Simple Single Transaction Process Template

# Introduction

It is a best practice to whenever possible start each new automation using a template. A template should be generic and process agnostic. This accomplishes several goals:

* Gives automations a common structure to make support easier
* Provides common terminology and context to make developer/architect collaboration more effective
* Increases speed to value by providing out-of-the-box functionality without additional effort on each automation. Much of the out-of-the-box functionalities are also best practices: Retry, Config file, exception handling, etc.

As a Technical Account Manager, I do probable 30-40 code reviews per year. I have observed many instances where multiple developers start automations from scratch and reinvent the wheel each time or engage in massive copy/paste activities at the start of each new project. In many cases, it is easy to understand the reasons for these decisions. With these instances in mind, I created this template.

# Template Selection

In a mature COE, the template selection and other design details should be discussed with the solution architect and documented the solution design. In a less mature COE, there may not be a formal process. In any case, making a good decision on the template can make a big difference in the success of the project and the growth of the program.

Some well-known factors affect which template should be selected.

1. The support team’s level of experience in software development and RPA – We shouldn’t expect a relatively new developer to see the value in or be willing to take on the complexity of a very heavy template.
2. Many Transactions and Parallel Processing Requirements – Automation with these requirements will almost certainly need to use Orchestrator queues. This is by nature more complex automation that most likely requires RE Framework or the Document Understanding Process Template. There are multiple versions of these available and customers tend to make their customizations as well.
3. Single Transaction (not transactional) – There are lots of automations that are a set of steps that just happens once per day or once per hour. In other words, there aren’t X number of transactions to be loaded into a queue. It is one all-or-nothing set of actions that produce a result.

It is the third factor above that has led to the creating of this template. I would like my customers to have a simple template to use for Single Transaction processes. One that doesn’t have the complexity or overhead of RE Framework but does provide out-of-the-box features that are best practices for any automation.

# Features of the Simple Single Transaction Process Template

Below I will describe the features of the template with screenshots. As with any other template, customize it to fit the needs of your company. Publish your template so that all developers in your program have access to it. If your ideas are generic and valuable to other customers, allow me to integrate them into the template.

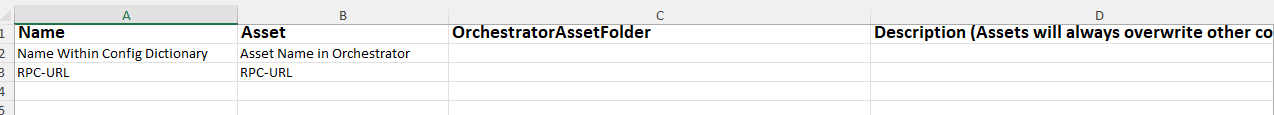
1. Config.xls and “in\_Config” dictionary variable – Best Practice – Don’t hardcode.

For those of use familiar with RE Framework, this is a direct copy of the same functionality. The Settings and Constants sheets are loaded into the Config dictionary variable. These should be used for values that don’t change per environment.

Table

Description automatically generated with medium confidence

The Assets sheet is processed by retrieving the assets (non-credential) from Orchestrator and loading the values into the Config dictionary variable. These should be used for things that are different for each environment and for things that are likely to be changed often. Examples: URL of the Dev, Test, or Prod website to be used, email addresses that the report is sent to, etc.



Graphical user interface

Description automatically generated with low confidence

1. Global Retry – Best Practice – Retry functionality should be used to increase the resilience of the automation.

Using a Retry Scope is much simpler to understand than the Retry in the heavier templates. The “number of retries” and the “pause between retries” are set in the config file. Since there is only one transaction, it is either all or nothing. The retry starts over at the beginning.

Graphical user interface, application

Description automatically generated

1. Exception Handling – Best Practice – A global exception handler should be used to catch all expected and unexpected exceptions.

Too many times I see code where a Try-Catch is implemented in a workflow and never tested. An inexperienced developer may expect the automation to end. Instead, only the current workflow ends, and execution continues on the next higher-level step causing unexpected results.

Having a single exception handler allows consistent well tested results and keeps the exception handling code in one place.

If you have code that detects a situation where you need the error to occur, use a Throw.

Graphical user interface, application, Word

Description automatically generated

If your requirement call for a Try-Catch at lower levels in the automation, minimize the copy-paste and use the global exception handler. Do the few things you need to do and then use a Rethrow activity to propagate the exception up to the main exception handling code.

Graphical user interface

Description automatically generated

Keep in mind that the Retry Scope will happen before any exception handling is engaged. As shown below, once the Retry is satisfied, the Try-Catch comes into play.

Graphical user interface, text, application

Description automatically generated

The exception handler takes a screenshot for diagnosis purposes and logs an error message. Depending on a config setting, it will either fail the job or not. This is usually driven by how you want to manage your response to a failure. If you primarily monitor logs, the Error log may be sufficient and you might not want to also fail the job.

Graphical user interface, application

Description automatically generated

1. Exception Screenshot – This is also directly leveraged from RE Framework.

The exception screenshot is very useful when troubleshooting issues on Unattended robots. The folder path is set in the Config file. By default, it is a relative path on the robot which makes it difficult to retrieve in some environments. It also supports a full path which can be a synced OneDrive folder or any shared file location.

Graphical user interface, text, application, table

Description automatically generated

# Conclusion

The Simple Single Transaction Process Template provides an easy-to-understand and lightweight template that can be used by experienced and inexperienced developers. It satisfies the best practice of starting each automation with a template. It also includes out-of-the-box functionality that meets several other best practices. Use this template as is or customize it for your company. In either case, publish your templates to Orchestrator and use them for your automations.