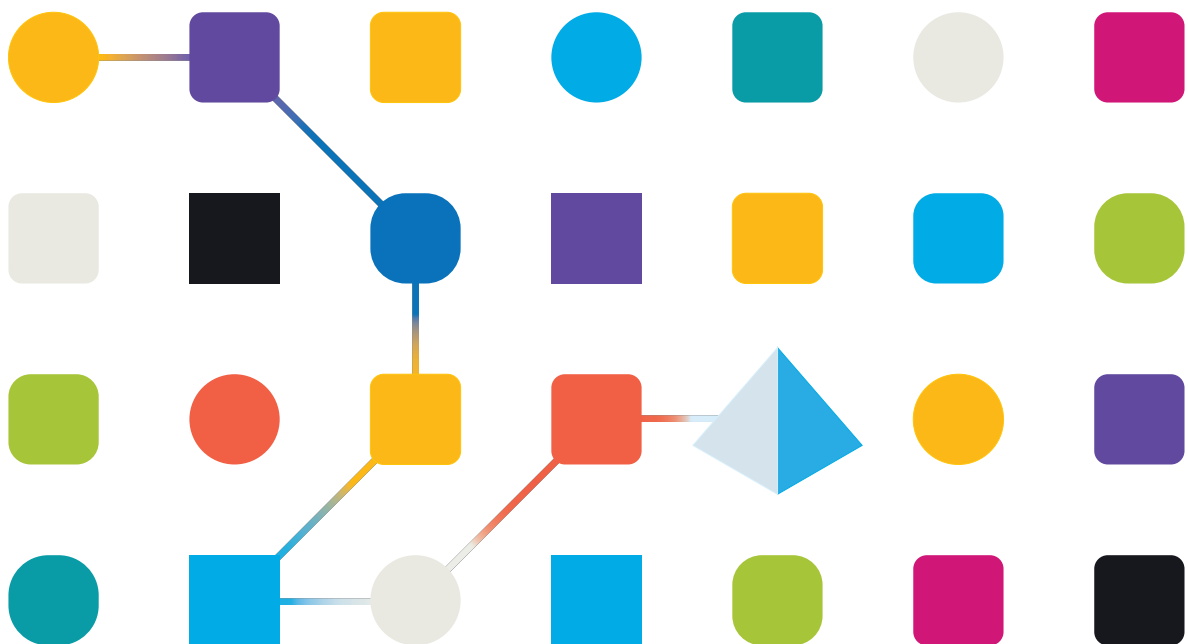


blueprism®

Blue Prism Cloud 3.0

User Guide

Document Revision: 1.0



Trademarks and copyrights

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.

© 2021 Blue Prism Cloud Limited. “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 help system.

Blue Prism Cloud Limited, 2 Cinnamon Park, Crab Lane, Warrington, WA2 0XP, United Kingdom.

Registered in England: Reg. No. 8534024. Tel: +44 870 879 3000. Web: www.blueprism.com

Contents

Trademarks and copyrights	2
IADA Orchestration – Standalone Edition	4
Overview	4
Scheduler	5
When to use the scheduler	5
Orchestration	6
What is orchestration?	6
Key definitions	6
How does orchestration work?	8
Orchestrator process	9
Designing for orchestration	11
Orchestration design	13
Collector process	13
Executor process	19
Orchestrator process	21

IADA Orchestration – Standalone Edition

This user guide covers Blue Prism Cloud Intelligent Automated Digital Assistant (IADA). This specific guide is for version 3.0 of the Blue Prism Cloud platform.

IADA allows automation developers to align business metrics to their varied workloads, driving priorities and service level agreements (SLAs) to determine the order that individual queue items should be executed in.

This guide outlines functionality and usage of IADA and how to apply orchestration across your workforce and into your automations. Whether this is existing automations or new automations.

This guide assumes that the user is familiar with, and has experience using, Blue Prism.

Overview

When we build process automation's we tend to only consider the process design/build at hand. We are taught and are proficient in being able to create a process automation to run well. When testing we test this single process automation for any errors or inconsistencies in performance.

What we are less likely to consider, or certainly spend not enough time considering, is how our process automation ecosystem will work as a whole; as a combined unit. An organization may employ a controller or task one of the developers with acting as a controller. The controller's role is to check on the automations, setup schedules and ensure that work items are being processed based on SLAs, priorities and demand. This is an inefficient use of resource and adds additional expense. Would it not be better to autonomously make decisions on what process automations to run and when to run them?

Sometimes, the perception of implementing the concept of orchestration is that there is a plug-and-play solution available or a solution that looks up against a dictionary of defined process rules. Unfortunately, to create a successful and effective solution which will return the desired benefits, there is a level of bespoke customization and design consideration that needs to be undertaken.

Scheduler

The scheduler available in Blue Prism is great for small to medium volume implementations of process automations. However, after a period of time you may find that the scheduler does not fully fit the need of your automations entirely and you may notice some limitations:

- Can I add additional complex logic to trigger schedule runs?
- Can I chain schedules together to remove general down time traditionally seen at the end of a scheduled process run?
- How can I react to changes in priority and SLA across queue items, without manual intervention?
- How can I get the more out of my digital workforce?

These, in addition to many more, are questions that clients and partners have asked us. A continued implementation of the traditional scheduler by itself may lead to a level of inefficiency, which in turn may stop your ability to grow scale without having to add further to your previous investment e.g. purchase more digital worker licenses when there is still more efficiency to be gained from the exiting workforce.

That being said, in some very specific circumstances the scheduler should be used exclusively or in conjunction with IADA orchestration, some of these are detailed below.

When to use the scheduler

There are instances where the scheduler is still the best solution for a given process:

- When it is imperative that a process needs to be run at a specific point in time and nothing can interfere with this;
- If a digital worker needs to be in a specific state (special permissions, access, abilities etc.) to run a process but other digital workers cannot be given the same access. In this situation it would be best to dedicate sections of the day (if not all day depending on the volumes associated with the process) specifically to schedule time for this process. An example of this may be when different windows accounts are required to run specific automation processes; and
- When software licensing restricts the digital workers all having the same software build installed on them.

When looking at utilizing orchestration it is imperative that each process should be assessed individually and a decision made as to whether or not it is fit to be included within orchestration or to use the traditional Blue Prism scheduler. Ultimately the best way to think of orchestration is as another tool to be utilized in conjunction with the scheduler; not to replace it in all instances.

Orchestration

The way Blue Prism Cloud utilizes orchestration is by leveraging the IADA service within a Blue Prism process called Orchestrator, the Orchestrator process will perform all the logic and intelligent decision making on which process to run; and when to run that process. The Orchestrator process will be running throughout the day, all day (aside from a small automation maintenance window), consuming all your digital workforce or as much as you want to allow it to utilize.

Implementing orchestration will allow you to use more advanced decision making, leveraging Blue Prism Cloud IADA service and Orchestrator process will ultimately allow you effectively utilize your Blue Prism Cloud Digital Workforce in the best way possible.

What is orchestration?

As we explained, within an organization not all process automations and their associated items have the same value to the organization, some business processes need to be prioritized over others. By default, items will simply be addressed and processed in a First-In-First-Out (FIFO) methodology.

This combined with the fact that there can be multiple queues with the Blue Prism Cloud Digital Workforce that need to be managed, means that work can get stalled in handling an individual queue or stuck processing automations with work items which are of a lower value.

IADA maintains, in effect, its own view, looking across not a single queue but all the queues and assessing assigned values (such as business priorities and SLAs), it will determine the next automation item that needs to run, whilst queuing the next best value item ready for execution by the next available digital worker.

For an automation to be given a priority over another automation, it has to be configured with a business priority and an SLA. There are multiple ways of adding these parameters into your automations and we will show you how in more detail later. This is the key element so when items are added to the work queue, the business metrics are added along with the process.

Key definitions

For clarification, let's define some key terms we use throughout this user guide:

Business priority

The business priority value must be above zero. The lower the value, the higher the business priority (e.g. a business priority value of 1, will be processed before an item with a business priority value of 2). This business priority value will override the SLA value.

IADA accepts a business priority value between 1 to 999.

Business SLA

The business SLA value is a timespan. This should be the amount of time that the work queue item should be completed and processed within. The SLA timespan value will then be used within the IADA algorithm to produce an SLA Datetime.

IADA accepts a business SLA in the format of DDD.HH:MM:SS (Days, Hours, Minutes, Seconds) e.g. 01.04:30:00 = 1 Day, 4 Hours, 30 Minutes, 0 Seconds. The minimum SLA is 1 second, the maximum SLA is 999 Days, 23 Hours 59 Minutes and 59 Seconds.

Environment locks

In order to ensure that there are no conflicts when reading from or writing to the queue item, environment locks are used.

Environment locks are used quite heavily within orchestration, as they are key to adding a control mechanism for decision making when getting the 'Get Next Best Item' to be worked via IADA.

Tag identification

This is a unique GUID that is assigned to a work queue item by the 'Get Next Best Item' function to allow the Executor function to correctly identify the work queue item for processing.

SLA timespan

If the Priority assigned to two items is the same, then the timespan before the SLA expires is used to determine the order in which items are processed. This is calculated from when the item is added to the queue and the current time.

For example, two items added to a queue with the same priority and different SLAs:

- Item A – Priority 10 – SLA 00:01:00:00 (1 hour) – added to queue at 09:30;
- Item B – Priority 10 – SLA 00:01:30:00 (1 hour 30 mins) – added to queue at 08:45.

If the current time is 10:00, then Item A has 30 mins remaining on its SLA timespan whereas Item B has only 15 mins remaining on its SLA timespan. Therefore, even though Item B has a longer SLA, it will be processed before Item A.

Whitelists and blacklists

IADA orchestration supports the use of whitelists and blacklists for the processes and queues used within orchestration.

The values set determines the action that the 'Get Next Best Item' function within the Orchestrator process performs, for example for the queue lists:

Flag State	Content	Description
True	Queue A and Queue B	Orchestration will only look for items with a Tag Identifier in Queue A and Queue B.
True	empty	Orchestration will look across all Queues for items marked with a Tag Identifier.
False	Queue A and Queue B	Orchestration will look across all items with a Tag Identifier in all queues, except for items in Queue A and Queue B.
False	empty	Orchestration will look across all Queues for items marked with a Tag Identifier.

The flags and the queue or process lists in the table above, are inputs to the IADA service and so control how the IADA 'Get Next Best Item' function finds data items.

The process lists work in a similar manner as the queue lists. Either only working on the process names listed as true, or ignoring defined processes if the flag is set to false.

How does orchestration work?

Before looking at the Orchestrator process operation, it is good to have a basic understanding of IADA and how the service is called.

IADA service

The IADA service is delivered as an Azure 'function app' within your Azure subscription. This is to ensure security and integrity of service.

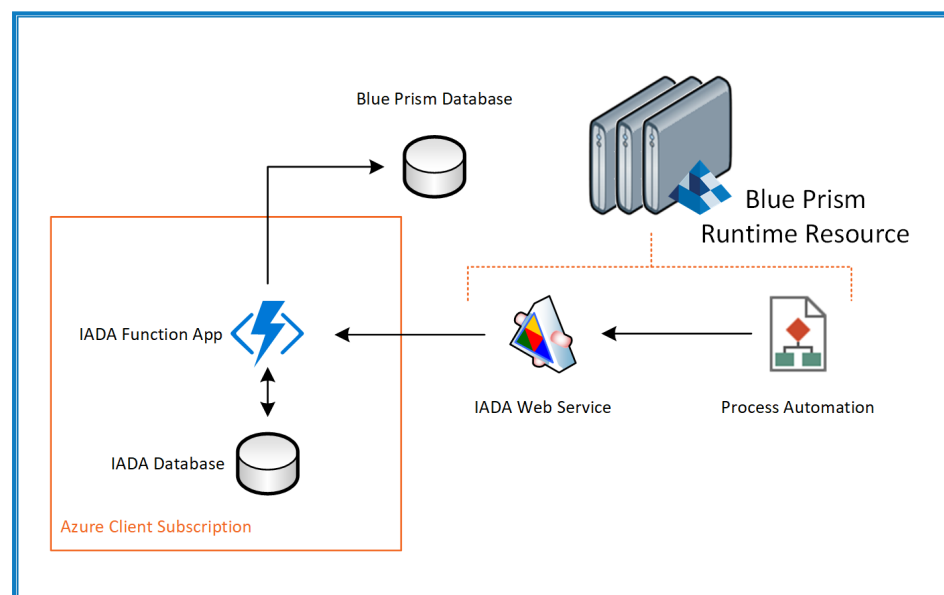
The Azure function app is built with two standard functions:

- Add to Queue
- Get Next Best Item

These functions are called via a Blue Prism RESTful web service which interacts with the IADA function app.

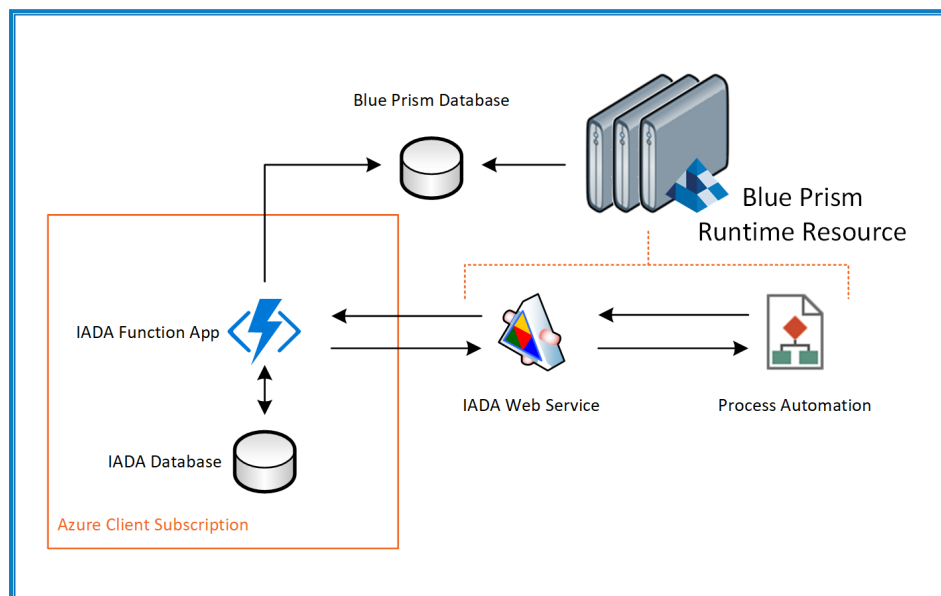
The IADA function app stores information relevant to orchestration in a dedicated IADA database. This database is an Azure SQL database held within your subscription.

The diagram below illustrates a typical data flow when using the 'Add To Queue' functions.



When the 'Add To Queue' function is called, the data item is added into the IADA database as well as the Blue Prism RPA database where it can be seen in the relevant Blue Prism queue.

To improve the performance of IADA orchestration, the 'Get Next Best Item' function reads from the IADA database to look for the next item most suitable for processing. This takes the orchestration load off the Blue Prism database. The Blue Prism queue is updated to include the Tag Identifier and mark it as complete once the item has been processed. This is illustrated in the data flow for 'Get Next Best Item' function below.



Orchestrator process

In a typical environment, the Orchestrator process would run continually on some digital workers, whilst other digital workers will be dedicated to performing the Collector process. Typically, the Orchestrator process runs and operates the same way across the digital workforce environment, except for those specified as Collectors.

It is recommended that the digital workers running the Orchestrator process are identical, with each digital worker capable of running every process.

The dedicated Collector digital workers use the 'Add To Queue' function to load work items into their respective queues.

1. The 'Add To Queue' function adds items to queues one by one.

This is different to the Blue Prism 'Add To Queue' which performs this task in bulk. The reason for the singular approach is to allow an item to be added with a specific priority and SLA rather than all items with the same priority and SLA.

If the item is added to a queue using the standard Blue Prism 'Add To Queue' it will NOT get written into the IADA database and therefore will not be considered for orchestration. Only items added into work queues using the IADA 'Add To Queue' function or via Interact with a defined priority and SLA will be considered for orchestration.

2. The Orchestrator process performs several steps:
 - a. It acquires a lock for a digital worker.
 - b. It runs the 'Get Next Best Item' function to identify the work item most suitable for processing, determined by the highest priority and the lowest SLA timespan.

The 'Get Next Best Item' function builds a collection using the information retrieved, with the following three parameters; Process Name, Queue Name and Tag Identifier. The Tag Identifier is a GUID, which is unique to the item, to mark it for execution. This GUID is also applied to the Blue Prism queue item to show the specific item within the specific queue during processing. The environment lock on the digital worker is then released.

3. The Orchestration control page of the process takes the built collection, acquires another environment lock and checks to see if the process to be called has just been run and is therefore still active. This is to save time in relaunching applications if they are still active and can be reused.
4. The Orchestrator process moves to the Process Control page to run the required process, releasing the environment lock before calling the Executor for that process. The Tag Identifier and the Queue Name is passed to the Executor from the built collection ensuring that the specific item with the highest priority and lowest SLA timespan is executed.
5. After the item is processed the GUID (Tag Identifier) is removed from the queue item and the item is marked as complete.

If the execution of the process results in a business exception, this should be handled within the actual business process and the item in the queue marked with an exception rather than completed.

Within the Orchestrator process there is a dedicated page to manage 'consecutive system exceptions'. This page provides a counter for when networking or failed digital workers cause an exception to the running of the process. By default, the maximum limit counter is set to three. The counter is set at '0' and incremented when a consecutive exception occurs. If there are three consecutive exceptions of the same system exception, the Orchestration process is halted.

Whilst this is happening on one digital worker with its own Orchestrator process, other assigned digital workers with their respective Orchestrator processes will be performing the same task on work items, ensuring they are getting processed in priority and SLA order.

The Orchestrator process continually runs and will continue to check to see whether there are items with an associate priority or SLA that needs to be processed.

As the Orchestrator process continually runs, it is recommended that the Orchestrator process is 'paused' once a day to allow maintenance work to be performed on the digital worker. This could include items such as reboots of digital workers and upgrades.

It is recommended that the Orchestrator process is run across specific digital workers within the workforce, whilst other digital workers are allocated the task of Collector or set aside to perform specific scheduled tasks. If a process has to be run at a specific time and is scheduled, it should not be included in the Orchestrator process and dedicated digital workers should be assigned to perform scheduled tasks.

Items will only be processed by the Orchestrator process, even if they exist in the IADA database, if the Queue and Process lists apply. See the definition on [Whitelists and blacklists on page 7](#) for more details.

Adding priority and SLA to queue items

There are several ways to add the priority and SLA to queue items. The main way is discussed in the section above, which is the 'Add To Queue' IADA function.

When using the 'Add To Queue' function you must stipulate a Priority and SLA, you cannot supply 'null' values. So this action cannot be used to add an item to a queue without a priority and SLA assigned.

The other main way to add items to the queue with priority and SLA is through Interact. When a form is created within Hub, you can stipulate the priority and SLA that the form will use when submitted.

Within the form creation you can decide whether to include these parameters or not. If they are:

- Not included, the submission is not added into the IADA database just the Blue Prism queue.
- Included, the submission is added to both the IADA database and the Blue Prism queue.

The 'Find Next Best Item' function will only consider items in the IADA database and items only get added into this database when they are assigned a priority and SLA.

Designing for orchestration

When implementing orchestration, it is common to feel apprehensive when considering how to approach the design, and it is sometimes thought that complex cognitive systems and Artificial Intelligence (AI) are required to complete the design.

Although cognitive systems like Machine Learning (ML) and AI implementations are important, it should be a secondary consideration after implementing a Logical and Algorithmic design. These technologies need to overlay a base foundation of logical decision making.

Logical decision making

Once you break down the requirements, you can start to understand what checks the process might want to perform when deciding whether an automation is suitable for orchestration. You should check each process in turn and consider the following:

- Does it need to run at a specified time?
 - This is important and should be one of the first checks that is performed. If the process automation has to be run at a specific time of day, and must be run at that time, then it may not be suitable for orchestration.
- What is its priority?
 - Where does this fit within the priority list compared to other process automations. Planning the priority list is a key factor to ensure that processes are added in the correct business order. We recommend that you do not set priorities as 1, 2 and 3, but start maybe in 10's or even in 100's. So, setting the highest priority with a 10 for example now, will allow you to still add a higher priority item at a later date without needing to adjust existing processes.
- What is its SLA?
 - Does the process, once initiated, need to be completed within a specific time frame? If so the SLA should be set to ensure that the item is added to a queue with a defined timeframe. This also needs to be in conjunction with an assigned priority to ensure that it gets the required attention.
- Are there processes and queues that either need to be specifically included or excluded?
 - Plan the whitelists or blacklists appropriately. Considering whether orchestration is only going to run on specific queues or all queues, or on all queues but a specific few. Then set the queue list appropriately to meet the business requirements. Repeat this for the list of processes to ensure that processes are either included or excluded from orchestration as appropriate.

- Can the business process be segregated into launch/login and execution?
 - For each business process added into orchestration, we need to look at whether the automation process can be segregated into a launch/login step and a standard execution stage. This will enable more efficient processing, especially if the application takes a long time to launch. The Orchestrator process will look to see if the last application it ran is still running and then will use that application without needing to relaunch and login back in again. If it is not, the same process it will terminate the previous application and launch the new application.

The same level of understanding and questioning should be performed for all processes to check suitability for orchestration.

Orchestration design

There are three processes within the orchestration environment; [Collector](#), [Executor](#) and the main [Orchestrator](#) process. These are provided as templates and will need to be modified to handle your organization's requirements and automations.

Collector process

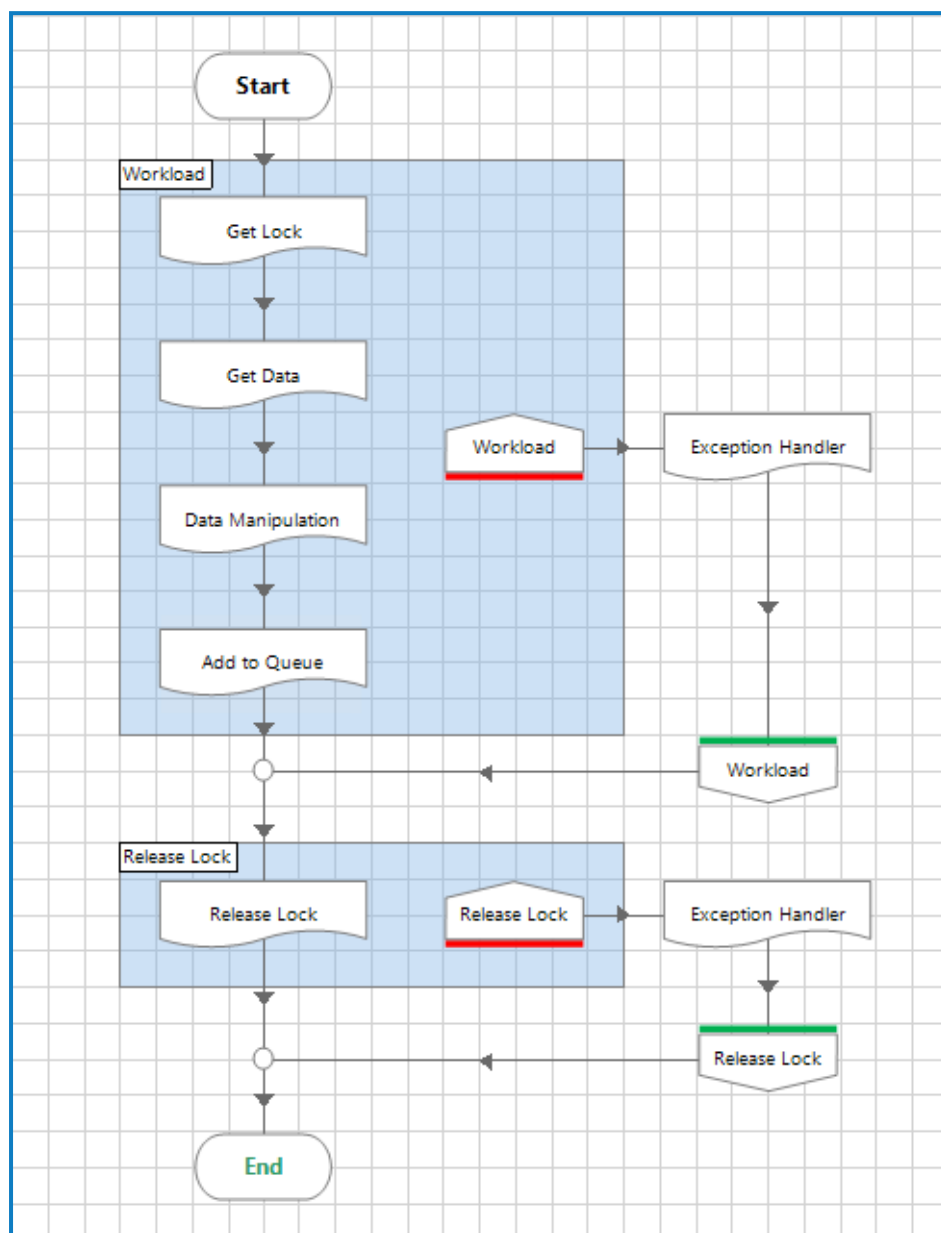
The Collector process template has to be adjusted to load the data into the specific queue for the business process with the required priority and SLA.

A Collector process is required for each business process. There are a number of pages in the Collector template that need to be modified to support your business process, these are described in detail below.

Remember that the 'Add To Queue' function in IADA adds a single item at a time to a queue.

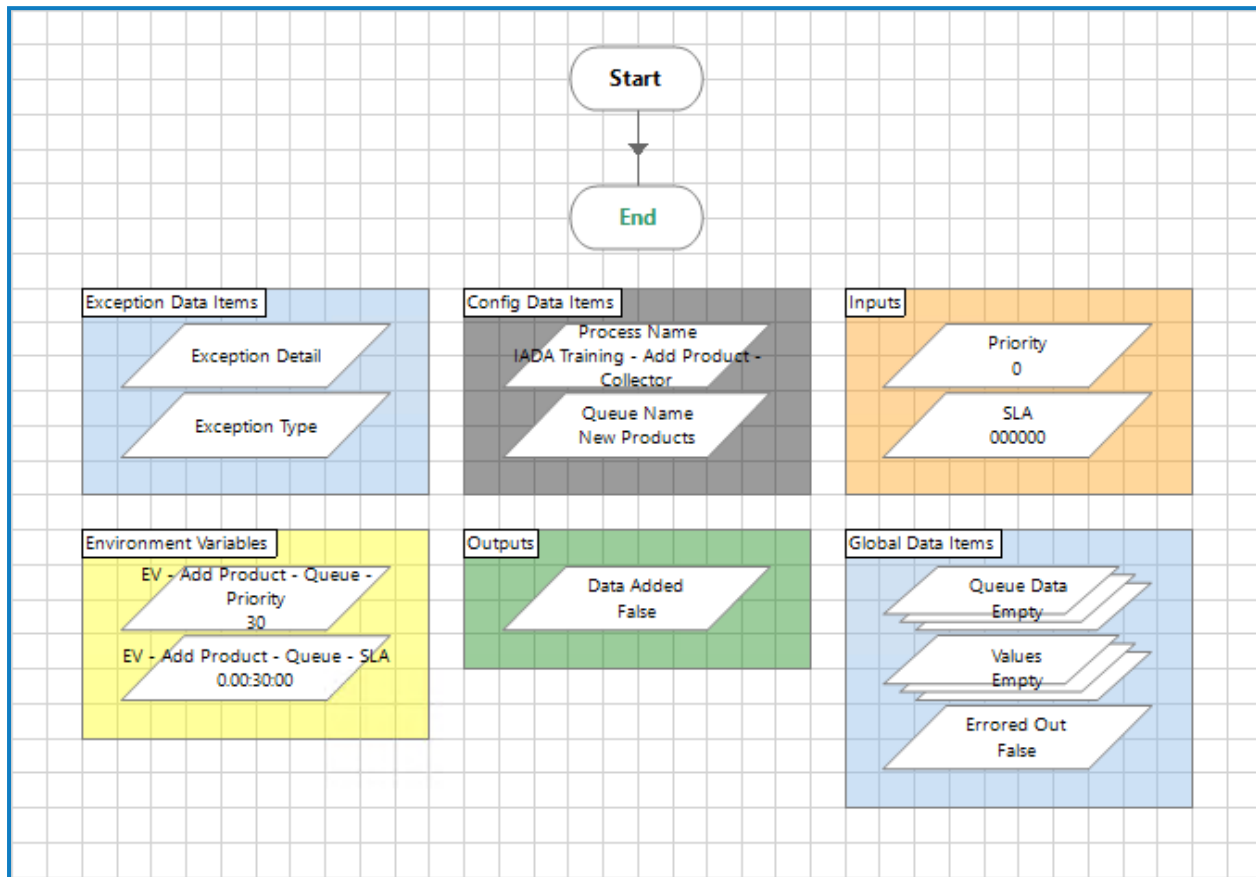
Main page

The main page invokes the Collector process, as illustrated below. This page calls the other pages and does not require any changes to support your business process.



Globals page

The Globals page is where the environment variables are called for the priority and SLA and where the process name and queue name are configured, as illustrated below.

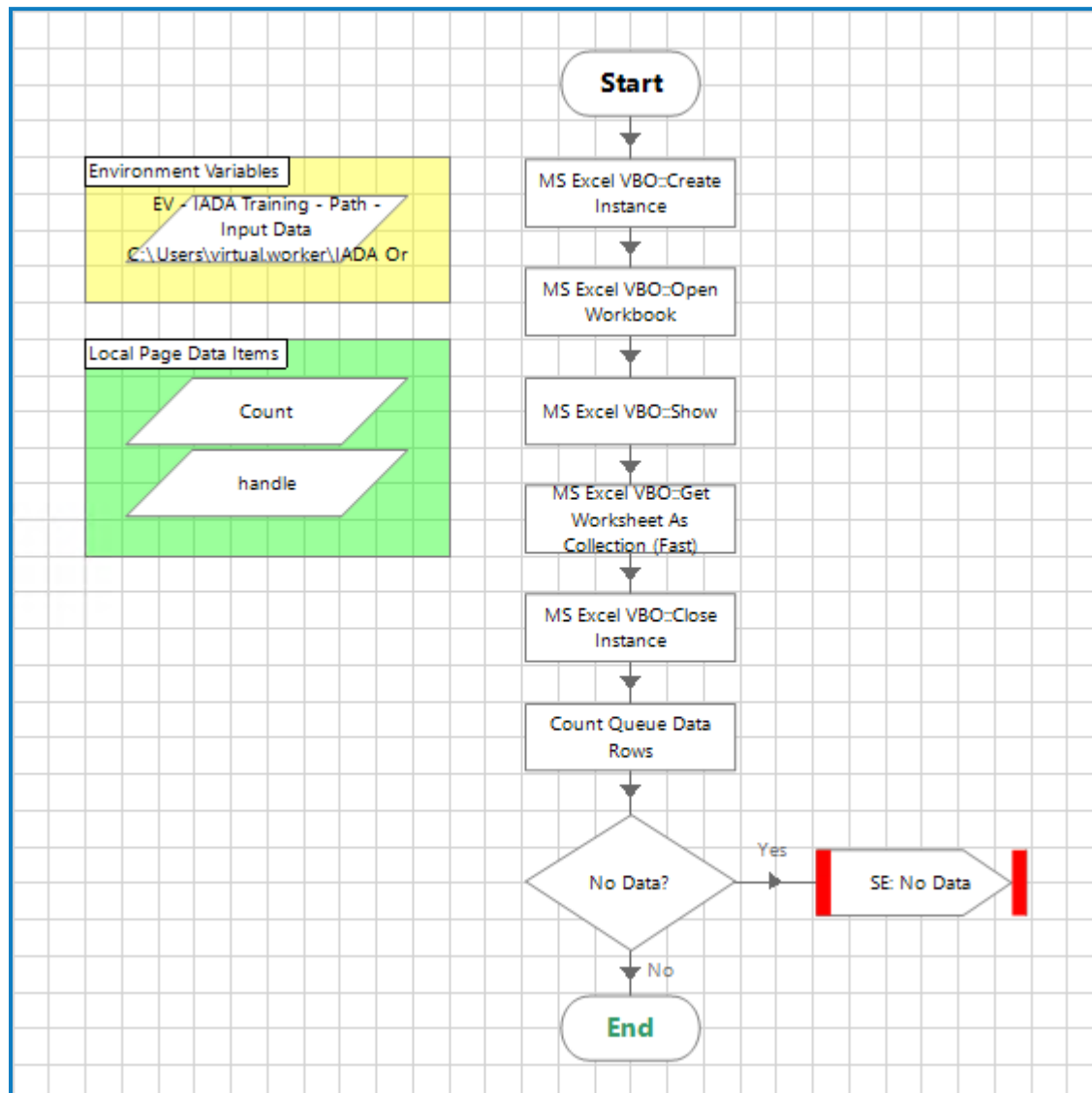


In the example above, the two environment variables are configured from system defined environment variables.

The 'Process Name' and 'Queue Name' need to be defined appropriately for the process you are configuring.

Get Data page

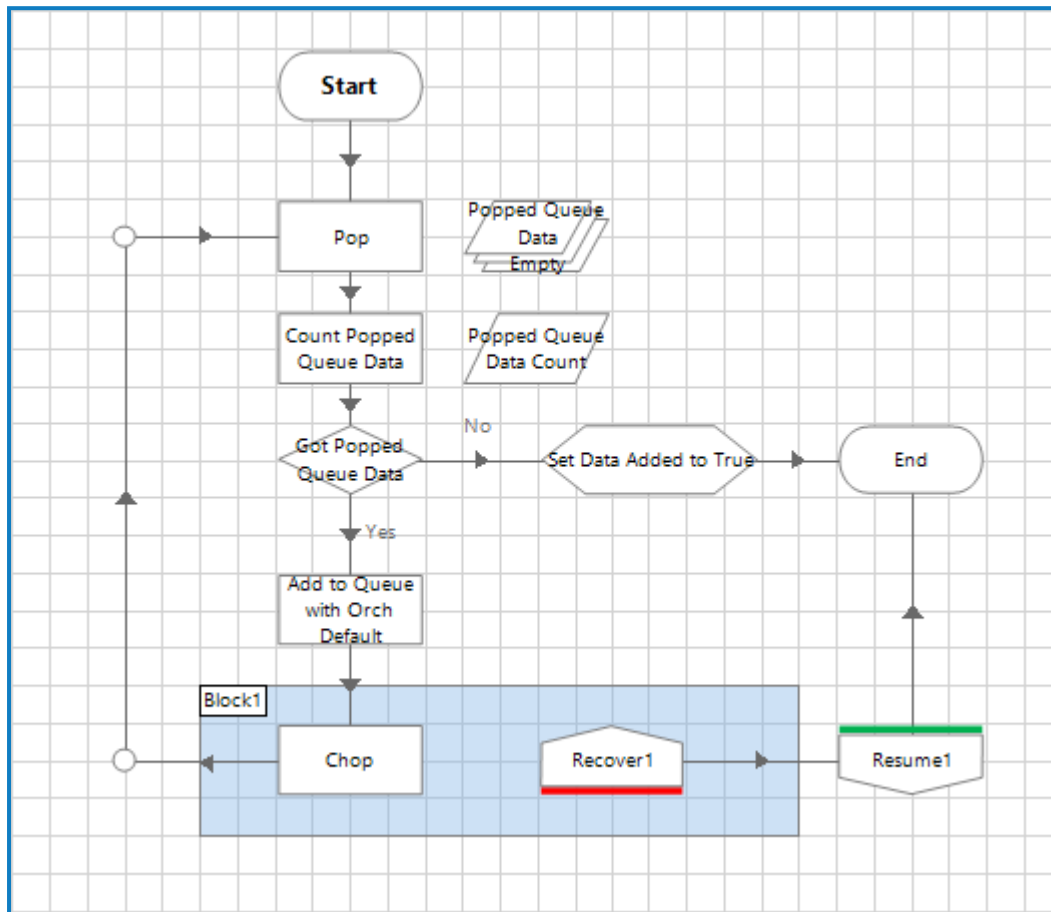
The Get Data page is the collection page. This page needs to be modified to support whatever is needed to gather the data required for the business process. The example below illustrates using Microsoft Excel to collect the data required to run the business process.



Add to Queue page

The Add to Queue page adds the item or items to the defined queue with the assigned priority and SLA.

It is important to remember the 'Pop' action is used to split the data collection into one row at a time to add to the queue. This is due to the IADA 'Add To Queue' which enables work items to be added with different priorities and SLAs, depending on conditions.



In the 'Add To Queue' action you must enter a process name and queue name. This process name also has to be configured in the [Orchestration process template](#), within the Process Control page, in the Choice stage to ensure that it is called correctly. This is illustrated in the properties form below.

The screenshot shows the 'Action Properties' dialog for the 'Add to Queue' action. The dialog is titled 'Action Properties' and has a dark blue header. Below the header, there are fields for 'Name' (set to 'Add to Queue with Orch Default') and 'Description'. The 'Business Object' is set to 'IADA Queue Management' and the 'Action' is 'Add to Queue'. The 'Inputs' tab is selected, showing a table of input parameters. The 'Outputs' and 'Conditions' tabs are also visible. On the right, there is a 'Group' section with checkboxes for 'Page' and 'Data Type', and a 'View All Items' button. Below this is a tree view showing the 'Add to Queue' action and its sub-items: Binaries, Collections, Popped Queue Data, Dates, DateTimes, Flags, Images, Numbers, Popped Queue Data Co, Passwords, Text, Times, TimeSpans, Globals, and Locks. 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
Queue Name	Text	[Queue Name]
Data	Collection	[Queue Data]
Process Name	Text	[Process Name]
Priority	Number	[EV - Add Customer - Queue - Priority]
SLA	TimeSpan	[EV - Add Customer - Queue - SLA]
Status	Text	
Tags	Text	"P="&[EV - Add Customer - Queue - Priority]&" S="&[EV...
EnvironmentName	Text	"RPA"

Executor process

The Executor process template has to be adjusted to run the specific business process.

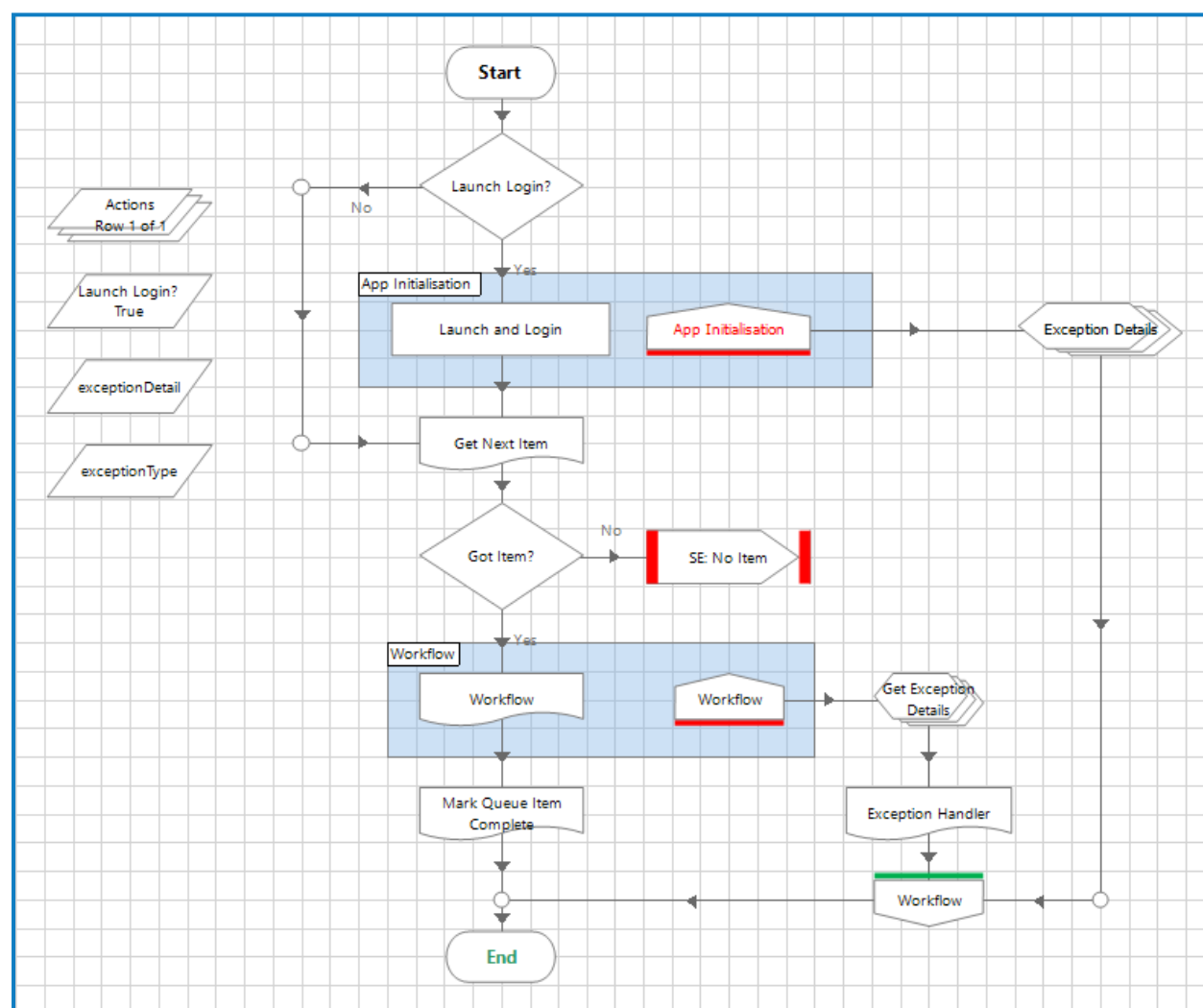
An Executor process is required for each business process to be run using the Orchestrator. There are a number of pages in the Executor template that need to be modified to support your business process, these are described in detail below.

Within the Executor a 'Create Launch Required' flag is configured as an input to the Executor by default. When this is set to True, it performs a 'Launch and Login' to the application from the [Orchestrator process template](#). For this to happen correctly, you must separate the launch and login actions from the rest of the process in the Executor and use the flag to decide whether to launch and log in or skip this function.

Main page

The Main page invokes the Executor process, as illustrated below. This page calls the other pages and does not require any changes to support your business process.

You can see that the Launch and Login are separate actions to the main tasks required to be performed by the Executor which are performed in the Workflow page.



Workflow page

The Workflow page is where the work is actually performed and needs to be configured to perform the specific business process. In this example, an action is called from an object which takes the necessary parameters and performs the business part of the process. The example illustrates 'Adding a Customer' to a sales application.

The screenshot displays the 'Action Properties' dialog box for the 'Add Customer' action. The dialog is divided into several sections:

- Name:** Add Customer
- Description:** (Empty text area)
- Business Object:** Component - IADA Training
- Action:** Add Customer
- Inputs/Outputs/Conditions:** A table with the following data:

Name	Data Type	Value
Customer Name	Text	[Data.Customer Name]
Customer ID	Text	[Data.Customer ID]
Customer Address	Text	[Data.Customer Address]
Customer Contact	Text	[Data.Customer Contact]
- Stage logging:** Enabled (dropdown), ☐ Don't log parameters on this stage
- Warning threshold:** System Default (dropdown), Number of minutes: 5 (spinner), (0 to disable)
- Group:** A tree view showing the hierarchy: Globals, Locks, Workflow, Binaries, Collections, Dates, DateTimes, Flags, Images, Numbers, Passwords, Text, Times, TimeSpans.
- Buttons:** OK, Cancel

In the background, a workflow diagram is visible, showing a 'Start' node, an 'Add Customer' action, and an 'End' node.

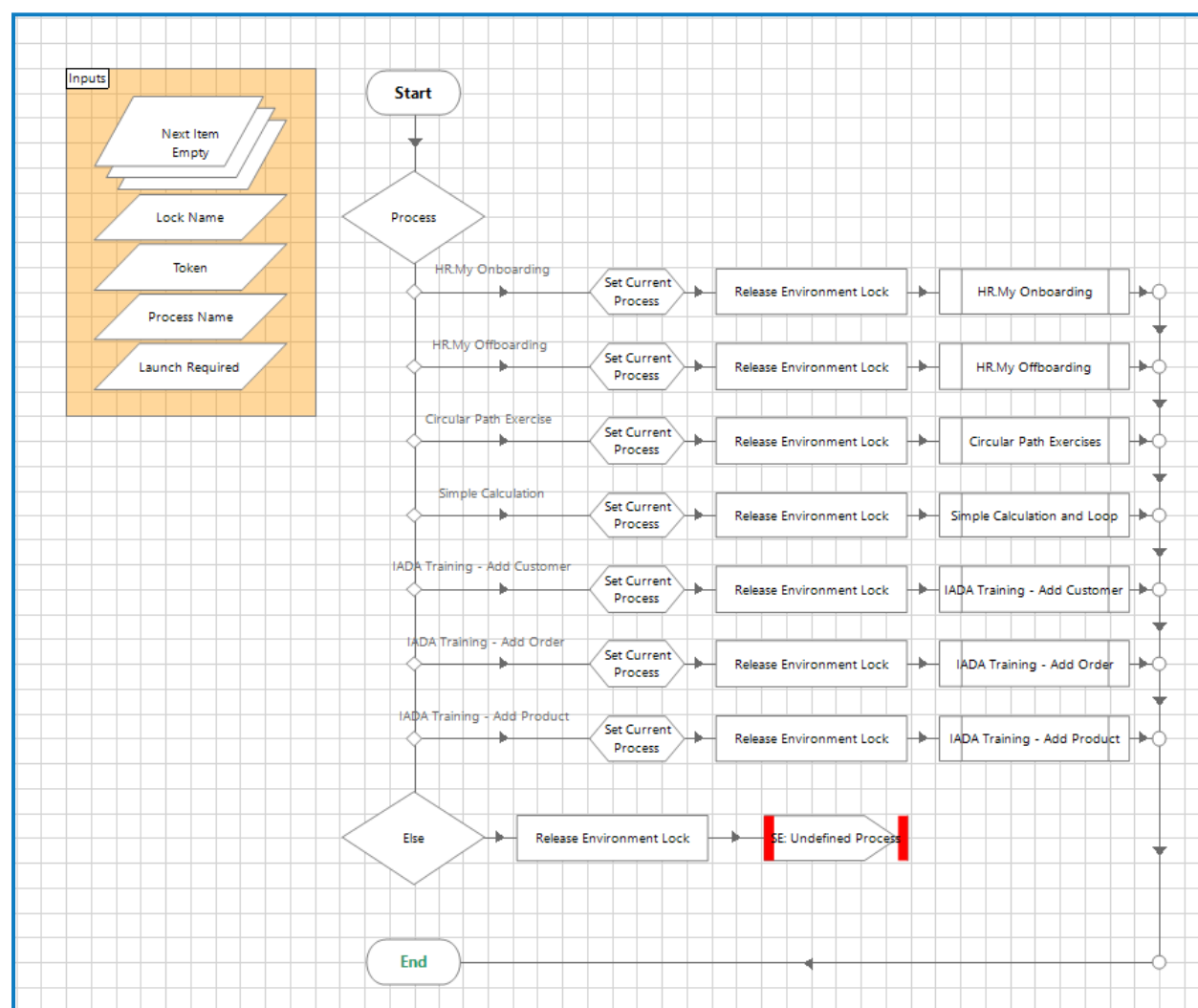
Orchestrator process

Like the other processes, the Orchestrator process template has to be adjusted to run the specific business processes.

There is one Orchestrator process required for each digital worker, though these are expected to be identical, with each digital worker capable of performing the business tasks required of the Orchestrator process. There are a number of pages in the Orchestrator template that will need to be modified to support your business process, these are described in detail below.

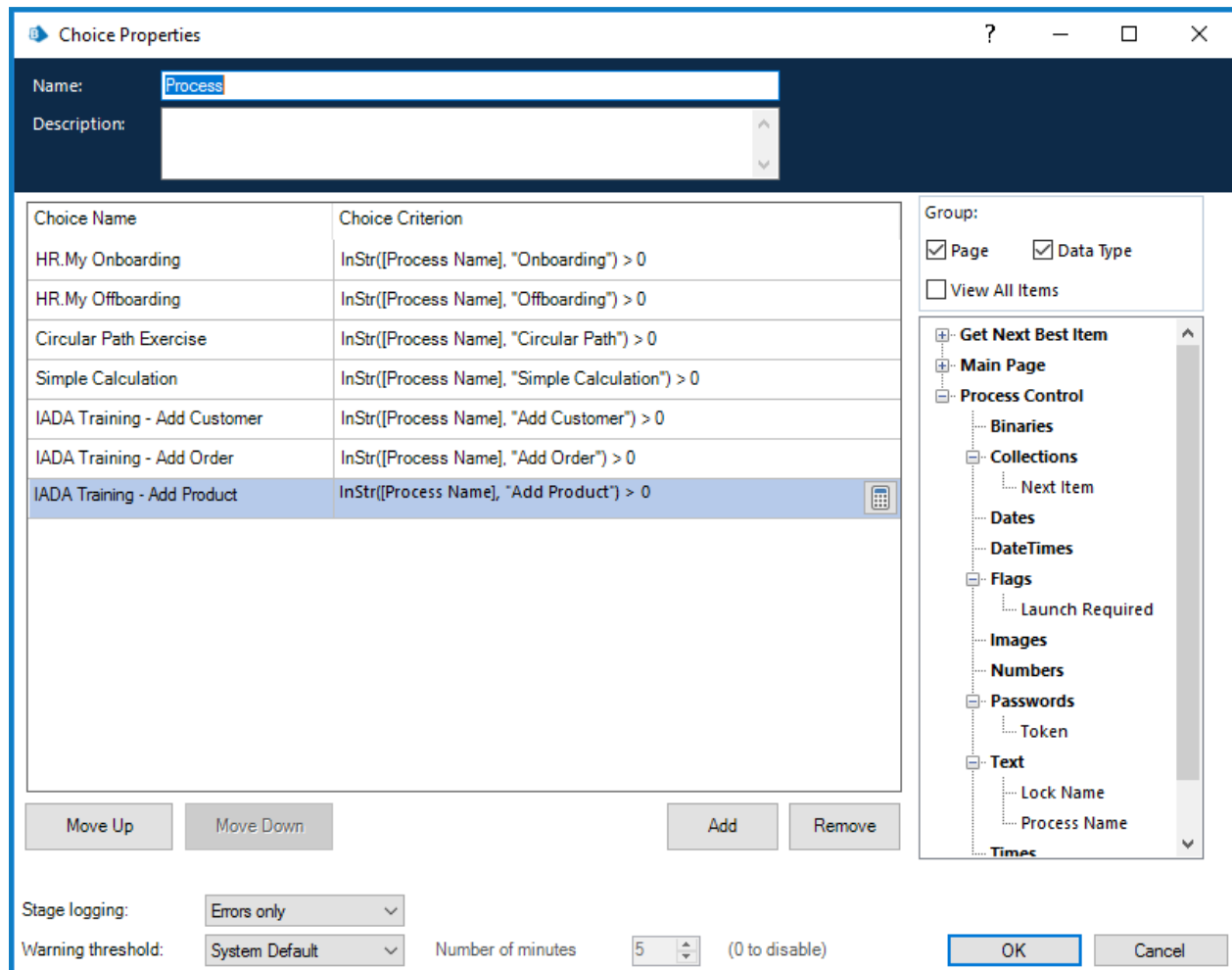
Process control page

The Process Control page of the Orchestrator template is where all the business processes are called. For each business process subject to orchestration, the process will need to be added in, as illustrated below.



The Choice action is used by the Orchestrator process to determine what actions the digital worker will perform. In the example above, there are seven processes configured. These all call their respective Executors.

The Choice stage is configured as illustrated below. This is where the check is performed to which Executor to run determined by the Process Name set in the Collector process.



The Choice Properties dialog box is shown. It has a title bar with a question mark, minimize, maximize, and close buttons. The dialog is divided into several sections:

- Name:** A text field containing "Process".
- Description:** A text area.
- Choice Table:** A table with two columns: "Choice Name" and "Choice Criterion".

Choice Name	Choice Criterion
HR.My Onboarding	InStr([Process Name], "Onboarding") > 0
HR.My Offboarding	InStr([Process Name], "Offboarding") > 0
Circular Path Exercise	InStr([Process Name], "Circular Path") > 0
Simple Calculation	InStr([Process Name], "Simple Calculation") > 0
IADA Training - Add Customer	InStr([Process Name], "Add Customer") > 0
IADA Training - Add Order	InStr([Process Name], "Add Order") > 0
IADA Training - Add Product	InStr([Process Name], "Add Product") > 0
- Group:** A section with checkboxes for "Page" (checked), "Data Type" (checked), and "View All Items" (unchecked).
- Tree View:** A tree view showing the hierarchy of the process. The "Process Control" folder is expanded, showing "Binaries", "Collections", "Dates", "DateTimes", "Flags", "Images", "Numbers", "Passwords", "Text", and "Times". The "Text" folder is expanded, showing "Lock Name" and "Process Name".
- Buttons:** "Move Up", "Move Down", "Add", and "Remove" buttons are located below the choice table.
- Stage logging:** A dropdown menu set to "Errors only".
- Warning threshold:** A dropdown menu set to "System Default".
- Number of minutes:** A numeric field set to "5" with a "(0 to disable)" note.
- OK and Cancel buttons:** Located at the bottom right.

Orchestration control page

This page is where the Launch Required flag is passed into the Process Control page. It needs to be configured for each process, as this determines whether the Launch and Login actions need to be processed by the Executor.

