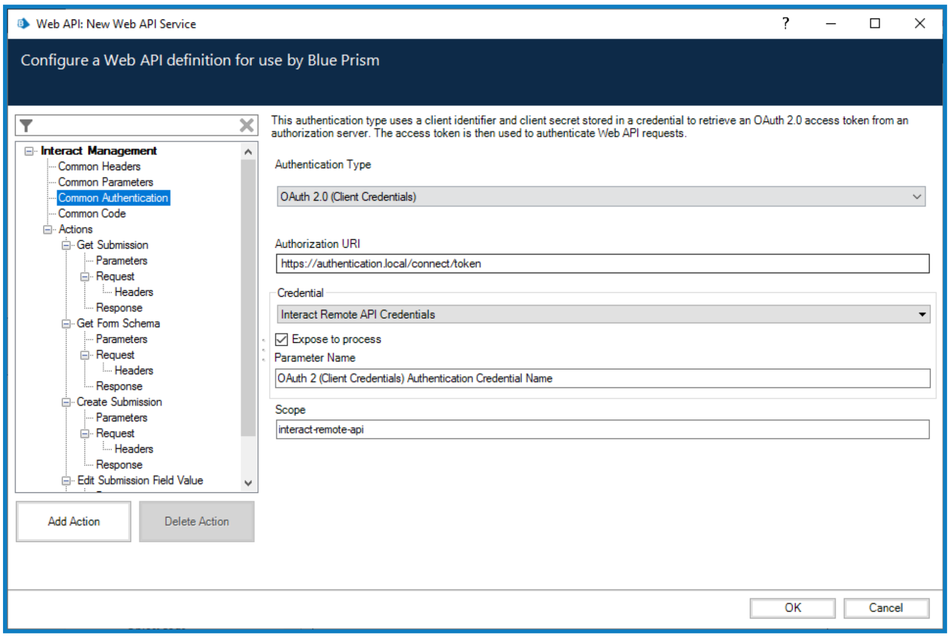
      `<Authentication Server URL>:<port if specified during install>/connect/token`

      For example, `https://authentication.blueprism.com:5000/connect/token`

      Or, if the default port was used, `https://authentication.blueprism.com/connect/token`.

      > If you have upgraded from a version earlier than 4.3, your system will still be using IMS. In this case, you should enter the information in the format:
      >
      > `<IMS URL>:<port if specified>/connect/token`
      >
      > For example, `https://ims.blueprism.com:5000/connect/token`.

   c. In **Credential**, select the credential you created in Set up credentials in Blue Prism on page 9.



Web API: New Web API Service

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://authentication.local/connect/token

Credential: Interact Remote API Credentials

☑ Expose to process

Parameter Name: OAuth 2 (Client Credentials) Authentication Credential Name

Scope: interact-remote-api

5. Click **OK** to save and complete the setup of the Web API Service.

Overview of the web service actions

The web service has 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	Edit Horizontal Rule Padding Edit <Type> Field Value, where the type is: • Checkbox Single • Checkbox Multiple • Date • Dropdown Single • Dropdown Multiple • Number • Paragraph • Radio • Table • Text • Text Area • Time • Upload
Update Config	Change Field State to Optional Change Field State to Mandatory Change Field State to Read Only Change Field State to Hidden Change Page State to Visible Change Page State to Hidden
Update Status	Move Submission to History Tab Move Submission to Approved Tab Move Submission to Review Tab Move Submission to Declined Tab Move Submission to Draft Tab Move Submission to Inbox Tab Move Submission to Archived Tab
Raise Submission	Raise Submission
Get Users by Form Name	Get Users by Form Name

Interact Web API	Blue Prism VBO: Utility - Interact API
Upload File	Upload File
Get Role ID by Role Name	Get Role ID by Role Name
Get Roles by Form Name	Get Roles by Form Name
Get Submission By Transaction	Get Submission By Transaction ID – Used internally by Create Submission, Raise Submission and Raise Submission to Role to return submission ID back to the digital workers.
Raise Submission to Role	Raise Submission to Role
Edit Submission Field Value by Transaction ID	Get Transaction ID – Used internally by Edit Horizontal Rule Padding and Edit Paragraph Field Value.

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

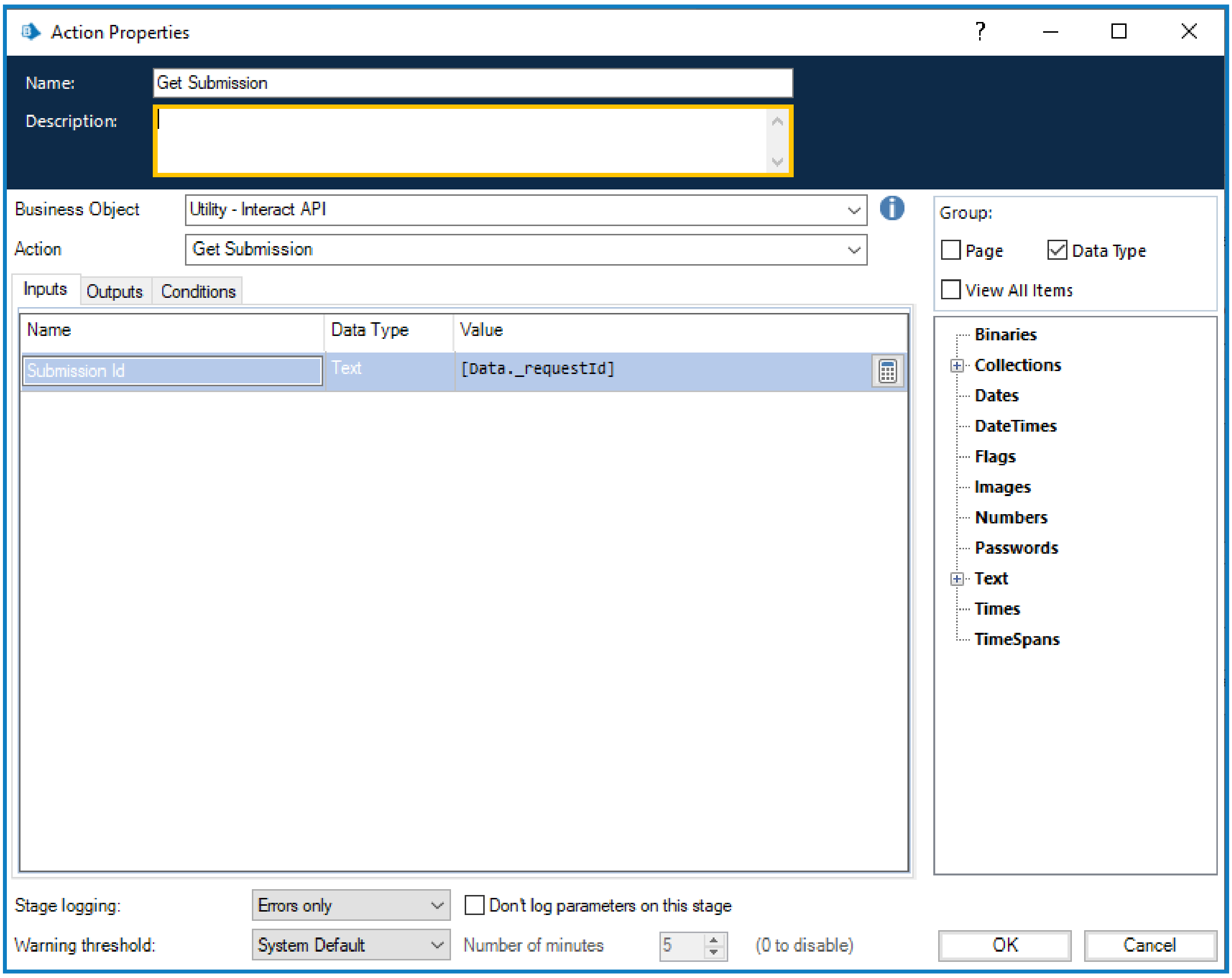
There are a number of actions within the Interact Web API Service, which are called using the Blue Prism Object (Utility - Interact API). Going through each of the actions in turn, the sections below explain how to use them in your automations.

> 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 action does a very similar action to 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.



Action Properties

Name: Get Submission

Description:

Business Object: Utility - Interact API

Action: Get Submission

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

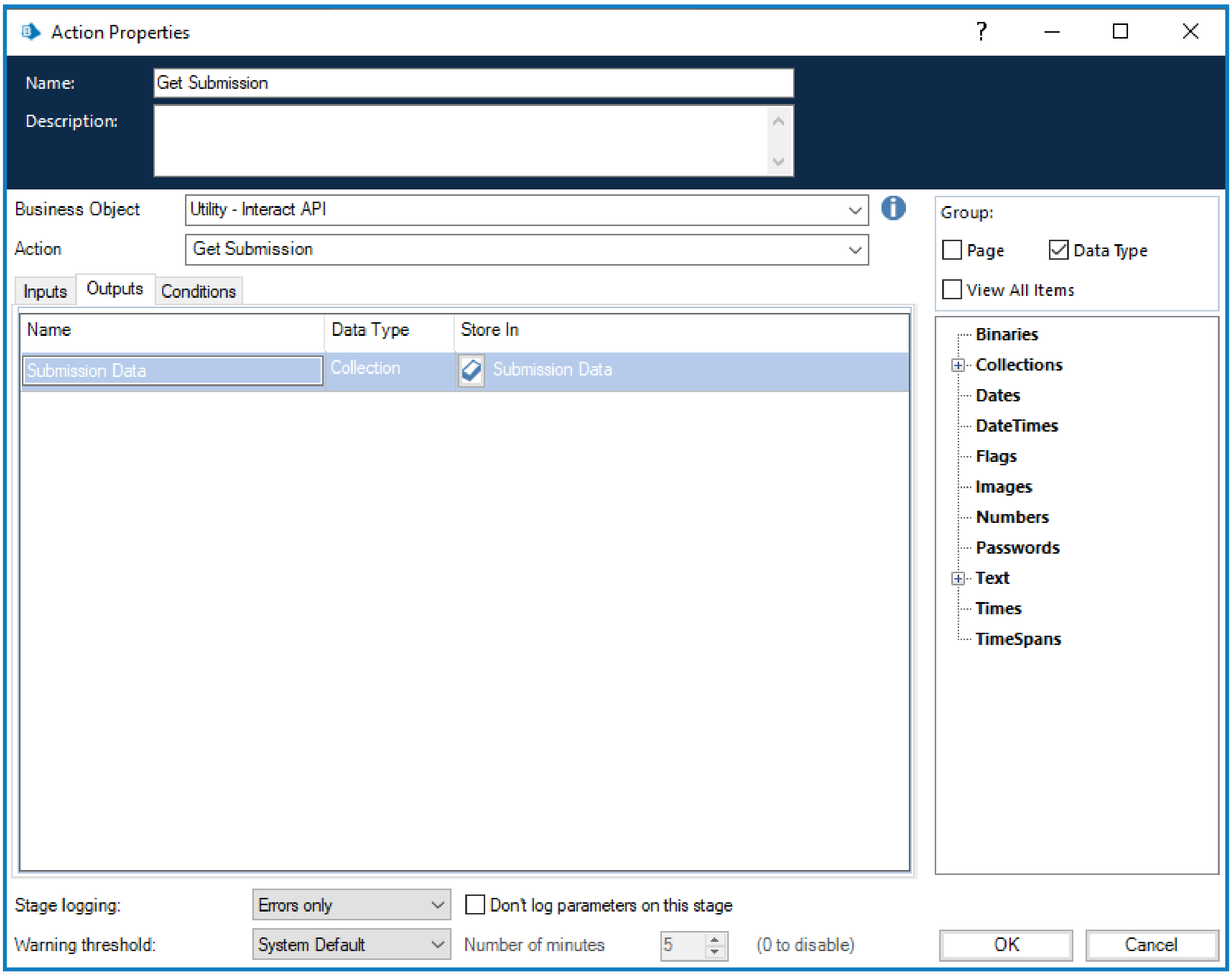
Stage logging: Errors only

□ Don't log parameters on this stage

Warning threshold: System Default

Number of minutes: 5 (0 to disable)

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



Action Properties

Name: Get Submission

Description:

Business Object: Utility - Interact API

Action: Get Submission

Name	Data Type	Store In
Submission Data	Collection	Submission Data

Group: □ Page ☑ Data Type

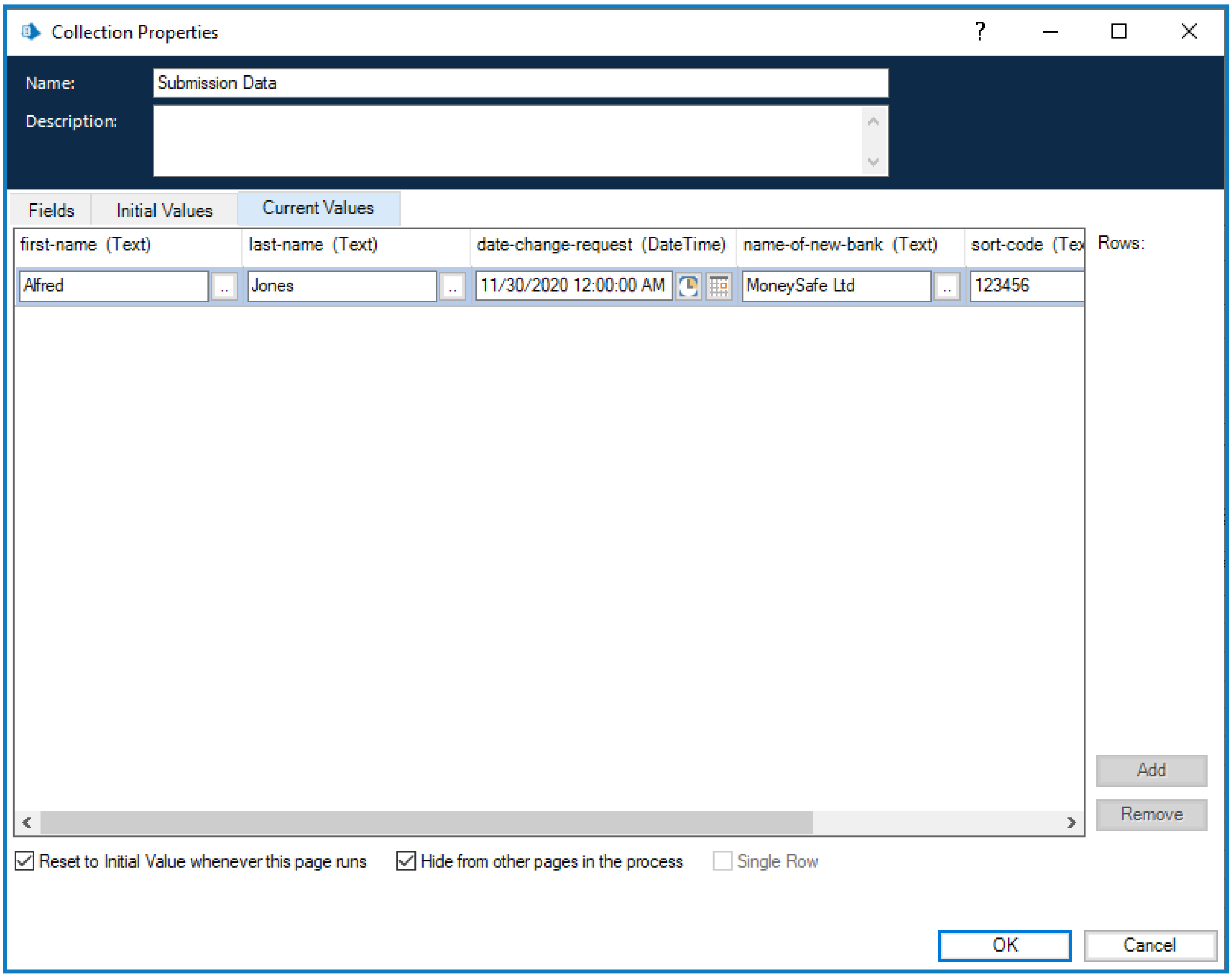
□ View All Items

Stage logging: Errors only

□ Don't log parameters on this stage

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

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.



Name: Submission Data

Description:

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

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

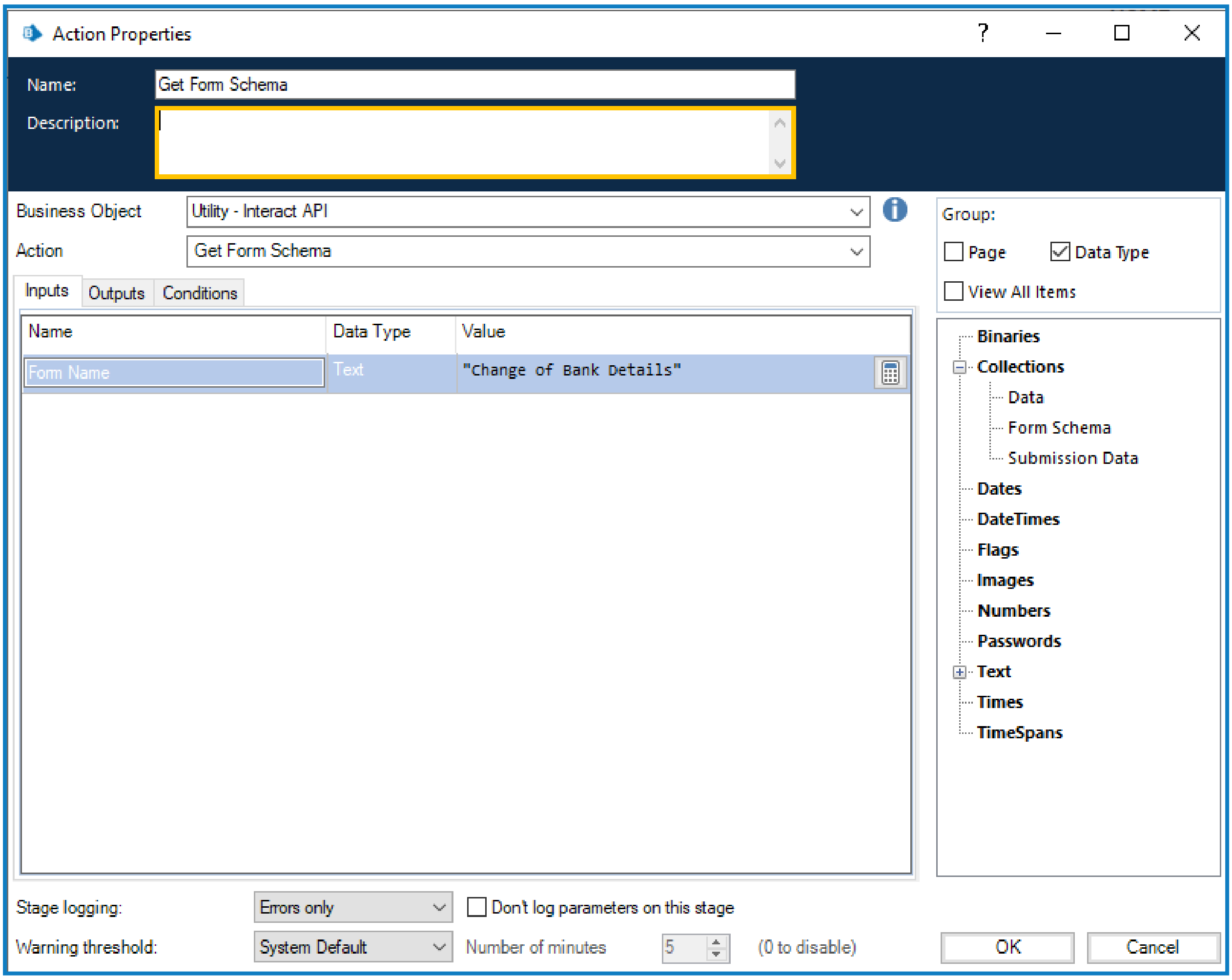
For a complete list of restrictions, see Interact Web API Service object restrictions on page 63.

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.



Name: Get Form Schema

Description:

Business Object: Utility - Interact API

Action: Get Form Schema

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

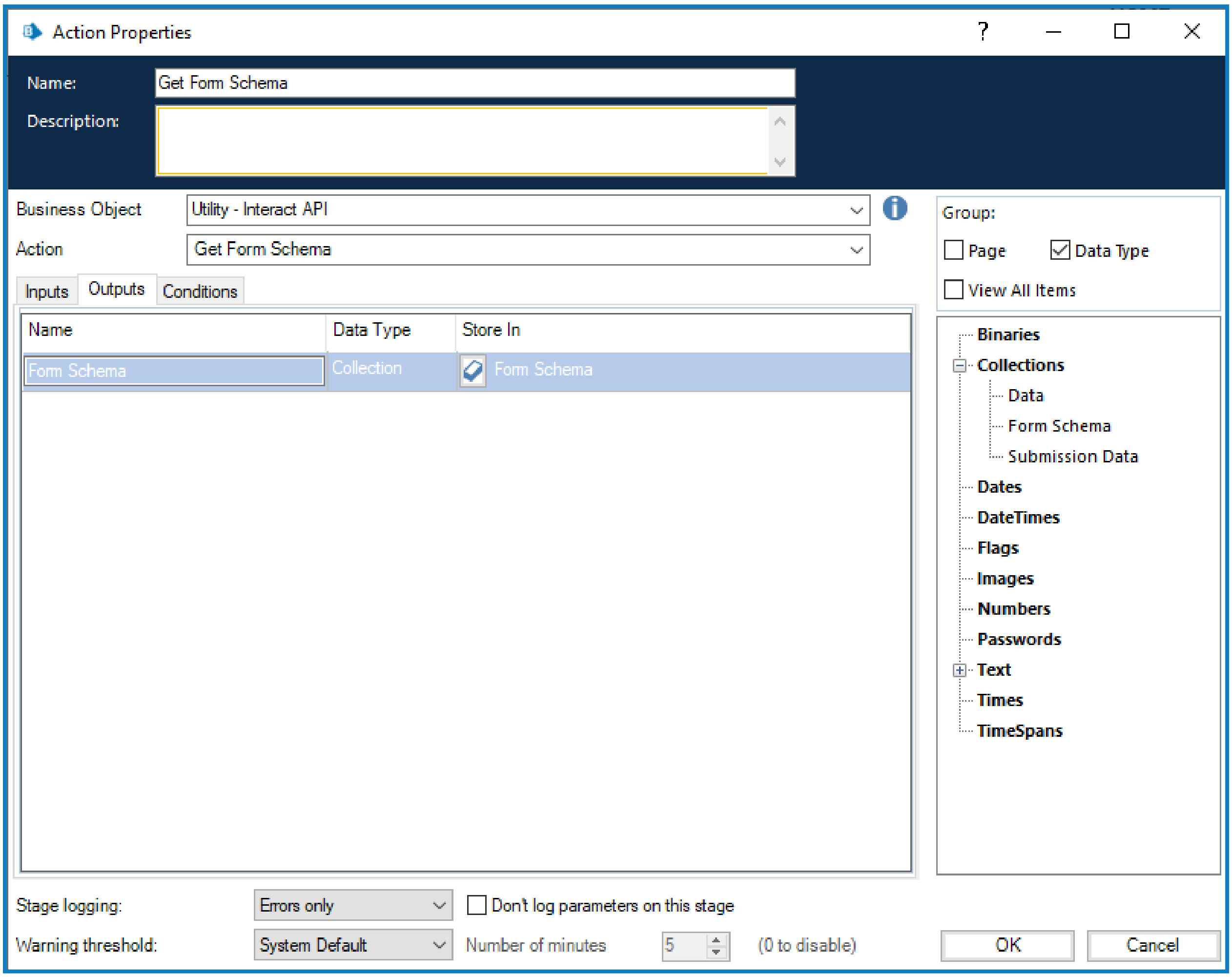
Group: □ Page ☑ Data Type

□ View All Items

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

Warning threshold: System Default Number of minutes 5 (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.



Name: Get Form Schema

Description:

Business Object: Utility - Interact API

Action: Get Form Schema

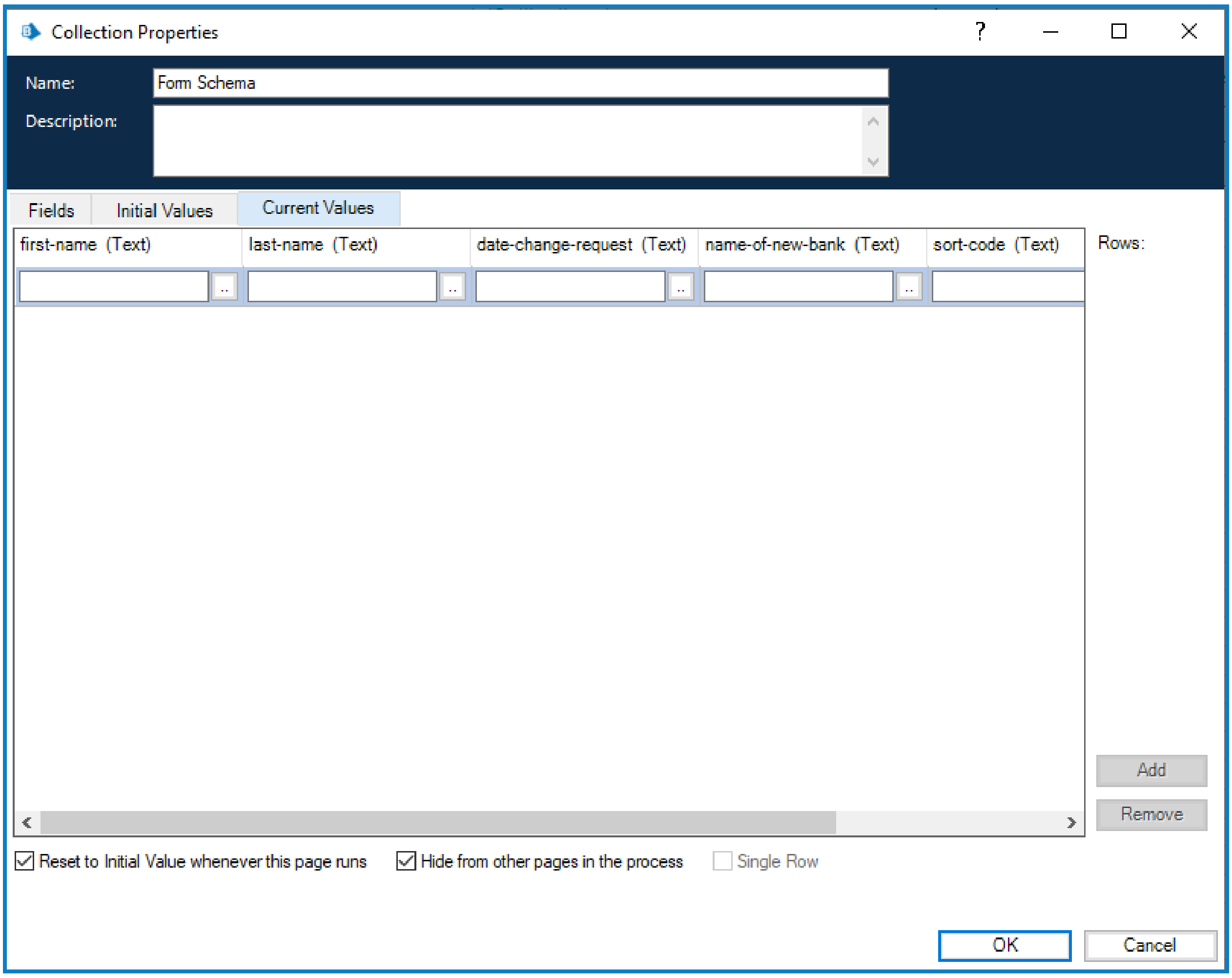
Name	Data Type	Store In
Form Schema	Collection	Form Schema

Group:
□ Page ☑ Data Type
□ View All Items

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

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

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



Name: Form Schema

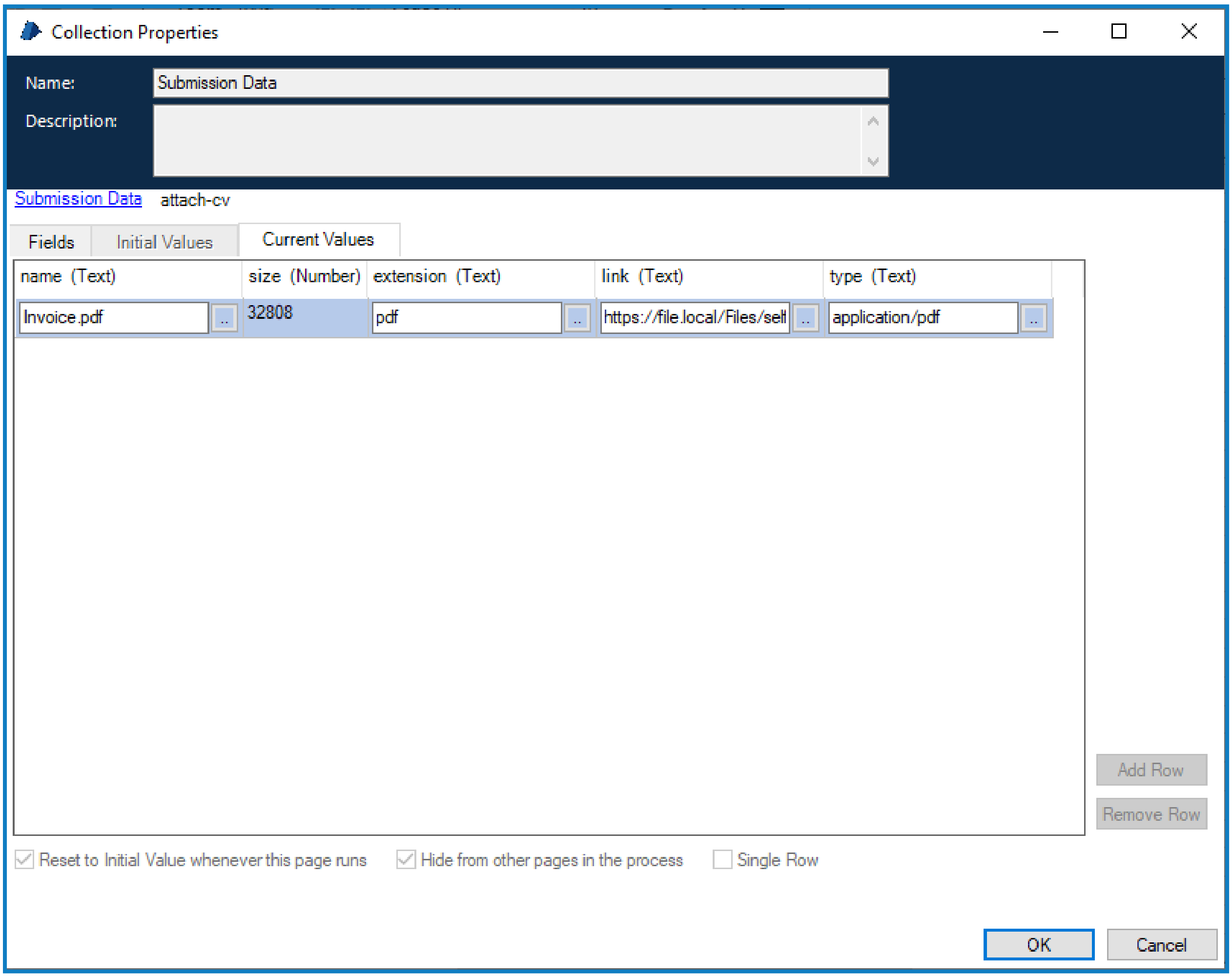
Description:

first-name (Text)	last-name (Text)	date-change-request (Text)	name-of-new-bank (Text)	sort-code (Text)

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

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.



Name: Submission Data

Description:

Submission Data attach-cv

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

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

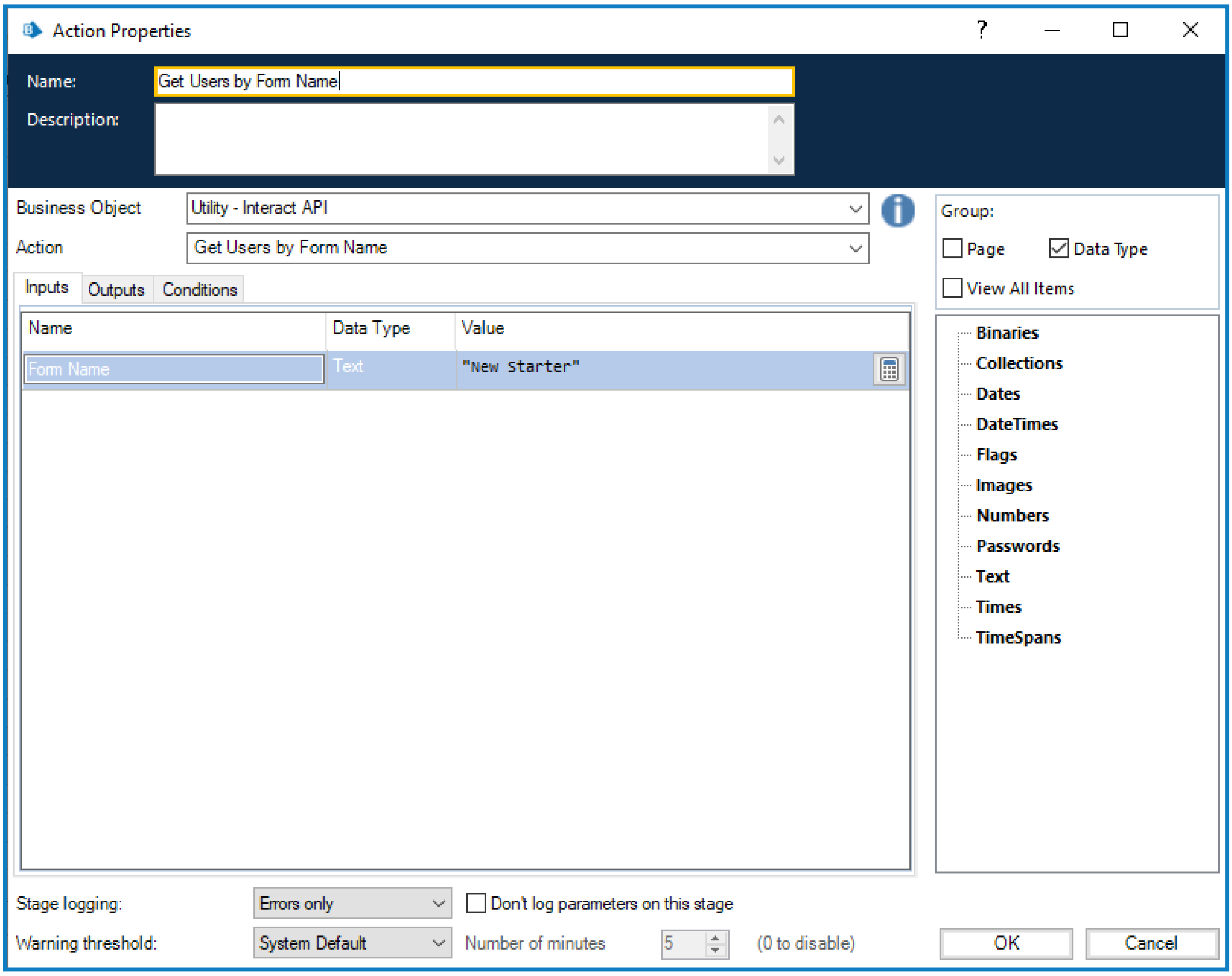
The format of the collection containing the details of the upload file must be set correctly, as illustrated above:

- **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’.

Get Users by Form Name

The Get Users by Form Name action 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: Get Users by Form Name

Description:

Business Object: Utility - Interact API

Action: Get Users by Form Name

Name	Data Type	Value
Form Name	Text	"New Starter"

Group:

□ Page ☑ Data Type

□ View All Items

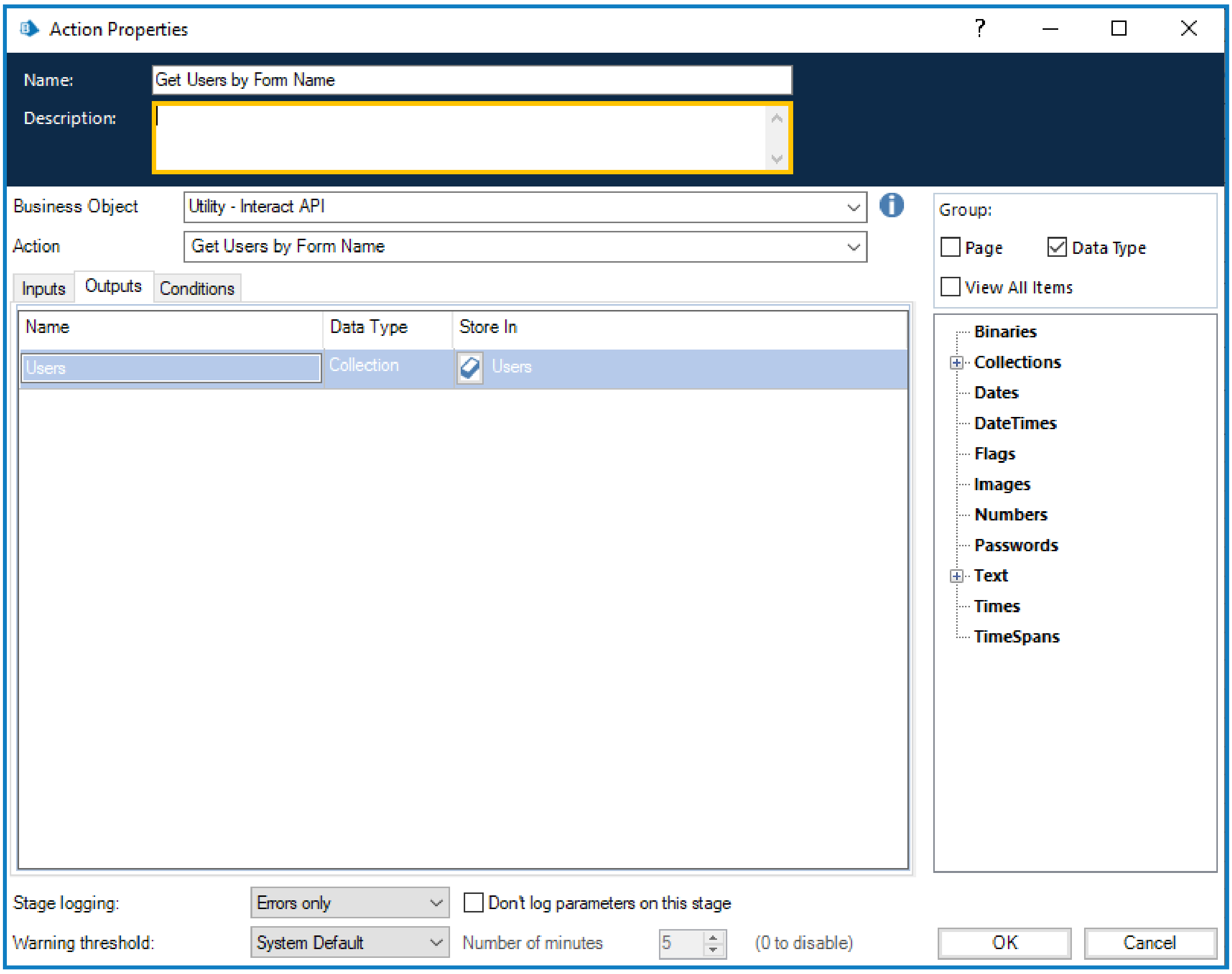
Stage logging: Errors only

□ Don't log parameters on this stage

Warning threshold: System Default

Number of minutes 5 (0 to disable)

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



Action Properties

Name: Get Users by Form Name

Description:

Business Object: Utility - Interact API

Action: Get Users by Form Name

Name	Data Type	Store In
Users	Collection	Users

Group:
☐ Page ☑ Data Type
☐ View All Items

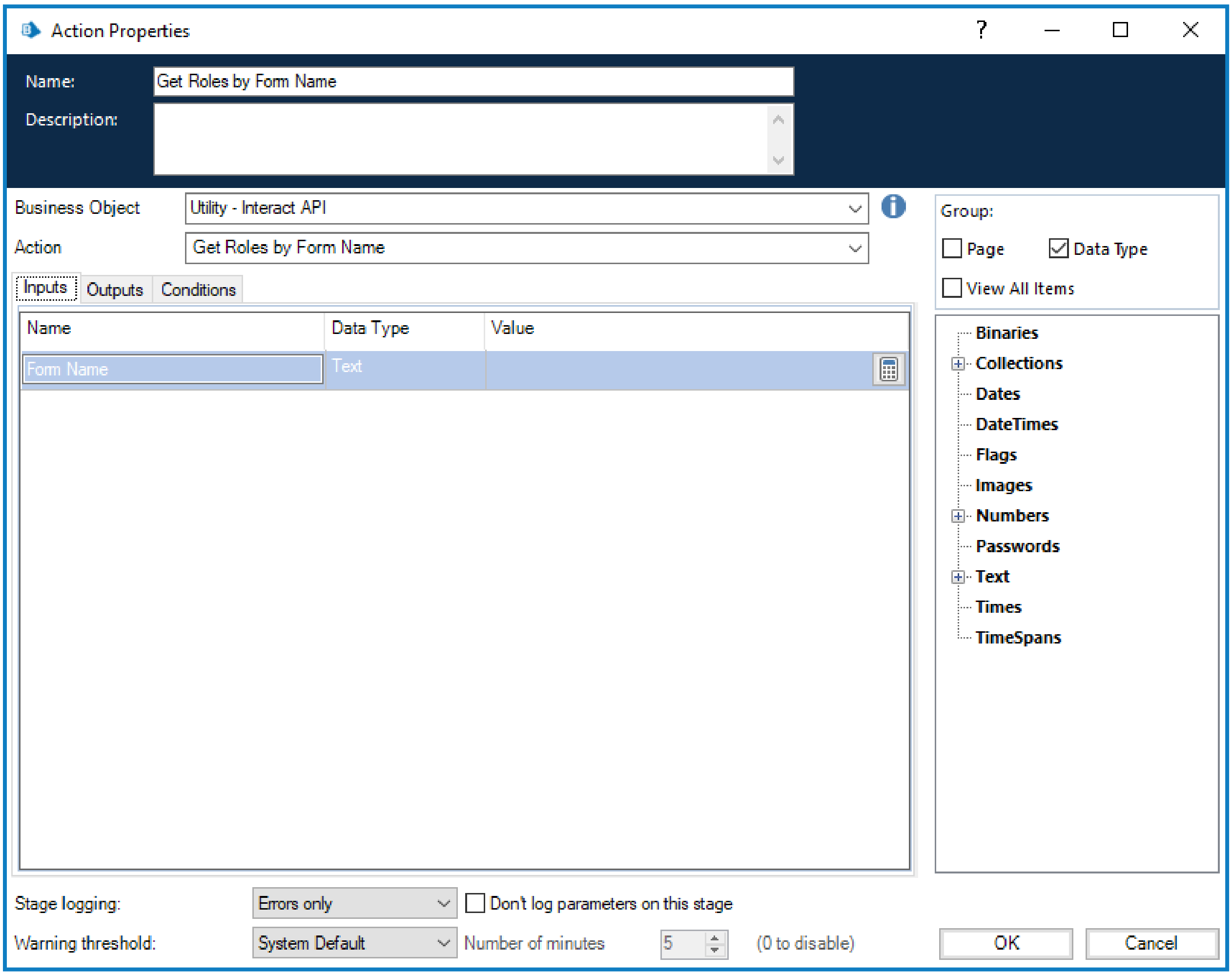
Stage logging: Errors only ☐ Don't log parameters on this stage

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

Get Roles by Form Name

The Get Roles by Form Name action allows you to collect a list of roles associated with a particular Form Name.

The input for the action is the Form Name, as illustrated below.



Name: Get Roles by Form Name

Description:

Business Object: Utility - Interact API

Action: Get Roles by Form Name

Name	Data Type	Value
Form Name	Text	

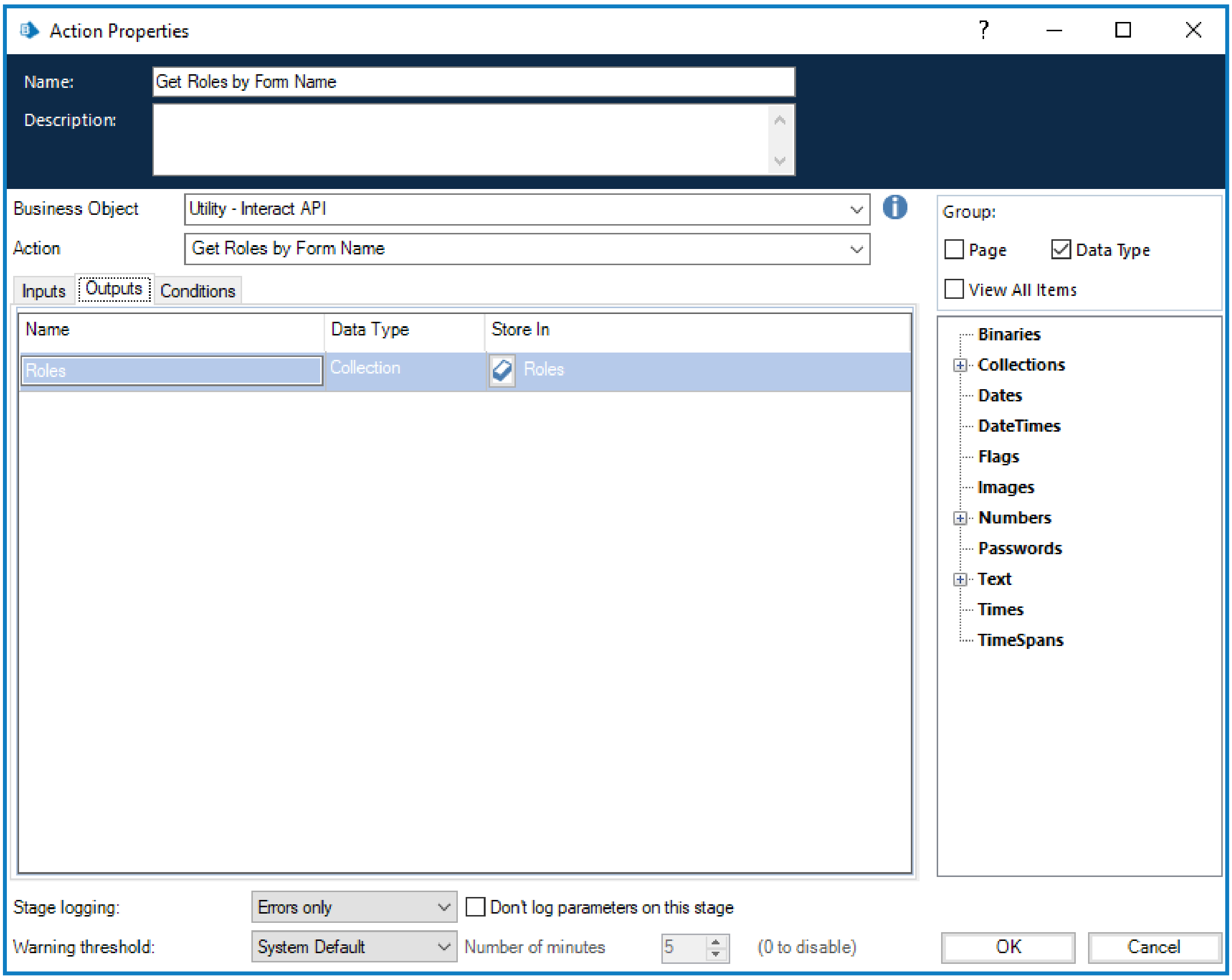
Group: □ Page ☑ Data Type

□ View All Items

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

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

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



Action Properties

Name: Get Roles by Form Name

Description:

Business Object: Utility - Interact API

Action: Get Roles by Form Name

Name	Data Type	Store In
Roles	Collection	Roles

Group: □ Page ☑ Data Type

□ View All Items

Stage logging: Errors only

□ Don't log parameters on this stage

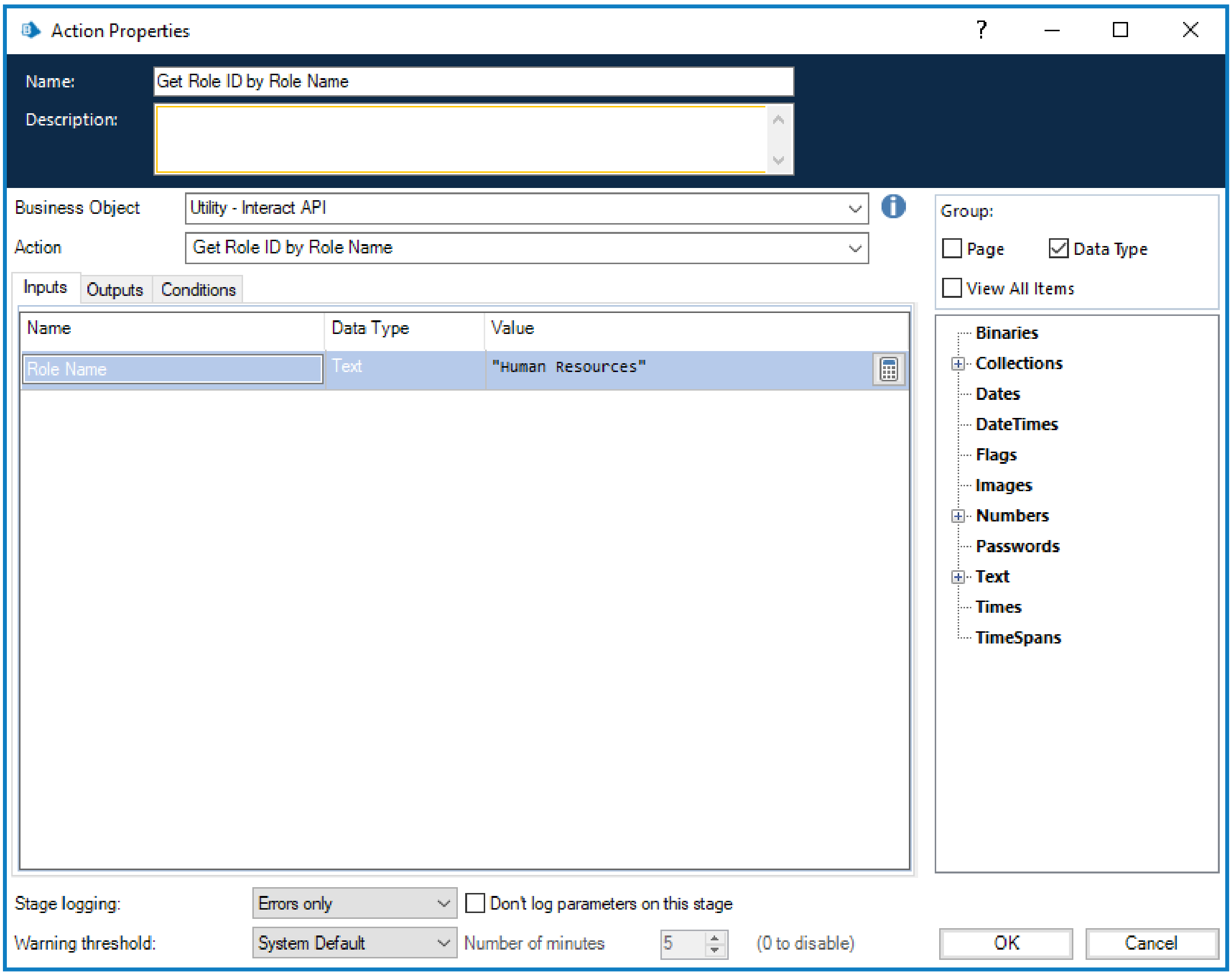
Warning threshold: System Default

Number of minutes 5 (0 to disable)

Get Role ID by Role Name

The Get Role ID by Role Name action allows you to retrieve the role identification number for a particular Role Name (which can be obtained by running Get Roles by Form Name).

The input for the action is the Role Name, as illustrated below.



Action Properties

Name: Get Role ID by Role Name

Description:

Business Object: Utility - Interact API

Action: Get Role ID by Role Name

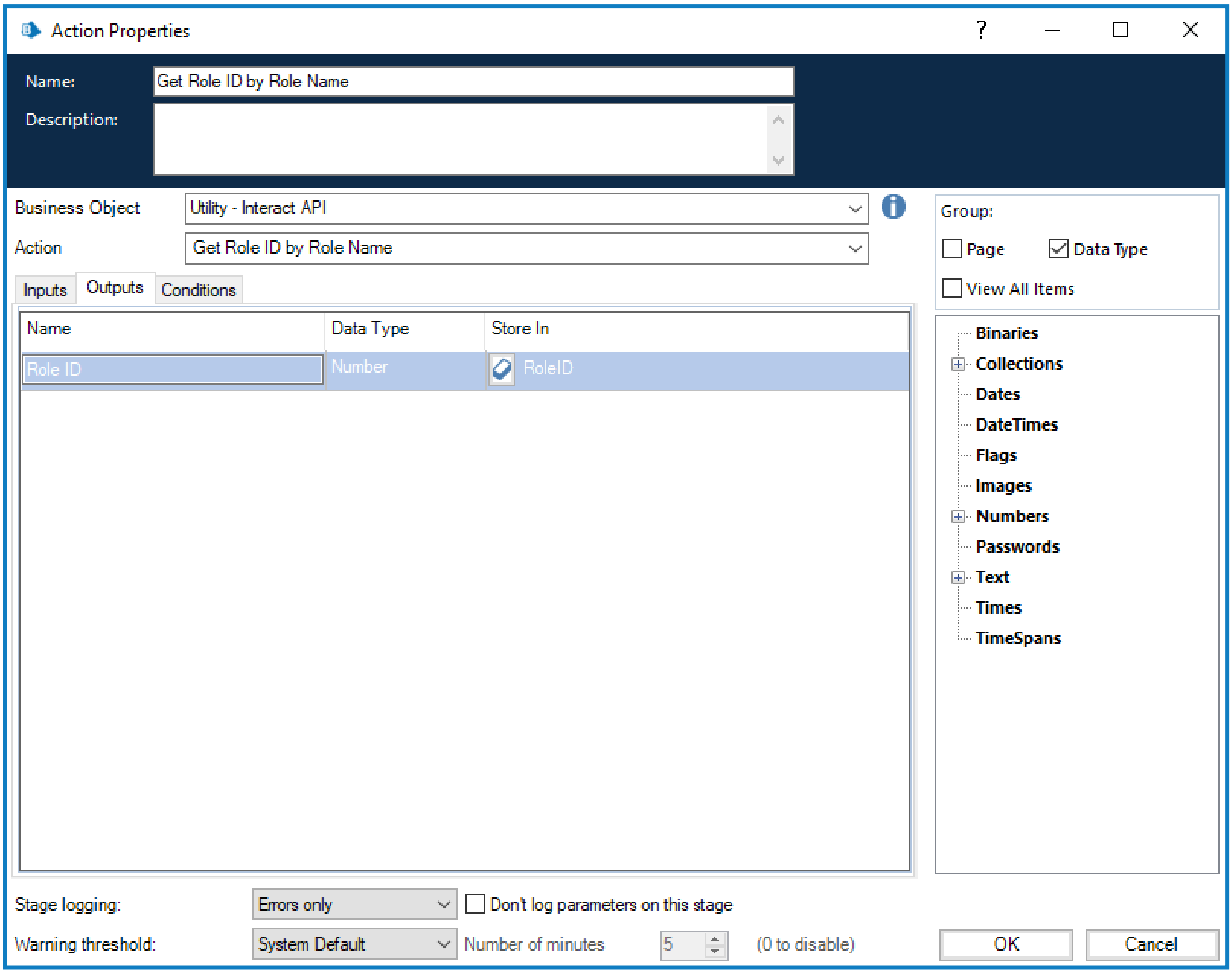
Name	Data Type	Value
Role Name	Text	"Human Resources"

Group: □ Page ☑ Data Type □ View All Items

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

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

The output is the Role ID, which is the identification number associated with that Role Name.



Name: Get Role ID by Role Name

Description:

Business Object: Utility - Interact API

Action: Get Role ID by Role Name

Name	Data Type	Store In
Role ID	Number	RoleID

Group: □ Page ☑ Data Type

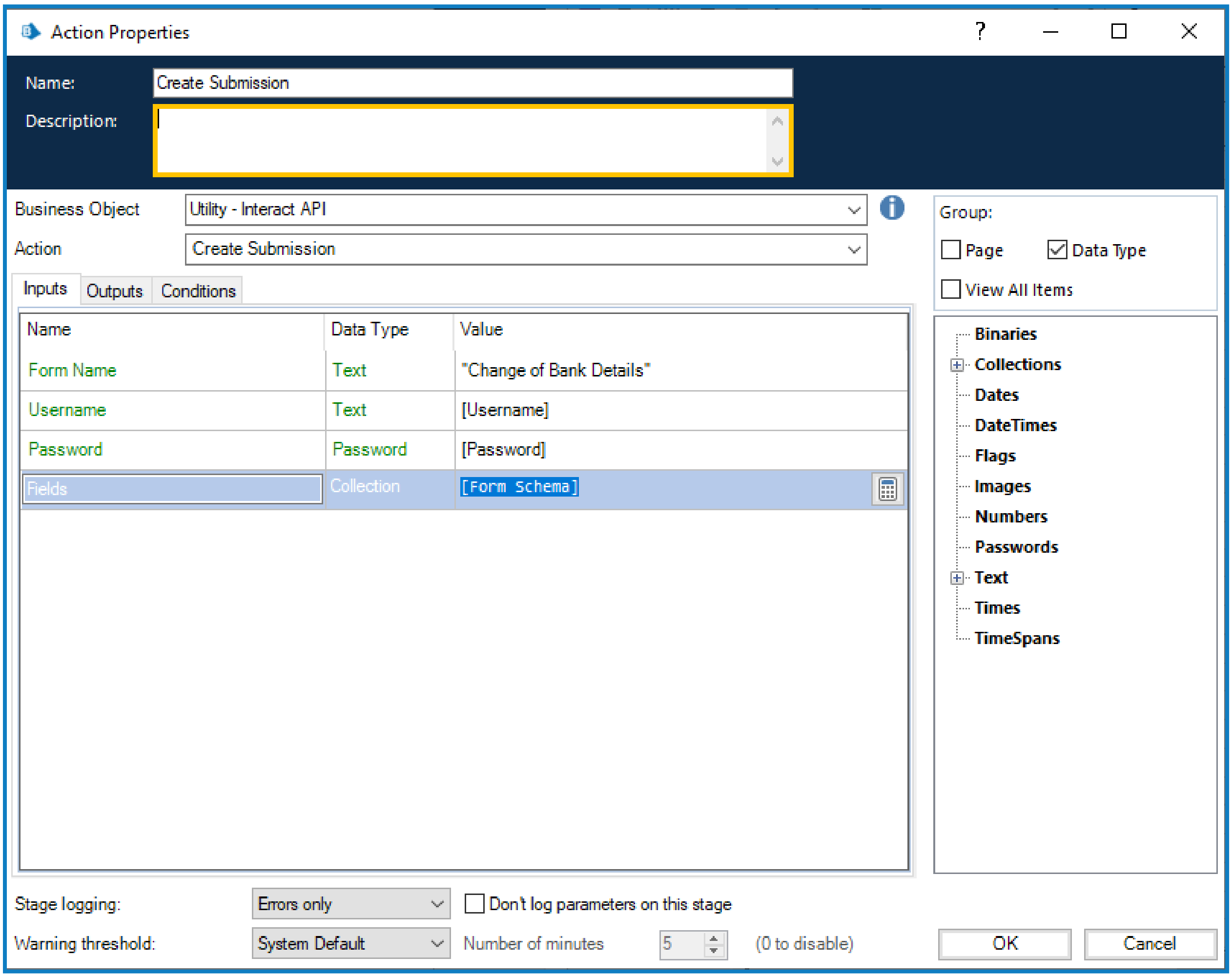
□ View All Items

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

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

Create Submission

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



Name: Create Submission

Description:

Business Object: Utility - Interact API

Action: Create Submission

Inputs | Outputs | Conditions

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

Group: □ Page ☑ Data Type □ View All Items

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

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

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. If you are using Interact Remote API v2, this is an optional parameter. If you are using Interact Remote API v1, this is a required parameter.
- **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 17 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.

The Create Submission action has the following optional output parameters:

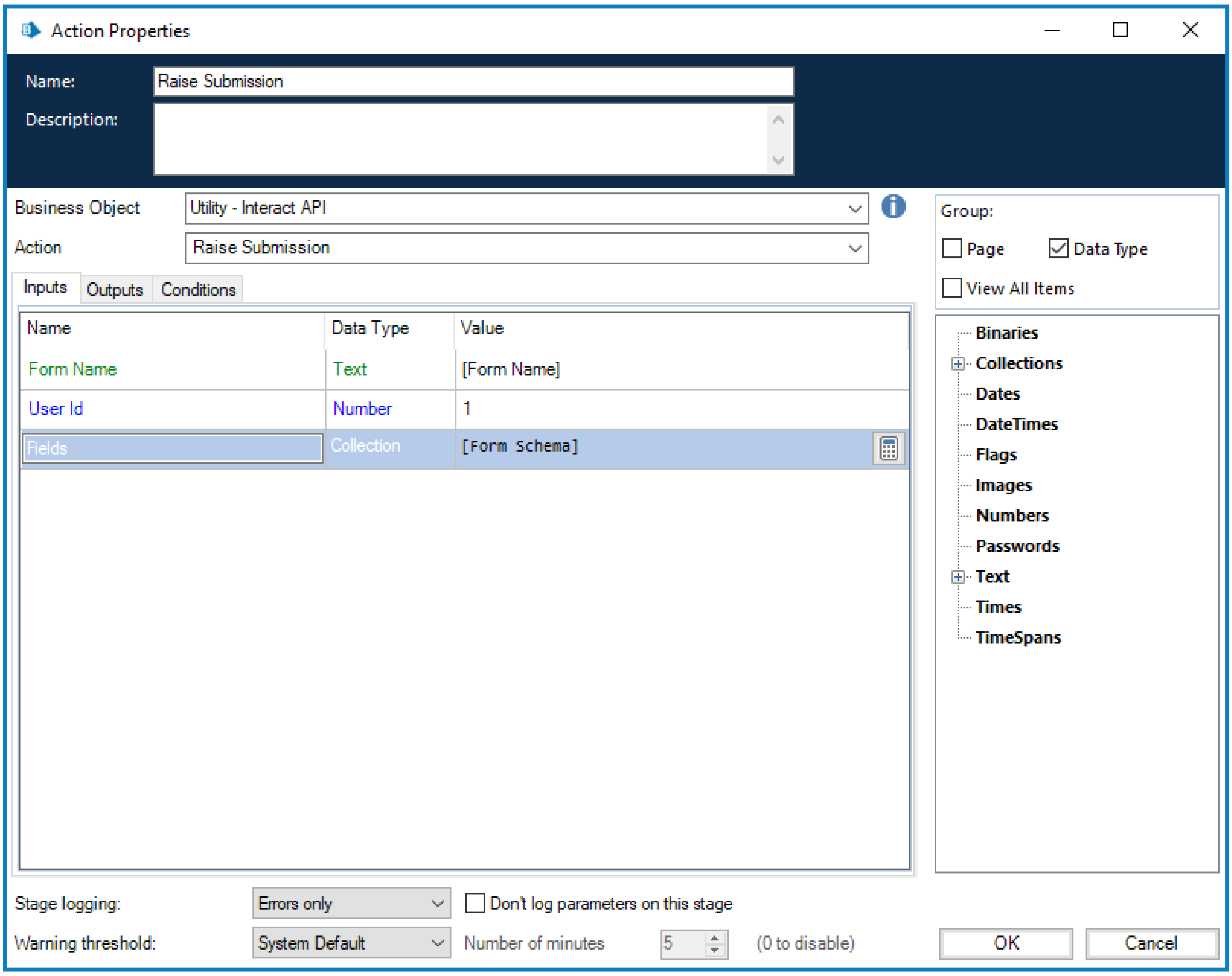
- **_requestId** – The unique identifier for the submission.
- **Submission Data** – Retrieves the contents of the Interact Form into a collection. This is in the same format as Get Submission on page 14.

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.

> If the form that your submission is based on contains the radio group capture type, you will be unable to raise an empty submission. The radio group requires a value to be set. For more information on form capture types, see the Interact Plugin User Guide.

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.



Action Properties

Name: Raise Submission
Description:

Business Object: Utility - Interact API
Action: Raise Submission

Inputs | Outputs | Conditions

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

Group: □ Page ☑ Data Type
□ View All Items

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

The Raise Submission action has the following optional output parameters:

- **_requestId** – The unique identifier for the submission.
- **Submission Data** – Retrieves the contents of the Interact Form into a collection. This is in the same format as Get Submission on page 14.

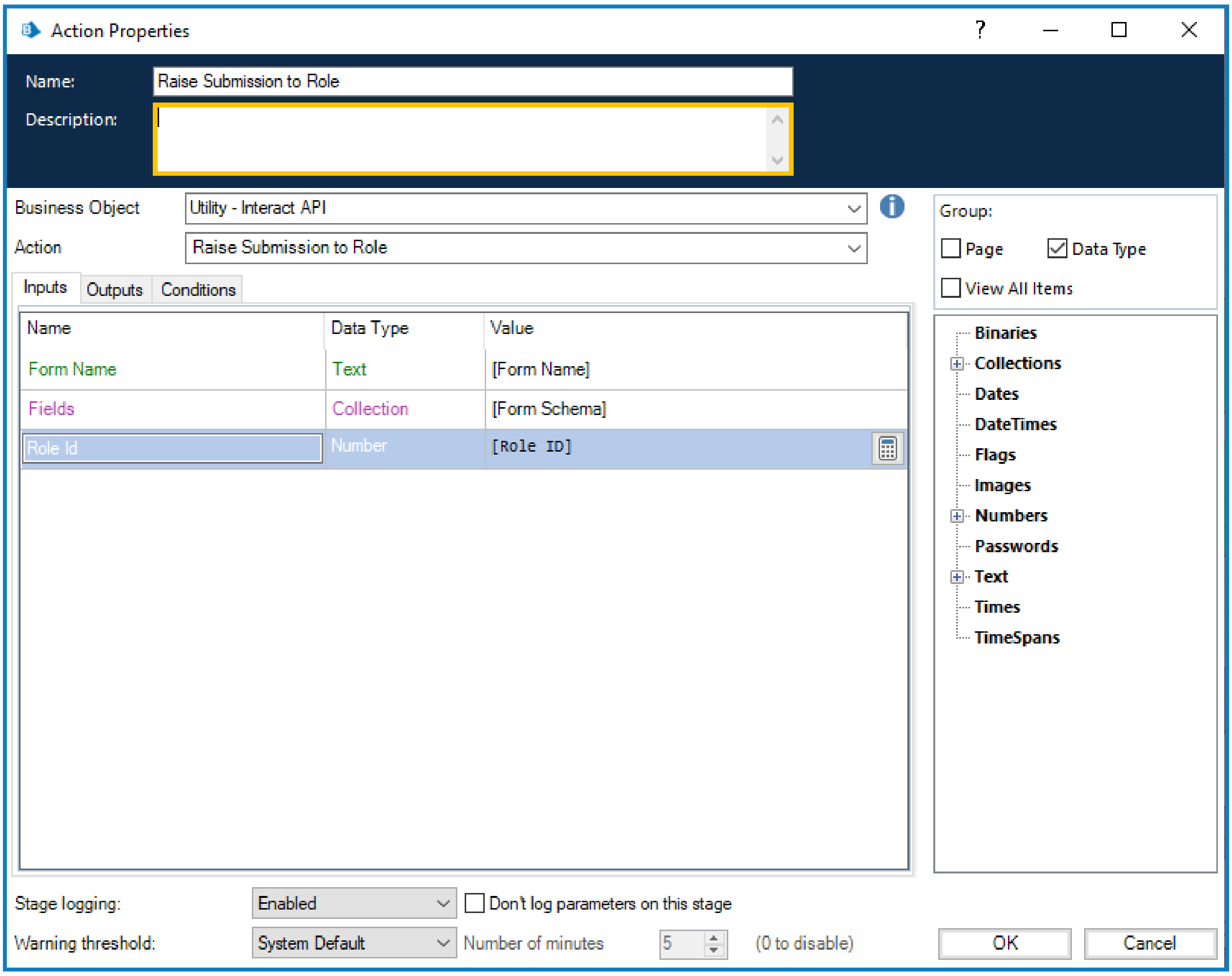
When using Raise Submission with Upload File on the next page, ensure you correctly format the collection containing the details of the file being attached. See Get Form Schema on page 17 for more details about the format of the collection.

Raise Submission to Role

The Raise Submission to Role action allows you to create a submission that appears in the Inbox folder of all users who are assigned to that role. The submission raised can be fully or partially completed, or even completely empty depending on your requirements.

If the form that your submission is based on contains the radio group capture type, you will be unable to raise an empty submission. The radio group requires a value to be set. For more information on form capture types, see the Interact Plugin User Guide.

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



Action Properties

Name: Raise Submission to Role

Description:

Business Object: Utility - Interact API

Action: Raise Submission to Role

Inputs | Outputs | Conditions

Name	Data Type	Value
Form Name	Text	[Form Name]
Fields	Collection	[Form Schema]
Role Id	Number	[Role ID]

Group: □ Page ☑ Data Type □ View All Items

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

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

The Raise Submission to Role action has the following optional output parameters:

- **_requestId** – The unique identifier for the submission.
- **Submission Data** – Retrieves the contents of the Interact Form into a collection. This is in the same format as Get Submission on page 14.

When using Raise Submission to Role with Upload File below, ensure you correctly format the collection containing the details of the file being attached. See Get Form Schema on page 17 for more details about the format of the collection.

Upload File

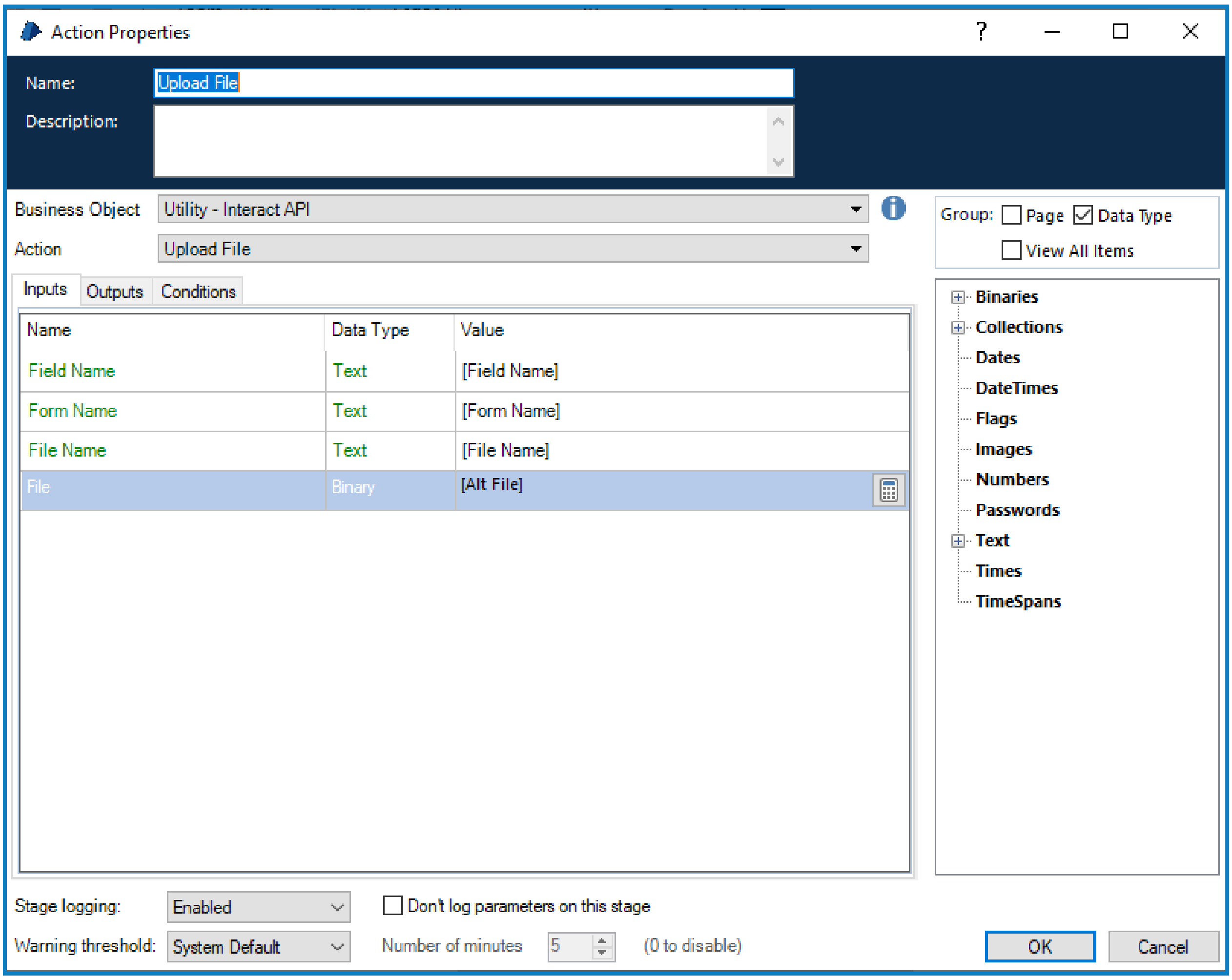
The Upload File action allows you to upload a file to local storage, for example, an image file or a document. It can be used in conjunction with:

- Create Submission on page 27
- Upload on page 47
- Raise Submission on page 28
- Raise Submission to Role on the previous page

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: Upload File

Description:

Business Object: Utility - Interact API

Action: Upload File

Group: □ Page ☑ Data Type
□ View All Items

Inputs | Outputs | Conditions

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

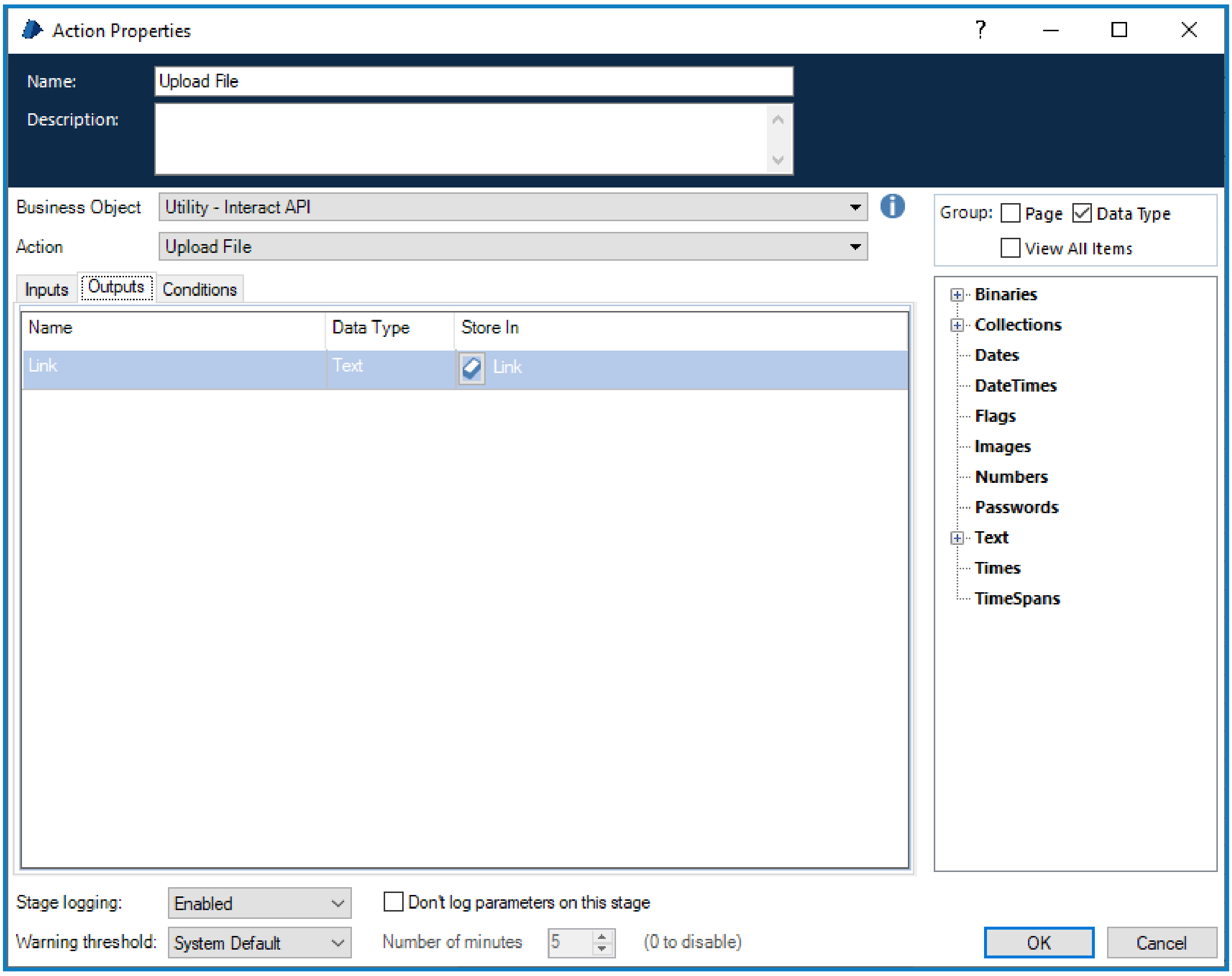
Stage logging: Enabled

□ Don't log parameters on this stage

Warning threshold: System Default

Number of minutes 5 (0 to disable)

When uploading a file, you are provided with a link which can then be used in the actions listed above.



Action Properties

Name: Upload File

Description:

Business Object: Utility - Interact API

Action: Upload File

Group: □ Page ☑ Data Type □ View All Items

Name	Data Type	Store In
Link	Text	Link

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

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

Once the file has been uploaded, you can attach the contents of the file to the appropriate action. To do this, you need to:

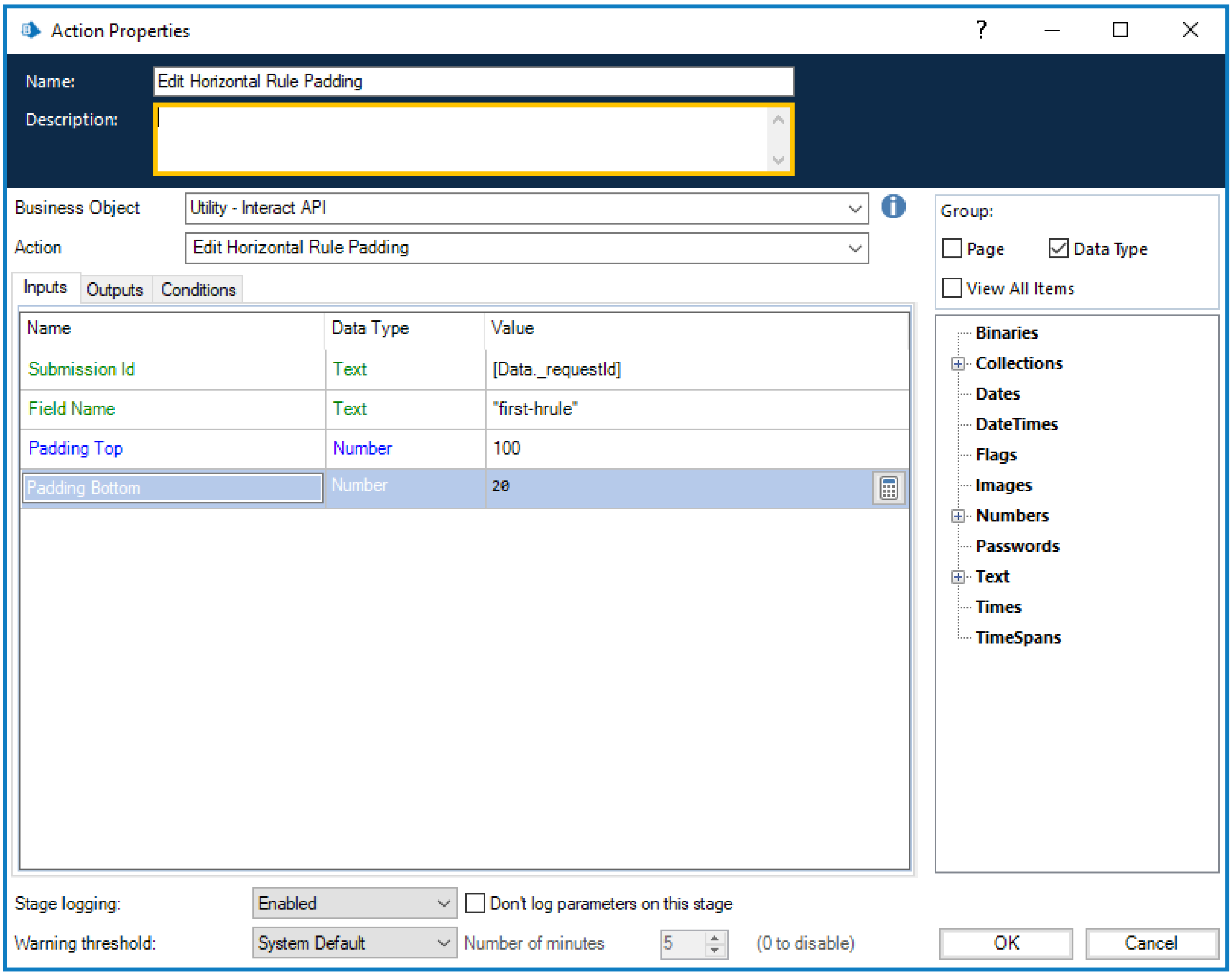
1. **Upload File** – Upload the file to the local storage.
2. **Get Form Schema** – Obtain the structure of the form where the file will be attached to.
3. **Create Submission or Raise Submission** – Which references the modified Form structure and attaches the uploaded file to the new submission.

> 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 Horizontal Rule Padding

The Edit Horizontal Rule Padding action complements the actions available in Edit Submission Field Value on the next page. This action links directly to the horizontal rule element on the form, enabling the padding values to be changed.

The Field Name is the Automation ID that was created within the Form designer. The padding fields accept a number, this defines the number of pixels of vertical space that will appear above and/or below the horizontal rule on the form.



Name: Edit Horizontal Rule Padding
Description:
Business Object: Utility - Interact API
Action: Edit Horizontal Rule Padding

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"first-hrule"
Padding Top	Number	100
Padding Bottom	Number	20

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

Edit Submission Field Value

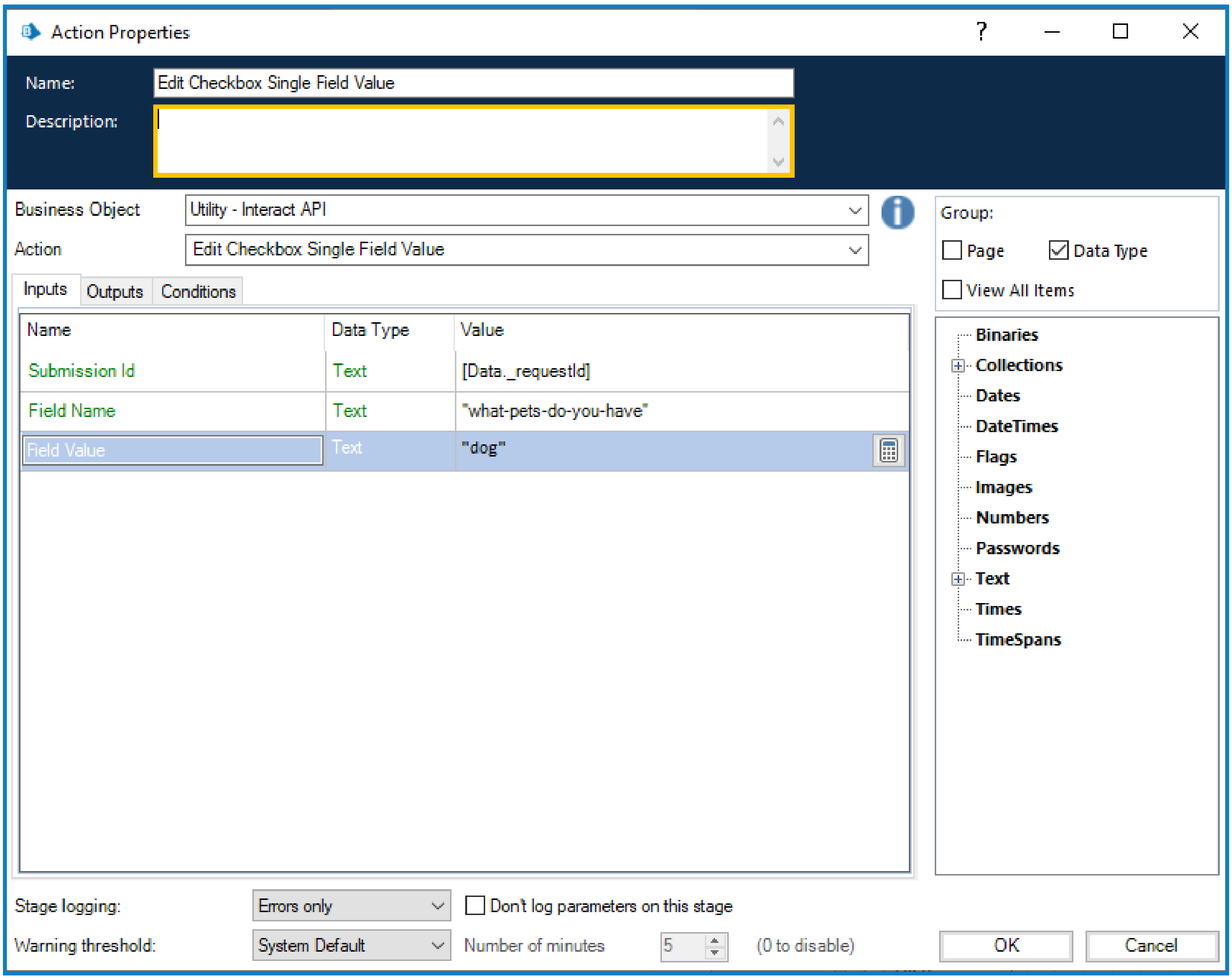
The Edit Submission Field Value action, like all the other actions, is called from the object. There are several actions within the object which allow you to update specific field elements, these actions link directly to the element type that is used in the Interact Form.

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.

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.



Name: Edit Checkbox Single Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Checkbox Single Field Value

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

Group: □ Page ☑ Data Type

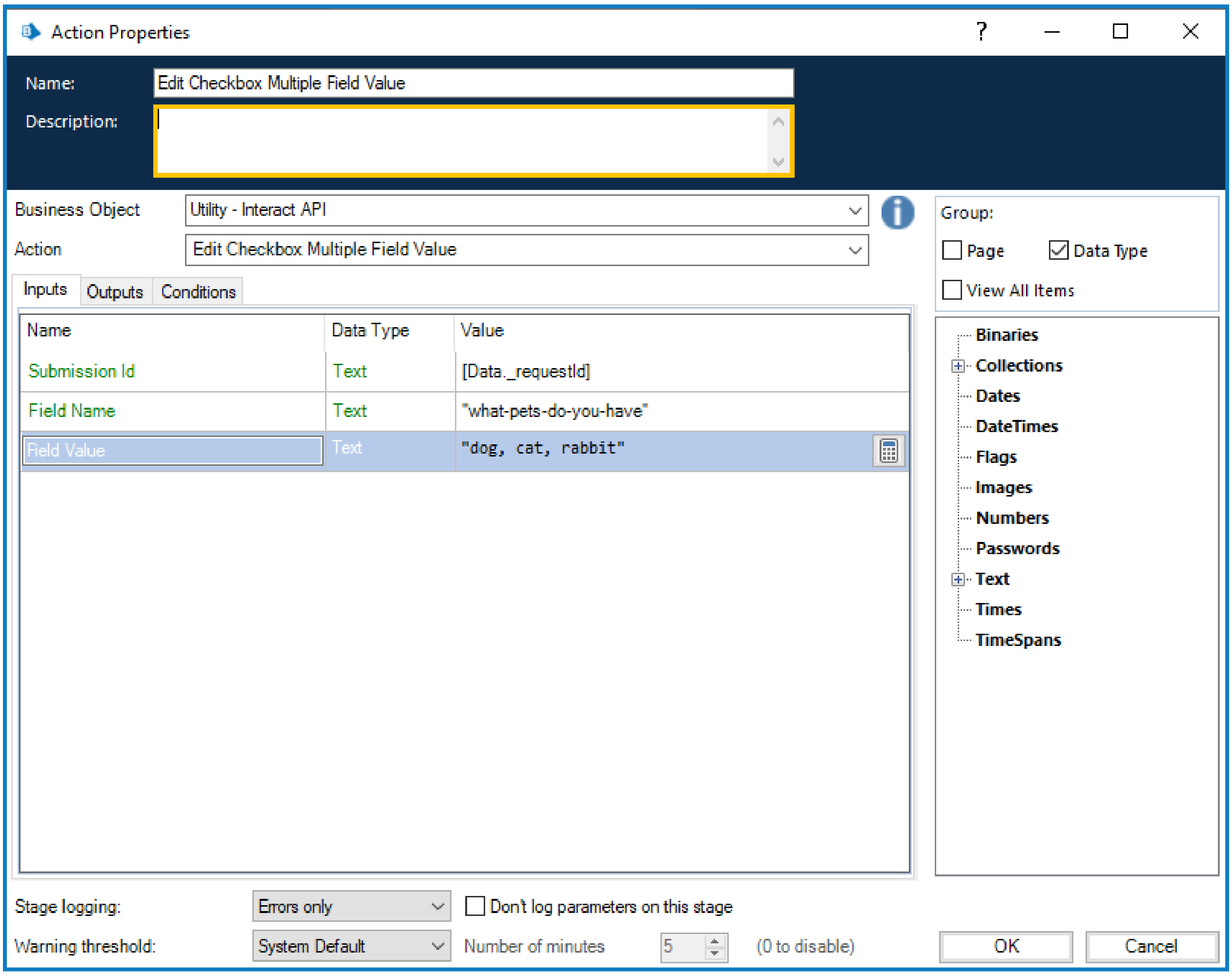
□ View All Items

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

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

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.



Name: Edit Checkbox Multiple Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Checkbox Multiple 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	"dog, cat, rabbit"

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

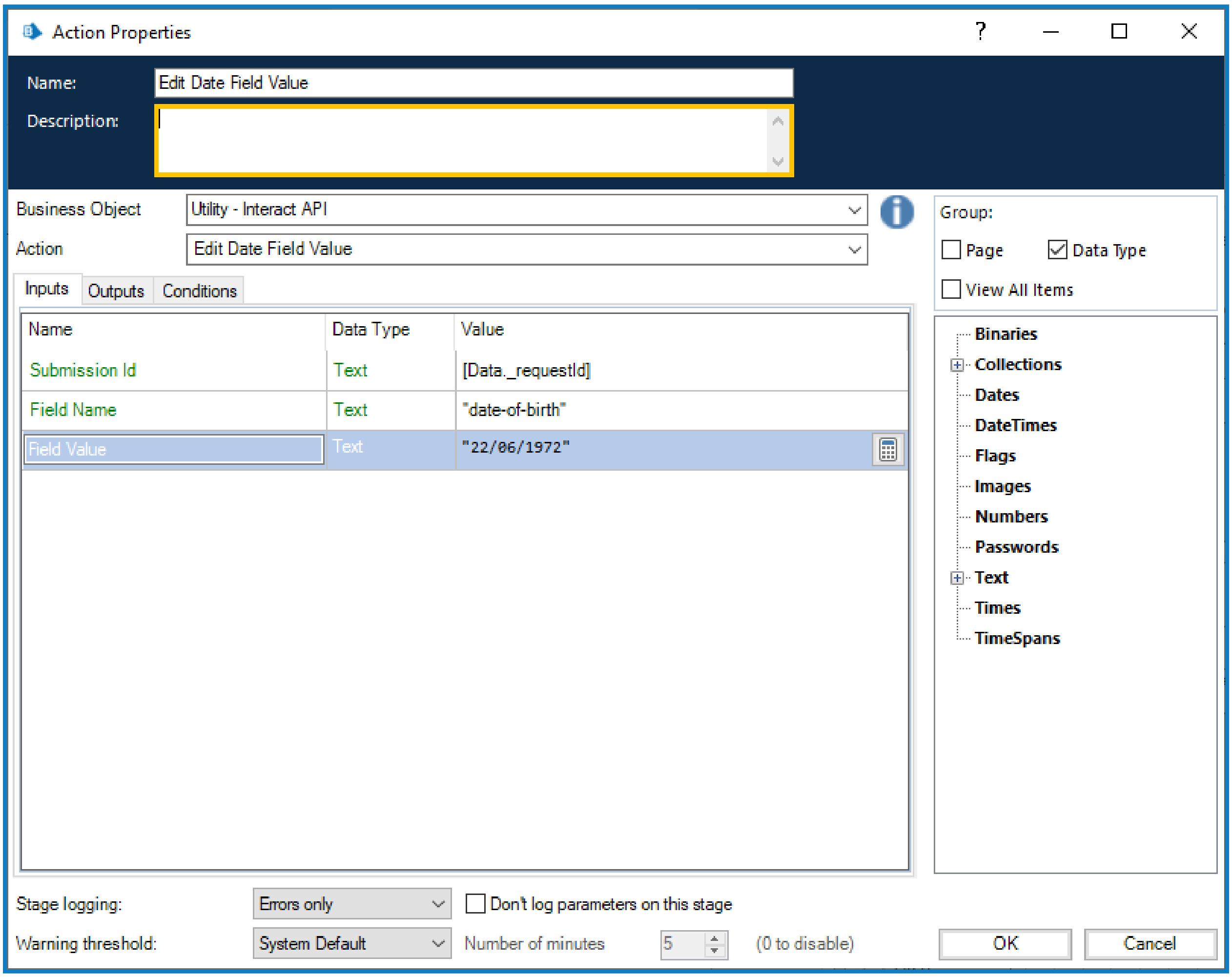
Warning threshold: System Default Number of minutes 5 (0 to disable)

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.



Name: Edit Date Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Date Field Value

Group: □ Page ☑ Data Type □ View All Items

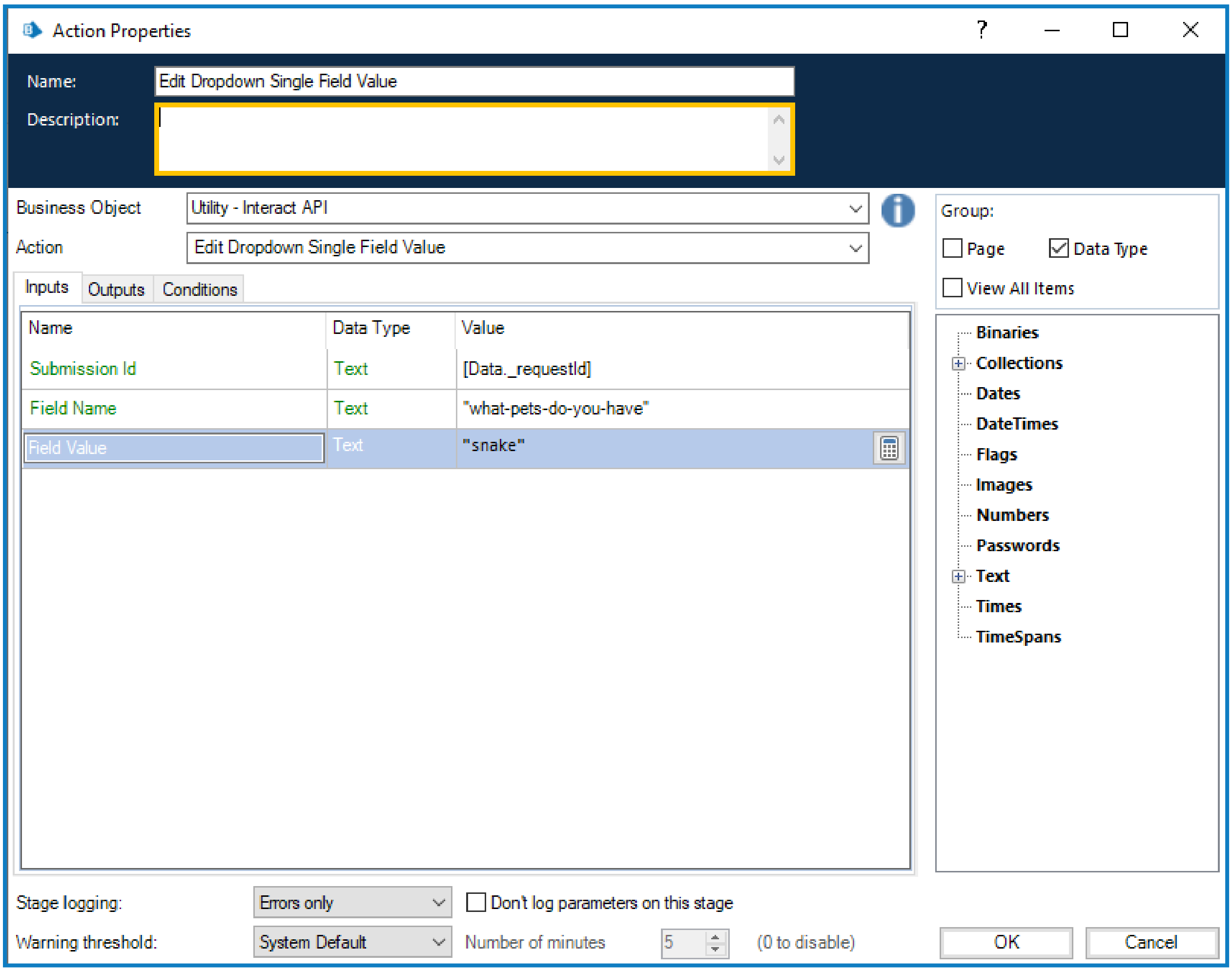
Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"date-of-birth"
Field Value	Text	"22/06/1972"

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

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

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:

Business Object: Utility - Interact API

Action: Edit Dropdown Single Field Value

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

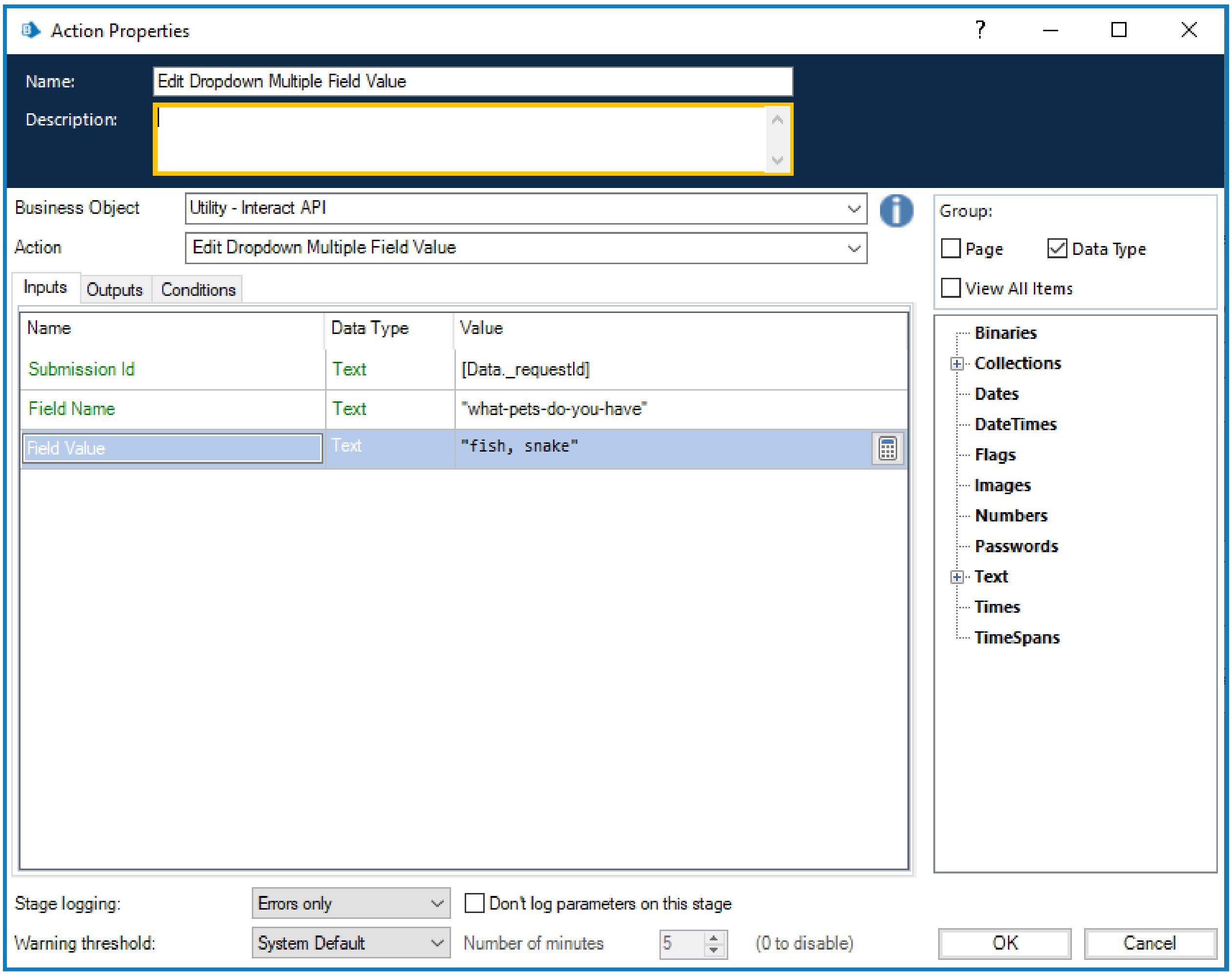
Group: □ Page ☑ Data Type
□ View All Items

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

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

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.



Action Properties

Name: Edit Dropdown Multiple Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Dropdown Multiple Field Value

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

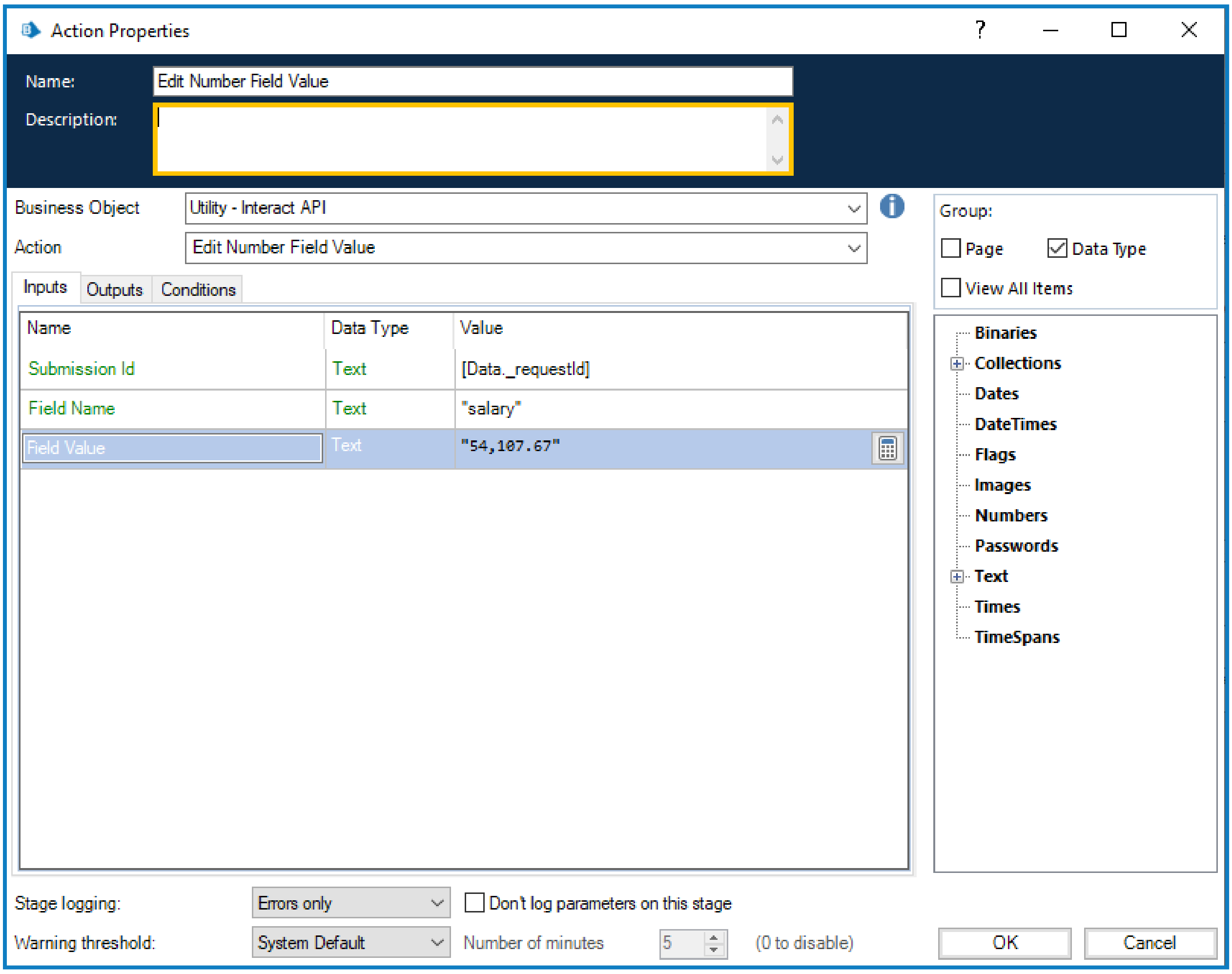
Group:
□ Page ☑ Data Type
□ View All Items

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

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

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: Edit Number Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Number Field Value

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"salary"
Field Value	Text	"54,107.67"

Group: □ Page ☑ Data Type

□ View All Items

Stage logging: Errors only

□ Don't log parameters on this stage

Warning threshold: System Default

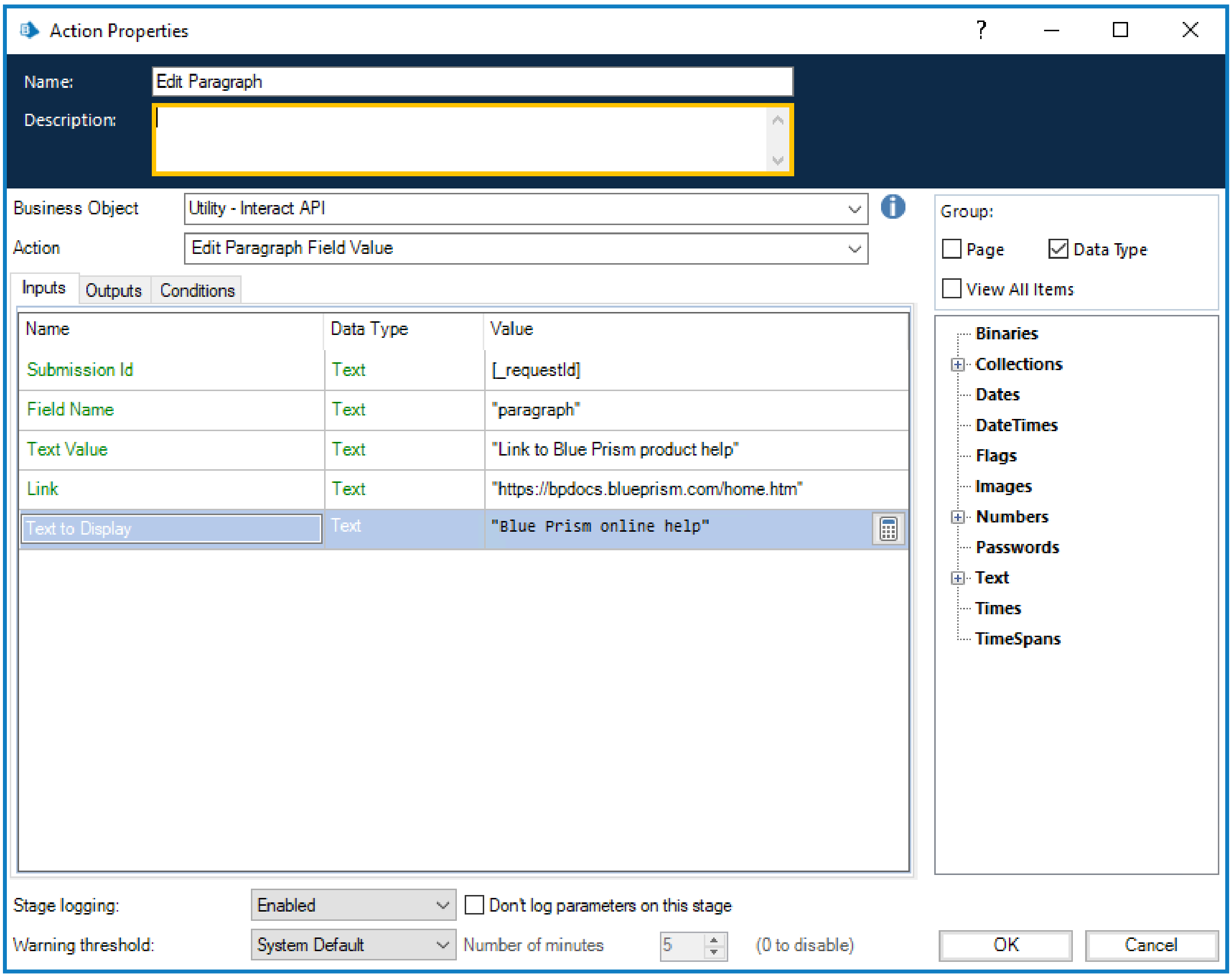
Number of minutes: 5 (0 to disable)

Paragraph element

The Paragraph element accepts a string of text or HTML, a hyperlink, and a label for the hyperlink. All of these fields are optional.

In its simplest form, the Text Value parameter can be used to update a paragraph of text, without any formatting. A hyperlink can also be added which will display on the line immediately below the paragraph of text entered in Text Value when displayed in the form. The hyperlink comprises the Link parameter for the URL and the Text to Display parameter for the text.

The example below shows the information being set on the action directly, however, it is more likely that you will use Data items in your automation to input the information into these fields.



Action Properties

Name: Edit Paragraph

Description:

Business Object: Utility - Interact API

Action: Edit Paragraph Field Value

Name	Data Type	Value
Submission Id	Text	[_requestId]
Field Name	Text	"paragraph"
Text Value	Text	"Link to Blue Prism product help"
Link	Text	"https://bpdocs.blueprism.com/home.htm"
Text to Display	Text	"Blue Prism online help"

Group: □ Page ☑ Data Type

□ View All Items

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

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

Alternatively, HTML can be entered in the Text Value parameter, which can be used to update a paragraph with formatted text. If HTML is to be entered directly into the input fields, any quotation marks inside the HTML must be escaped using a backslash (for example, when inserting links or using styles) as the entry in the Text Value parameter is wrapped in quotation marks. This can also be built up using the expression validation. The HTML must be correctly formatted using opening <> and closing </> tags.

For example:

```
<p>Link to <a href="https://bpdocs.blueprism.com/home.htm">Blue Prism product
help</a></p>
```

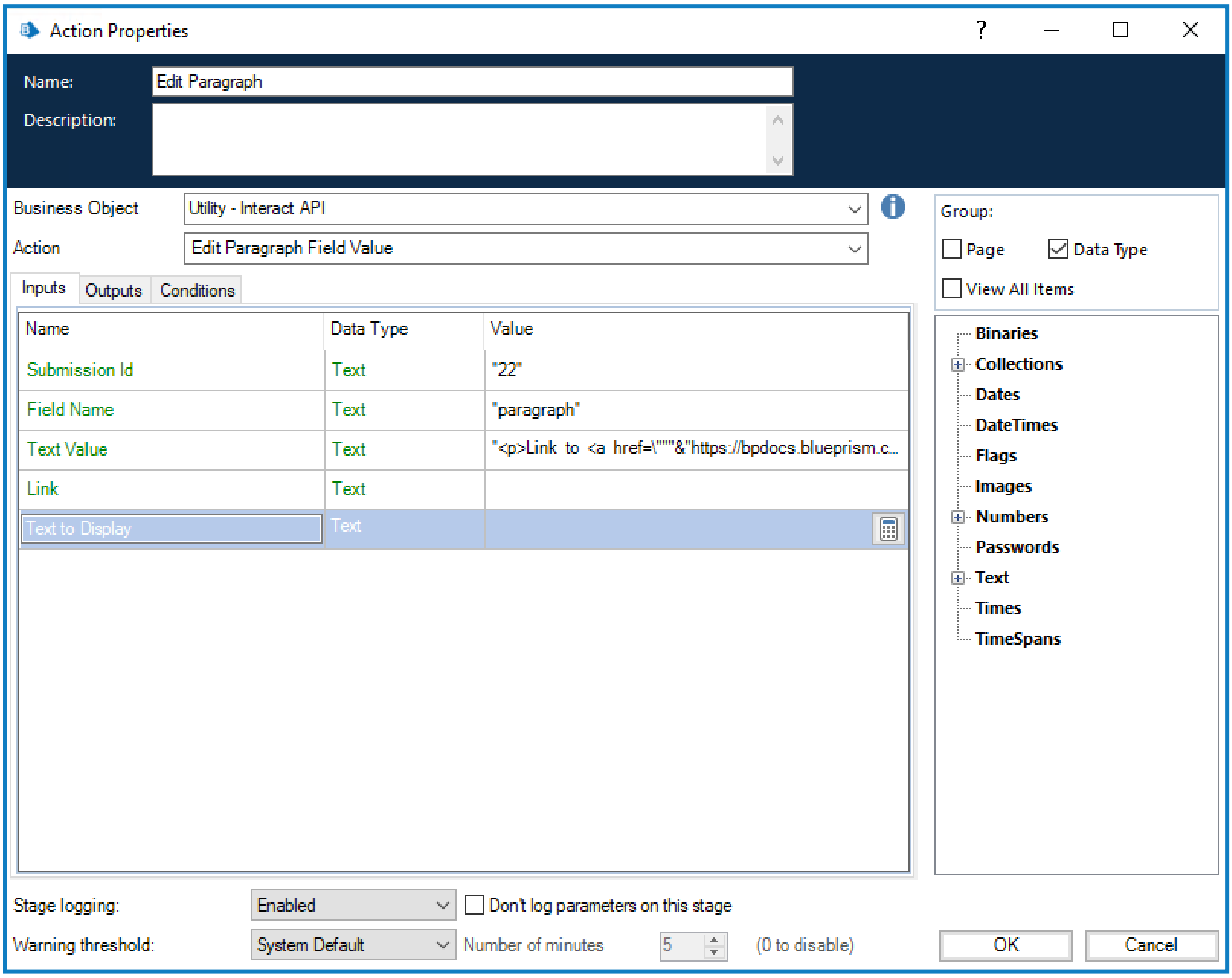
When using a Data item to input the information into the action, this becomes:

```
"<p>Link to <a href=\"https://bpdocs.blueprism.com/home.htm\">Blue Prism product
help</a></p>"
```

It is recommended that you use Data items to input the information into these fields as this avoids the formatting issue with escaped characters required by the Blue Prism fields.

For example, entering the above example directly into the field will require further escaping of the quotation marks:

```
"<p>Link to <a href=\"""&"https://bpdocs.blueprism.com/home.htm\"""&">Blue Prism product help</a></p>"
```



Action Properties

Name: Edit Paragraph

Description:

Business Object: Utility - Interact API

Action: Edit Paragraph Field Value

Inputs | Outputs | Conditions

Name	Data Type	Value
Submission Id	Text	"22"
Field Name	Text	"paragraph"
Text Value	Text	"<p>Link to <a href=\"""&"https://bpdocs.blueprism.c...
Link	Text	
Text to Display	Text	

Group: □ Page ☑ Data Type □ View All Items

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

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

To reflect the formatting that can be entered in the paragraph field in the Interact Forms user interface when building a form, the following HTML tags can be used:

- Paragraph – <p> –The HTML must always be wrapped by the paragraph tag <p>, for example, `<p>This is my paragraph.</p>`.

  The Text Value parameter can contain more than one paragraph, for example, `<p>This is my first paragraph. This is my next sentence in my first paragraph.</p><p>This is my second paragraph.</p>`.
- **Bold** – <b> – Wrap the required text using the bold tag <b>, for example, `<p>this is <b>bold</b></p>`.
- *Italic* – <i> – Wrap the required text using the italic tag <i>, for example, `<p>this is <i>italic</i></p>`.
- <u>underline</u> – <u> – Wrap the required text using the underline tag <u>, for example, `<p>this <u>text is underlined</u></p>`.

- Highlight – <mark> – Wrap the required text using the highlight tag <mark>, for example, `<p>this <mark>text is highlighted</mark></p>`. By default, this displays the text as black font with a yellow background.

  Alternatively, a span tag can be used with the style attribute defining the background color of the text, for example, `<p>this<span style="background-colour:blue;">text is highlighted</span></p>`.

  Remember that the quotation marks must be escaped, so this becomes: `<p><span style=\"background-color:blue;\">text is highlighted</span></p>`.

- Text color – <span style> – To change the color of certain words, the text needs to be wrapped in a span tag with the style attribute applied defining the color, for example, `<p>this text is <span style="color:red;">red</span></p>`.

  Remember that the quotation marks must be escaped, so this becomes:`<p>this text is <span style=\"color:red;\">red</span></p>`

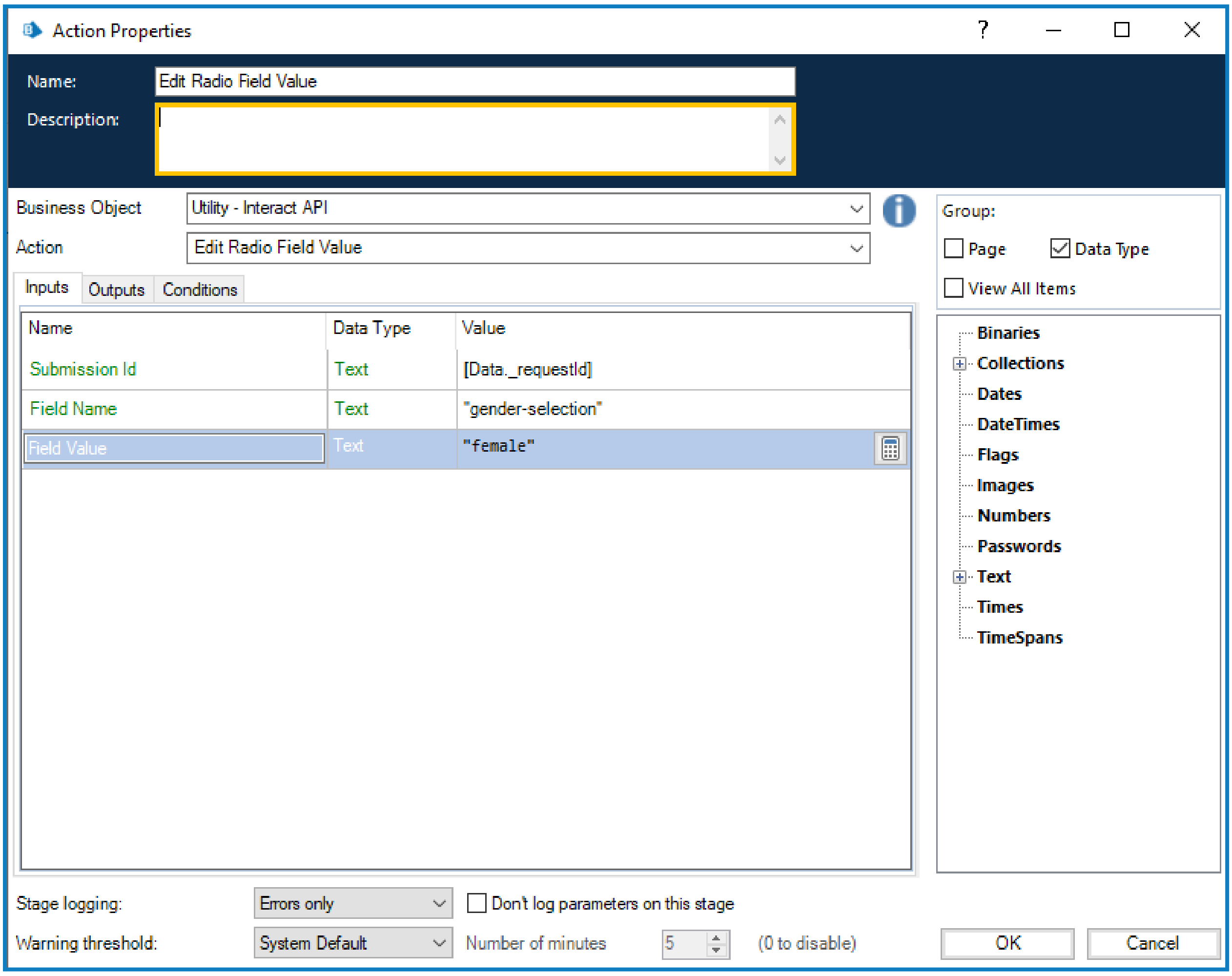
  Alternatively, to have all the text in the paragraph display in a different color to the standard form, the style can be specified on the paragraph tag.

- Hyperlinks – < a href=""> – Wrap the text that will be linked using the hyperlink tag and attribute <a href>, for example, `<p>this is a <a href="http://www.blueprism.com">hyperlink</a></p>`.

  Remember that the quotation marks must be escaped, so this becomes: `<p>this is a <a href=\"http://www.blueprism.com\">hyperlink</a></p>`.

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: Edit Radio Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Radio Field Value

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

Group: □ Page ☑ Data Type

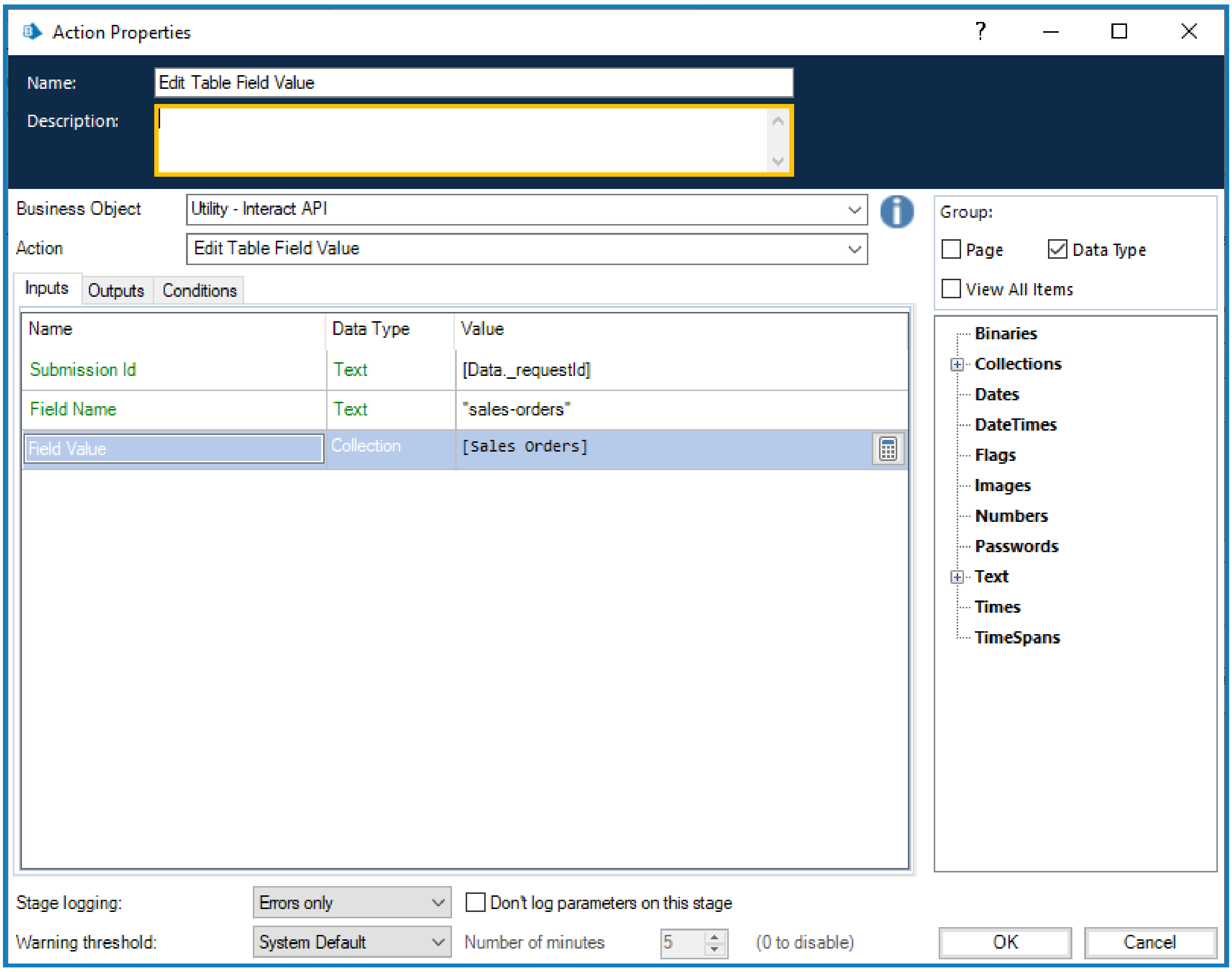
□ View All Items

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

Warning threshold: System Default Number of minutes 5 (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. If the form specifies a data type for a cell, such as number, the input must match the data type otherwise the update with fail.



Name: Edit Table Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Table Field Value

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

Group: □ Page ☑ Data Type □ View All Items

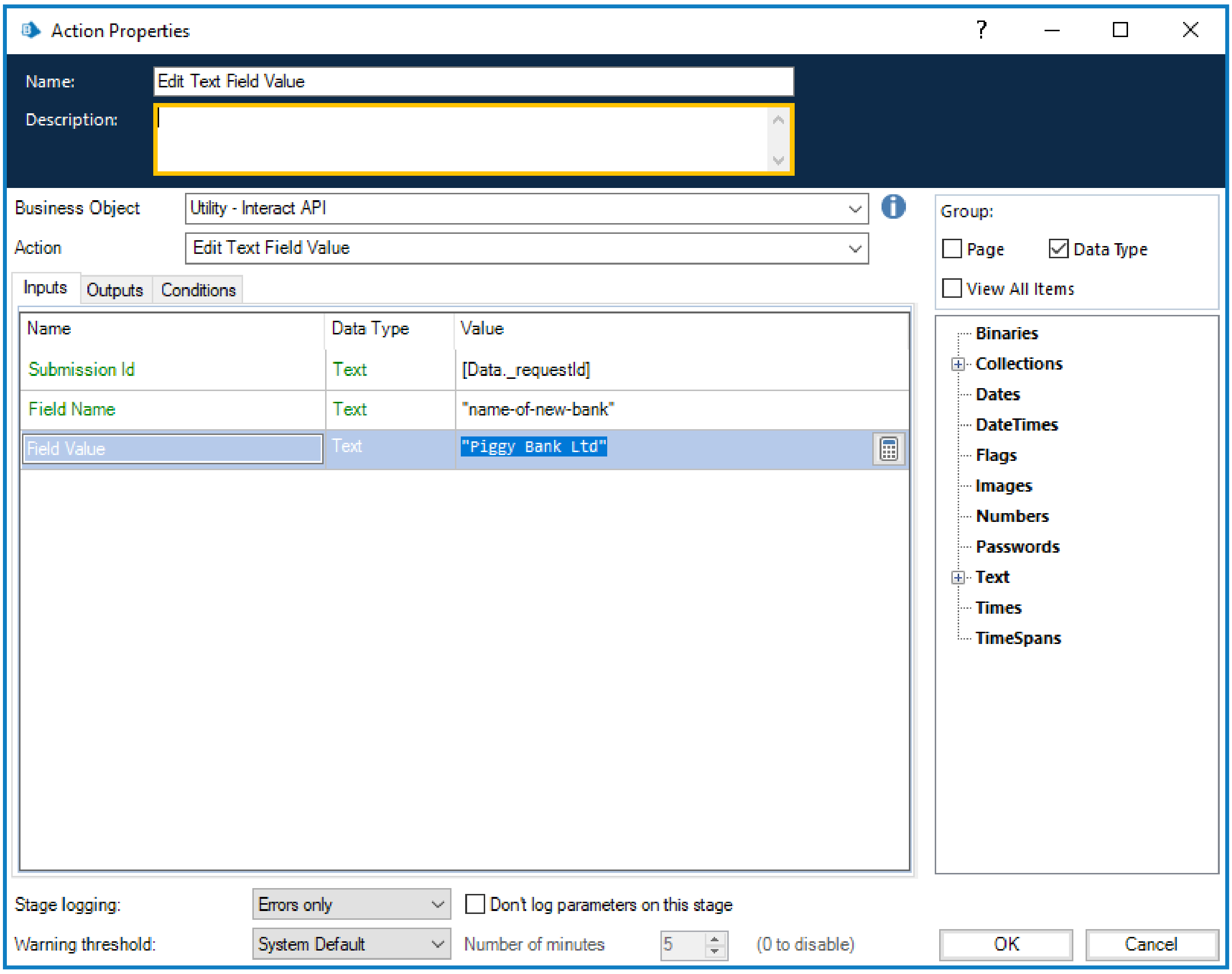
Stage logging: Errors only □ Don't log parameters on this stage

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

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

Text element

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



Name: Edit Text Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Text Field Value

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"name-of-new-bank"
Field Value	Text	"Piggy Bank Ltd"

Group: □ Page ☑ Data Type

□ View All Items

Stage logging: Errors only

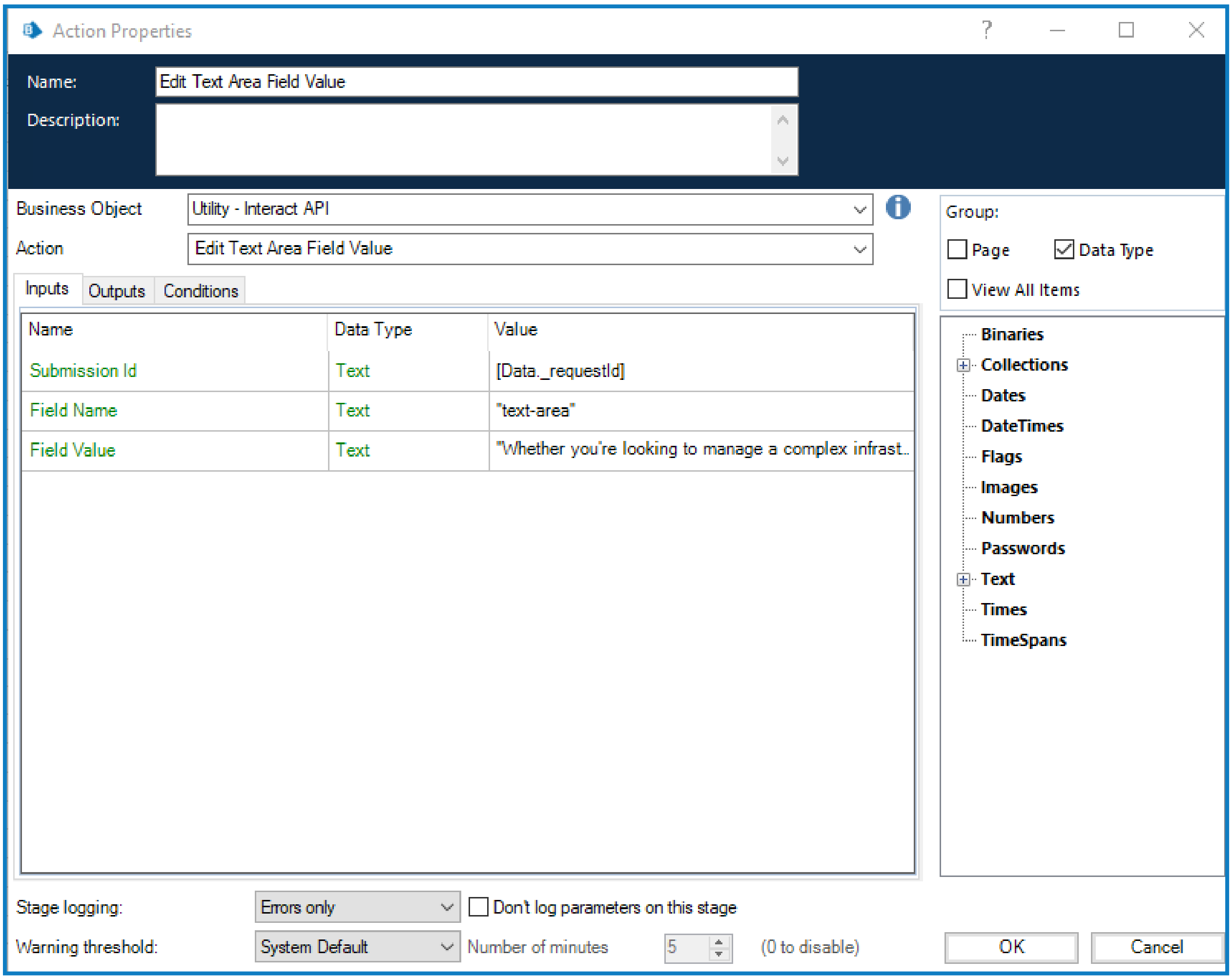
□ Don't log parameters on this stage

Warning threshold: System Default

Number of minutes 5 (0 to disable)

Text Area element

The Text Area element accepts a string of text up to 3500 characters in length.



Name: Edit Text Area Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Text Area Field Value

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"text-area"
Field Value	Text	"Whether you're looking to manage a complex infrast..

Group:

□ Page ☑ Data Type

□ View All Items

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

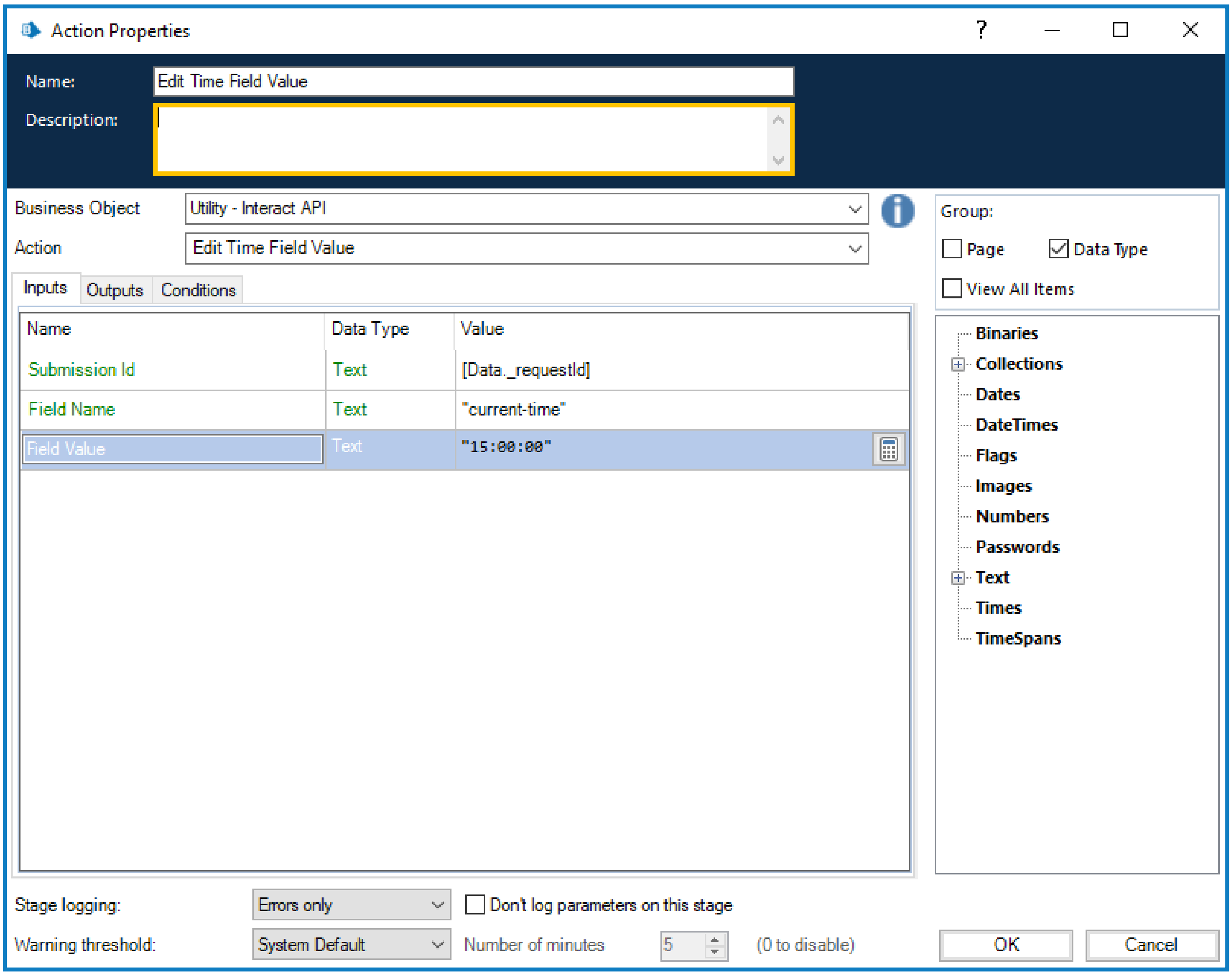
Warning threshold: System Default Number of minutes 5 (0 to disable)

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:

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"

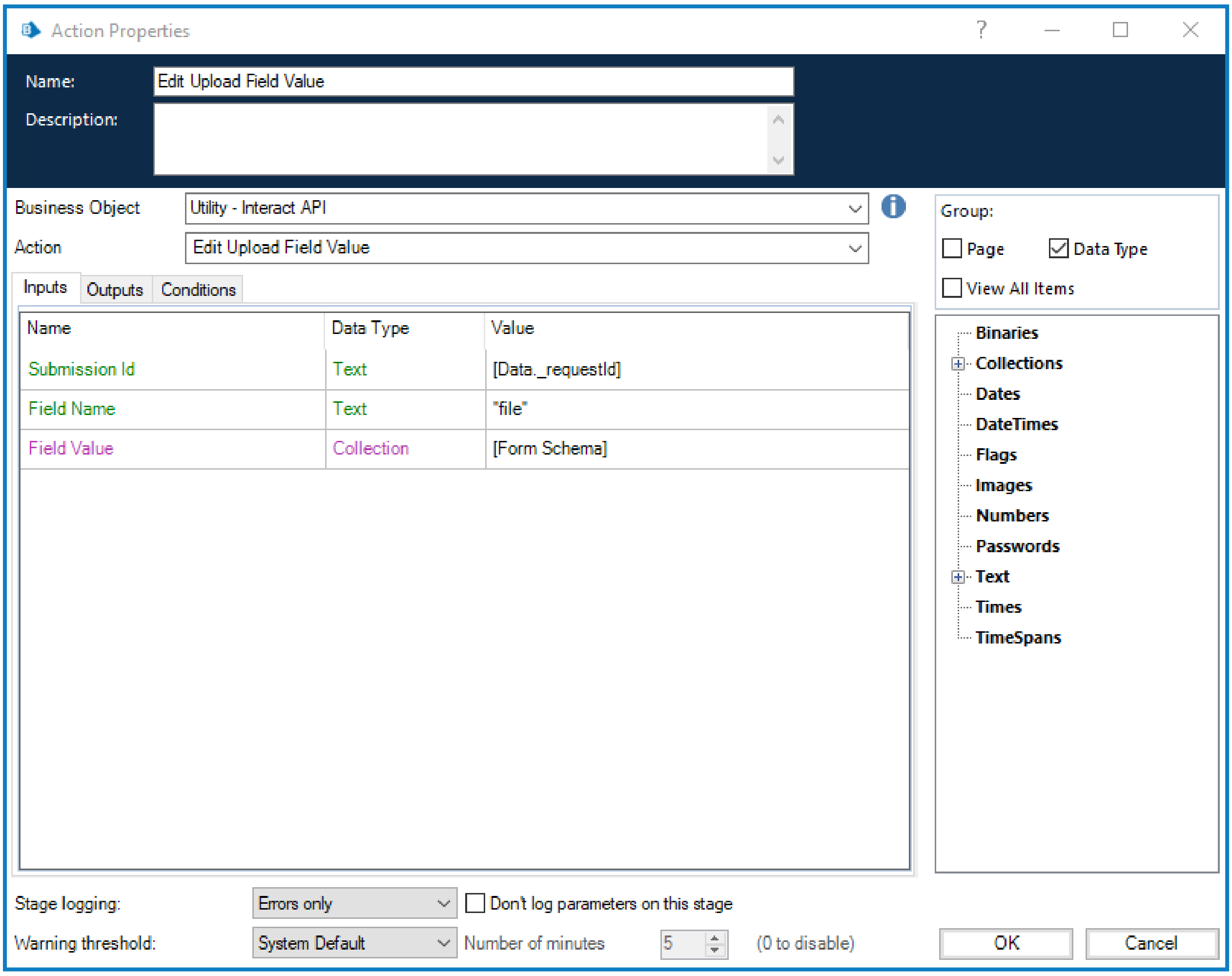
Group: □ Page ☑ Data Type □ View All Items

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

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

Upload

The Edit Upload Field Value action enables an existing file, such as an image or document, to be replaced with a new file.



Name: Edit Upload Field Value

Description:

Business Object: Utility - Interact API

Action: Edit Upload Field Value

Name	Data Type	Value
Submission Id	Text	[Data._requestId]
Field Name	Text	"file"
Field Value	Collection	[Form Schema]

Group: □ Page ☑ Data Type
□ View All Items

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

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

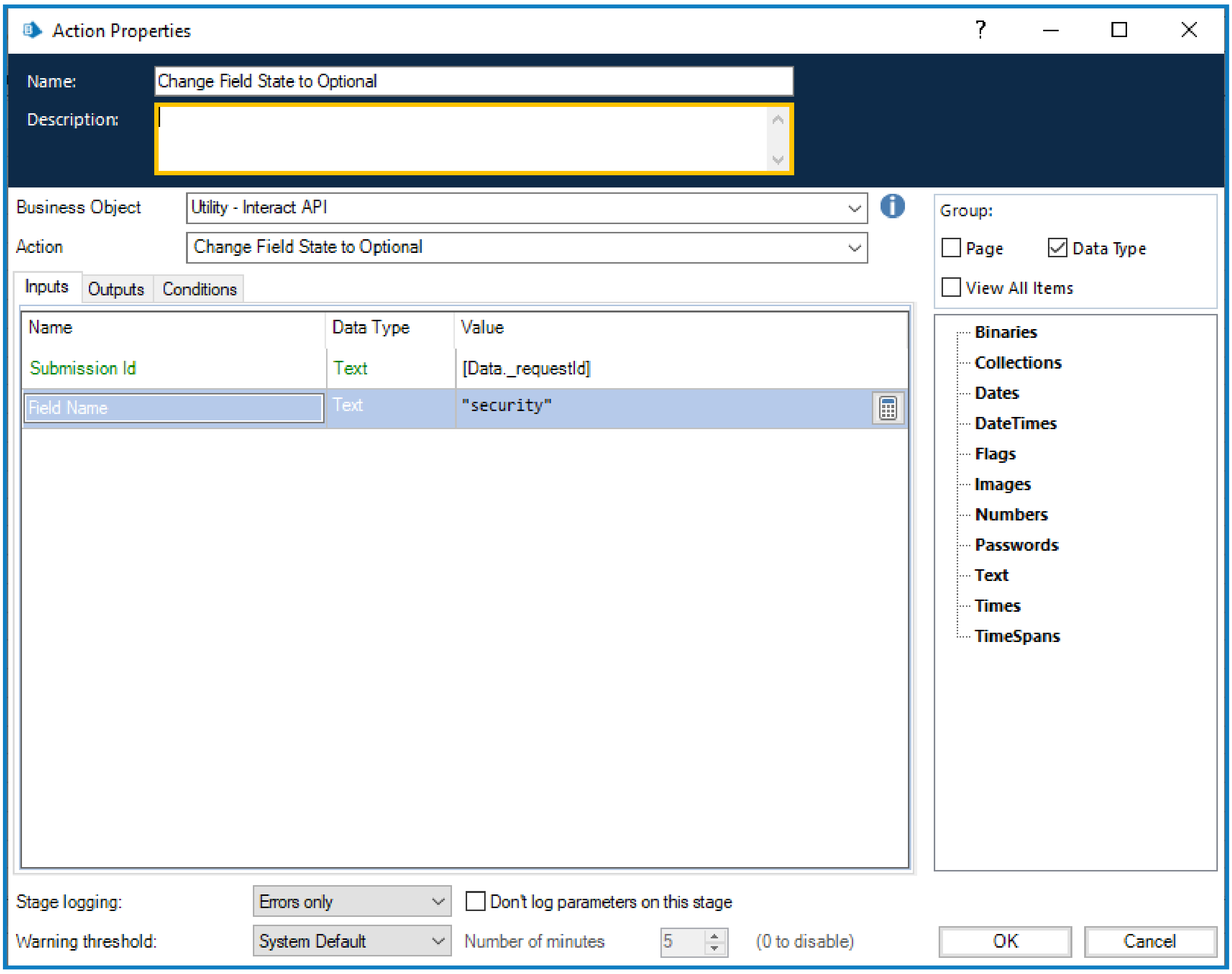
If this action is used with:

- Get Form Schema, you can start with a blank submission and use this action to upload a file.
- Get Submission or **Get Next Item** (from the Blue Prism Internal Business Objects, Work Queues), there must already be a file present to be replaced by Edit Upload Field Value.

Change Field State to Optional

The Change Field State to Optional action 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.



Name: Change Field State to Optional

Description:

Business Object: Utility - Interact API

Action: Change Field State to Optional

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

□ Page ☑ Data Type
□ View All Items

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

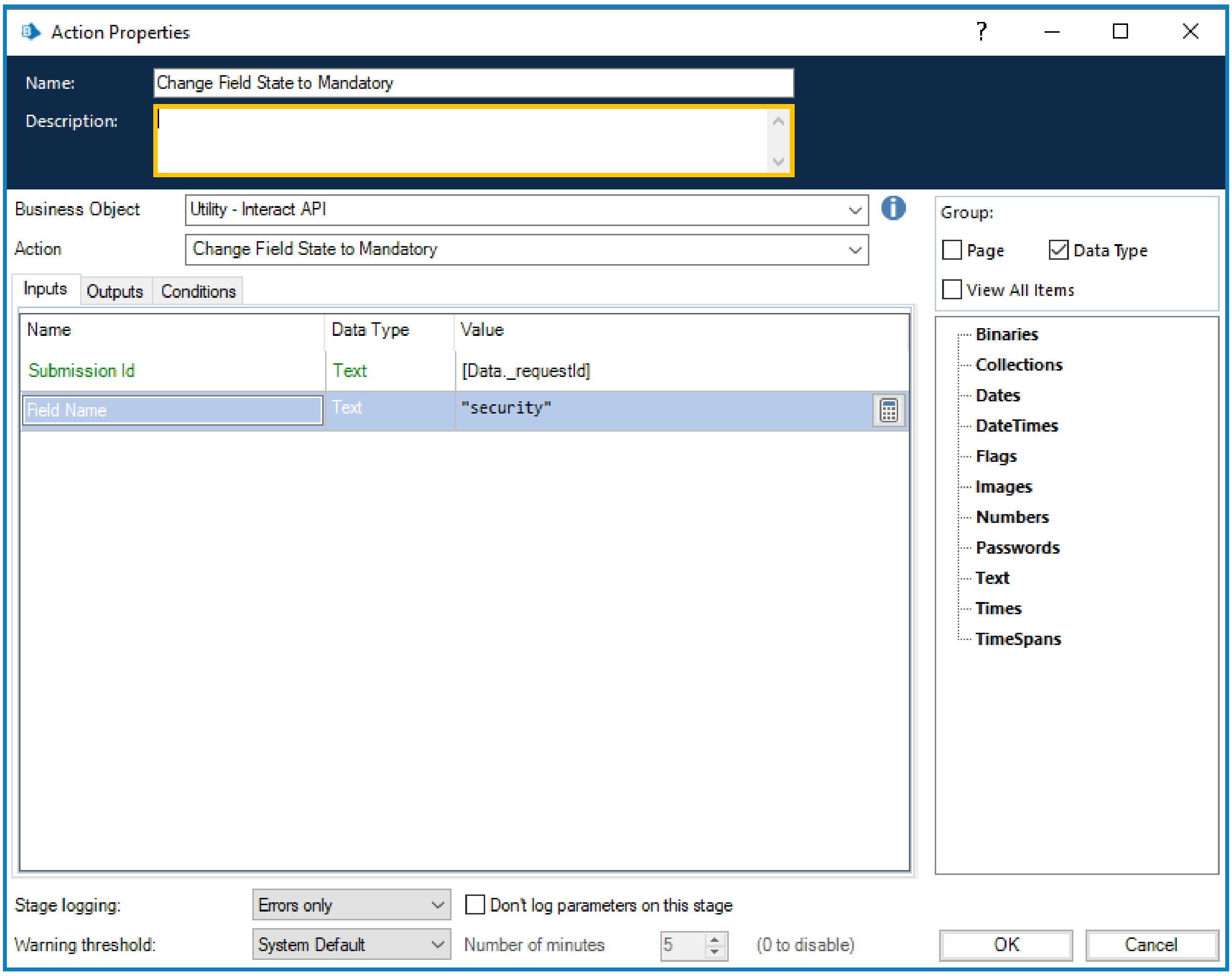
Warning threshold: System Default Number of minutes 5 (0 to disable)

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

Change Field State to Mandatory

The Change Field State to Mandatory action 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.



Action Properties

Name: Change Field State to Mandatory

Description:

Business Object: Utility - Interact API

Action: Change Field State to Mandatory

Group: □ Page ☑ Data Type □ View All Items

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

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

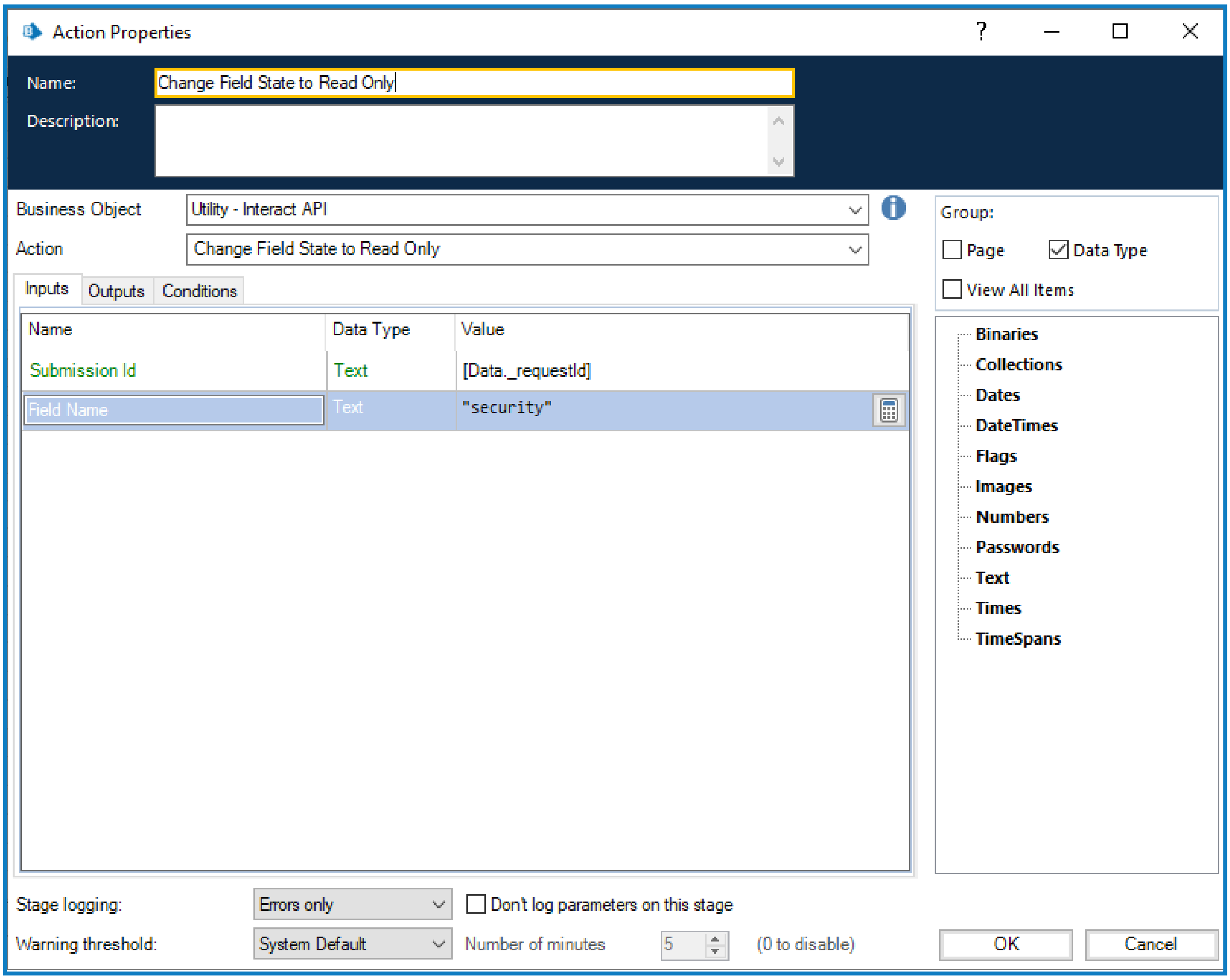
Warning threshold: System Default Number of minutes 5 (0 to disable)

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

Change Field State to Read Only

The Change Field State to Read Only action 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.



Action Properties

Name: Change Field State to Read Only

Description:

Business Object: Utility - Interact API

Action: Change Field State to Read Only

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

Stage logging: Errors only

□ Don't log parameters on this stage

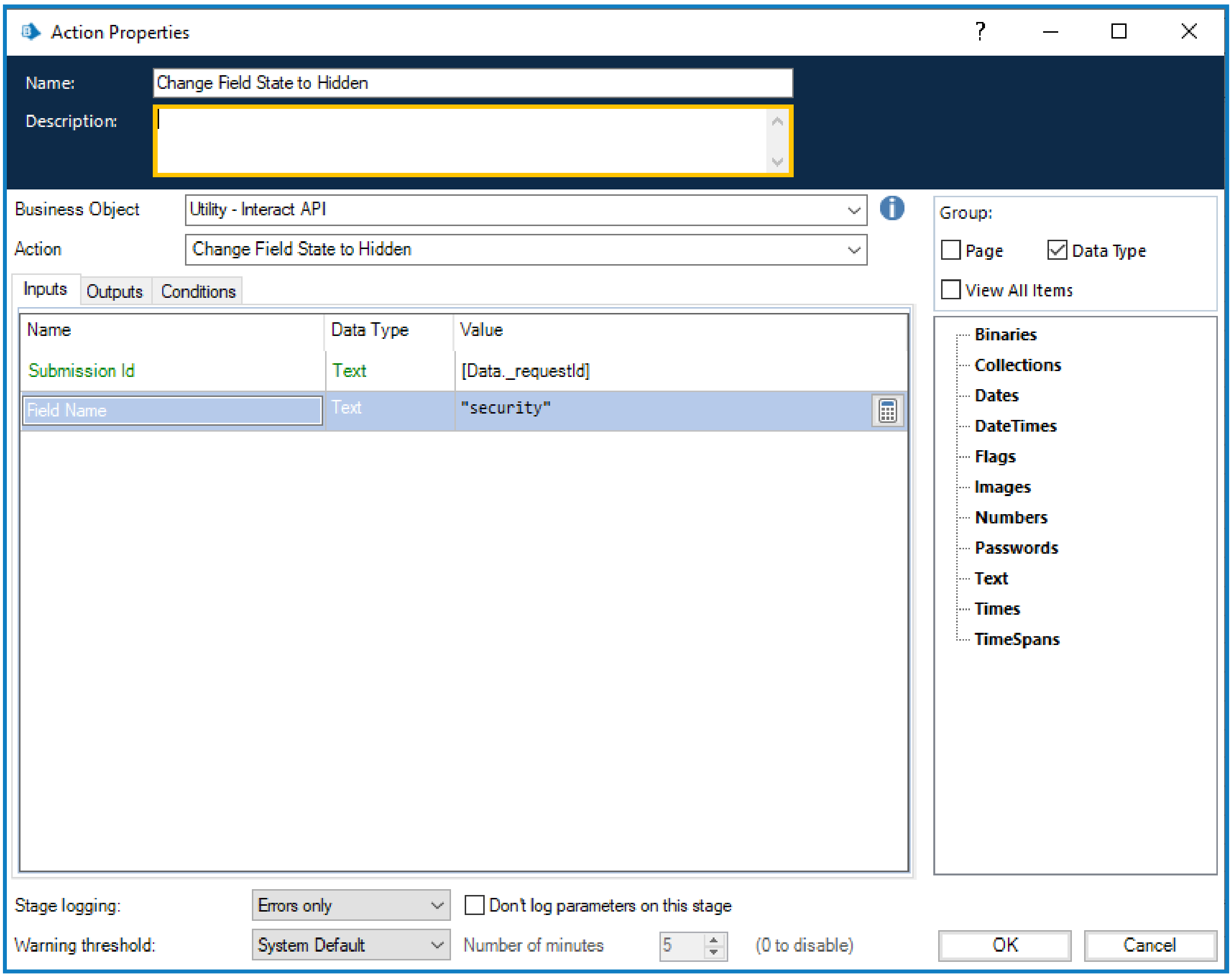
Warning threshold: System Default — Number of minutes 5 (0 to disable)

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

Change Field State to Hidden

The Change Field State to Hidden action 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.



Name: Change Field State to Hidden

Description:

Business Object: Utility - Interact API

Action: Change Field State to Hidden

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

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

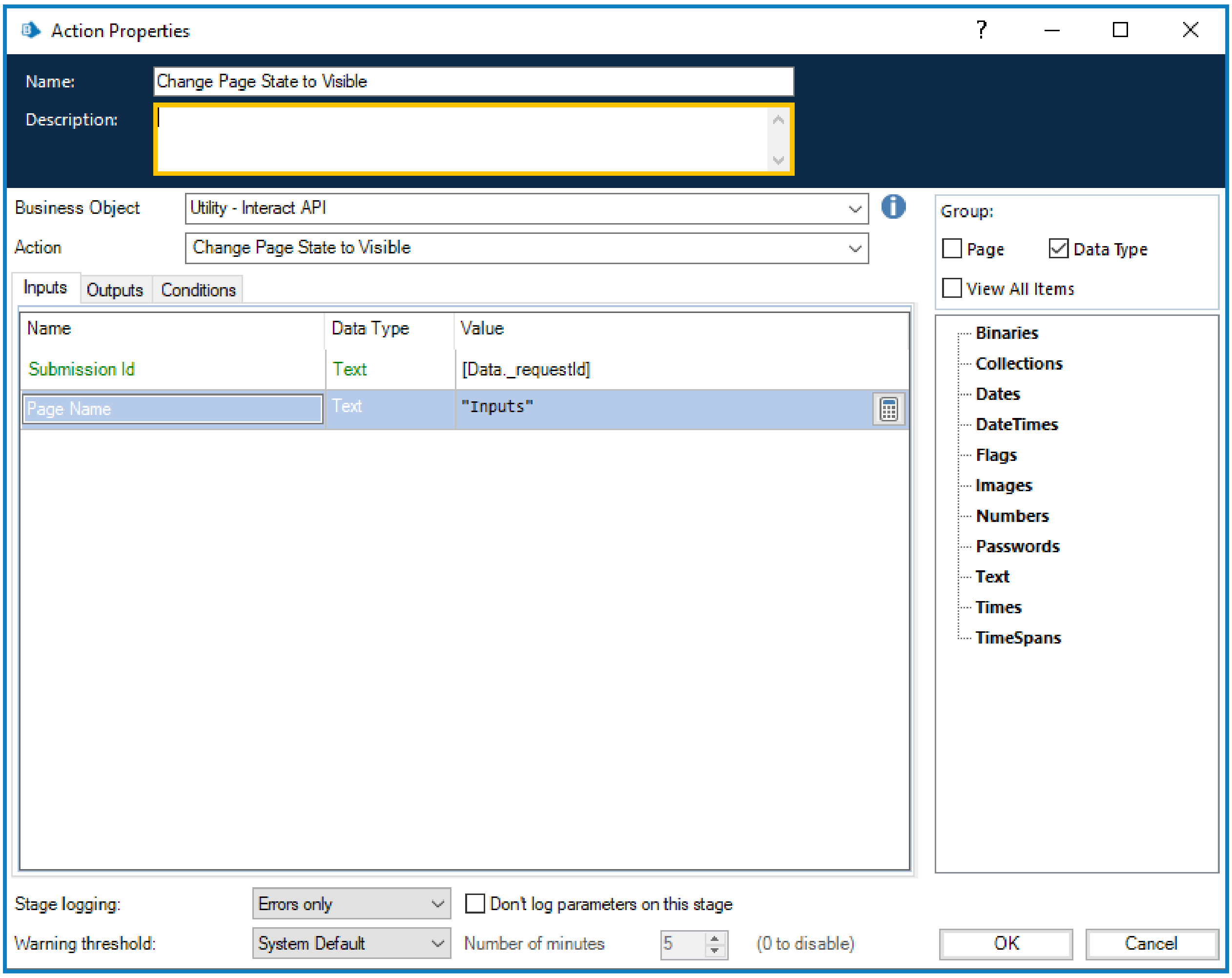
Warning threshold: System Default Number of minutes 5 (0 to disable)

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

Change Page State to Visible

The Change Page State to Visible action 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

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

Group: □ Page ☑ Data Type □ View All Items

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

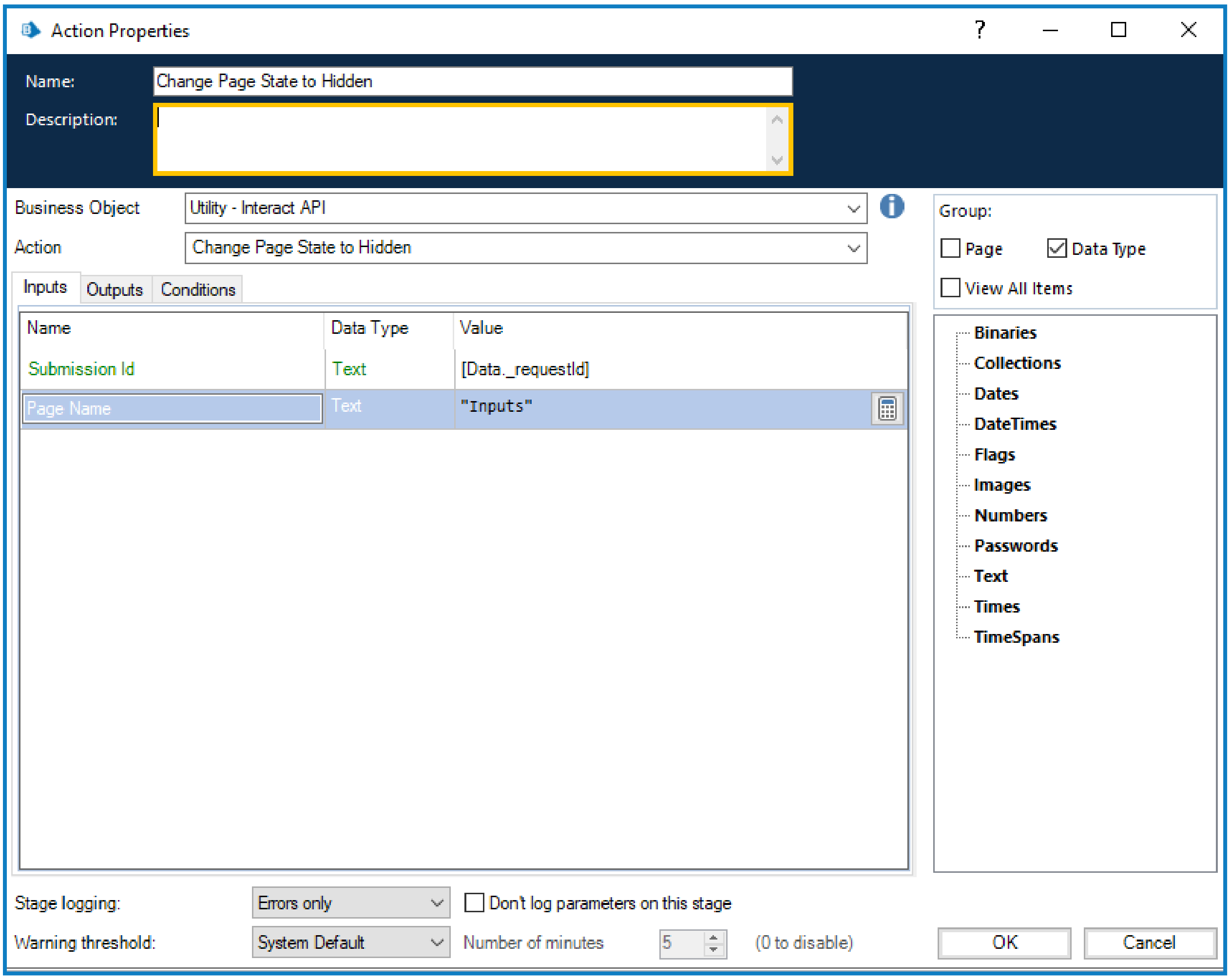
Warning threshold: System Default Number of minutes 5 (0 to disable)

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

Change Page State to Hidden

The Change Page State to Hidden action 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

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

Group: □ Page ☑ Data Type

□ View All Items

Stage logging: Errors only

□ Don't log parameters on this stage

Warning threshold: System Default

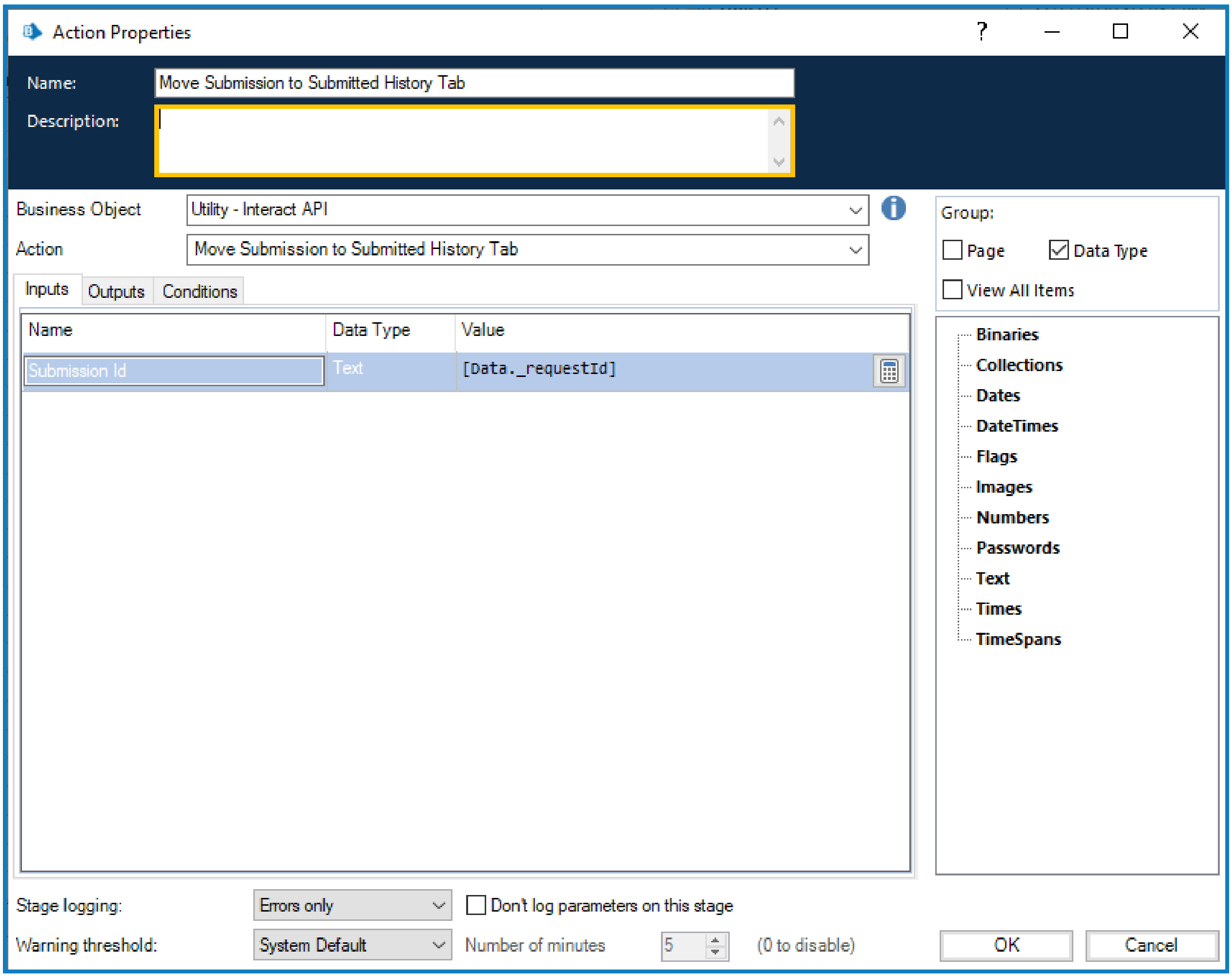
Number of minutes: 5 (0 to disable)

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

Move Submission to History Tab

The Move Submission to History Tab action 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.



Action Properties

Name: Move Submission to Submitted History Tab

Description:

Business Object: Utility - Interact API

Action: Move Submission to Submitted History Tab

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

Group: □ Page ☑ Data Type □ View All Items

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

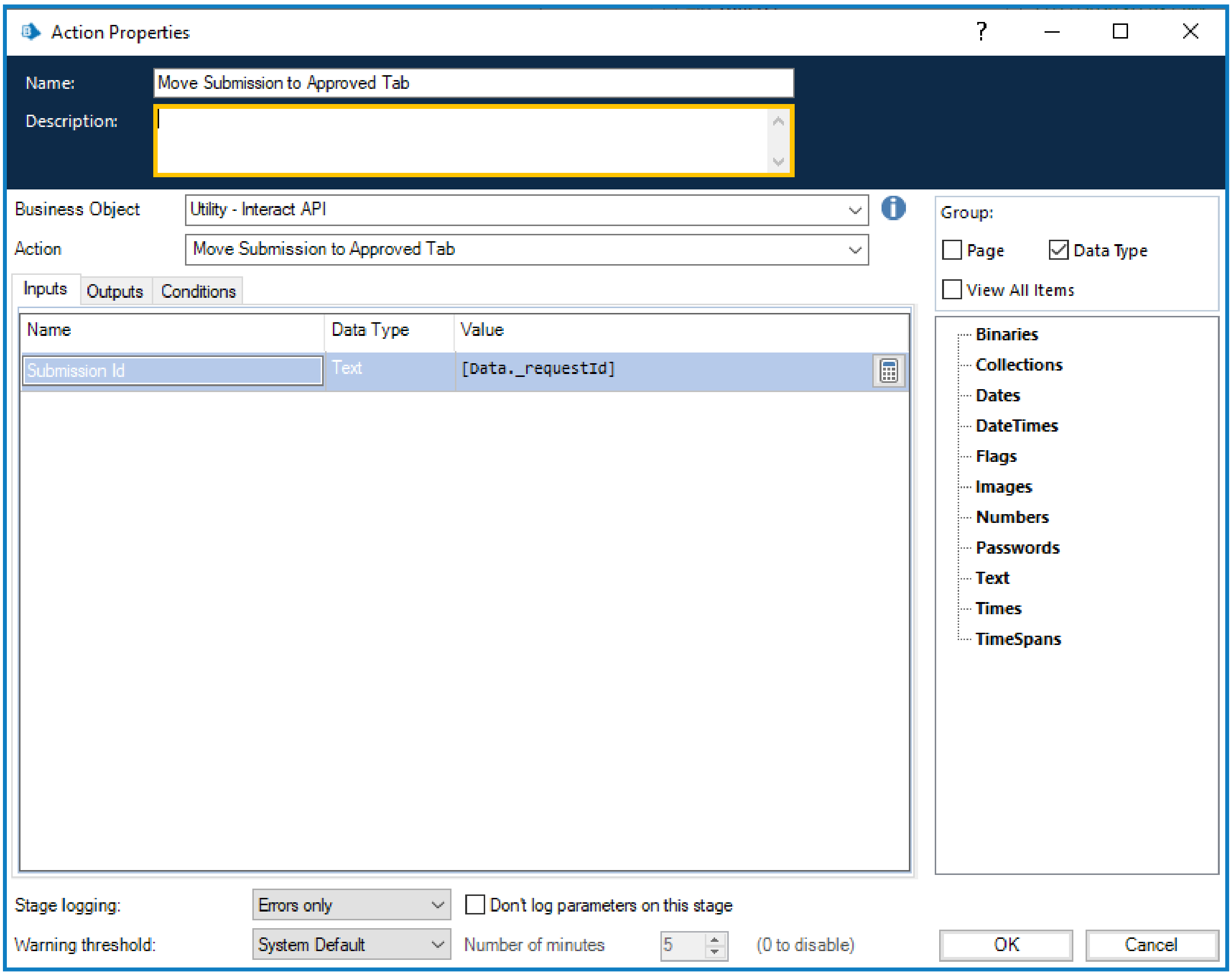
Warning threshold: System Default Number of minutes 5 (0 to disable)

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

Move Submission to Approved Tab

The Move Submission to Approved Tab action 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.



Action Properties

Name: Move Submission to Approved Tab

Description:

Business Object: Utility - Interact API

Action: Move Submission to Approved Tab

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

Group: □ Page ☑ Data Type □ View All Items

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

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

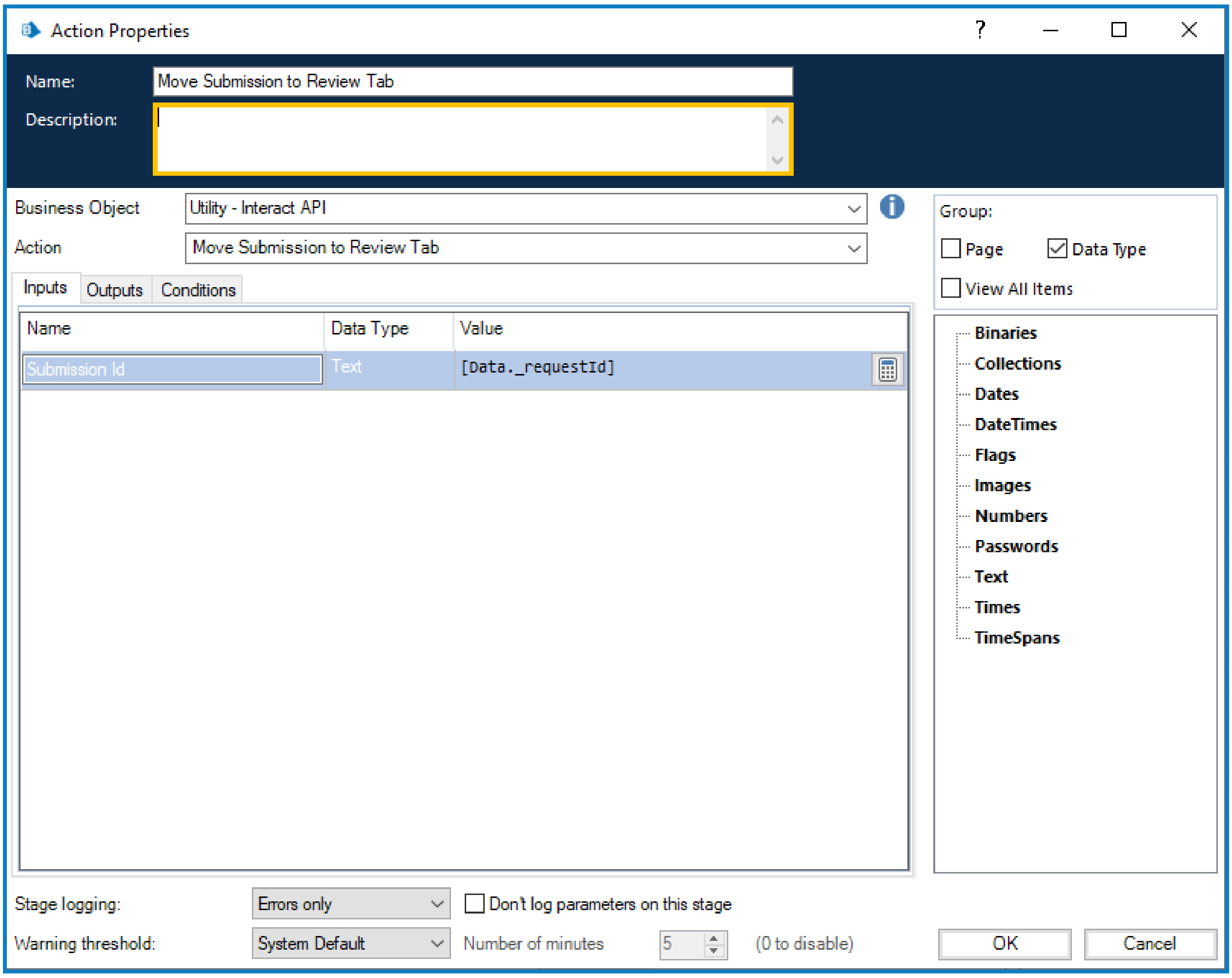
There are no outputs for the Move Submission to Approved.

Move Submission to Review Tab

The Move Submission to Review Tab action 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.



Action Properties

Name: Move Submission to Review Tab

Description:

Business Object: Utility - Interact API

Action: Move Submission to Review Tab

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

Group: □ Page ☑ Data Type
□ View All Items

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

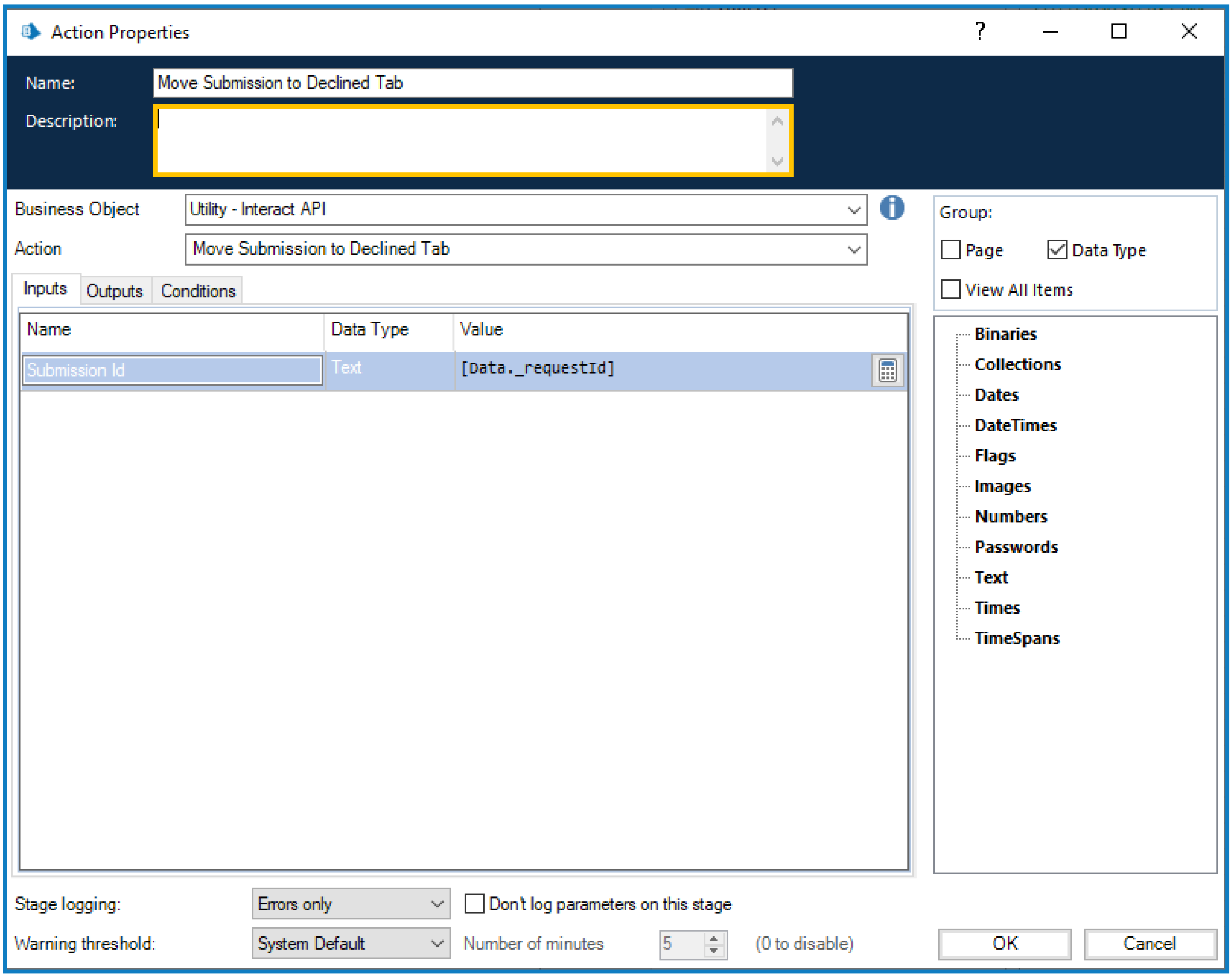
Warning threshold: System Default Number of minutes 5 (0 to disable)

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

Move Submission to Declined Tab

The Move Submission to Declined Tab action 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.



Name: Move Submission to Declined Tab

Description:

Business Object: Utility - Interact API

Action: Move Submission to Declined Tab

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

Group: □ Page ☑ Data Type □ View All Items

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

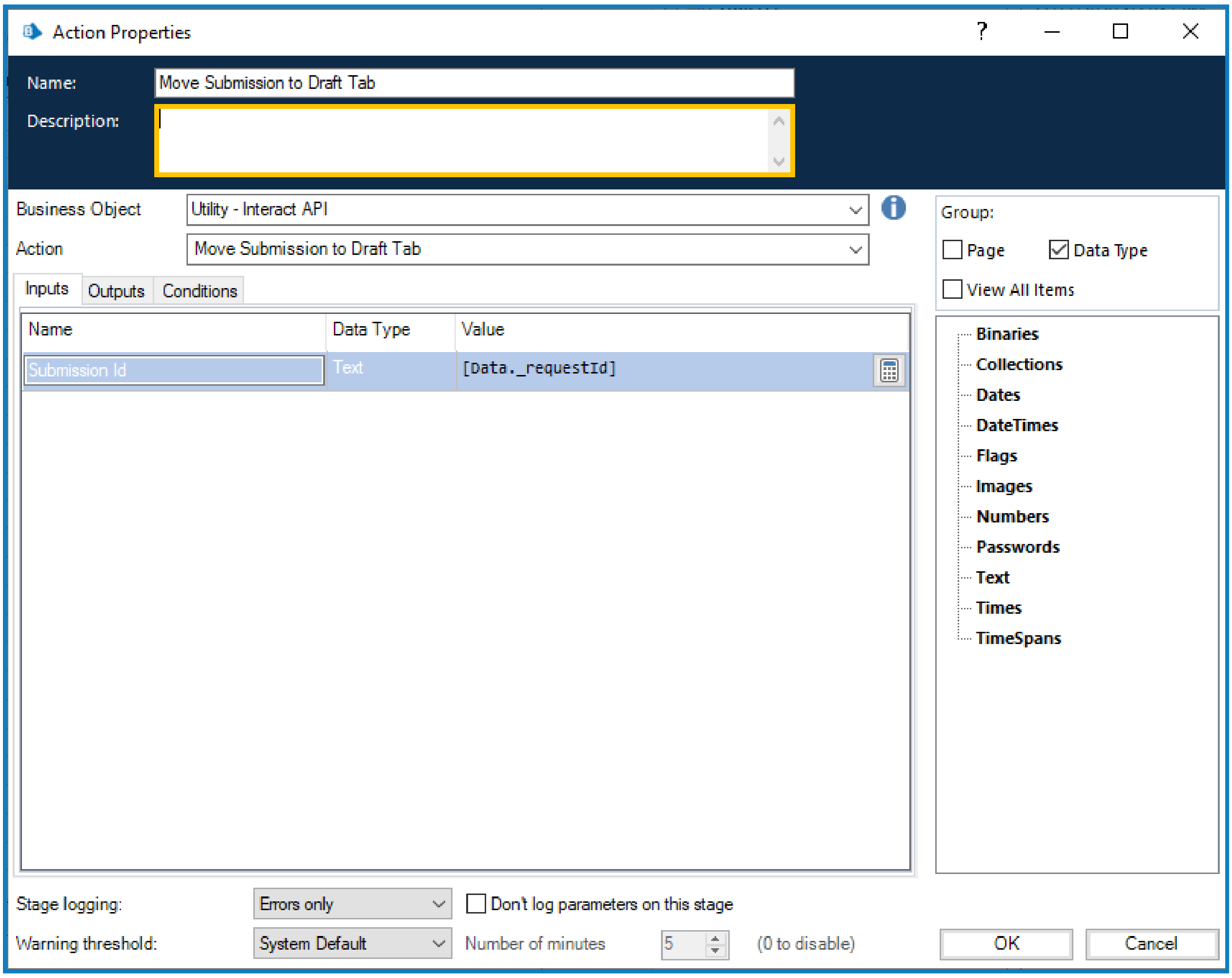
Warning threshold: System Default Number of minutes 5 (0 to disable)

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

Move Submission to Draft Tab

The Move Submission to Draft Tab action 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.



Action Properties

Name: Move Submission to Draft Tab

Description:

Business Object: Utility - Interact API

Action: Move Submission to Draft Tab

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

Group: □ Page ☑ Data Type
□ View All Items

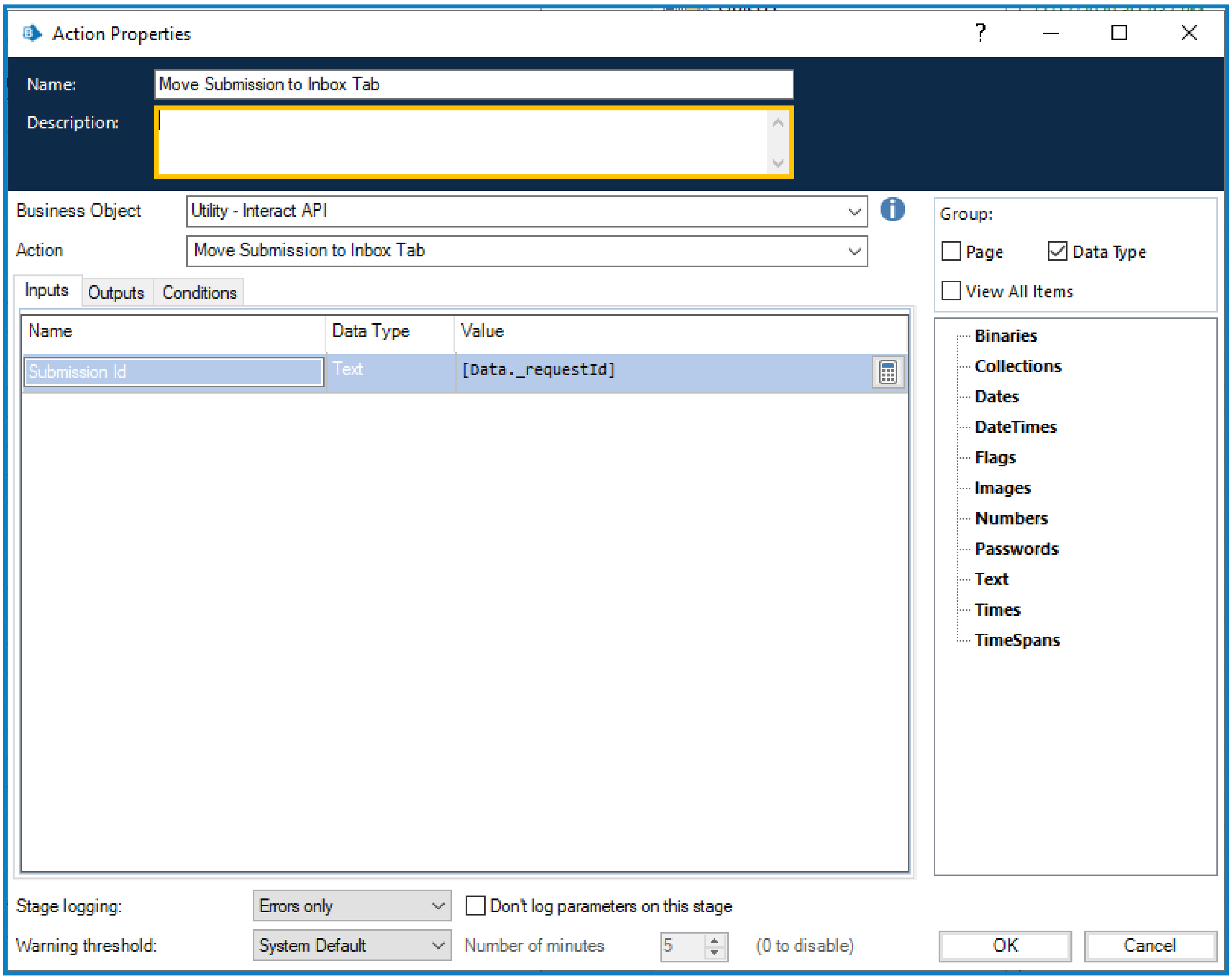
Stage logging: Errors only
□ Don't log parameters on this stage
Warning threshold: System Default
Number of minutes 5 (0 to disable)

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

Move Submission to Inbox Tab

The Move Submission to Inbox Tab action 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.



Action Properties

Name: Move Submission to Inbox Tab

Description:

Business Object: Utility - Interact API

Action: Move Submission to Inbox Tab

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

Group: □ Page ☑ Data Type
□ View All Items

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

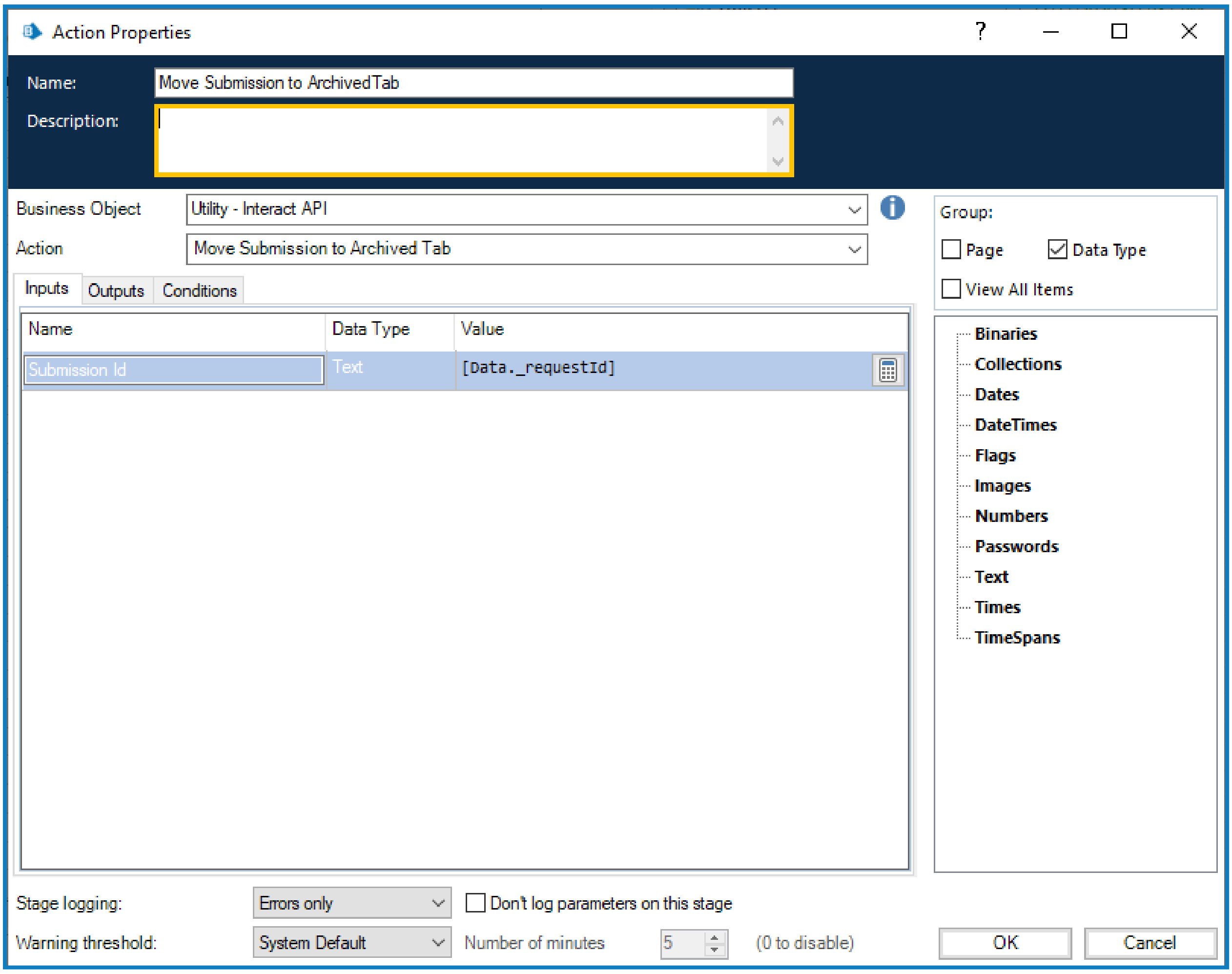
Warning threshold: System Default Number of minutes 5 (0 to disable)

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

Move Submission to Archived Tab

The Move Submission to Archived Tab action 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.



Action Properties

Name: Move Submission to ArchivedTab

Description:

Business Object: Utility - Interact API

Action: Move Submission to Archived Tab

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

Group: □ Page ☑ Data Type

□ View All Items

Stage logging: Errors only

□ Don't log parameters on this stage

Warning threshold: System Default

Number of minutes 5 (0 to disable)

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.

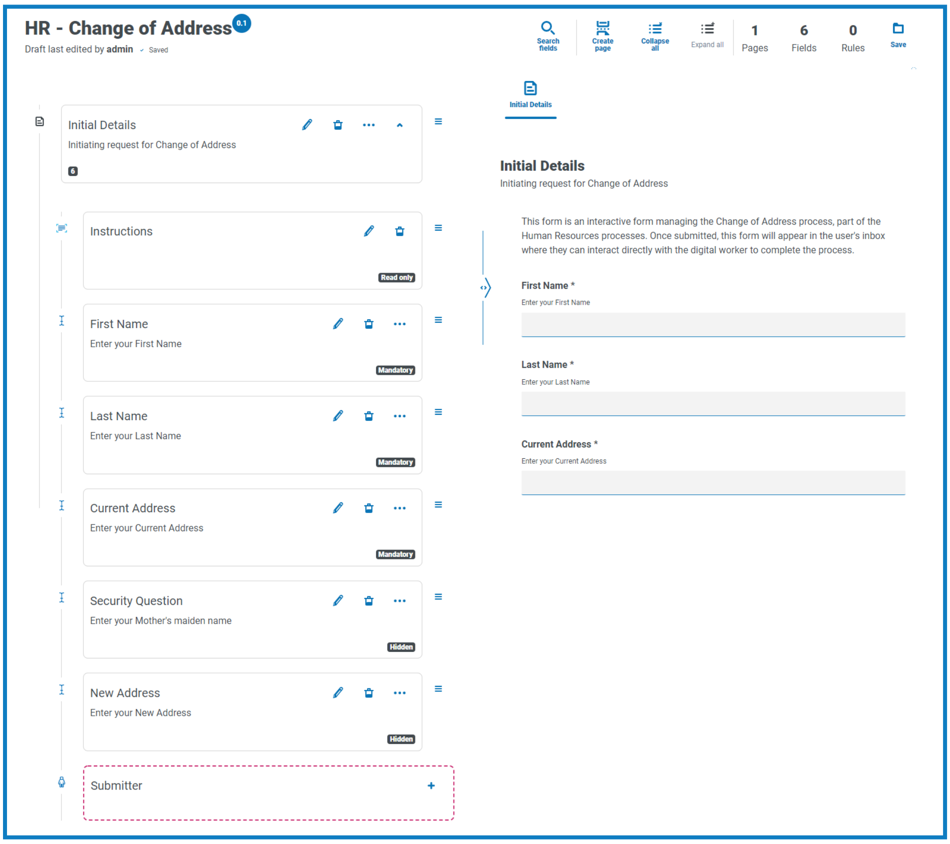
The steps on a high-level are detailed below:

- 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 initially captures the First and Last Names, as well as the current address. The other fields, as you can see in the testing area, are hidden. These are the security question and the new address fields.



HR - Change of Address 0.1
Draft last edited by **admin** Saved

Initial Details
Initiating request for Change of Address
6

Instructions
Read only

First Name
Enter your First Name
Mandatory

Last Name
Enter your Last Name
Mandatory

Current Address
Enter your Current Address
Mandatory

Security Question
Enter your Mother's maiden name
Hidden

New Address
Enter your New Address
Hidden

Submitter

1 Pages
6 Fields
0 Rules

Initial Details
Initiating request for Change of Address

This form is an interactive form managing the Change of Address process, part of the Human Resources processes. Once submitted, this form will appear in the user's inbox where they can interact directly with the digital worker to complete the process.

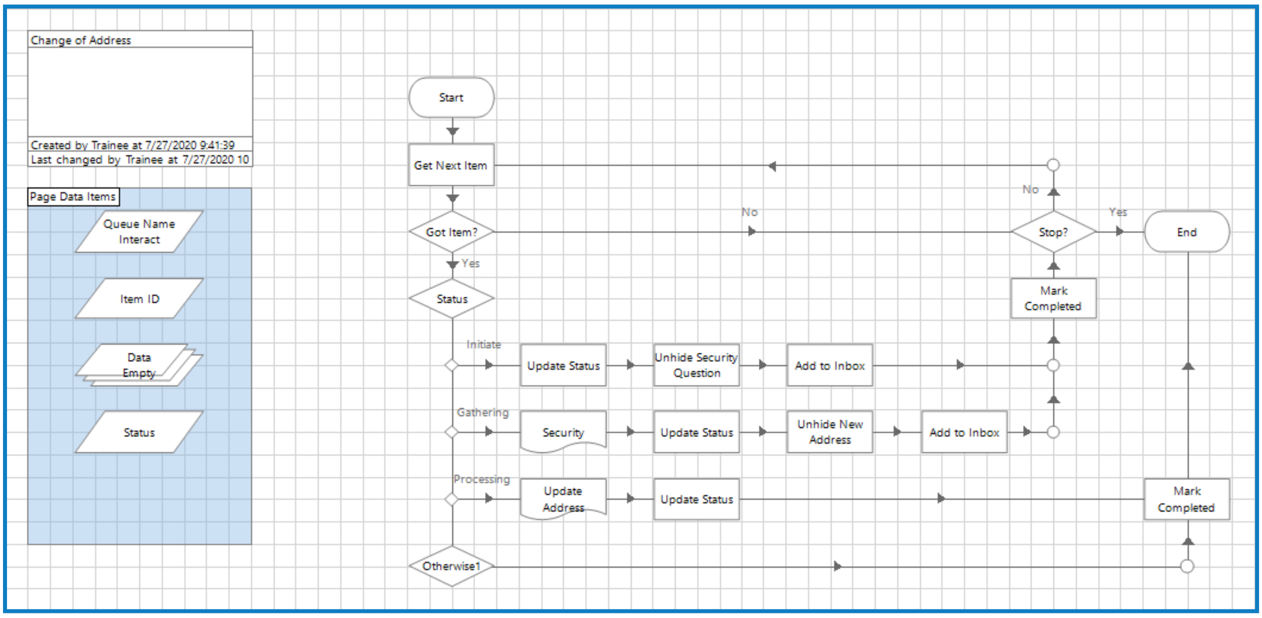
First Name *
Enter your First Name

Last Name *
Enter your Last Name

Current Address *
Enter your Current Address

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	Table	When editing a submission using the Web API Service, you must ensure that any inputs match the expected data type defined in the form, otherwise the update will fail.
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 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.