



# Rampiva Automate

## Using Parameters

Driving Change Whitepaper Series

December 8, 2021

# Disclaimer

This document is provided “as-is.” Information and views expressed in this document, may change without notice. This document does not provide any legal rights to any intellectual property in any Rampiva product. This material does not represent a commitment from Rampiva to provide software or services to the recipient. Recipient may copy and use this document for internal, reference purposes.

Please do not distribute without written permission from Rampiva.

© Rampiva Global LLC, 2021 All Rights Reserved

<b>What Are Parameters</b>	1
<b>Designing Workflow Parameters</b>	1
<b>Exploring Different Types of Parameters</b>	2
General Parameters	2
Dynamic Parameters	3
Scheduler Parameters	3
Protected Parameters	4
Masked Parameters	4
Lock Parameters	4
Dropdown Parameters	4
Scripted Parameters	4
<b>How to Use Parameters</b>	5
Setting Case, Source and Database Directories	5
User Input	5
Referencing a File Name and Location	6
Setting a Username and Password	7
<b>4 Reasons Why You Should Maximize the Use of Parameters</b>	8

---

# What Are Parameters

Rampiva Automate delivers multiple layers of automation to Nuix Workstation. The first, Workflow Operations, allow users to define a specific order of operations for Nuix tasks, such as Load, OCR, Search & Tag, Deduplication, Reporting, and Exporting. This allows for repeatable, easy-to-use workflows that process and export data the very same way, every time.

The second, Job Scheduling, allows users to build a queue of prioritized jobs that execute the tasks defined in a Workflow, as Nuix processing resources become available. Job Scheduling allows users to utilize different resource pools, assign job priority, confidentiality, and when the job will run.

The final layer, Parameters, are settings that allow the other two automation layers to interact with the processing environment based on the unique input of the workflow designer or user. Like coding languages that use attributes, variables and statements to 'store' data for later use in the code, Rampiva uses Parameters to store dynamic information that will be used in a Workflow.

# Designing Workflow Parameters

When designing a robust workflow that utilizes Parameters, designers will need to consider how the operating environment is configured, including machine profiles like Memory, Disk and CPU. They should also consider which user accounts to use when connecting to things like SMTP, SQL or Relativity. Workflow Parameters can have any name that the designer designates (if a System Parameter with that name does not already exist).

The syntax for creating a workflow Parameter is:

1. All lower-case letters, numbers and underscores.
2. No special characters or spaces; and
3. Enclose the value with curly brackets, for example {parameter\_sample01}.

It's good practice to use values that are meaningful to what the Parameter does. Additionally, you can also set friendly names for Parameters which are presented to technicians when scheduling Jobs.

To understand the value of Parameters you first need a basic understanding of when and where they can be used. Parameters can be used in two distinct ways:

- **System Configuration**—These parameters most often prescribe the location of fixed assets, such as a case or source location, database location & credential or a binary storage location. By design, these Parameters are often hidden from the technicians operating the platform. System Configuration Parameters limit user input, while simultaneously creating or maintaining workflow standards. System Configuration Parameters also ensure that environmental objects are treated similarly on each Nuix processing machine. They allow each machine to copy data, authenticate and interact with each other in a consistent and prescribed manner.
- **Workflow Operations**—These Parameters can be used to set any number of variables such as data time zone, document numbering, evidence name or a specific folder name to process. These Parameters are instrumental in allowing Workflows to scale and be used interchangeably between clients and matters. For slight variations in a standard Workflow – such as time zone preference for a client, a workflow Parameter allows the technician to set this critical data point at job setup, and it will inform all future steps in the workflow.

# Exploring Different Types of Parameters

## General Parameters

These parameters do not depend on the previous workflow executions or user operations. They can be used at any point in the workflow. Usually, these Parameters are static configuration items and not user-input items.

**Examples include:**

{case_name}	The name of the case
{date_time}	The current date and time at execution, formatted using YYYYMMDD-HH:mm:ss
{workflow_name}	The name of the workflow
{server_name}	The name of the server running the workflow

## Dynamic Parameters

These parameters only apply to operations performed during the execution of the workflow and will not retain values from previous executions or manual operations. Dynamic Parameters have two sub-categories. First is the input category. They are designed to control how the operations run (for example {data\_timezone\_id}). The second is the output category. They are system built-in and capture the output of operations as data flows through the system (for example {last\_batch\_load\_guid}).

**Examples include:**

{data_timezone_id}	The source time zone of the data to use when adding evidence
{last_batch_load_guid}	The GUID of the last batch loaded into the case
{last_item_set_name}	The name of the item set to which items were last added
{export_production_set_name}	The name of the production set being exported

## Scheduler Parameters

These parameters only apply to workflows ran through Rampiva Scheduler. These Parameters automatically set by Rampiva when scheduling a Job and assist in reporting, automation and data structuring.

**Examples include:**

{job_submitted_by}	The name of the user who submitted the job
{job_submitted_date}	The time and date when the job was submitted
{matter_name}	The name of the matter under which the job was submitted
{client_name}	The name of the client under which the job was submitted

## Protected Parameters

The values of these parameters can be set using the Configuration operation but can only be accessed by the Rampiva operations which have a password field. Protected Parameters cannot be accessed by users, scripts or other systems.

## Masked Parameters

Parameters with the name ending in '\_masked' will have their values masked in the user interface but can be retrieved by any operation. This includes system configurations, user inputs and scripts. As a result, when a workflow includes these Parameters, they will be displayed as `*****` in the workflow execution log.

## Lock Parameters

Parameters with the name ending in '\_lock' can be used to create interdependencies between jobs running in Rampiva Scheduler. When Lock Parameters are set, no two jobs with the same Lock Parameter with the same value will run at the same time. This is very helpful when you want to break large datasets into smaller more manageable tranches, but also use a single case per matter workflow model. This also allows other jobs to be processed in parallel with the job using the Job Lock Parameter. This Parameter can accept any value before the '\_lock' attribute. You can use the job name, a jobid or gibberish, either work. Any job with the same '\_lock' attribute will be grouped and processed in series – one after the other.

## Dropdown Parameters

Parameters using regular expressions that have values separated with a pipe character “|” will have a dropdown with those values in Scheduler. This type of Parameter is most used when users are asked to input a value from a list, but the workflow operation requires precision, such as setting the data source time zone, or case time zone.

## Scripted Parameters

Scripted Parameters are an advanced type of Parameter. They combine the flexibility of Parameters with the power of coding. Scripted Parameters can perform multiple actions at once. For example, a Scripted Parameter could access a database with a Python script, evaluate the active users' authentication context, then populate the value of the Parameter based on the user account supplied – this is helpful, for example, if each team has their own Relativity Workspace. When a scripted Parameter is used, the Parameter could first query the Relativity server to identify which workspaces the logged-on user has access to. Based on that information, the Scripted Parameter will return only the paths available to that user. They will be presented to the user in the Scheduler interface as Dropdown Parameters.

# How to Use Parameters

## Setting Case, Source and Database Directories

Processing environments benefit from consistent naming conventions, locations and processes. In Nuix, setting static locations for source data, database and temp locations are also very important in order to maximize performance. Parameters can be used to accomplish both goals. Typically, designers will build the Add Evidence and Use Case operations to use a series of parameters together to designate the location in the operation.

For example, the Add Evidence operation could look something like this: {base\_source\_location}\{client\_name}\{matter\_name}\{data\_set\_id}

User notes: Add Evidence by Evidence Listing File

Prompt at execution

Predefined

Source path: {base\_source\_location}\{client\_name}\{matter\_name}\{data\_set\_id}\{evidence\_list} Browse

Scope: Evidence listing ?

## User Input

While automation can drastically reduce the amount of user input and clicks required to iterate through a workflow, sometimes you need users to input certain variables. Most commonly, users need to input the data time zone, Evidence name and starting document numbering and prefix.

User notes: Main Configuration

Prompt for configuration settings at execution

Profiles

Configuration profile Browse

Supports Evidence Processing Settings (Data Processing, MIME Type, Parallel Processing), Legal Export (Export Type - Partial, Load File - Partial, Numbering and Files -Partial, Parallel Processing)

Processing profile: Full Text Processing

Execution custom parameters

+ - Load from TSV

Parameter name	Value	Friendly name	Description (optional)	Regex (optional)
{data_set_id}	001	Data Set Id	Input the Data Set ID to be Processed	.+
{docid_prefix}	ABC	Docid Prefix	Input the Production Set Numbering to be Used	.+
{evidence_id}	EV.0001.5435	Evidence Id	Input the Evidence ID Used During Collection	.+
{case_timezone_id}	UTC	Case Timezone Id	Input Case Timezone	(Etc\GMT\+12 Etc\GMT\+11 Pacific\Apia)
{data_timezone_id}	America/Denver	Data Timezone Id	Input Source Timezone	(Etc\GMT\+12 Etc\GMT\+11 Pacific\Apia)
{relativity_username}		Relativity Username	Input Relativity Username	.+
{relativity_password}		Relativity Password	Input Relativity Password	.+

## Referencing a File Name and Location

Several Rampiva Automate operations allow workflow designers to specify a file to be used as part of the operation or where to save a file in the operation. These include, Add Evidence, Search & Tag, Reports, Generate Duplicate Fields and Exports to name a few. Rather than building the workflow with a static location to these files, users can input the values or parameters to dynamically 'look-up' the location of the file for the operation. In the image below, we're loading data from a tab separated evidence listing file. When this workflow runs, a user will be prompted to input the location of the list. This allows a user to select a list that contains multiple custodians, evidence items and time zones.

### Configuration Operation:

User notes: Main Configuration

Prompt for configuration settings at execution

Profiles

Configuration profile

Supports Evidence Processing Settings (Data Processing, MIME Type, Parallel Processing), Legal Export (Export Type - Partial, Load File - Partial, Numbering and Files -Partial, Parallel Processing)

Processing profile: Meta Scan - Index PST contents

Execution custom parameters

Parameter name	Value	Friendly name	Description (optional)	Regex (optional)
{evidence_list}	{matter}_list.txt	Evidence List		

This workflow will use the value provided by the user when it arrives at the Add Evidence Operation. In the image below, we're defining the source path to include the Evidence List parameter. Doing so will allow the user to enter only the filename of the Evidence List when running the workflow.

## Setting a Username and Password

Rampiva Automate can interact with several external systems that require username and password entry. Including Relativity, SQL, Email and Messaging systems. In order to interact with these systems, workflow designers can allow users to input their username and password when the workflow starts. The passwords will be masked and stored in an encrypted state but will still be available to the operation that requires the username and password values.

User notes: Set the Relativity Hostname and Login Details

Configuration help:

Host name: {relativity\_hostname}

Service endpoint: /relativitywebapi

Endpoint type: HTTPS

User name: {relativity\_username}

Password: {relativity\_password}

Workspace identifier: Name {matter}

Folder path: {relativity\_top\_folder}



# 4 Reasons Why You Should Maximize the Use of Parameters

- 1. Simplicity.** Yes, it's true that automation can increase performance, efficiency and quality, but they also make things easier on their operators. Parameters take that concept to the next level. Parameters allow you to take dynamic input that would usually be performed by a user in a series of mouse clicks and keyboard inputs, converting that into a set and prescribed series of operations. A workflow that utilizes parameters abundantly and effectively can easily reduce the user clicks and inputs by 95% or more. You'll be able to do more with Nuix and significantly cut down on user inputs and errors.

---

- 2. Adaptability.** Parameters can be used to define dynamic information on the fly - as the user starts the workflow. This allows you to use the same workflow template for different customers, jobs and processing machines. Using Parameters will allow users to run any workflow, from any location in the processing environment and achieve consistent results.

---

- 3. Security.** Any complete workflow designed in Rampiva Workflow, with or without parameters, will improve the throughput and efficiency of your Nuix processing environment. However, a workflow designed to extensively utilize Parameters considers environmental variables that end users may not be aware of. It allows the workflow to connect to objects outside of Nuix, such as SQL and Relativity. While you can technically accomplish these functions without Parameters, doing so would mean you would be storing the username and password in clear text. When users run workflows with Parameters, they will be prompted to enter in the Parameter details that the workflow designer prescribed. Such as username and password to a resource like SQL or Relativity. This additional security layer ensures that users have access to the resources they need, without compromising the integrity of the security mechanisms in other systems.

---

- 4. Standardization.** Many Parameters are designed to be input by a technician when starting a job. You can combine the power of Parameters with the standardization of Regular Expressions and scripts. Workflows can be configured to allow users to free-fill the values for the Parameters or force users to select a value from a Dropdown Parameter. When Dropdown parameters are used, the workflow designer has limited the acceptable inputs for the Parameter. This is especially valuable when setting client directories, or other variables that require the input to be precise.

---



[rampiva.com](https://rampiva.com)

© 2021 Rampiva Global LLC. All rights reserved.