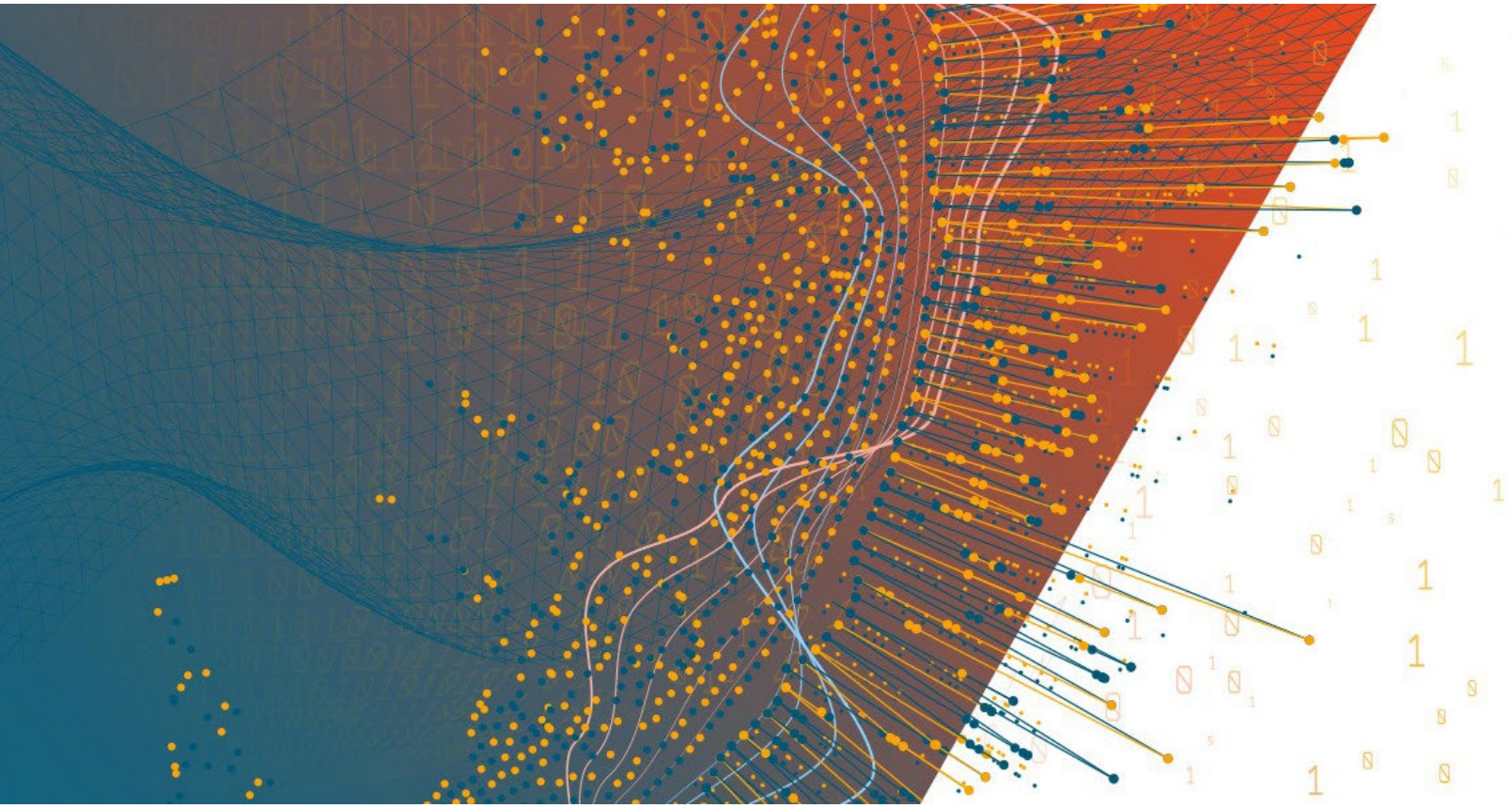


# Altair® Monarch Server™ v2025.0

## AUTOMATOR - TUTORIAL

## TABLE OF CONTENTS

**Introduction** ..... **1**

Overview of Automator Objects ..... 1
Automator Workflow ..... 2
Visual Process Designer ..... 3

**Overview of the Tutorial Lessons** ..... **4**

**Lesson 1 - Working with Project Files** ..... **7**

About Project Files ..... 7
Creating a Project File with the Project Editor ..... 7
Specifying General Settings for a Project File ..... 8
Assigning a Model to a Project File ..... 8
Selecting Input Files for a Project File ..... 10
Configuring a Project File Export ..... 11

**Lesson 2 - Working with Processes** ..... **17**

About Processes ..... 17
Creating a Process ..... 17
Specifying General Settings ..... 17
Specifying Advanced Settings ..... 19
Specifying Job Log Disposition Settings ..... 21
Specifying Monitoring Settings ..... 22
Creating a Custodian Schedule ..... 25
Creating Alerts ..... 26

**Lesson 3 - Running and Scheduling Processes** ..... **28**

Running a Process ..... 28
About Jobs ..... 28
Viewing a Job's Log File ..... 29
Viewing the Output File ..... 30
Scheduling a Process ..... 30

**Lesson 4 - Working with Distributions** ..... **33**

About Distributions ..... 33
Adding a Global Distribution ..... 33
Adding an RSS Global Distribution ..... 35
Defining a Global Translate Distribution ..... 35

Configuring the RSS Global Distribution ..... 36

**Lesson 5 - Working with Alerts ..... 41**

About Alerts ..... 41
Adding an Alert ..... 41
Re-running a Failed Job ..... 43

**Lesson 6 - Working with Visual Processes ..... 45**

Creating a Process with VPD ..... 45
Instructions for Process Work Flow Building ..... 46
Adding and Assigning Items to the Process ..... 46
Configuring a Process Run ..... 49
About Process Logs ..... 53
Verifying Workspaces ..... 54
Working with the Hub Element ..... 61
Working with Scripts ..... 69

**Additional Resources ..... 75**

ALTAIR © Altair Engineering, Inc. All Rights Reserved. / altair.com / Nasdaq: ALTR

# INTRODUCTION

Altair Monarch Server Automator Edition is designed to be used in conjunction with Altair Monarch Complete, a data extraction and analysis tool that lets you view, analyze, extract, and print data from existing computer reports. Any structured report that is used in your organization can be utilized by Monarch Complete. When Monarch Complete "reads" a report file, instead of producing a hard copy printout with words and numbers frozen on the page, Monarch Complete creates a soft copy of the report on screen, with live data you can work with.

### Features and Functions

Automator has the following capabilities:

- Automated conversion of data from TXT, PDF, ISAM, XPS, and OLE DB/ODBC sources to an ISAM, or OLE DB/ODBC (for example, SQL Server, Oracle) target by scheduled or event-driven execution via monitored folders.
- Email distribution of PDF, ISAM, and Open XML Spreadsheets
- File system and SharePoint distribution of input and export files
- RSS publication
- Creation of logs and alerts resulting from operations
- User creation of pre-processes and post-processes using VB.NET Scripting
- Scheduled administration of job logs and database entries

# OVERVIEW OF AUTOMATOR OBJECTS

Throughout this Tutorial, we shall be dealing with the following:

### Project

A project is a key component of a process. When you create a process in Automator, one of the key steps you perform is to assign a project (or projects) to it.

A project file is a special file created with the Project Editor (or with Monarch Complete) that has an XPRJ or DPRJ extension and which contains all of the information necessary to execute one or more Automator operations. A project uses a text or database input, extracts data from the input via Monarch model files, and exports the extracted data to various file types (e.g., XLS, PDF, MDB, TXT, HTML, XML, portable reports, etc.).

### Process

A process is a set of one or more XML project files (i.e., XPRJ or DPRJ files) that are to be executed to perform a desired Automator operation. Processes utilize Monarch Complete models and project files to extract data from text or database sources and then export them to various file types (e.g., XLS, MDB, TXT, HTML, XML, portable reports, etc.).

### Job

A job is a resulting task for one or more projects that comprise the associated process. When a process runs, it ultimately spawns one or more jobs, with each job appearing as a row on the Jobs page.

These are explained in detail in the succeeding lessons.

# AUTOMATOR WORKFLOW

To help you better understand of what a typical Automator session might consist, let's look at the steps you would likely perform during one.

> **NOTE** This sequence of events does not have to be followed exactly. Depending on your needs, you may perform fewer or more steps than those listed below, and the order in which you perform them may differ.

### Create an XML Project File

Before we can begin working with Automator, we need to create an XML project file. Automator can utilize XML project files created either via Monarch Complete or Project Editor. We will create the **Automator Tutorial** project file via Project Editor.

When creating a project file you may perform the following steps:

- **Specify general settings -** General settings include an author for the project file, its name and a description of it.
- **Specify input file** - Input settings include the input file type (text or database), the input file(s) you want to assign to the project, how multiple input files should be grouped, etc.
- **Select a model file** - You should assign a model file to the project to specify which data will be extracted from the input file.
- **Specify export settings** - When you specify project export settings, you also specify an export (or number of exports) of output files and specify any distribution actions you want to apply to them.
- **Specify prerequisite settings** - When you specify prerequisite settings, you also add a file prerequisite or a command line prerequisite to the project. Any prerequisite you add must first be met if the associated process is to run.
- **Specify script settings** - Script settings allow you to specify pre-export and post-export scripts to modify the input and output files, respectively.

### Create a Process

Once you have created or identified the project file(s) you want to work with, you can create a process. Creating a process includes multiple steps, such as the following:

- **Specify general settings** - General settings include a name and description of the process, how many projects per job setting (i.e, one per job or multiple ones per job), the project(s) to be assigned to the process, etc.
- **Specify advanced settings** - Advanced settings include a priority setting, and whether to allow parallel jobs and process retries.
- **Specify job log disposition settings** - These settings determine what will be done with the job log after the job has run, and when (i.e., under what circumstances) the specified action will carried out.
- **Specify input file monitoring settings** - These settings enable you to tell Automator to monitor a particular location, then trigger the running of a process when a specific input file (or any file of the specified file type) appears in that location.

### Run the Process

Once you have completed the process configuration, you can

- Run the process manually.
- Schedule the process to run.
- Configure an input file specification which will trigger the process to run when a particular input file arrives in the specified location.

# VISUAL PROCESS DESIGNER

Automator has a Visual Process Designer (VPD) tool that allows you to create the process and define its work flow in a fast and convenient way.

You have full set of tools for process creation that is used in standard processes but you have greater capabilities to apply these tools and therefore set up more sophisticated scenarios.

Data acquisition, conversion and distribution is visualized in the Process Designer through use of graphic elements and connectors arranged by threads.

By using library of graphic elements and a workspace, you can quickly assemble different process components to set the required process. The created process can be represented as a graph in the Process Designer window.

# OVERVIEW OF THE TUTORIAL LESSONS

Before you attempt to use Automator, we encourage you to walk through this tutorial, which is designed to provide you with a quick introduction to Automator's features and capabilities. While not exhaustive in scope, the tutorial should enable you to begin using Automator effectively.

The tutorial includes the following lessons:

### Lesson 1

Lesson 1 focuses on project files. It explains what they are and then describes how to create them.

Lesson 1 contains:

- Information about project files
- How to create a project file with the Project Editor
- How to assign a model to the project
- How to select an input file for the project
- How to configure a project export

### Lesson 2

Lesson 2 covers processes. It explains what they are and then describes how to create them.

Lesson 2 contains:

- Information about processes
- How to create a process
- How to assign a project file to a process
- How to specify general settings
- How to specify advanced settings
- How to specify job log disposition settings
- How to specify and enable a Custodian schedule

### Lesson 3

Lesson 3 explains how to run processes, both manually and via a scheduled runtime. It also introduces us to jobs.

Lesson 3 contains:

- How to run a process manually
- Information about jobs
- How to view a job
- How to view the job’s log file
- How to view the output file
- How to schedule a process to run

### Lesson 4

Lesson 4 defines distributions and describes how to create them. It also explains how to re-run a job.

Lesson 4 contains:

- Information about distributions
- How to add a global distribution
- How to add an RSS global distribution
- How to define a global translate distribution
- How to configure an RSS global distribution

### Lesson 5

Lesson 5 defines alerts and describes how to create them. It also explains how to re-run a job.

Lesson 5 contains:

- Information about alerts
- How to add an alert
- How to view the job's log file for the failed process
- How to re-run a failed job

### Lesson 6

Lesson 6 describes how to create, configure, and run processes using VPD.

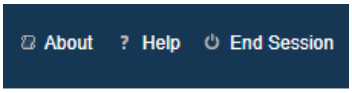
Lesson 6 contains:

- Information about creating a process with VPD
- Instructions for process work flow building
- Information about adding and assigning items to the process
- Information about configuring a process running
- Details about project logs
- Steps to view the log file of a job for a failed process

### Using the web help

Automator comes with a web help that provides information not included in this tutorial. If you encounter an item or topic you would like additional information on while going through the tutorial lessons, there is a good chance you will find it in the web help.

To access the web help, click the **Help** icon at the top of the page.




**NOTE** Certain steps in the tutorial proper will require the use of specific files. These files are only available in the single server installation of Automator. For details, refer to

Monarch Server Automator Edition Single Server Installation Guide.

# LESSON 1 - WORKING WITH PROJECT FILES

This lesson deals with project files. It describes what they are and what they do, then takes you through the process of creating one with the Project Editor.

## ABOUT PROJECT FILES

A project file is a configuration file – created via Monarch Complete or the Project Editor – that has an XPRJ or DPRJ extension and contains all of the information necessary to execute one or more Automator operations.

A project file stores:

- Information about the project's data source (e.g., one or more report files, an external database, etc.)
- The path and filename of the model file
- The name of the active filter, sort order and summary
- Information about project exports, portable report settings, and process prerequisites

When you create a process in Automator, one of the key steps you perform is to assign a project file (or files) to it. Projects generated within Monarch Complete can be used "as is" in and can also be enhanced with the use of the Project Editor to add Automator-specific settings, such as process prerequisites, to them.

For more information on project files, consult the Automator Help file.

## CREATING A PROJECT FILE WITH THE PROJECT EDITOR

Let's try creating our project file via the Project Editor. To open the Project Editor, on the **Navigation Tree**, click **Automator** > **Standard Processes**, and select **Project Editor**.

When creating a new project file via the Project Editor, you may perform the following steps:

- Specify general settings - General settings specify an author for the project file, its name and provide a description of it.
- Specify a model setting - When you specify a model setting, you assign a model to the project, and specify the desired verification settings, if any.
- Specify input settings - Input settings specify the input file type (i.e., text or database), the input file(s) you want to assign to the project, how multiple input files should be grouped, etc.
- Specify export settings - When you specify project export settings, you specify an export (or exports) of export files and specify any distribution actions you want to apply to them.
- Specify prerequisite settings - When you specify prerequisite settings, you add a file prerequisite or a command line prerequisite to the project. Any prerequisites you add must first be met if the associated process is to run.

- Specify script settings - Script settings allow you to, among other things, specify pre-export and post-export scripts to modify the input and export files.

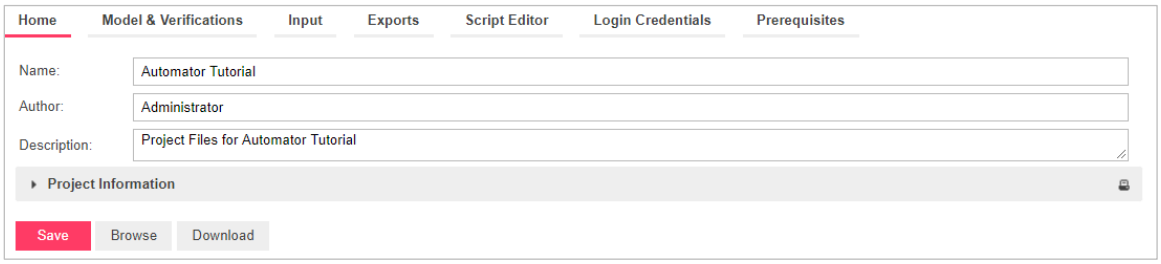
> **NOTE** Depending on your needs, not all of these steps may have to be performed. Specifying general and model settings is enough to create a project file.

# SPECIFYING GENERAL SETTINGS FOR A PROJECT FILE

When you specify general settings for a project, you specify an author for the project file, its name and provide a description of it.

### To specify general settings for a project

1. On the Navigation Tree, click **Automator** > **Standard Processes** and select **Project Editor**. The Home tab is displayed.




2. In the **Name** field, enter a project name, e.g., *Automator Tutorial*.
3. In the **Author** field, type in your name, e.g., *Administrator*.
4. In the **Description** field, enter a description for the project, e.g., *Project files for Automator Tutorial*.
5. Proceed to the **Model and Verifications** tab to specify model settings for the project.

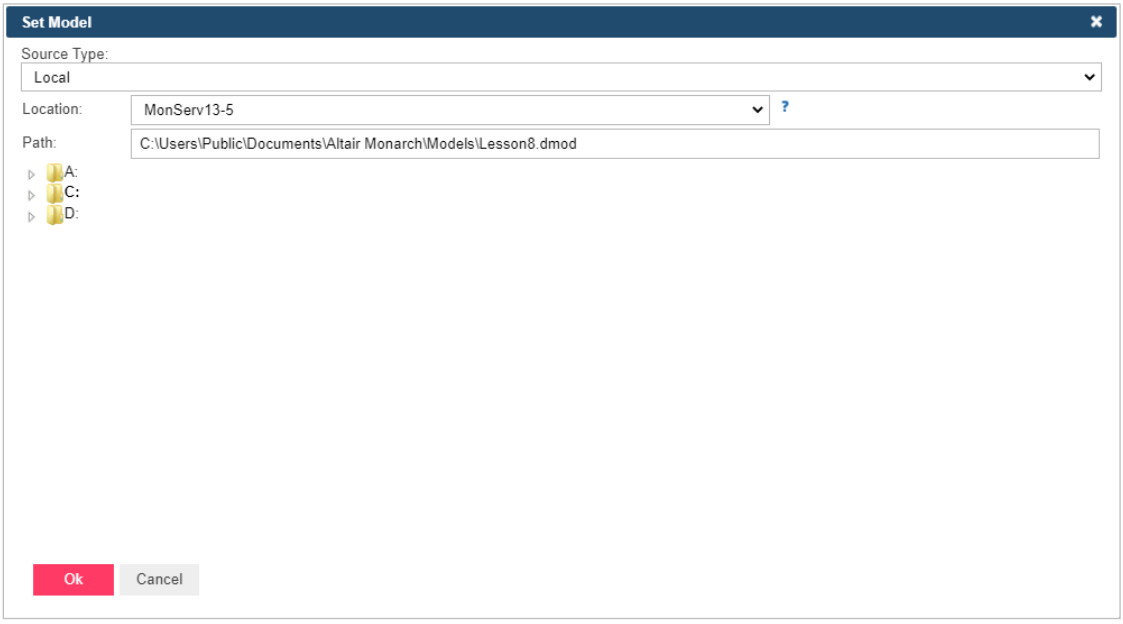
# ASSIGNING A MODEL TO A PROJECT FILE

Now we need to assign a model file to specify what data will be extracted from the input file. Models, which are created within Monarch Complete, contain all of the definitions created during a Monarch Complete session, such as data extraction templates, field properties, filter, sort and calculated field definitions, and so on. Models save work by allowing you to apply the same settings to a periodic report each time it is generated.

> **NOTE** When selecting a model file, be sure to select a model that successfully traps the data you want to extract from the input file.

### Steps

1. On the **Model and Verifications** tab of the Project Editor, click the **Browse** button next to the **Location** field to select the desired model.
2. In the displayed window, from the data source type drop-down list, select **Local**.




3. In the **Location** field, select the location type.

> **NOTE** If the required server is not on the list, you need to create it. For information about adding a new location, refer to the Locations Management topic in the web help.
>
> In case of a single server configuration, refer to Monarch Server Configurator Guide to change location settings.

4. In the **Path** field, enter the file path of your target model, e.g., C:\Users\Public\Documents\Altair Monarch\Models\Lesson8.dmod.
5. Click **OK**. The model file path is displayed in the **Location** text field.

> **NOTE** Model files in compressed archive format (e.g., zip and gzip) can be used and specified in the Project Editor.

The next step in creating a project file is to specify input settings for it.

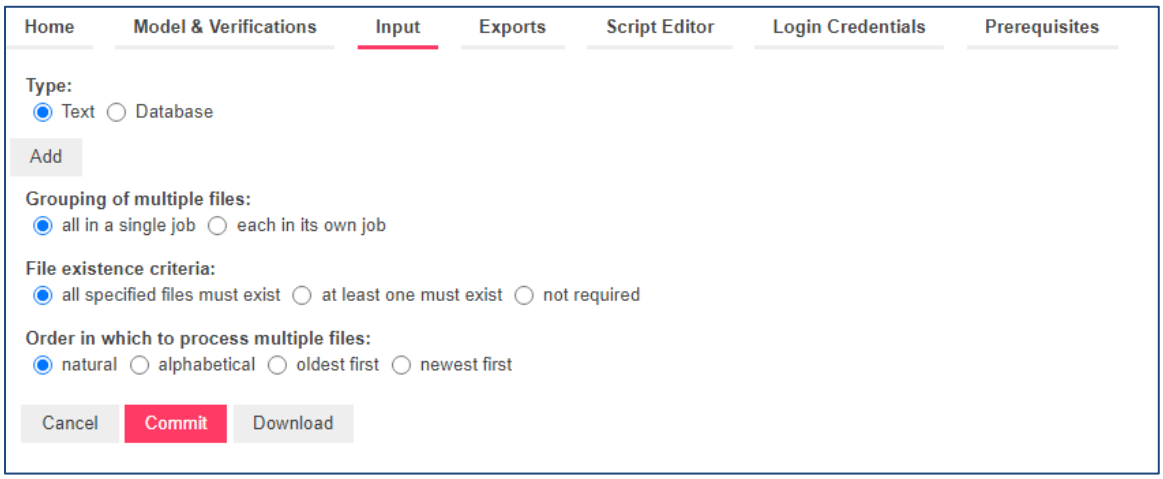
# SELECTING INPUT FILES FOR A PROJECT FILE

Now we're ready to specify what input file we want to assign to the project. The input file is the source of the data that we want to extract, export, etc. For this lesson, we'll use one of the Monarch Complete report files, *ClassJan.prn*, which is a list of fictitious music orders shipped in January of 2010.

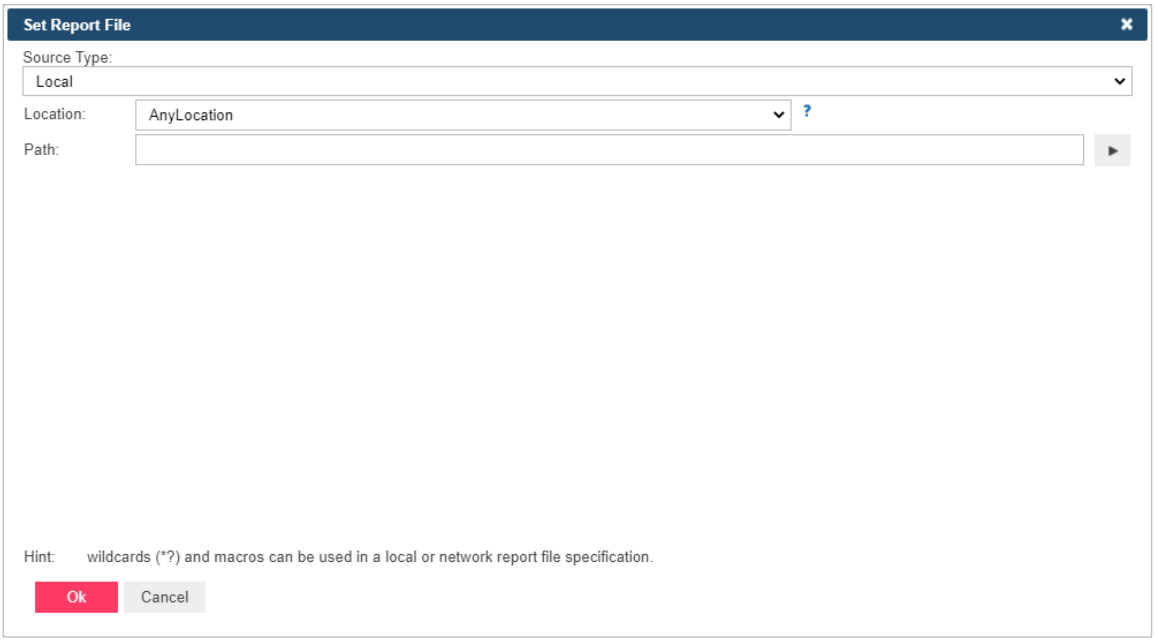
### Steps

1. On the **Input** tab of the Project Editor, select the **Text** file type.

   We choose the **Text** option in this step because we will be using a Monarch Complete report file. If our input was a database file we would choose the **Database** option instead, and then select the appropriate **Select a Data Source** option (i.e., Local or Web File, Data Link Wizard, Data Link File).




2. Click the **Add** button to add an input file to the project via the displayed dialog.




   - Select the **Local** data source type from the drop-down list.

- In the **Location** field select the location type.

> **NOTE** If the required server is not on the list, you need to create it. For the information about adding a new location, refer to the Locations Management topic in the web help.
>
> In case of a single server configuration, refer to Monarch Server Configurator Guide to change location settings.

- In the **Path** field, enter the file path of the report, e.g., C:\Users\Public\Documents\Altair Monarch\Reports\ClassJan.prn.

> **NOTE** Input files in compressed archive format (e.g., zip and gzip) can be used and specified in the Project Editor.

The remaining options on the Input tab have to do with multiple input files, and since we are currently only working with one input file, we can safely ignore them. If you would like more information on these options, refer to the Automator web help.

> **NOTE** If we wanted to, we could now specify a distribution for the project export. Distributions are actions that distribute or handle information or files outside of Automator. Distributions can include the sending of e-mails (with or without attachments), RSS feeds, moving, copying, or deleting of files, and writing to the Windows Event Log. Since we shall be working with distributions in Lesson 4, we won't specify one now.

The next step in configuring a project file is specifying export settings.

# CONFIGURING A PROJECT FILE EXPORT

The next step is to configure our project export.

A project export specifies the table or summary data that will be exported when the project is run, what filters or sort orders (if any) will be applied to the data, and the output file (or database) to which the data will be exported.

Project exports are configured via the Export Wizard. Depending on the view you wish to export (i.e., report, table, or summary) and the output export format (e.g. XLS, PDF, or TXT), a different set of screens will be displayed by the Export Wizard.

For instance:

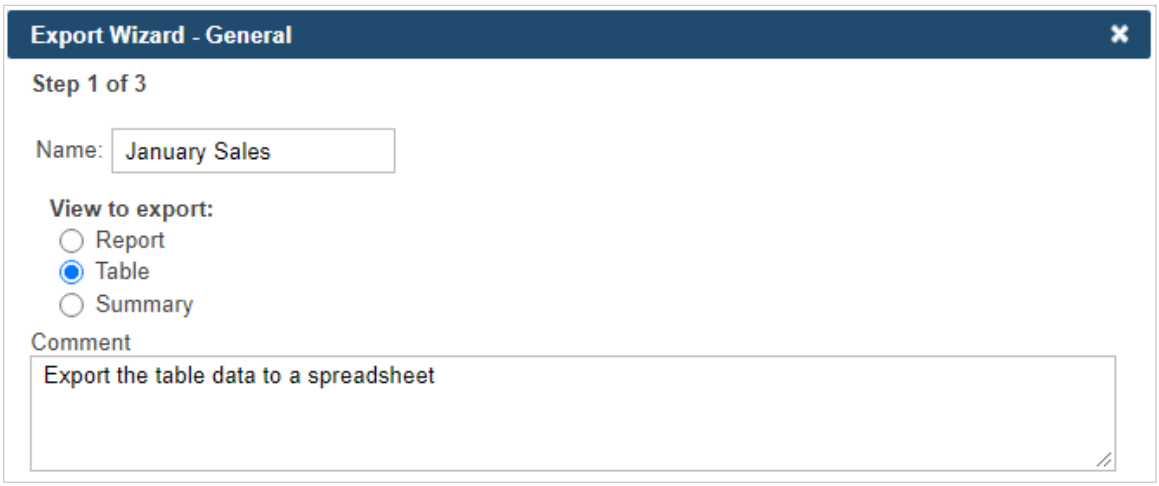
- If you export a Summary, you will be asked to select the summary, drill state, and measure.
- If you export a Table, you will be asked to select the table to export.

- If you opt to export to PDF, you will be asked to supply information about PDF security, such a password to use and permissions to apply.
- If you opt to export a Table to a spreadsheet (XLS), you will be asked to select fields to export and the columns to map to.

The following steps describe a simple case of exporting a table to a spreadsheet.

### Steps

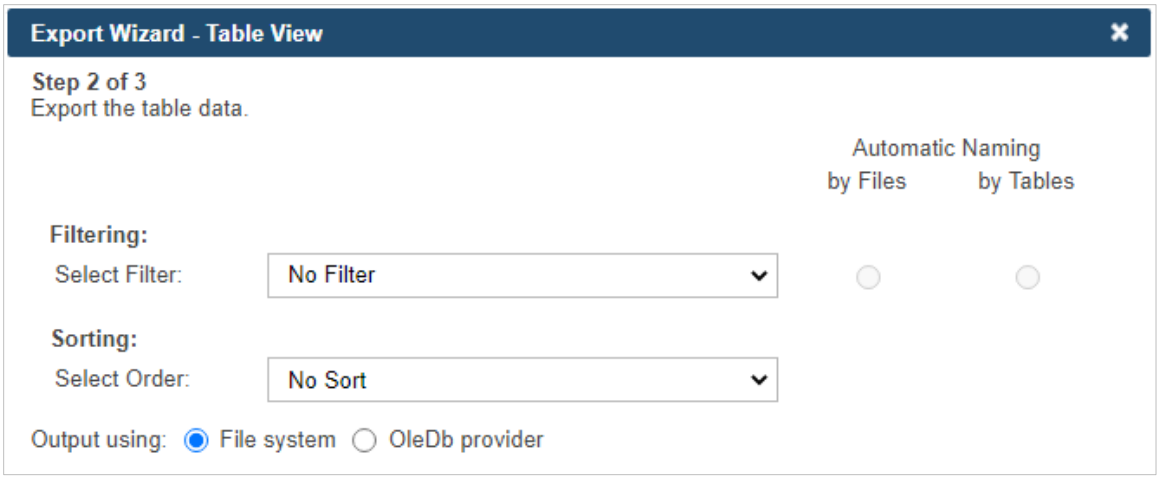
1. Select the **Exports** tab of the Project Editor to display it.
2. On the **Exports** tab click the **Add** icon + to display the Export Wizard.




   On the displayed General screen we can:

   - Name the export
   - Select the view to export: Report, Table, or Summary
   - Enter a comment to annotate the export

3. In the **Name** field, enter *January Sales*.
4. Under **View to export**, leave the **Table** option selected as is by default. This means we want to export the table data. We could also select:
   - **Report**, if we wanted to export the report(s) to a file
   - **Summary**, if we wanted to export the data of a particular summary
5. In the **Comments** field, enter any comment, such as *Export the table data to a spreadsheet.*
6. Click the **Next** button. The **Table View** screen is displayed.




On the displayed **Table View** screen we can:

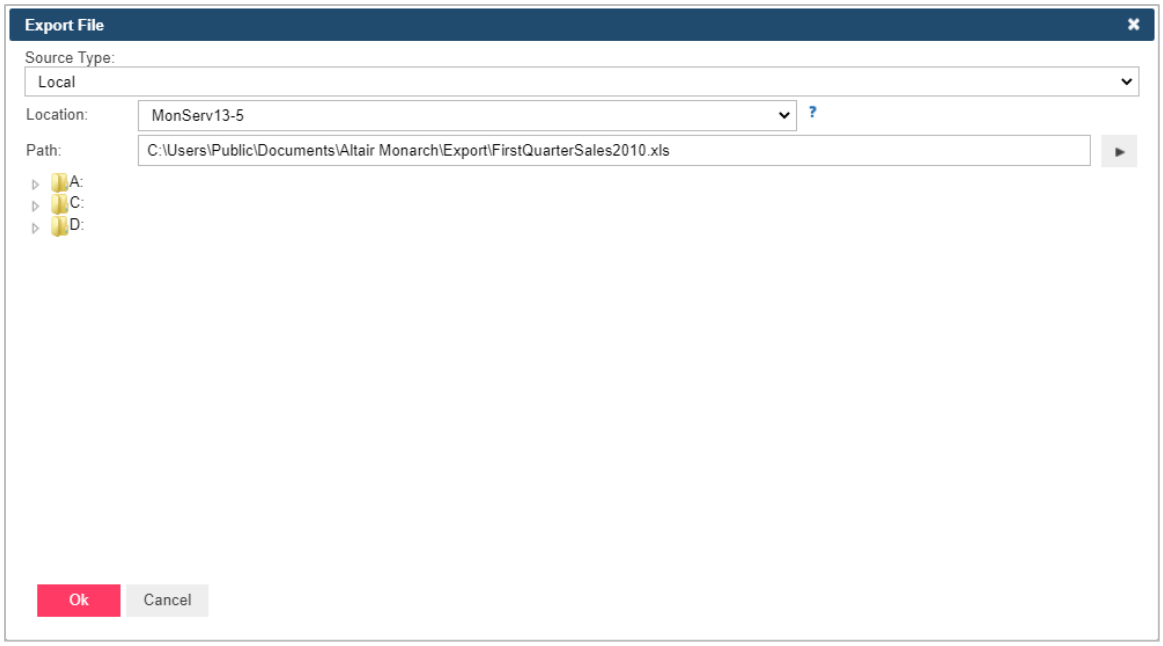
- Select a filter to apply to the export
- Select a sort order to apply to the export
- Specify an output method for the export (i.e., either file system or OleDb provider)

7. Select **No filter** and **No Sort** for filtering and sorting correspondingly.
8. Leave the output as **File system**.
9. Click the **Next** button. The **Output File** screen is displayed.

This dialog box allows you to:

- Specify where we want the table to be exported to
- Select a **When Output Files Exist** option, which tells Automator how to handle an already existing output file of the same name
- Set the format for text files, which is, either fixed length or delimited

10. Click the **Browse** button and use the dialog box that displays to select the export location:




- Select the **Local** data source type from the drop-down list.
- In the **Local** field select the location type.

> **NOTE** If the required server is not on the list, you need to create it. For the information about adding a new location, refer to the Locations Management topic in the web help.
>
> In case of a single server configuration, refer to Monarch Server Configurator Guide to change location settings.

- In the **Path** field enter C:\Users\Public\Documents\Altair Monarch\Export\FirstQuarterSales2010.xls.

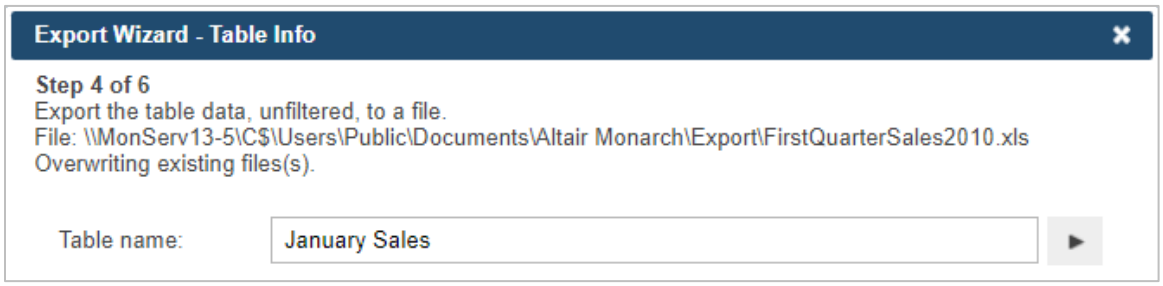
11. Under the **When output files exist** option, leave the option **Overwrite file** selected. This tells Automator that if the current project file is run more than once, data from subsequent exports should overwrite any data found in the existing output file.

> **NOTE** We are selecting this option because, during the tutorial, we will utilize this project file several times.

12. Under the **Format for Text File(s)** option, leave the option **Fixed Text** selected.

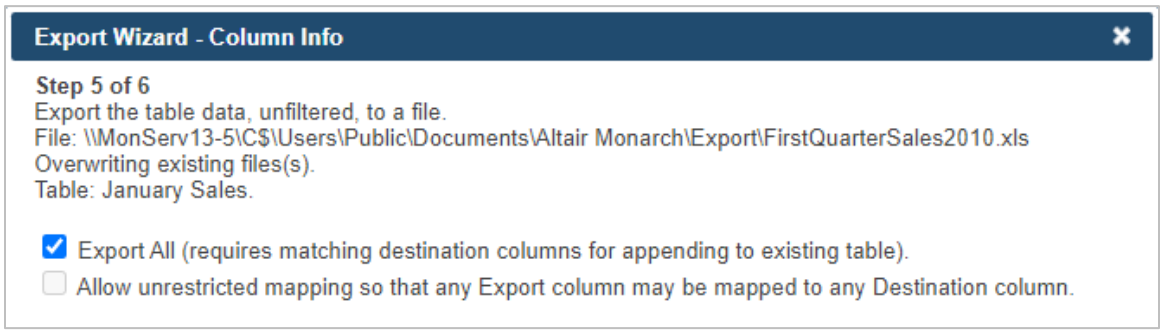
> **NOTE** When creating a new export job or modifying a non-text export job, the default text export format is fixed-length for "txt" extensions and delimited for other extensions. This assures backward compatibility with old projects.

13. Click **Next** to display the Table Info screen. We'll use this screen to name the table we want to export the data to.



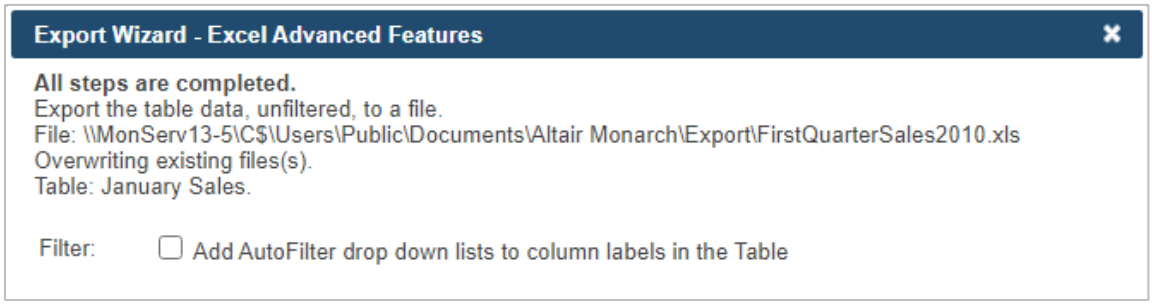

14. Highlight *Untitled* in the **Table name** field and then type in *January Sales*.

15. Click **Next**. The **Column Info** screen is displayed.




16. You will be asked to specify how many columns to export and map to the Excel spreadsheet. Ensure that the **Export all** option is selected.

17. Click **Next** to set Excel advanced features. The Excel Advanced Features screen is displayed.

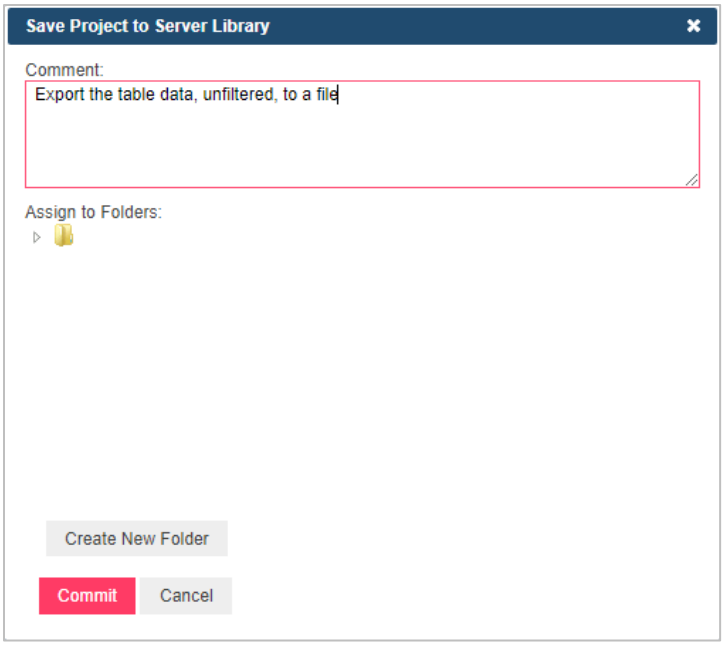



18. Since we do not wish to specify any advanced options for now, click the **Finish** button.

    Automator saves the export settings and closes the Export Wizard. Note that our export, which we named January Sales, is now listed on the Project Editor's Exports tab. On the Prerequisites and Script Editor tabs we could specify process prerequisites and define project-specific scripts for pre-export or post-export operations. On the Login Credentials tab we could enter the credentials necessary to send the export to an FTP site or website. At this point, we can save the current project settings and close the Project Editor.

19. Click the **Save** button to save the project file and close the Project Editor.

20. In the displayed **Save Project to Server Library** dialog box, type in any comment, such as *Export the table data, unfiltered, to a file*, and click the **Commit** button. The Automator Projects page is displayed.




21. Click the **Save** button save the project file to the Automator database.

Now that we have configured a project file to extract the January 2010 sales data from the ClassJan report and export it to a FirstQuarterSales2010 spreadsheet, we can go on to Lesson 2, where we will learn about processes, create a process, and then assign our project file to it.

# LESSON 2 - WORKING WITH PROCESSES

This lesson deals with processes. It describes what they are and shows how to create a process and assign project files (created via Monarch Complete) to it.

## ABOUT PROCESSES

A process contains a set of one or more XML project files (for example, XPRJ files) that are to be executed to perform a desired Automator operation. Processes utilize Monarch Complete models and project files to extract data from text or database sources, and export it to various file types, such as, XLS, MDB, TXT, HTML, and XML.

When configuring a process you do the following:

- Name it
- Assign a project file or files to it
- Set a schedule for it to specify when it is to be run

Once a process is configured and saved, it is added to the Processes page.

## CREATING A PROCESS

Now that we know what a process is, let's try creating one.

Processes are configured on the Processes page. To view the Processes page, on the **Navigation Tree**, click **Automator**, and select **Standard Processes**.

To create a process, we will perform the following steps:

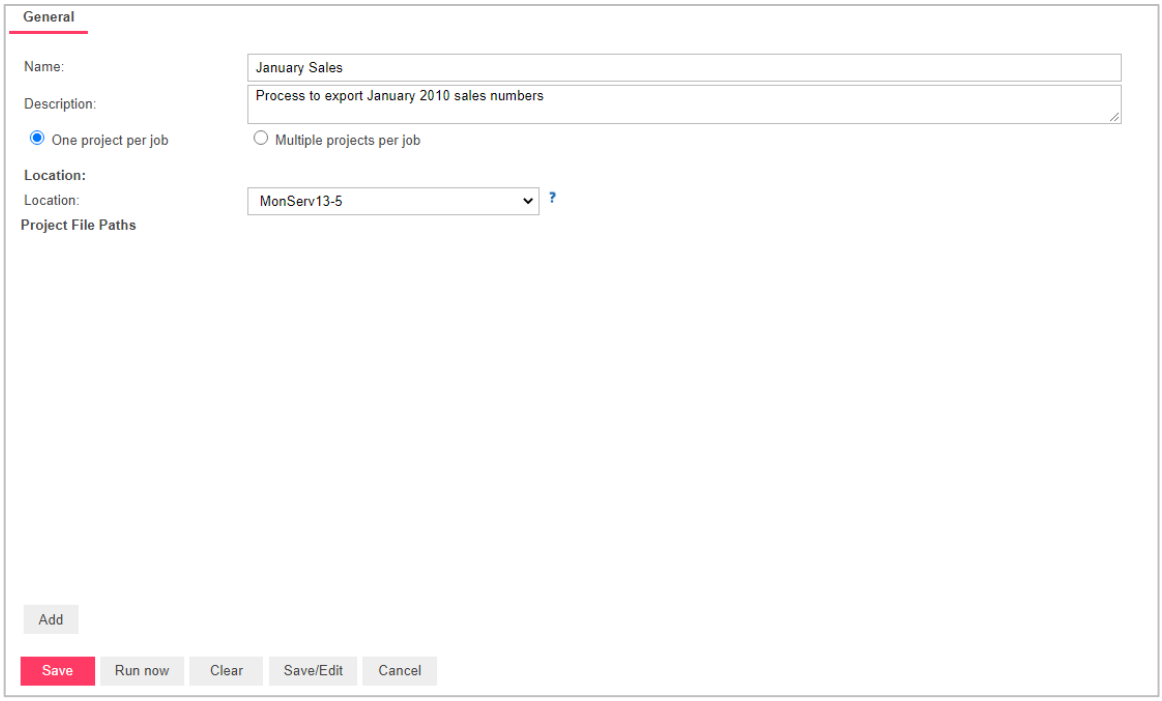
1. Specify general settings – General settings include a name and description of the process, the number of projects per job setting, the project or projects to be assigned to the process, and other settings.
2. Specify advanced settings – Advanced settings include a priority setting and whether or not to allow parallel jobs and process retries.
3. Specify job log disposition settings – These settings determine what will be done with the job log after the job has run and when (that is, under what circumstances) the specified action will be carried out.

## SPECIFYING GENERAL SETTINGS

Let's get started by specifying general settings for our process.

**Steps**

1. On the Processes page, click the **Add** icon . The page with general process settings will be displayed.



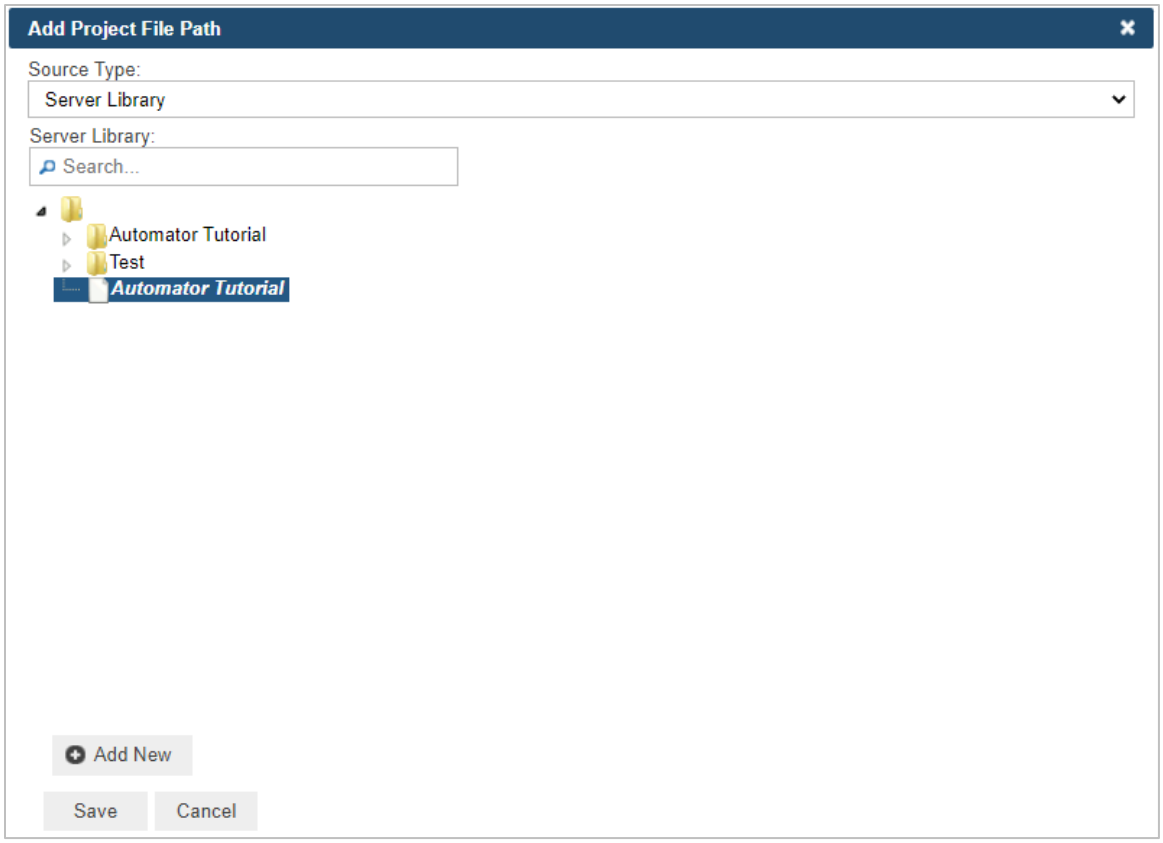

2. In the **Name** field, enter *January Sales*.
3. In the **Description** field, enter *Process to export January 2010 sales numbers*.
4. Select the **One project per job** option since we will only be adding the Automator Tutorial project file to the process.

> **NOTE** For more information about choosing the number of projects per job, see the "Process Handling Scenarios" topic in the web help.

5. In the **Location** field, select the location of the **Automator Tutorial** project file.

> **NOTE** If the required server is not on the list, you need to create it. For the information about adding a new location, refer to the Locations Management topic in the web help.
>
> In case of a single server configuration, refer to Monarch Server Configurator Guide to change location settings.

6. In the **Project File Paths** box, specify the path to the project file. In this lesson we will use the path to the project file we have created. For this click the **Add** button and the **Add Project File Path** dialog box is displayed:




a. From the **Source Type** list, select the **Server Library** data source type.

b. Below the **Server Library** field, select **Automator Tutorial**.

c. Click **Save**.

7. Click the **Save** button to save the process. The system saves the new process to the database and displays the updated Process page, if the operation is completed successfully.

After you have created a process you can specify its priority, whether to allow parallel jobs, and whether to enable process retries. For instructions, refer to the "Specifying Advanced Settings" topic.

# SPECIFYING ADVANCED SETTINGS

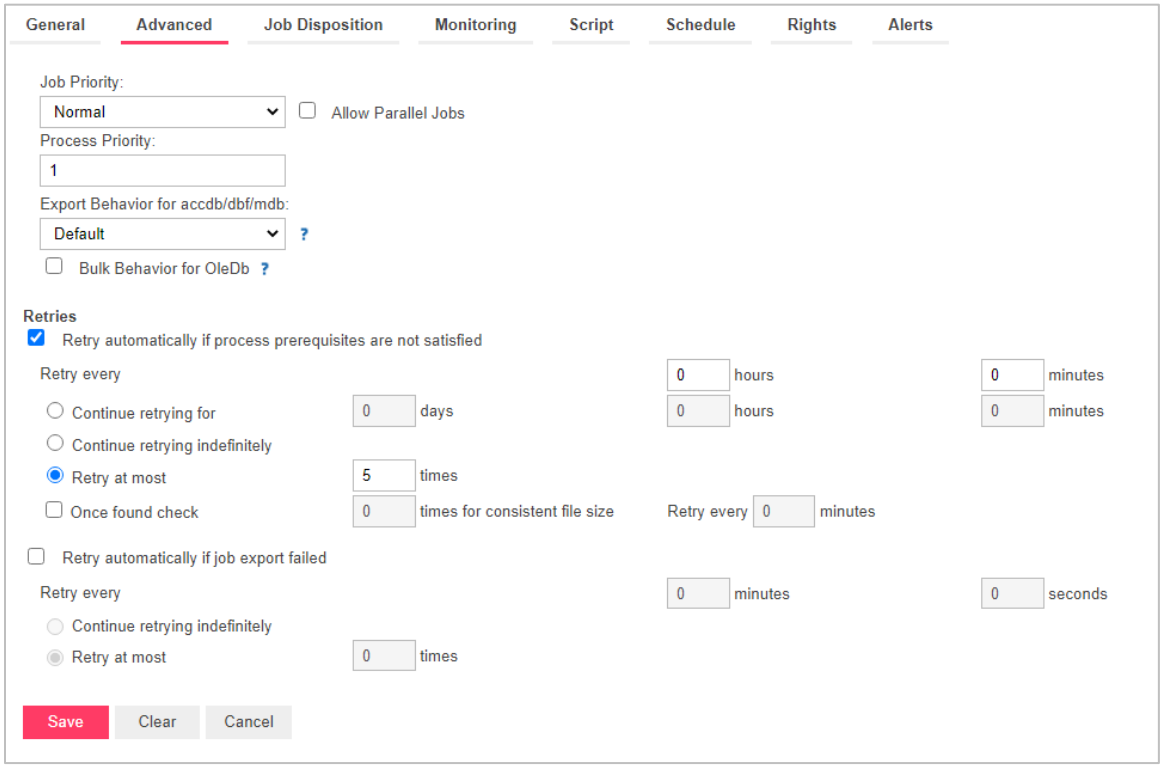
After you have created a process you can specify its priority, whether to allow parallel jobs, and whether to enable process retries.

- **A priority setting** – If two or more processes are scheduled to run at the same time, their priority settings determine the order in which they will actually be run. Any number from 1 to 100 is a valid priority setting. Processes are run in ascending order according to their priority setting (for example, a process with a priority of 1 will run before one with a priority of 2 or higher).
- **Whether or not to allow parallel jobs** – If we choose to allow parallel jobs, Automator will create a job for each of the input files assigned to the process (for example, if there are two input files assigned to the process, two jobs will be created). These jobs will then execute in parallel, up to the amount of concurrent jobs specified.
- **Whether or not to enable process retries** – Enabling process retries allows a process to be retried if it fails to run successfully (due to a prerequisites failure when a file is missing, for example). We may specify the interval for the retry in hours and minutes, as well as the number of retries to attempt. This is especially important to set

when selecting monitoring options, as Automator will try to open a file the second that file starts being written to a monitored directory. Setting process retry rules will allow Automator to pick up the file after the file is completely written to the directory.

### Steps

1. On the Navigation Tree, click **Automator**, and select **Standard Processes**.
2. On the Processes page, select the **January Sales** process by clicking anywhere on the row, and then click .
3. On the displayed page, select the **Advanced** tab.




4. In the **Process Priority** field type *1*.

   This priority setting of "1" specifies that if there were any other processes scheduled to run concurrently with our *January Sales* process, our process would run before any processes with a priority setting of "2" or higher.

> **NOTE** We don't need to select the **Allow Parallel Jobs** option because we are only working with one input file. If we were to add another input file, selecting this would ensure that Automator would create a separate job for each of them.

5. In the **Export Behavior for accdb/dbf/mdb** drop-down list, you can set the behavior of the process when exporting to .accdb, .dbf, and .mdb. The following options are available.
   - **Default** - Use this option for all new processes.

- **ACE DAO** - Use this option for better performance for large datasets. All exports in the system with this behavior will be executed concurrently. ACE OleDb will be used as fallback when output tables exist option is "Update existing rows" or "Update existing rows and append new."
- **ACE OleDb** - Use this option for better reliability. This behavior is slower than ACE DAO, but it allows to handle exports in parallel.

For this step, choose **Default.** Additional information is available in Monarch Server Automator Administrator's Guide.

6. Select the Retry automatically if process prerequisites are not satisfied check box. This activates other retry settings:
   - After **Retry every** in the **hours** field enter *1* and in the **minutes** field enter *30*, to set the amount of time that should lapse between retries.
   - Select the **Retry at most n times** option and enter *5* in the **times** field.

   These settings specify that if the process fails to run successfully, Automator should attempt to rerun it up to five times at an interval of every 90 minutes.
7. Click the **Save** button to save the current settings. The system saves the changes and displays the Process page, if the operation is completed successfully.

# SPECIFYING JOB LOG DISPOSITION SETTINGS

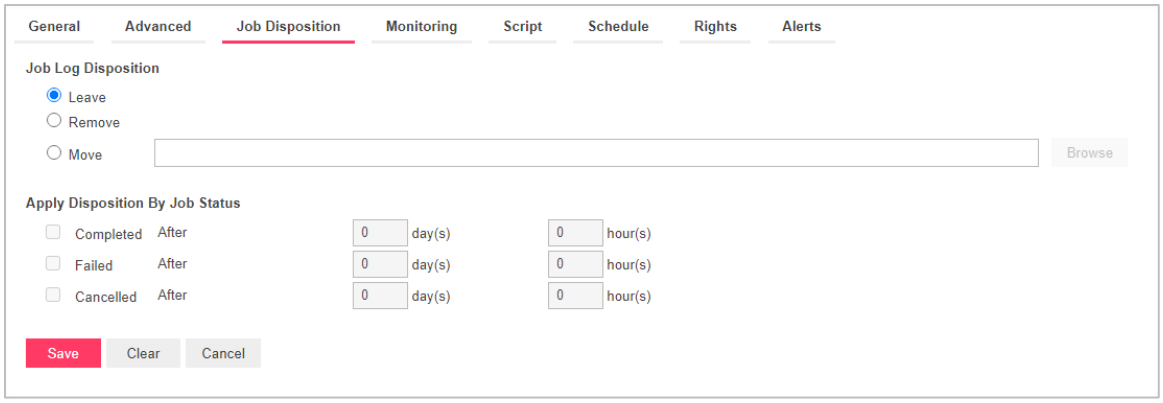
We can now specify job log disposition settings for our process. When a process runs, it spawns a job, for which a log file is generated. Job log disposition settings determine what will be done with the job log after a process is run and under what circumstances the specified disposition will occur.

After specifying job log disposition settings, we will specify a Custodian schedule. The Custodian is responsible for carrying out the specified job log disposition, which does not automatically occur when jobs are run. The Custodian allows you to schedule these operations so that a convenient time that does not conflict with busy processing times can be chosen.

**NOTE** It is possible to set default settings for job log disposition which are automatically applied to all new processes created. However, you can still override them on a per process basis if you want. For more information about configuring default settings for job log disposition, refer to the Default Job Log Disposition topic in the web help.

**Steps**

1. On the Navigation Tree, click **Automator**, and select **Standard Processes**.
2. On the Processes page, select the **January Sales** process by clicking anywhere on the row, and then click .
3. On the displayed select the **Job Disposition** tab.




4. Under **Job Log Disposition** we can choose one of the following:
   - Leave the job log where it is
   - Remove it (delete from the database)
   - Move it to another location

   The default value is **Leave**. This specifies that the job logs will remain in the database. We have no need to delete or move the job logs at this point, so we will leave this setting as it is.

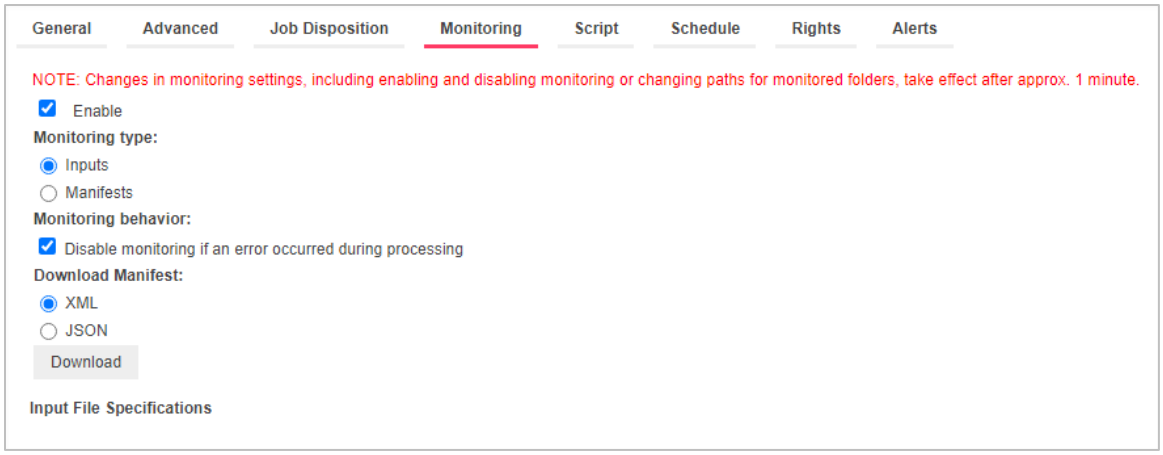
> **NOTE** If we had chosen to delete or move the job logs, we would then specify **Apply Disposition by Job Status** settings. For information on these settings, refer to the Specifying Job Log Disposition Settings topic in the web help. In addition, we would need to specify a Custodian schedule which is responsible for implementing the job log disposition settings.

# SPECIFYING MONITORING SETTINGS

When you specify monitoring settings for a process, you specify the input files or manifest XML files that you want Automator to monitor a particular location for. When the file arrives in the specified location, Automator will trigger running of the associated process.

You can enable monitoring via the **Monitoring** tab.

1. On the Navigation Tree, click **Automator**. Under **Standard Processes**, choose **Processes**.
2. Click the name of the process you want to change (*January Sales* in this case) or click anywhere on its row, and then click .
3. On the displayed page, select the **Monitoring** tab.




4. Select the **Enable** check box to enable monitoring.
5. Select the monitoring type.
   - Choose **Inputs** if the file to be monitored is an input file. Proceed to **Step 6**.
   - Choose **Manifests** if the file to be monitored is an XML file. Proceed to **Step 7**.

**NOTE** In manifest monitoring in Standard Processes, a manifest XML file should be prepared in the following format:

```
<manifest process="ProcessName">
    <inputs>
  <item name="InputName">
    <uri>file:Input path</uri>
  </item>
    </inputs>
</manifest>
```

The process for manifest monitoring is as follows:

a. Create a project file as described in Lesson 1 (the input file is a manifest XML file, which should be renamed as “manifest.xxx,” where “xxx” is the item name from the XML file).

b. Create a process. Add the project file from Step a.

c. In the **Monitoring** tab, click the **Enable** check box, and select **Manifests** under **Monitoring Type**.

d. Add the manifest XML file you prepared (refer to sample above).

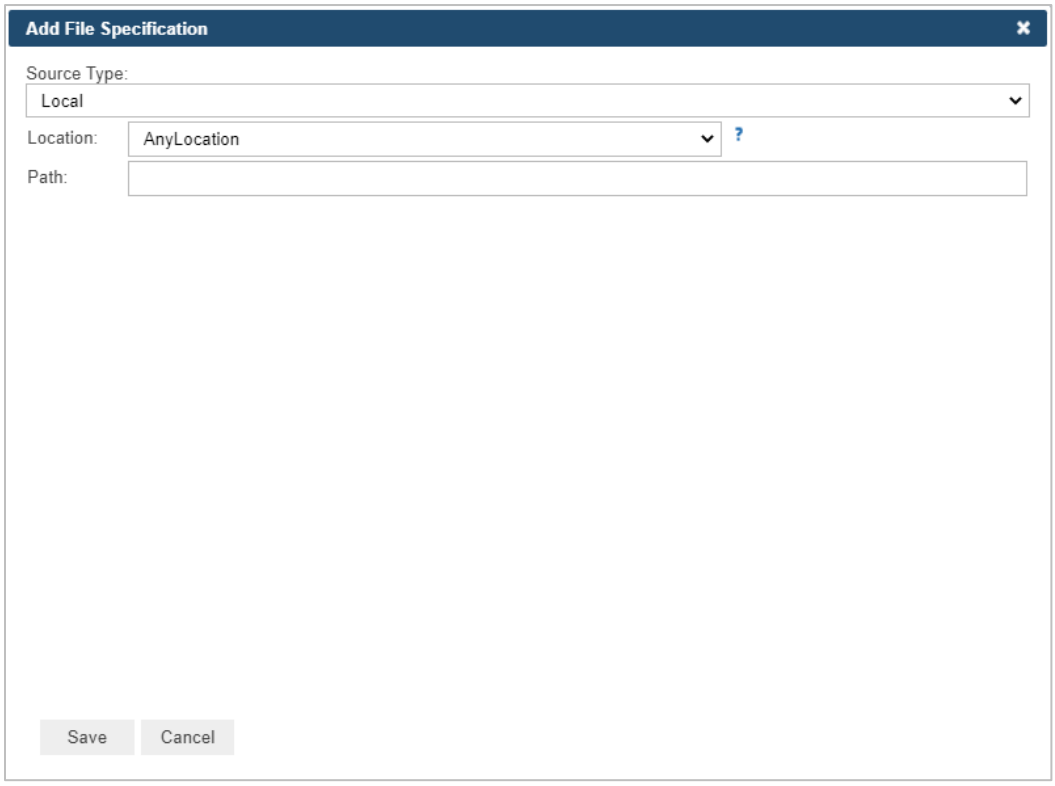
e. Select one of two options in Manifest PostProcessing.
   - Delete (selected by default): Removes the XML file after processing
   - Move (and set the folder path): Moves the XML file after processing

f. **Save** the process.

g. Proceed to the **Jobs** page under **Standard Processes** to view the job log or re-run the job.

For further details about Standard Processes with manifest monitoring, refer to the Automator Administrator's Guide.

6. Click the **Load File Specifications from Project** button to specify the way specifications will be added from the following:
   - Append to the end of the list, or
   - Replace the list
7. Click the **Add** button to select the folder you want Automator to monitor for reports. The **Add File Specification** dialog box will appear.
8. In the **Add File Specification** dialog box, specify the following parameters:




- **File Specification**: Enter the file path with the name of the file you want Automator to monitor for at the end, or select the file from the Named Paths list.

**NOTE** If you're unsure of the file's exact name or if you want Automator to monitor for any file of a particular file type, you can use a wildcard character, either an asterisk (*) or a question mark (?), so that the arrival of any file of the specified type will trigger the

running of the process. An asterisk may replace multiple characters while a question mark may replace any single character.) For example, adding *.prn to the end of the folder path instructs Automator to trigger the running of the associated process when any report file with a .prn extension arrives in the specified location.

- **Location**: Select the location type from the drop-down list.

**NOTE** If the required server is not on the list, refer to Locations Management topic to add a new location.

- From the Named Paths list, select the appropriate named path, and select the project file from the list displayed in the Named Path Content box.
- Click **Save** to add the file.

9. Click the **Save** button to save the monitoring settings.

The file from the monitoring folder can sometimes be processed with an error and be left in this folder. By selecting **Disable monitoring if an error occurred during processing,** Automator does not monitor the corresponding folder for this process. When this option is not selected, the monitoring is left enabled; when new files appear in the monitoring folder, the process running is triggered again.

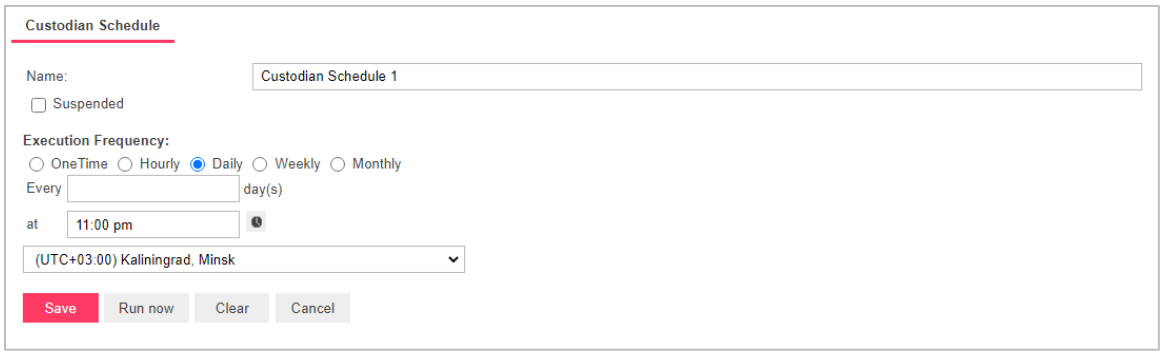
# CREATING A CUSTODIAN SCHEDULE

Since we chose the **Leave** job log disposition setting, we don't need to specify a Custodian schedule. But let's see how easily we could do it if we needed to.

For example, if we specified the **Move** job log disposition setting, we might want to have the job logs moved daily at 11 o'clock p.m.

### Steps

1. On the Navigation Tree, click **System Settings**, and select **System Preferences**.
2. At the top of the page under the **Automator** tab, select the **Custodian Schedules** tab.
3. On the displayed page, click the **Add** icon .



4. On the displayed page specify the name you want to assign to the schedule. We will enter *Custodian Schedule 1*.
5. Under **Execution Frequency**, select the **Daily** option.
6. Set the **Every…day(s)** field to 1 to indicate that the current process will recur every day.
7. Set the start time of *11:00 PM*.
8. Select the appropriate timezone from the drop-down list. We will select *(UTC+03:00) Kaliningrad, Minsk*.
9. Click the **Save** button, to save the schedule.

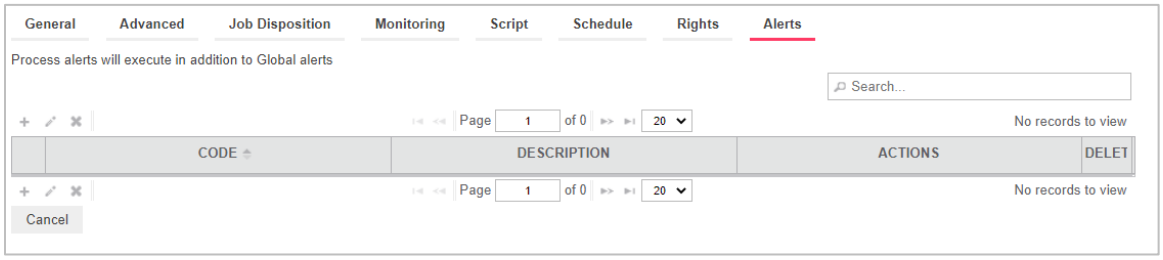
So far we have configured a project file (Automator Tutorial) and a process (January Sales) and assigned the project file to the process. Now we are ready to try running the process. Processes can be run manually or they can be scheduled to run on a particular day at a specific time. Let's go on to Lesson 3, where we will try running our process to see if it successfully extracts and exports the data we want.

# CREATING ALERTS

Automator provides an alerting mechanism whereby you can be notified based upon certain conditions. You can, for example, be notified via email if a process fails to run. These notifications are called alerts. Some alerts are pre-defined within Automator itself, such as the process failure (SystemAlertProcessFailed) and job failure (SystemAlertJobFailed) alerts, while others can be defined by users in global, pre-export or post-export scripts, using the AddEvent method from the JobLog object.

### Steps

1. On the Navigation Tree, click **Automator**. Under **Standard Processes**, choose **Processes**.
2. On the Processes page select the January Sales process, by clicking anywhere on the row, and then click the Edit icon.
3. On the displayed select the **Alerts** tab.




4. On the **Alerts** page, click the **Add** icon to create a new alert.
5. In the **Code** box specify the alert type.

**NOTE** You may type a new alert name or select one of the following predefined alert types:

- SystemAlertProcessFailed: Indicates that a process has failed.
- SystemAlertJobFailed: Indicates that a job has failed.

- SystemAlertNoInputToExporter: Indicates that there was no input to the Exporter (which means, that no input fileexists).
- SystemAlertDistributionError: Indicates that a distribution has failed.
- SystemAlertVerifyFailed: Indicates a project verification failure.

If a predefined alert (for example, the SystemAlertJobFailed) has already been selected, it will no longer be listed in the Code drop-down list, as you cannot specify two predefined alerts of the same type.

6. In the **Description** field enter alert description.
7. Click **Add Actions** to add actions to the alert.

# LESSON 3 - RUNNING AND SCHEDULING PROCESSES

In Lesson 2, we learned how to create and configure a Automator process. In this lesson, we will concentrate on running processes. Processes can be run in different ways:

- Manually
- Via a process schedule
- Via input file monitoring

In this lesson, we will learn how to run the process manually and via a schedule. We will also be introduced to jobs.

The input file monitoring settings enable you to set monitoring of a particular location, then trigger the running of a process when a specific input file (or any file of the specified file type) appears in that location. For the information about specifying input file monitoring settings for a process, refer to the corresponding topic in the web help.

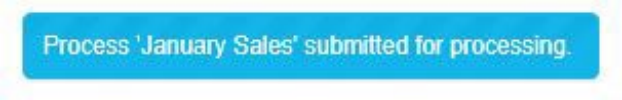
## RUNNING A PROCESS

After a process has been configured, you may want to run it manually to make sure that it works properly. Once you have determined that it does, you could then schedule it to run when you actually need it to.

Let's try running our process manually.

**Steps**

1. On the Navigation Tree, click **Automator**. Under **Standard Processes**, choose **Processes**.
2. On the displayed page select the **January Sales** process and click the **Run Now** icon .
3. Automator displays a Submitted for Processing message.




When a process is launched Automator generates and executes a job (or jobs) for the process. Let's take a few moments and learn about jobs.

## ABOUT JOBS

A job is the resulting task for one or more projects assigned to the job's associated process. When a process runs, it ultimately spawns one or more jobs. Each job then appears as a row on the Jobs page. You can view the list of jobs, their states, and their associated job logs.

To view the Jobs page, on the Navigation Tree, double-click **Automator** and select **Jobs**. You may delete jobs, cancel jobs, and even re-run jobs that have failed due to process prerequisites that haven't been met, for example.

The Jobs page displays the following information:

- The job's unique ID number

- The name of its associated process (January Sales)
- The status of the job, which is currently “Job completed without errors”
- The date and time when the job was registered
- The date and time when the job was last processed

To re-run the job, click the **Re-run** icon .

To view the jobs log, click the **View Job Log** icon .

To check the most current job status, click the **Reload Grid** icon .

To delete the job, click the **Delete Selected Job** icon .

To cancel the job, click the **Cancel** icon .

Depending on the time needed for jobs to execute and other various factors, the following statuses may be returned:

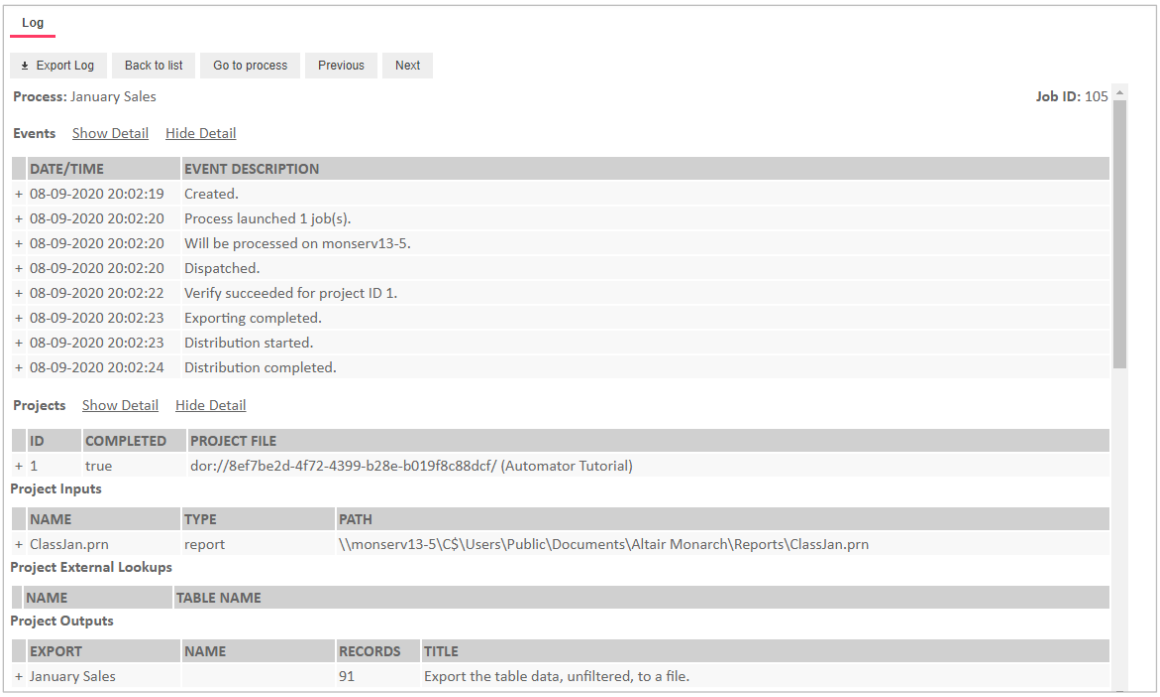
- If the job is not yet complete, the status of the job will read "Job running".
- If the job has finished running, it will read "Job Completed without errors".
- If the job fails to run successfully, consult web help for more information.

# VIEWING A JOB'S LOG FILE

For each job that runs, a log file is generated. A log file provides detailed information about the job, such as its associated process, the time at which it was run, the project file(s) utilized, the export(s) performed, and so on. Let’s take a look at our job’s log file.

**Steps**

1. On the Navigation Tree, click **Automator.** Under **Standard Processes**, select **Jobs**.
2. On the displayed page, select the job with the ID number *101*.
3. Click the **View Job Log** icon . The job's Log will be displayed.



Log

Export Log | Back to list | Go to process | Previous | Next

**Process:** January Sales **Job ID:** 105

**Events** Show Detail Hide Detail

| DATE/TIME | EVENT DESCRIPTION |
|---|---|
| + 08-09-2020 20:02:19 | Created. |
| + 08-09-2020 20:02:20 | Process launched 1 job(s). |
| + 08-09-2020 20:02:20 | Will be processed on monserv13-5. |
| + 08-09-2020 20:02:20 | Dispatched. |
| + 08-09-2020 20:02:22 | Verify succeeded for project ID 1. |
| + 08-09-2020 20:02:23 | Exporting completed. |
| + 08-09-2020 20:02:23 | Distribution started. |
| + 08-09-2020 20:02:24 | Distribution completed. |

**Projects** Show Detail Hide Detail

| ID | COMPLETED | PROJECT FILE |
|---|---|---|
| + 1 | true | dor://8ef7be2d-4f72-4399-b28e-b019f8c88dcf/ (Automator Tutorial) |

**Project Inputs**

| NAME | TYPE | PATH |
|---|---|---|
| + ClassJan.prn | report | \\monserv13-5\C$\Users\Public\Documents\Altair Monarch\Reports\ClassJan.prn |

**Project External Lookups**

| NAME | TABLE NAME |
|---|---|

**Project Outputs**

| EXPORT | NAME | RECORDS | TITLE |
|---|---|---|---|
| + January Sales | | 91 | Export the table data, unfiltered, to a file. |

4. When you have finished examining the job log, click the **Back to list** button to exit.

Now that we have run the process and viewed the resulting job and job log, let's see if the export we configured when creating the Automator Tutorial project file was successful.

# VIEWING THE OUTPUT FILE

When we created the Automator Tutorial project file, we configured an export to extract data from the ClassJan report and export it to the FirstQuarterSales2010 output file. Let's see if the export was successful.

1. Using Windows Explorer, navigate to the **Exports** folder typically located in *C:\Users\Public\Documents\Altair Monarch\Export*.

2. You will see that it now contains the **FirstQuarterSales2010** spreadsheet. Let's examine the spreadsheet to make sure that the ClassJan report data were successfully extracted and exported.

   Double-click the **FirstQuarterSales2010** spreadsheet to open it in MS Excel. The ClassJan report data was extracted and exported to a "JanuarySales" worksheet within the output file.

3. Close the **FirstQuarterSales2010** spreadsheet.

# SCHEDULING A PROCESS

In addition to running processes manually, as we already did with the Automator Tutorial, we can schedule a process to run at a specific time on a particular day and at a definite interval. Let's try scheduling a process to run.

Since we have already run the **January Sales** process manually, we need to create a new process to specify a schedule for it. Let's create a new **February Sales** process.

### Steps

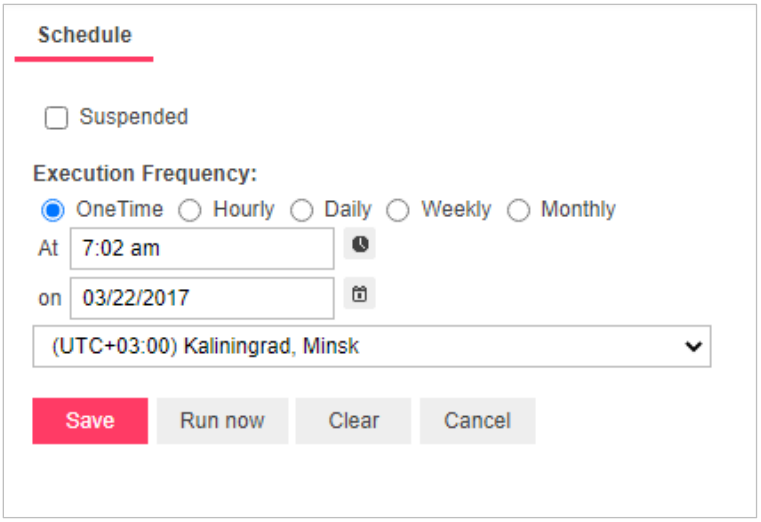
1. Create a new ClassFeb project. For instructions, refer to the "Creating a Project File with the Project Editor" topic.

> **NOTE** Specify the ClassFeb.prn as an input file for the ClassFeb project located at *C:\Users\Public\Documents\Altair Monarch\Reports\ClassFeb.prn*.

2. Create a new February Sales process on the Processes page according to the workflow described in the "Creating a Process" topic.

> **NOTE** When assigning a project file to the process select the ClassFeb project from the Server Library.

3. After you specify parameters of the **February Sales** process, click the **Save/Edit** button.
4. On the displayed page, select the **Schedule** tab.
5. On the Schedule tab, click the **Add New Schedule** icon .




6. On the displayed page, under **Execution Frequency** select the **One Time** option since we only need to run our process once.

> **NOTE** If we needed to run the process on a hourly, daily, weekly, or monthly basis we could select the corresponding option. For detailed information about execution frequency settings, refer to the "Execution Frequency" topic in the web help.

7. In the **At** field, set the time you want the process to run. We will add five minutes to the current time (for example, if the current time is *6:57*, enter *7:02*).

8. In the **On** field, set the start date of the process. For this click the **Calendar** icon adjacent to the field and choose the current date.

9. Select the appropriate timezone from the drop-down list. We will select *(UTC+03:00) Kaliningrad, Minsk*.

10. Click the **Save** button, to save the schedule.

In five minutes, when the specified run time arrives, the process will be run, spawning a job on the Jobs page. If the process fails to run, you can consult the web help for possible explanations. You can also refer to the project's log file for additional information.

After the **February Sales** process runs, check the **FirstQuarterSales2010** output file to see if the data were successfully extracted and exported from the ClassFeb report.

#### Steps

1. Using Windows Explorer, navigate to the **Exports** folder typically located in *C:\Users\Public\Documents\Altair Monarch\Export*.

2. Double-click the **FirstQuarterSales2010** spreadsheet to open it in MS Excel. You will see that the output file now contains **January Sales** and **February Sales** worksheets.

3. Proceed to the **February Sales** worksheet to make sure that the data from the ClassFeb report has been successfully extracted.

4. Close the **FirstQuarterSales2010** spreadsheet.

Now that we've run processes both manually and via a scheduled runtime, let's go on to Lesson 4, where we will work with distributions.

# LESSON 4 - WORKING WITH DISTRIBUTIONS

This lesson deals with distributions and explains what they are then shows you how to configure them.

## ABOUT DISTRIBUTIONS

Distributions are actions that distribute or handle information or files outside of Automator.

For example, distributions can include:

- The sending of e-mails (with or without attachments)
- RSS feeds
- Moving, copying and deleting files (to both the file system and to FTP and WebDAV-compliant servers using http)
- Writing to the Windows Event Log

Distributions can be specified for input files, output files, and portable reports, and are configured with the Distribution Editor.

There are two types of distributions that can be specified:

- **Global Distributions** – These distributions are defined in the Automator application and can apply to any input files that are used in all processes and any output files and portable reports that are produced by all processes in the specified folders. For more information about global distributions, consult the web help.
- **Project-Specific Distributions** – Project distributions are defined in the Monarch Complete and apply to the input, output, or portable report files of only the specified project. They do not apply across all projects in a process, as global distributions do.

## ADDING A GLOBAL DISTRIBUTION

Let's try adding a global distribution to the project file. Remember that global distributions apply to any input files that are used in all processes when run, as well as to any export files and portable reports that are produced by all processes in the folders specified.

### Steps

1. On the Navigation Tree, click **Show Advanced options**. Under **Automator**, click **Standard Processes**, and select **Distributions**.
2. Click the **Add** icon to display the Distribution page.

3. In the **Location** field select your server location name from the drop-down list.

> **NOTE** If the required server is not on the list, refer to the "Locations Management" topic in the web help.

In case of a single server configuration refer to "Monarch Server Configurator Guide" to change location settings.

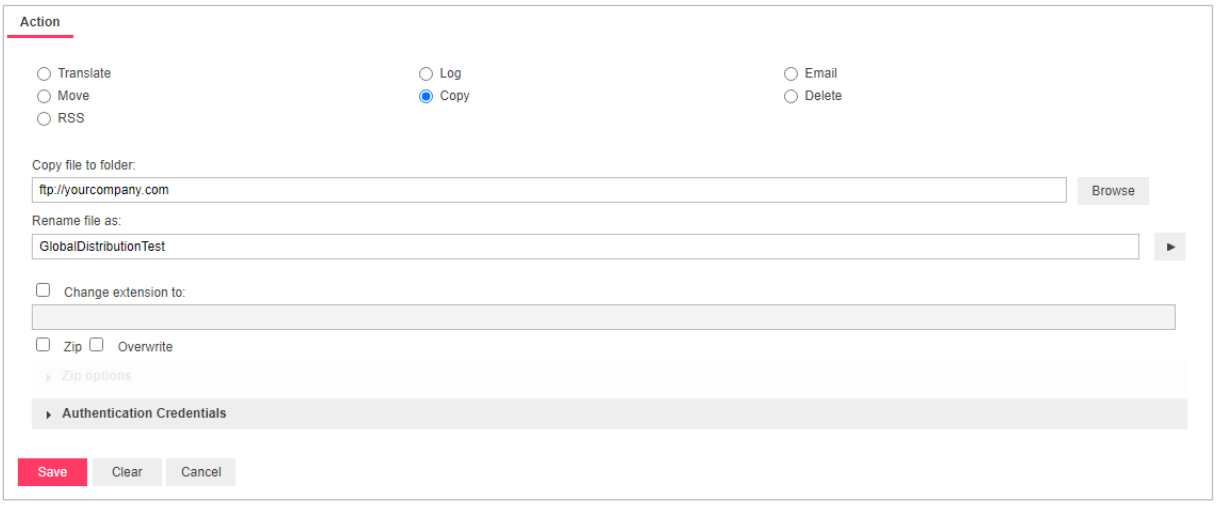
4. In the **Path** field specify the path to the Exports folder: *C:\Users\Public\Documents\Altair Monarch\Export*.

   This field is used to specify the folder where the file(s) that we want to distribute will be located. We entered the path to the output file in order to distribute the output file of a process.

5. Click the **Add Actions** button to add action to the distribution.

6. On the displayed page click the **Add New Action** icon . The **Action** page where we can select the kind of distribution action to occur will be displayed.

7. Let's assume that we want our global distribution to copy project output files. To do so, we should select the **Copy** option.




8. In the **Copy file to folder** field, specify the folder to which we want to copy the output file(s). Let's assume we want to copy the output file(s) to a company's FTP site so we enter the URL of the company's FTP site (for example, ftp://yourcompany.com).

> **NOTE** You may not follow along with the tutorial if you don't have an FTP site to which you can copy files. If you prefer, you may copy the files to a local or network folder by using the **Browse** button to select the location. For more information, refer to the

Copy Distribution Action topic in the web help.

9. In the **Rename file as** field enter *GlobalDistributionTest*.

10. Click the **Insert Macro** button to display the drop-down list of file-naming macros, and then select **Job Number**. This will add the job number to the file name we specified.

11. Click the **Authentication Credentials** drop-down arrow, which appears when you enter "ftp://" or "http://" in the folder path, to specify the authentication credentials:

    - Enter your username and password in corresponding fields
    - Specify domain (if required)
    - Select the Ascii check box to specify an ASCII mode transfer for your FTP distribution rather than binary mode, which is the default

**NOTE** ASCII Mode should only be used for text-only files such as TXT, ASP, HTML, while binary mode can be used for anything else, for example, DOC, ZIP, XLS. If in doubt as to which transfer mode to use, go with the default mode of binary (to do so, do not select this check box).

12. Click the **Save** button to add the action and save the distribution.

    Note that the distribution is now listed on the Distributions page. If you were to run this process, the output file would be copied to your company's FTP site.

# ADDING AN RSS GLOBAL DISTRIBUTION

One of the features of Automator is the ability to create RSS feeds. RSS (Really Simple Syndication) is an XML-based format for content distribution. It is a widely supported standard and a very useful "push" method of distributing information from Automator. RSS can be read by a variety of free and commercial client products, such as Internet Explorer, RSS Bandit, SharpReader, and Newsdesk, as well as be integrated with portals and other web server software to provide server-side viewing of RSS. Automator supports the creation of RSS 2.0.

**NOTE** For more information on RSS version 2.0, consult the specifications at http://blogs.law.harvard.edu/tech/rss. Further information on RSS is freely available on the Internet.

Let's explore this capability of Automator by creating an RSS global distribution that will "publish" information and the location of the project's output file to an RSS feed.

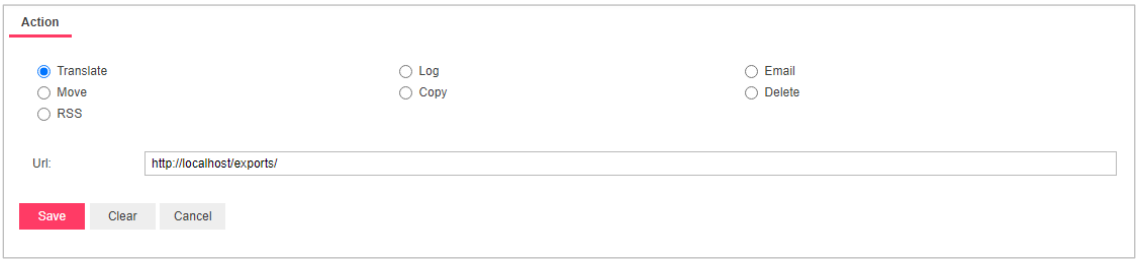
# DEFINING A GLOBAL TRANSLATE DISTRIBUTION

Before we actually add our RSS distribution, we must define a global translate distribution to assign a URL to the output file. This translate distribution translates a file system location, where the file is to be exported, to a URL.

> **NOTE** The file system location must be Internet accessible (for example, through an IIS Virtual Folder) for an RSS distribution to work. It is possible to perform a translate distribution even when the eventual location of the file may be on a remote web server, for example, a UNIX web server running Apache. You can simply create the translate distribution to point to the initial export folder on the local machine, but create the translate distribution to refer to the UNIX web server and use a copy distribution to FTP the output file to the UNIX machine.

### Steps

1. On the Navigation Tree, click **Show Advanced options**. Under **Automator**, click **Standard Processes**, and select **Distributions**.
2. On the Distributions page, select the distribution we have created and then click to open it for editing.
3. On the displayed page, click the **Add New Action** icon . The **Action** page where we can select the kind of distribution action to occur will be displayed.
4. Select the **Translate** option.




5. In the **URL** field enter *http://localhost/exports/.* In this field we specify the internet address we want to apply to the selected folder when using a global RSS distribution.

> **NOTE** We are using "localhost" as the web server name here, which refers to the web server on the local machine. Instead of "localhost", you can enter the name of your web server.

6. Click **Save** to save the URL.

# CONFIGURING THE RSS GLOBAL DISTRIBUTION

Now we can specify our global distribution to distribute the output file as an RSS feed.

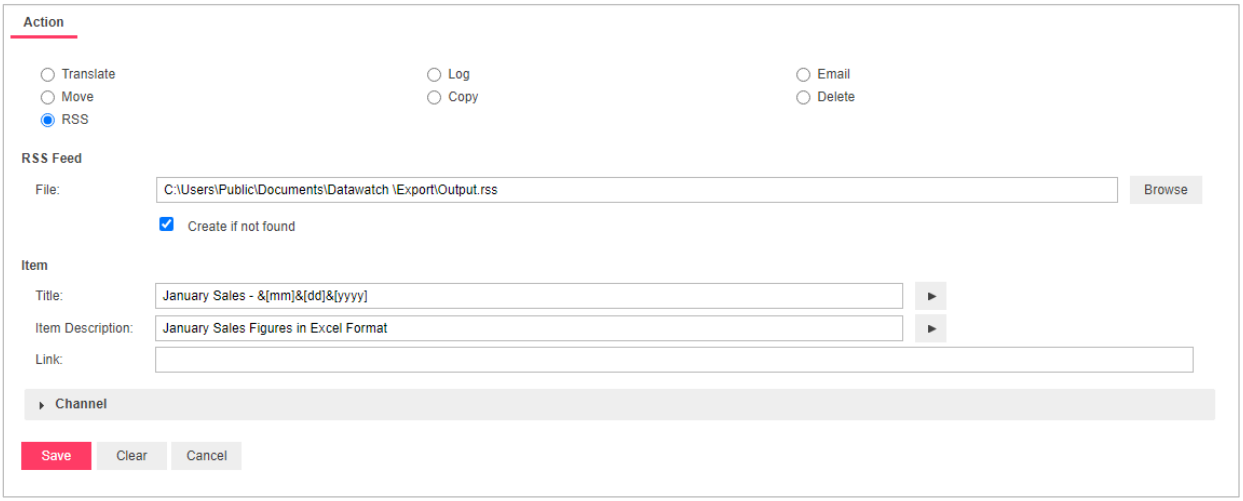
### Steps

1. On the Navigation Tree, click **Automator**. Under **Standard Processes**, select **Distributions**.

2. On the Distributions page, select the distribution we have created and then click , to open it for editing.

3. On the displayed page, click the **Add New Action** icon . The **Action** page where we can select the kind of distribution action to occur will be displayed.

4. Select the **RSS** option.




5. In the **File** field specify the RSS file (also known as RSS channel or feed) we want to create. We will enter the path to the file in the Export folder: *C:\Users\Public\Documents\Altair Monarch \Export\Output.rss*. This file doesn't exist yet, but when the process is run, Automator will generate it.

> **NOTE** The folder we specify should be web accessible at *http://YourWebServerNameHere/exports*. An RSS file must be web accessible, whether or not on the company intranet or the public Internet.

6. Select the **Create if not found** check box, since the **Output RSS** file does not currently exist. If we had selected an already existing RSS file, we would not select this check box.

7. In the **Title field**, enter a title for the RSS item. You can enter your own text here and also use the **Insert Macro** icon to add a naming macro. You could, for example, enter something like *January Sales - &[mm]/&[dd]/&[yyyy]*.

8. In the **Description** field, enter a description of the RSS item. You can add your own text here and also use the **Insert Macro** icon to add a naming macro. For example, you could enter something like *January Sales Figures in Excel Format*.

9. Leave the **Link** field blank so that Automator can automatically assign a URL, based on the global translate distribution we defined earlier, from the output file that is being processed.

10. We can also click the **Channel** drop-down arrow to display the advanced properties, used to customize the title, description and the URL of the RSS channel we have created, as well as specify how many items we want to allow in the channel before overwriting old items.

11. Click the **Save** button to add the action and save the distribution.

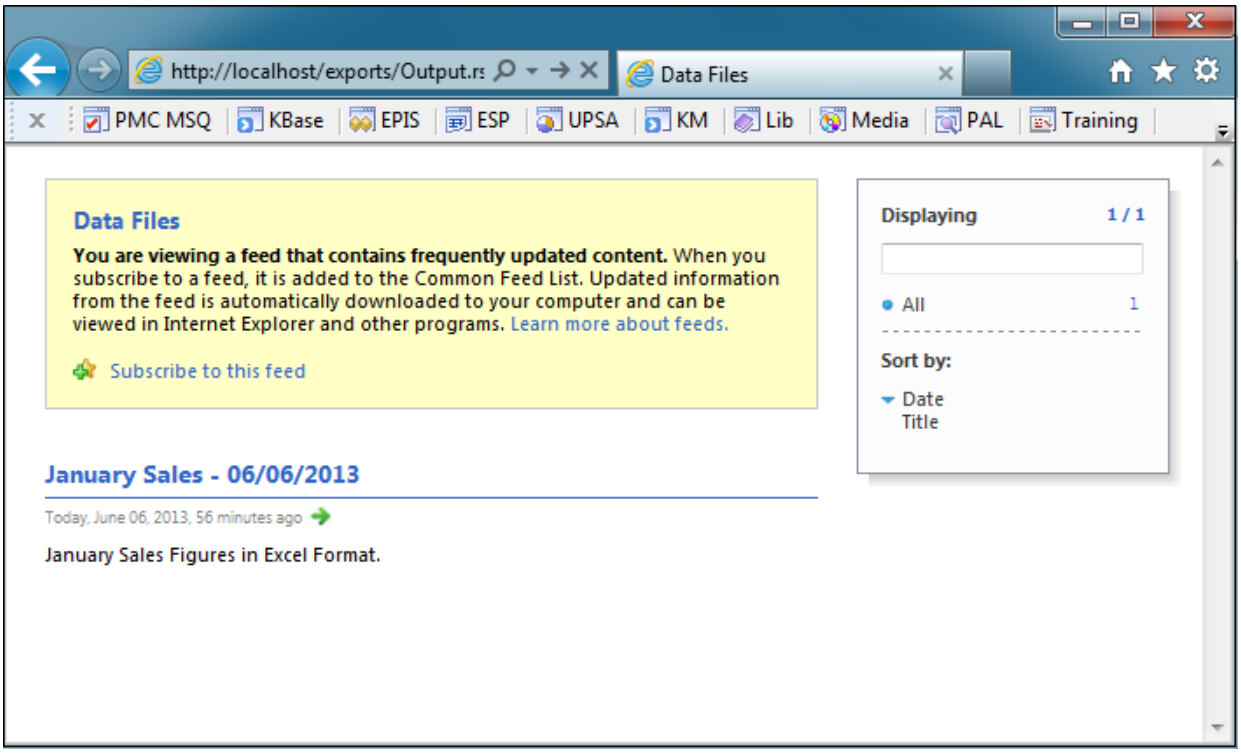
If you want to, you can now run the process by selecting it and clicking the **Run Now** icon . The export and RSS file will then be created. In order to take a look at the RSS file, we can use Internet Explorer 7 or higher. (There are many free RSS readers available and RSS can also be served from a web server or portal software.)

> **NOTE** If you don't have an RSS-capable application, simply follow along with the tutorial.

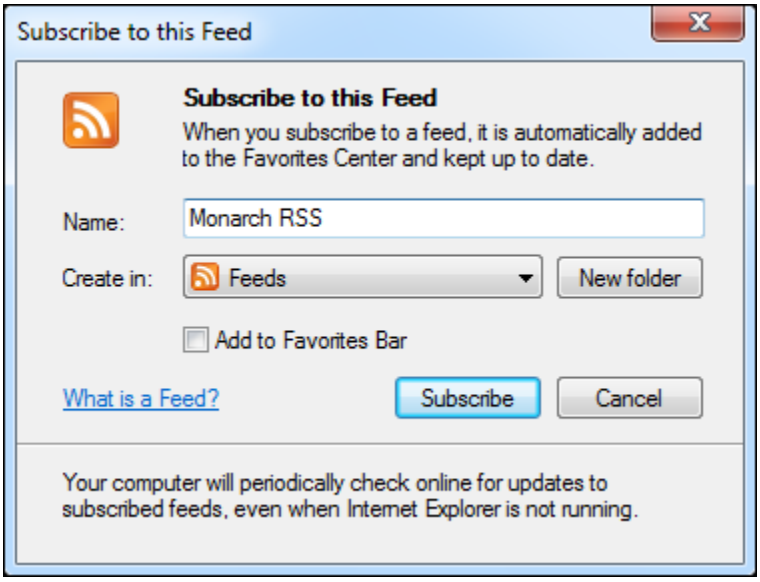
### Adding an RSS Feed

In Internet Explorer, we need to add a new feed by entering the URL to the feed, which in our case is *http://localhost/exports/output.rss*.

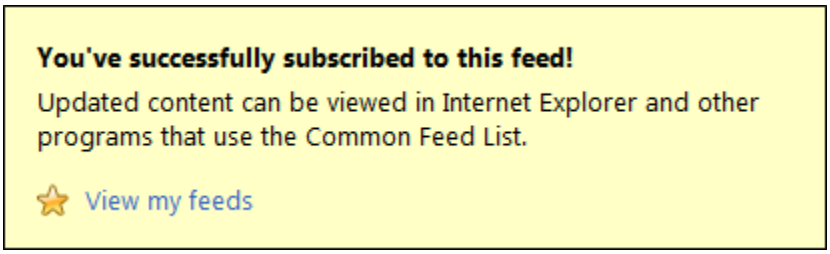
1. Open Internet Explorer and in the URL field, enter the URL to our RSS feed, for example, *http://localhost/exports/output.rss*.



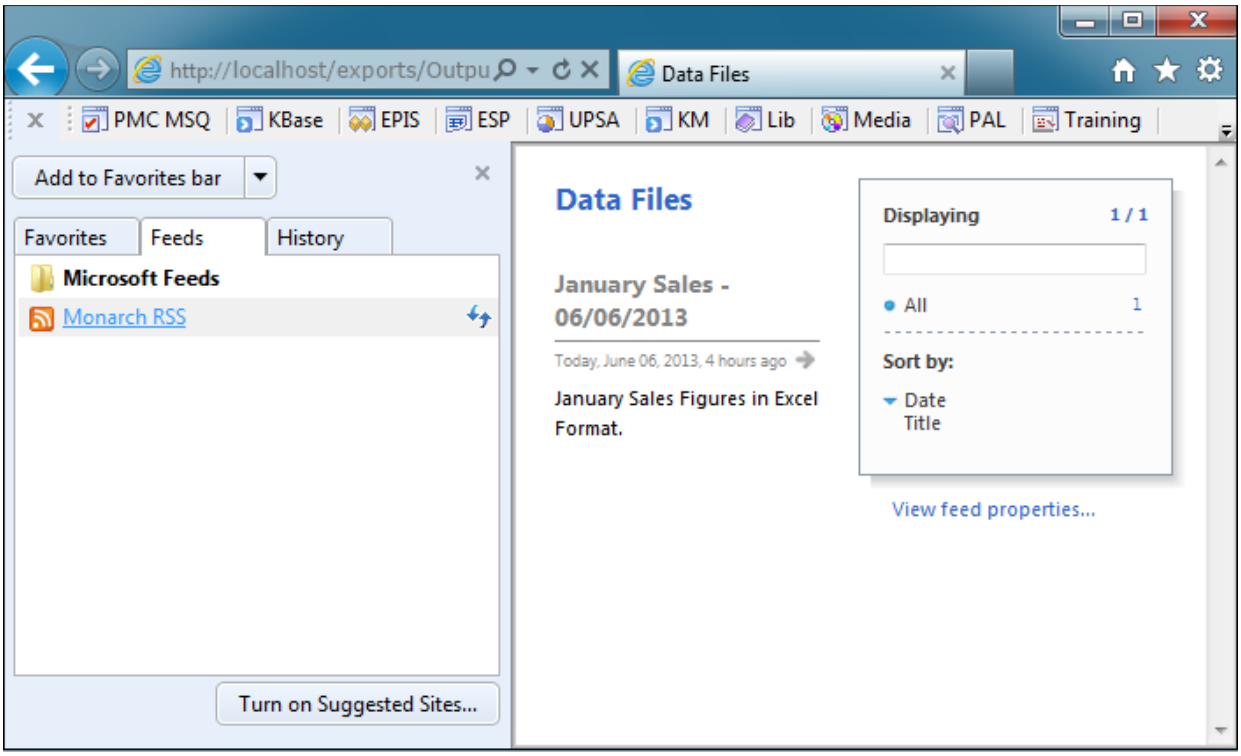

2. Click the Subscribe to this feed link.




3. In the **Name** field, we can manually enter whatever we would like, such as *Monarch RSS*.
4. Click **Subscribe** to save the settings, subscribe to the feed, and close the dialog. Internet Explorer will inform you that you have successfully subscribed to the feed.




5. Open Internet Explorer and click the **Favorites** icon to proceed to Favorites Center.
6. In the Favorites Center section, select the **Feeds** tab and select **Monarch RSS** to view the feed.




Now that we have familiarized ourselves with distributions, let's take a few minutes to learn about alerts.

# LESSON 5 - WORKING WITH ALERTS

This lesson deals with alerts and explains what they are then shows you how to configure them.

## ABOUT ALERTS

Automator provides an alerting mechanism that sends notifications based on certain conditions. You can, for example, be notified via email if a process fails to run. These notifications are called alerts. Some alerts are pre-defined within Automator itself, while others can be defined in global, pre-export or post- export scripts, using the AddEvent method from the JobLog object.

Automator includes the following pre-defined alerts:

- **SystemAlertJobFailed** - Provides notification that a job has failed.
- **SystemAlertNoInputToExporter** - Provides notification that there was no input to the Exporter (that is, input file does not exist).
- **SystemAlertProcessFailed** - Provides notification that a process has failed.
- **SystemAlertDistributionError** - Indicates that a distribution has failed.
- **SystemAlertVerifyFailed** – Indicates that a project verification failure has occurred.
- Existing alerts are listed on the Alerts page. Once an alert has been defined, one of the following distribution actions should be assigned to it:
- **Email** - Select this to specify an email action.
- **RSS** - Select this to specify an RSS action.
- **Log** - Select this to specify an Application Event Log action.

## ADDING AN ALERT

Let's add an alert to our Automator Tutorial project file.

**Steps**

1. On the Navigation Tree, click **Show Advanced options**. Under **Automator**, click **Standard Processes**, and select **Alerts**.
2. On the Alerts page, click the **Add** icon + to create a new alert.
3. From the Code drop-down list, select **SystemAlertProcessFailed**. This predefined alert will notify us if any process fails to run successfully.

   Note that Automator automatically enters *A process has failed* in the **Description** field.
4. Click the **Add Actions** button to add actions to the alert.
5. On the displayed page click the **Add New Action** icon +. The Action page where we can select the kind of alert action to occur will be displayed.
6. Let's choose the **Email** action, which is selected by default. By doing so, we will be alerted, via an email message, if a process fails to run.

7. Specify the recipients' email addresses in the corresponding boxes: **To**, **Cc**, **Bcc**.

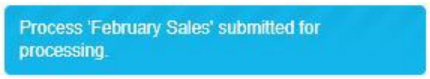
> **NOTE** You can also select recipients from the contacts list by clicking the **Contacts** icon . For more information, refer to the "Email Alert Action" topic in the web help.

8. In the **Subject** field enter *A process has failed*.
9. Click the **Save** button to add the action and save the alert. The "SystemAlertProcessFailed" alert is now listed on the Alerts page.

Let's see how this alert will work.

### Steps

1. On the Navigation Tree, click **Server Library**, and select **Projects**. The Projects page is displayed.
2. Select the **ClassFeb** project file from the list, and click the **Edit Selected Project** icon .
3. On the project file page of ClassFeb, click the **Open in Project Editor** button. The Project Editor window displays.
4. In the Project Editor, select the **Input** tab.
5. Enter a non-existent input file, e.g., *Classic999.prn*, by double-clicking on the field for the file path. Click **Commit,** and click **Commit** again in the **Save Project to Server Library** dialog box.
6. Let's return to Automator. On the Navigation Tree, click **Automator,** and select **Processes** under **Standard Processes.**
7. **Select** the **February Sales** process, and then click the **Run Now** icon .
8. Automator displays a Submitted for Processing message.




Proceed to the **Jobs** page. You will notice that the status text for the job you just ran displays as "Process prerequisites failed, but no retry information is available. Process will not be retried."

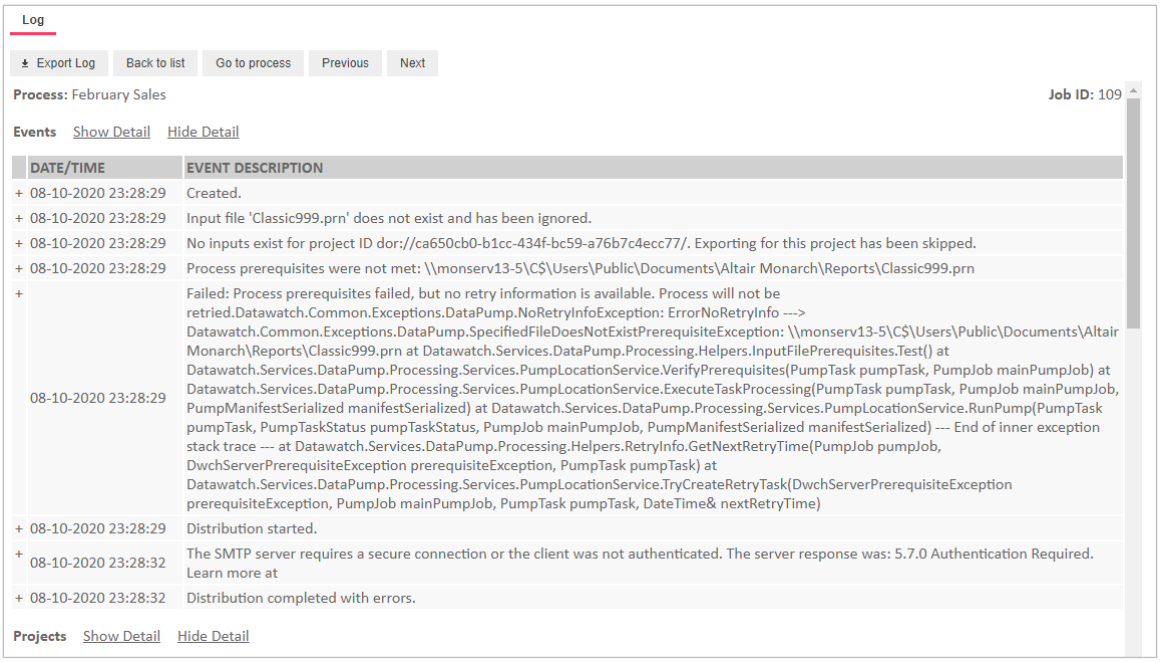
In addition to the message on the Jobs page, you should receive an email message with "A process has failed" in the **Subject** line. This is the result of the SystemAlertProcessFailed alert that we configured earlier, and for which we specified an email action.

You can view the log file for failed process through the Jobs page.

Let's view the job's log file for the failed process we created previously.

### To view the job's log file for the failed process

1. On the Navigation Tree, click **Automator**. Under **Standard Processes,** select **Jobs**.
2. On the displayed page, select the failed process.
3. Click the **View Job Log** icon . The Job's log will be displayed.



Log

Export Log | Back to list | Go to process | Previous | Next

**Process:** February Sales **Job ID:** 109

**Events** Show Detail Hide Detail

| | DATE/TIME | EVENT DESCRIPTION |
|---|---|---|
| + | 08-10-2020 23:28:29 | Created. |
| + | 08-10-2020 23:28:29 | Input file 'Classic999.prn' does not exist and has been ignored. |
| + | 08-10-2020 23:28:29 | No inputs exist for project ID dor://ca650cb0-b1cc-434f-bc59-a76b7c4ecc77/. Exporting for this project has been skipped. |
| + | 08-10-2020 23:28:29 | Process prerequisites were not met: \\monserv13-5\C$\Users\Public\Documents\Altair Monarch\Reports\Classic999.prn |
| + | 08-10-2020 23:28:29 | Failed: Process prerequisites failed, but no retry information is available. Process will not be retried.Datawatch.Common.Exceptions.DataPump.NoRetryInfoException: ErrorNoRetryInfo ---> Datawatch.Common.Exceptions.DataPump.SpecifiedFileDoesNotExistPrerequisiteException: \\monserv13-5\C$\Users\Public\Documents\Altair Monarch\Reports\Classic999.prn at Datawatch.Services.DataPump.Processing.Helpers.InputFilePrerequisites.Test() at Datawatch.Services.DataPump.Processing.Services.PumpLocationService.VerifyPrerequisites(PumpTask pumpTask, PumpJob mainPumpJob) at Datawatch.Services.DataPump.Processing.Services.PumpLocationService.ExecuteTaskProcessing(PumpTask pumpTask, PumpJob mainPumpJob, PumpManifestSerialized manifestSerialized) at Datawatch.Services.DataPump.Processing.Services.PumpLocationService.RunPump(PumpTask pumpTask, PumpTaskStatus pumpTaskStatus, PumpJob mainPumpJob, PumpManifestSerialized manifestSerialized) --- End of inner exception stack trace --- at Datawatch.Services.DataPump.Processing.Helpers.RetryInfo.GetNextRetryTime(PumpJob pumpJob, DwchServerPrerequisiteException prerequisiteException, PumpTask pumpTask) at Datawatch.Services.DataPump.Processing.Services.PumpLocationService.TryCreateRetryTask(DwchServerPrerequisiteException prerequisiteException, PumpJob mainPumpJob, PumpTask pumpTask, DateTime& nextRetryTime) |
| + | 08-10-2020 23:28:29 | Distribution started. |
| + | 08-10-2020 23:28:32 | The SMTP server requires a secure connection or the client was not authenticated. The server response was: 5.7.0 Authentication Required. Learn more at |
| + | 08-10-2020 23:28:32 | Distribution completed with errors. |

**Projects** Show Detail Hide Detail

Note that in the illustration above the first line states:

*Input file 'Classic999.prn' does not exist and has been ignored.*
*No inputs exist for project…*
*Failed: Process prerequisites failed, but no retry information is available. Process will not be retried.Datawatch.Common.Exceptions.DataPump.NoRetryInfoException: ErrorNoRetryInfo*

# RE-RUNNING A FAILED JOB

Let's restore the original input file for the **February Sales** process and try re-running the job for the ClassFeb project.

### Steps

1. On the Navigation Tree, click **Server Library,** and select **Projects.** The Projects page is displayed.
2. Select the **ClassFeb** project file from the list, and click the **Edit Selected Project** icon .
3. On the project file page of ClassFeb, click the **Open in Project Editor** button. The Project Editor window displays.
4. In the Project Editor, select the **Input** tab.
5. In the input file field, enter the correct file name, i.e., *ClassFeb.prn.* Click **Commit,** and click **Commit** again in the Save Project to Server Library dialog box.
6. Navigate to the **Jobs** page. On the Navigation Tree, click **Automator**, and select **Jobs** under **Standard Processes**.
7. On the Jobs page, select the failed job, and click the **Re-run** icon .
8. Automator displays the **Job re-run successfully started** message.
9. Click the **Refresh Jobs** icon button on the Jobs page. Continue clicking the **Refresh Jobs** icon until the "Job completed without errors" message displays.

Because the process successfully ran this time, you will also notice that you didn’t receive an email notification of its failure.

# LESSON 6 - WORKING WITH VISUAL PROCESSES

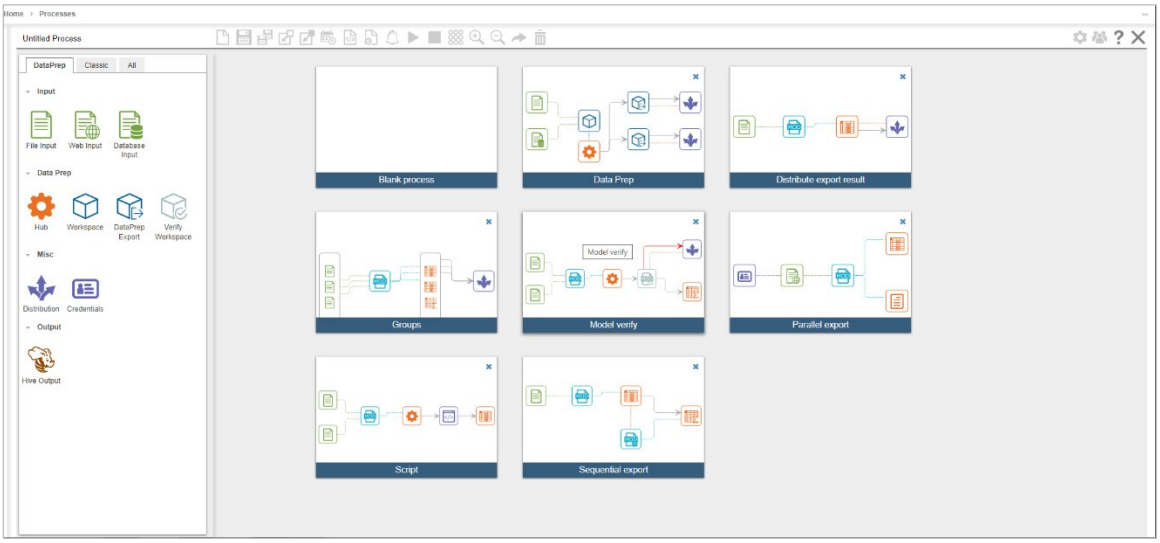
This lesson describes how to create and configure processes using Visual Process Designer, how to run processes.

## CREATING A PROCESS WITH VPD

Let's create a process with Visual Process Designer.

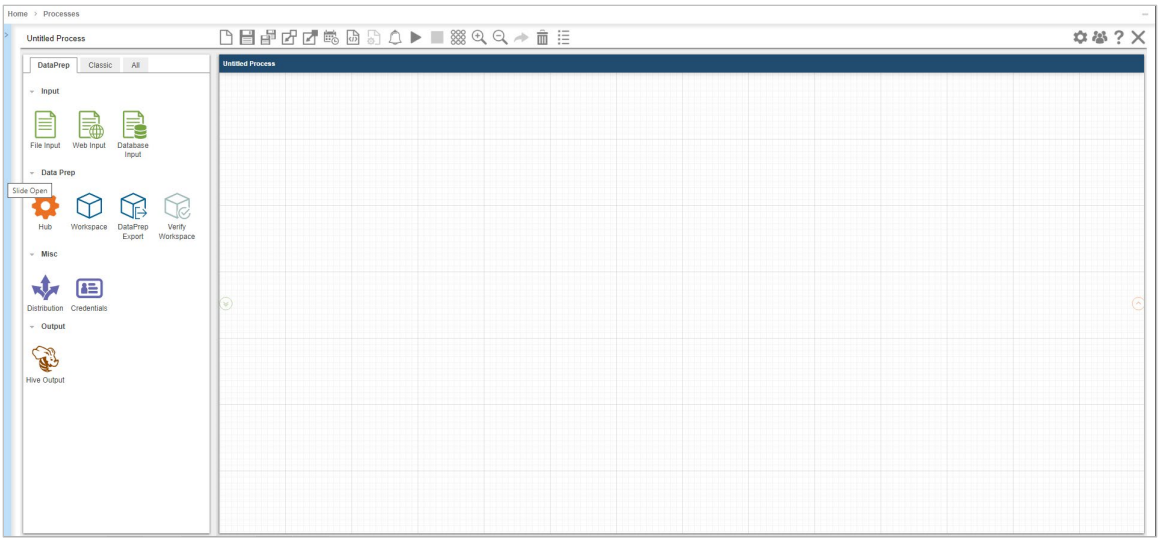
### Steps

1. To open the Visual Process Designer, on the **Navigation** Tree, click **Automator**, click **Visual Processes**, and select **Processes**. Click the **Add** icon .




> **NOTE** You can either create a new process or choose among the existing templates.
>
> If you exceed the number of processes allowed by your license, the following message appears:
>
> "The license for Monarch Server is used improperly. The maximum number of processes per your license (XX) is exceeded as you have NN processes currently defined and no more processes can be defined. You may delete unused processes or contact your Altair sales representative for extended options."

2. Click the Blank Process.




3. The Visual Process Designer interface consists of ribbon buttons, **Process Designer Library**, **Workspace** panel, and **Properties** panel.

   The Process Designer ribbon contains the name of the process (on the left) and the Designer Toolbar. If needed, you can change the process name.

   Type the process name, e.g., *Visual Process*.

# INSTRUCTIONS FOR PROCESS WORK FLOW BUILDING

- There are no mandatory graphical elements that must be used for wore flow building.
- Starting work flow item is defined by absence of input connector. Process execution starts with the starting work flow item.
- Final work flow item is defined by absence of output connector. Process execution finishes on processing final work flow item.
- If there two or more starting items, execution starts synchronously and is divided into two parallel threads.
- You can fork work flow using connectors if you need to branch one thread of process execution into two or more parallel threads.
- Successful execution of work flow stage is defined by gray connector, unsuccessful execution of work flow stage is defined by red connector.
- For export operations, output work flow item can be an input item for subsequent work flow item.

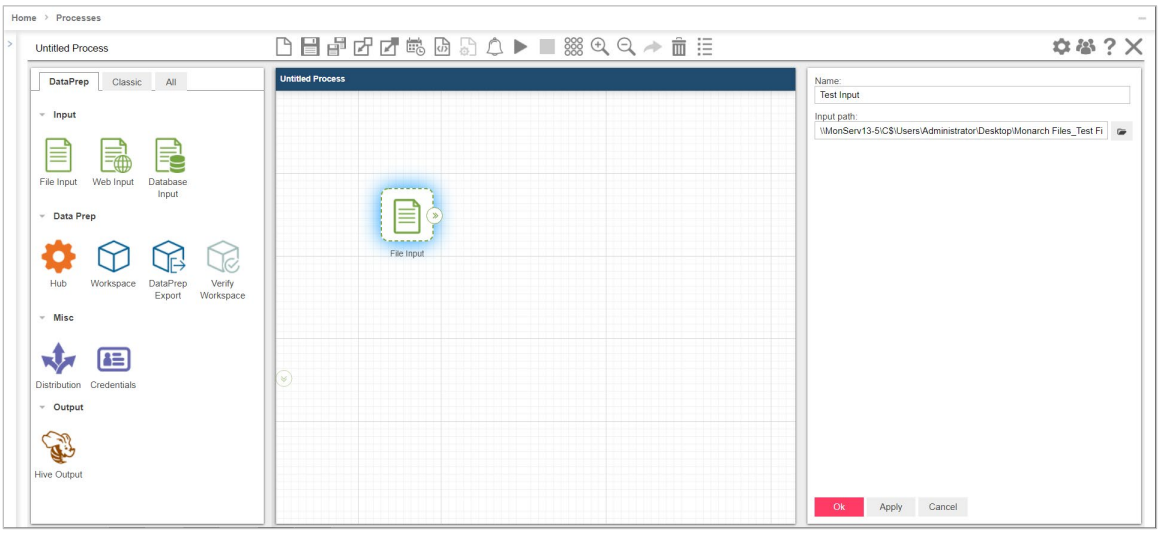
# ADDING AND ASSIGNING ITEMS TO THE PROCESS

1. To add items to the process, drag the needed item from the **Process Designer Library,** and drop it on the **Workspace** panel. Alternatively, you can double click the selected item in the **Library**.

2. Add an input item (e.g., File Input) to your process. Input elements are data sources used for exporting. For this tutorial, we will use the same files used in Standard Processes above.

   Enter the element's name (e.g., *Test Input*), and indicate the file path (e.g., C:\Users\Public\Documents\Altair Monarch\Report\ClassJan.prn).

   Click **Apply**.

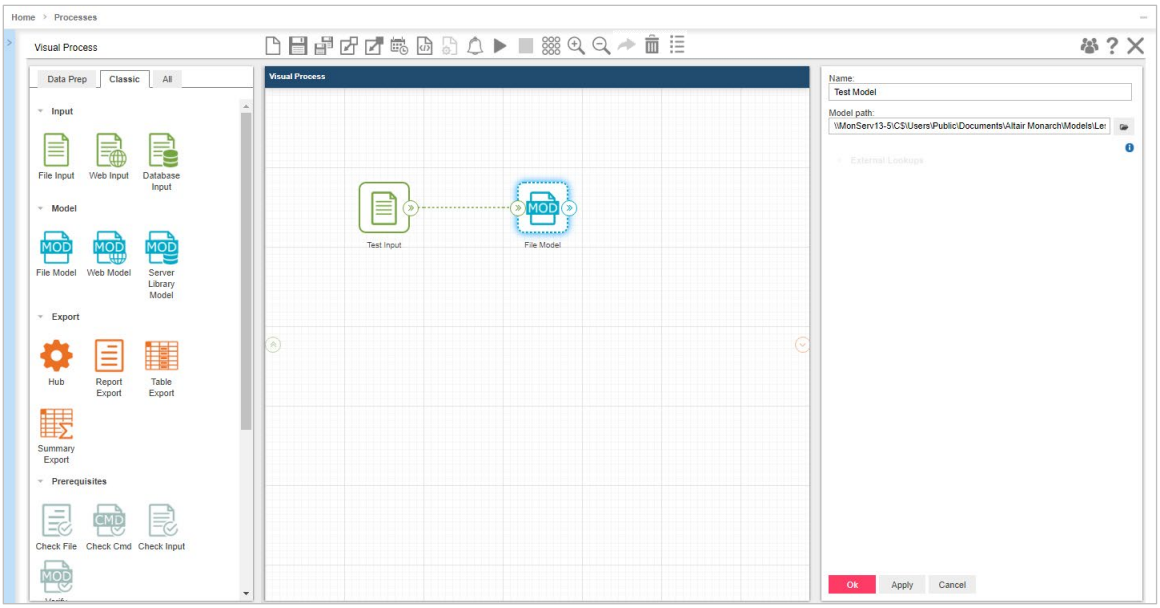



> **NOTE** When using a Database as an input item, you need to indicate the connection string and table name.
>
> When using a Web input, you need to add valid item credentials.

3. Add a model item (e.g., File Model), and connect your input item to it. Model elements, found under the **Classic** tab in the **Library**, are sets of instructions how input files are parsed by Automator.

   Enter the model name (e.g., *Test Model*), and indicate the file path of the model (e.g., C:\Users\Public\Documents\Altair Monarch\Models\Lesson8.dmod). You may also select the required model from the available models in the **Server Library**.

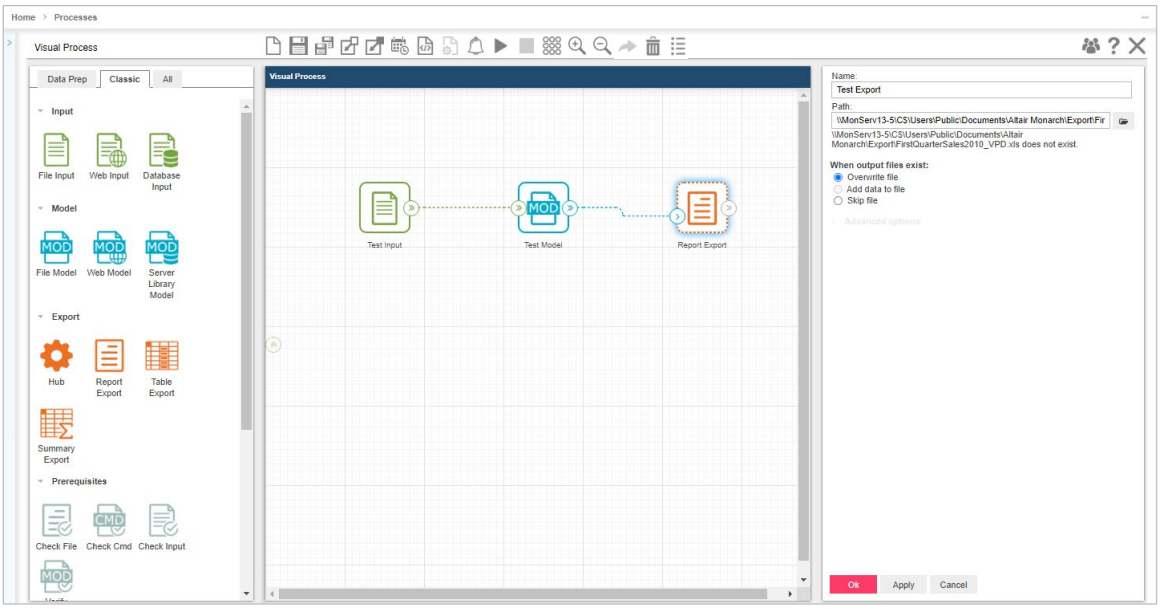
   Click **Apply.**




4. Add an export item (e.g., Report Export), and connect your model to it. Export elements, found under the **Classic** tab in the **Library**, are sets of instructions about how the parsed data are exported by **Automator**.

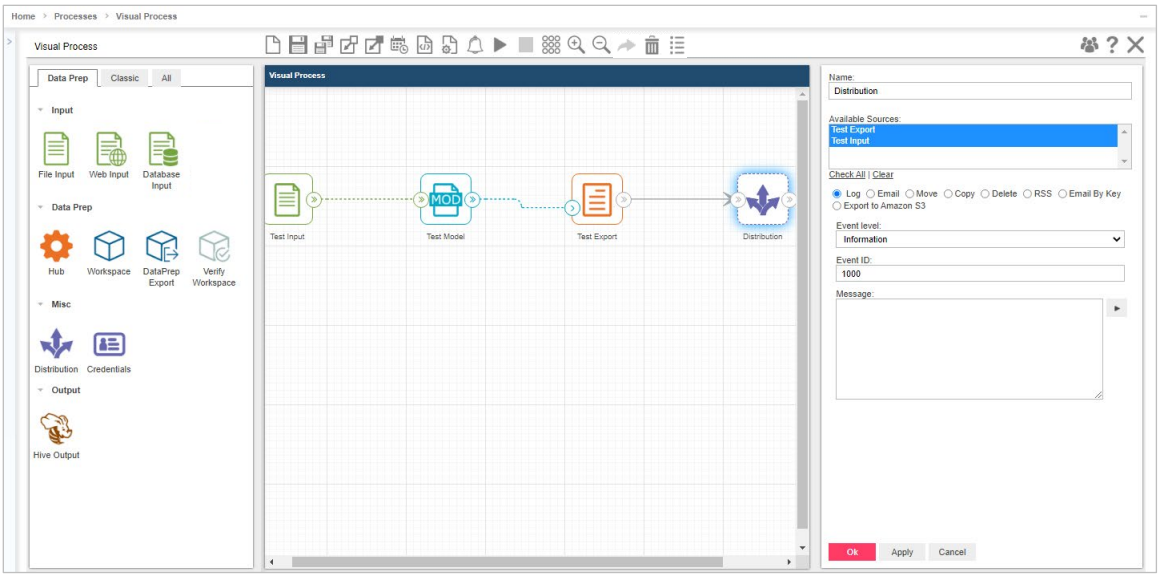
   Enter the element's name (e.g., *Test Export*), and indicate the file path (e.g., C:\Users\Public\Documents\Altair Monarch\Export\FirstQuarterSales2010_VPD.xls).

   Click **Apply**.




5. Add a distribution item (e.g., Distribution), and connect your export item to it. Distributions, found under **Misc** in the **Library**, specify actions that distribute or handle files and information outside of **Automator**.

   Enter the element's name (e.g., *Test Distribution*), select the required action, and indicate the necessary settings.




The Available Sources panel shows the sources that are available for distribution. In this case, the “Test Input” and “Test Export” sources are shown.

For the other configurations, you can refer to Lesson 4. For this part of the tutorial, let us choose the Log distribution.

**NOTE** To get an email with specific macros (for example, Item File Path, Directory, Name, Extension, or URL) and/or attached files, add a dashed connector (dashed orange line) between the export item and distribution item for sending data.

6. Click the **Save** icon to save your process.

# CONFIGURING A PROCESS RUN

Processes can be run in different ways:

- Manually
- Via a process schedule trigger
- Via inputs monitoring trigger
- Via manifests monitoring trigger
- Via Report Warehouse file receiving

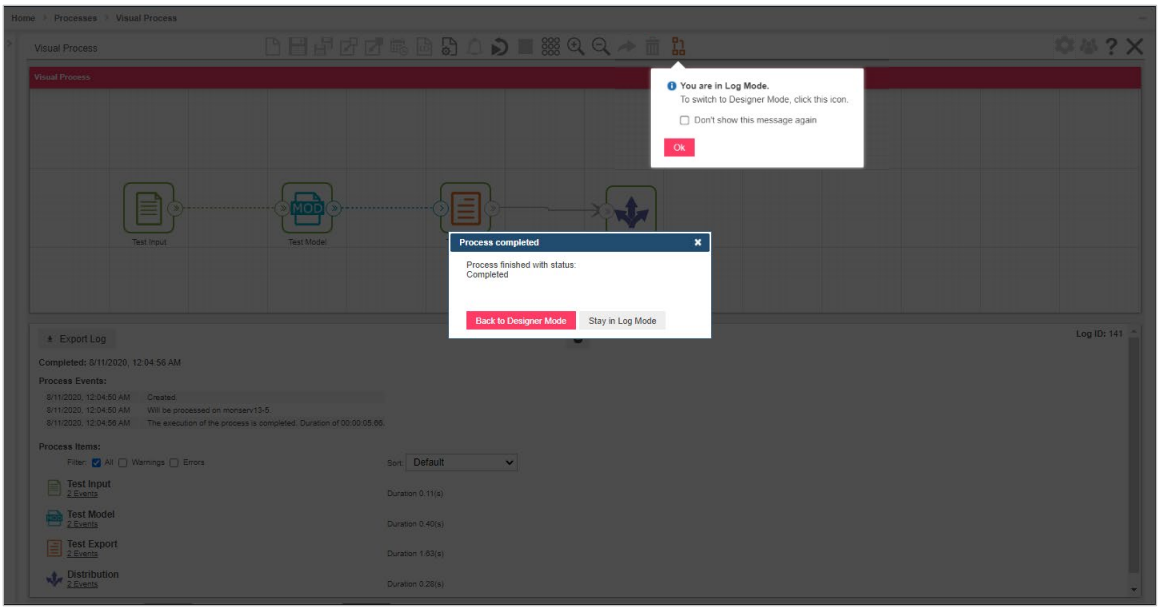
### To configure a manual process run

1. On the **Processes** page, select the **Visual Process** process by clicking anywhere on the row, and then click the **Edit** icon .

2. On the Visual Process Designer toolbar, click the **Save and Run** icon to save the process and launch its execution.

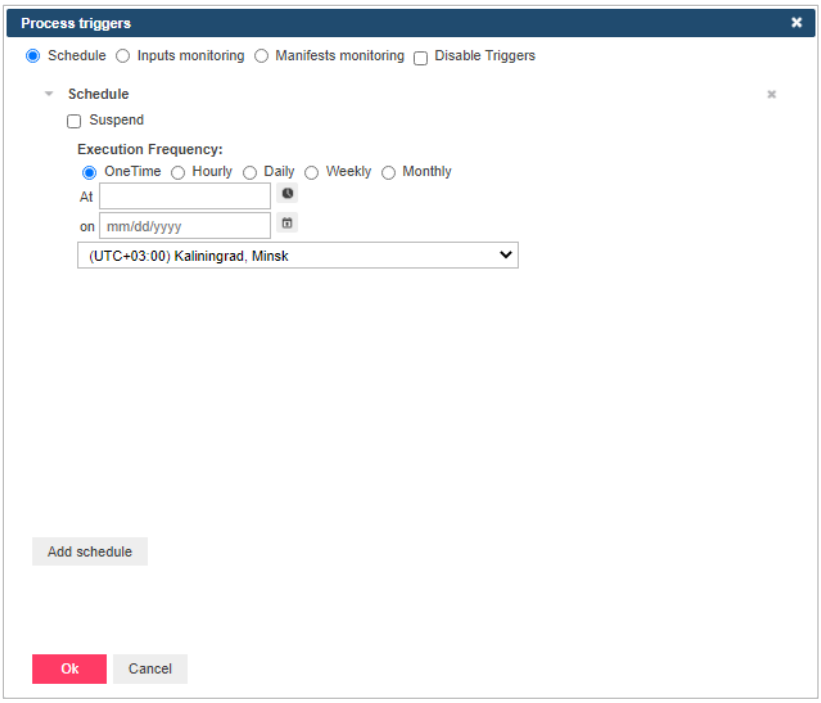
3. The notification of the status of the process run is displayed. In our example, the **Process Completed** dialog box is shown.

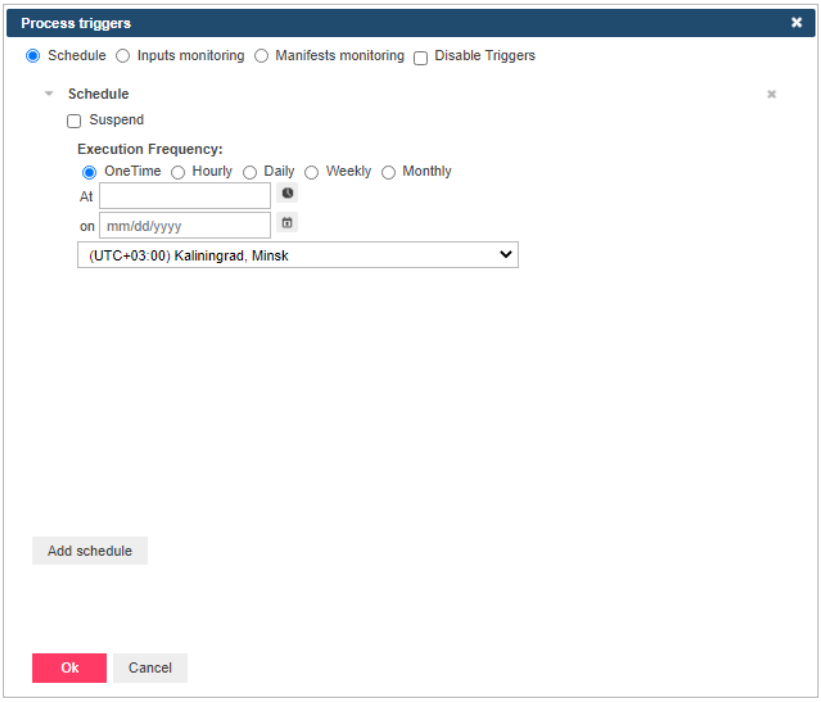
   You can choose to go **Back to Designer Mode** or **Stay in Log Mode**. For details about Log Mode, refer to the About Process Logs topic below.




### To configure a process run via triggers

1. Click the **Triggers** icon to modify the process triggers settings.

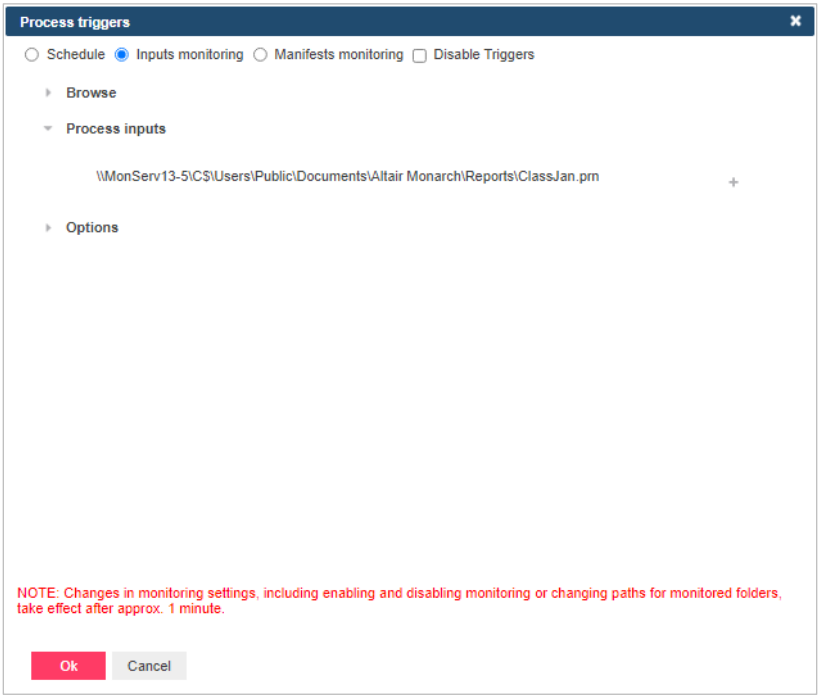


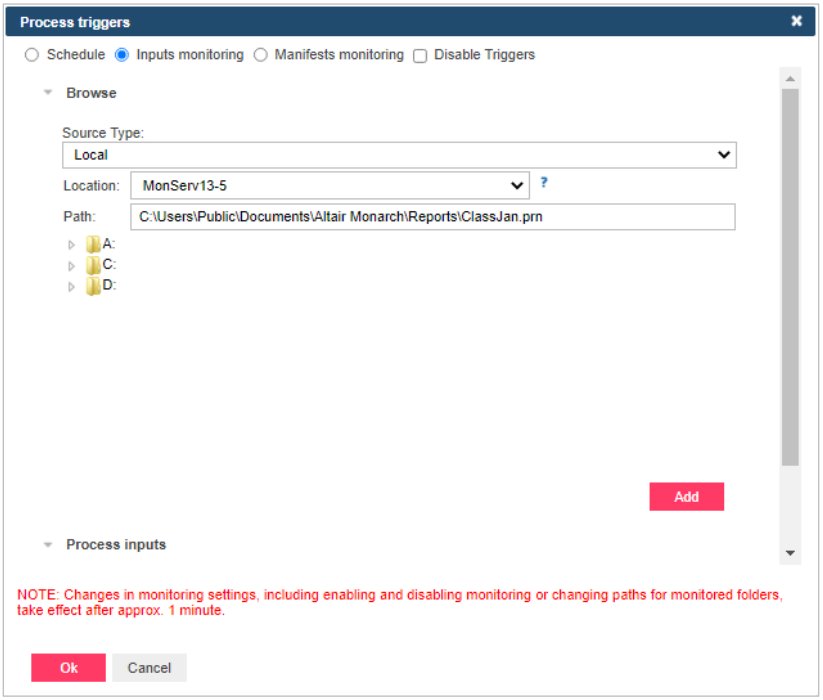

2. Select the required trigger (e.g., inputs monitoring trigger).

**NOTE** For details about Visual Processes with manifest monitoring, refer to the Automator Administrator's Guide.

3. To add a file from the process, expand the **Process inputs** section, and press the **Add** icon near the input path. In our example, the file path for *ClassicJan.prn* appears. This is added to the **Monitoring Paths** list.




To add a file not from the process, expand the **Browse** section, set the file path, and click the **Add** button.




When the file arrives in the specified location, **Automator** will trigger the running of the associated process.

4. Click **Ok.**

> **NOTE** Select the **Disable Triggers** button to disable triggers.

# ABOUT PROCESS LOGS

Process logs are used to view information about process execution, including the start and end dates and times, duration, and results of execution.

User can view the Logs on the **Logs** list. To open it, on the Navigation Tree, click **Automator**, click **Visual Processes**, and select **Logs**. Open a log item to view it.

Click the **Text** tab to display a summary of the process execution. The user may choose the **Visual** tab, which displays the visual process of the selected process as well as the summary of the process execution.

**Process Events** displays general information about the recently run process, including the start and end times.

**Process Items** allows you to filter the process information by selecting and clearing the following check boxes:

- All — displays all file sources.
- Warning — displays the file sources with warnings.
- Error — displays the files sources with errors.

Click the *n Events* link below each file source to expand the relevant information, including the start and end times.

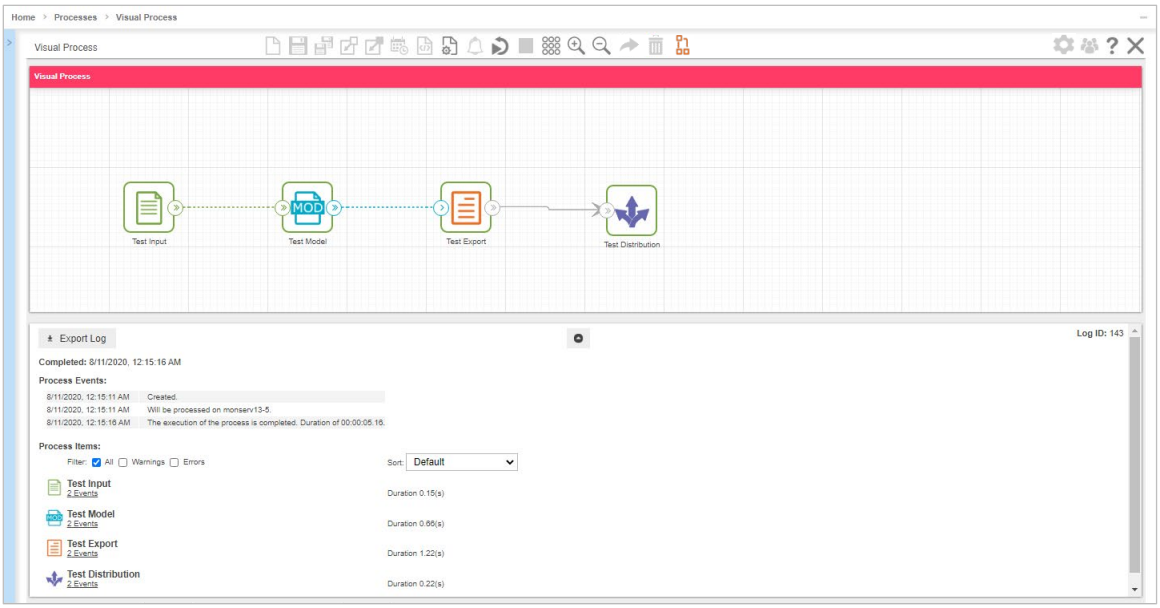
The items may be sorted according to duration (ascending and descending), start time (ascending and descending), and type.

Click the **Export Log** button

to export the log to an XML file.

The user can also view the log under manual run mode. To open it, on the Navigation Tree, click **Automator**, click **Visual Processes**, and select **Processes**.

1. On the **Processes** page, select the **Visual Process** process by clicking anywhere on the row, and then click the **Edit** icon .
2. On the Visual Process Designer toolbar click the **Save and Run** icon to save the process and launch its execution.
3. The Log Mode view shows the results of the last run of current processes.




4. The Log Mode allows the user to edit visual elements and change their respective settings. After changes are made, click the **Save and Run** icon to actualize your changes by running a process with updated settings.

5. Click the **Designer Mode** icon to leave the Log Mode view.

# VERIFYING WORKSPACES

In Automator, results in a workspace can be verified so that users can stop, stop without error notifications, proceed, or proceed with notification. In the previous lessons, you learned about the basics of building a Visual Process. Here, let's try creating a new Visual Process that involves a workspace and the new element "Verify Workspace."

To try out the steps below, prepare a couple of input files and a workspace that we could work on. The goal is to understand the new Verify Workspace element and how it can help you generate high-quality results.

For this lesson, we will demonstrate how the Verify Workspace element works and what the results will be if set conditions are satisfied or not satisfied.

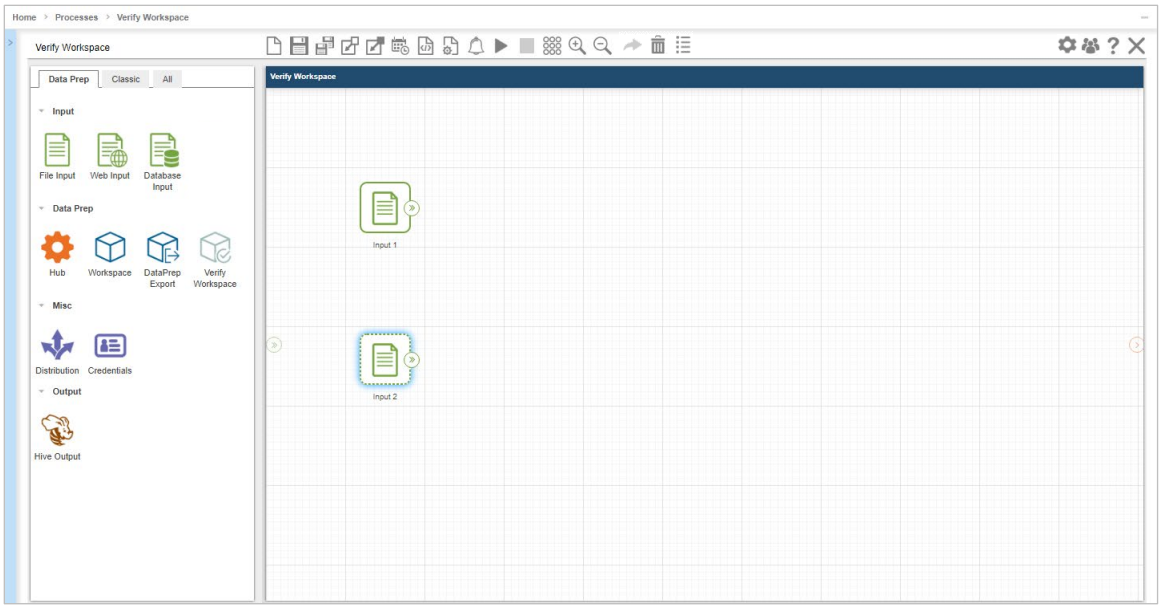
### Steps

1. On the **Navigation** Tree, click **Automator**, click **Visual Processes**, and select **Processes**. Click the **Add** icon .

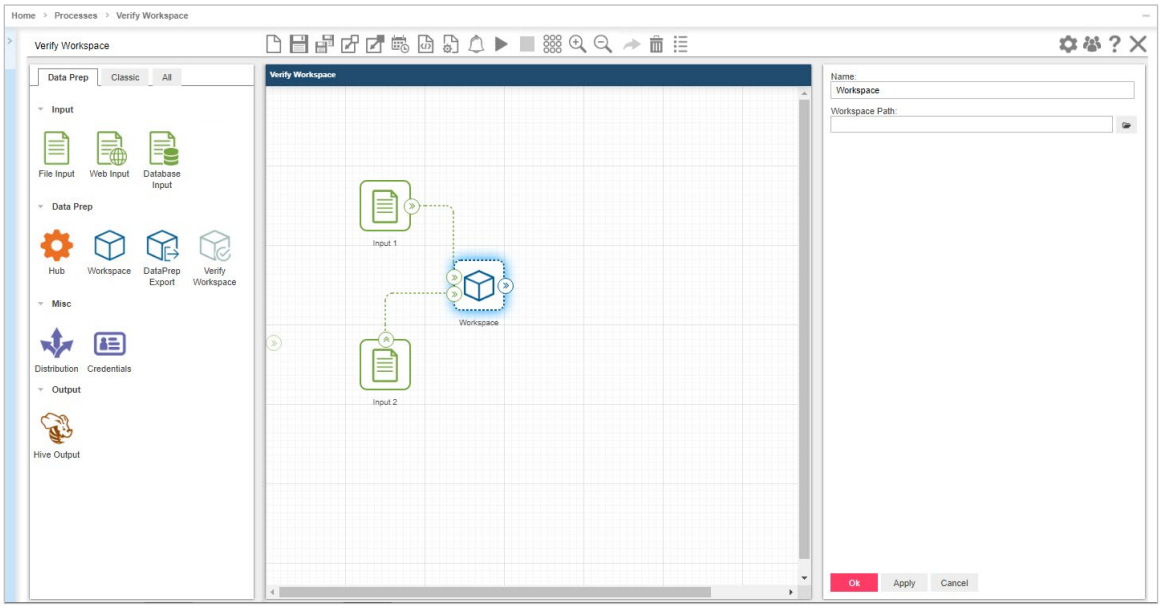
2. Name the Visual Process (e.g., *Verify Workspace*).

3. Add one or more input elements to the workspace. Similarly, provide a name for the elements, and supply the correct input path.

   Click **Apply** and then click **Ok.**



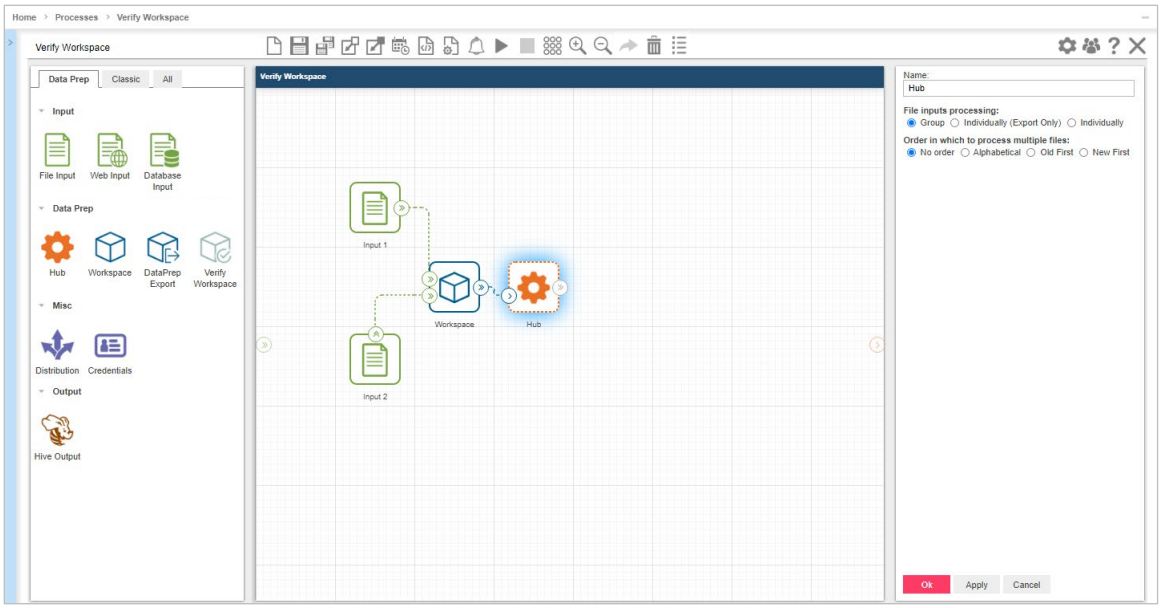

4. Add a workspace element and connect to it the input elements you previously added. As usual, provide a workspace name, and supply the correct workspace path.




Click **Apply** and then click **Ok.**

5. As we are dealing with more than one input file, it is ideal to add a Hub element that could process the multiple input files individually or by group.

   Add a Hub to the Visual Process and connect the workspace to it.



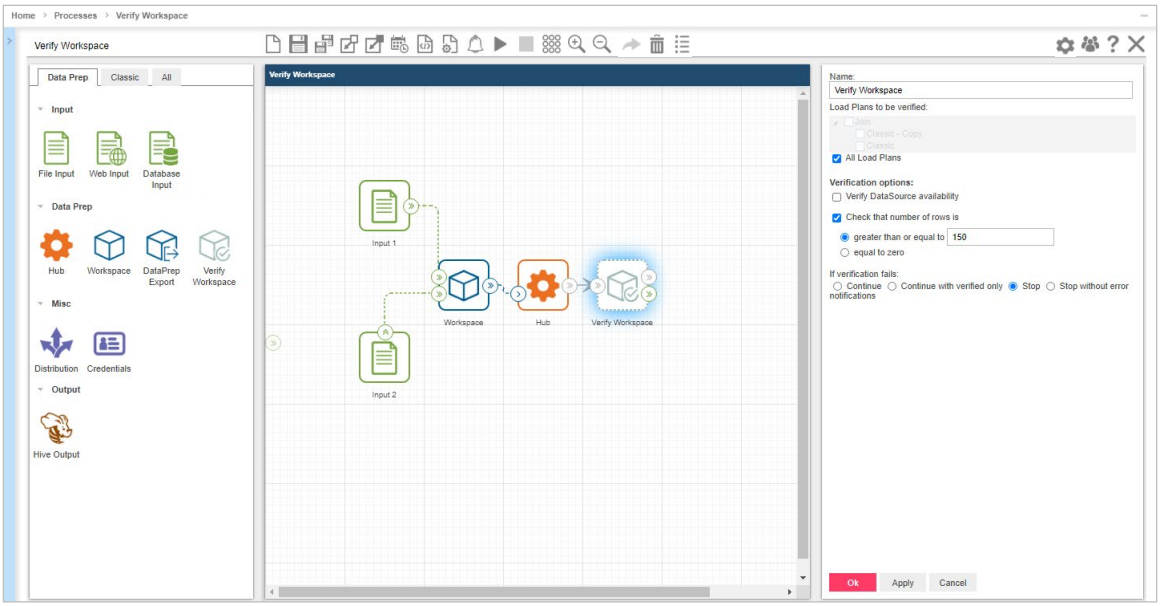

Rename the Hub element accordingly, and apply the settings appropriate for your Visual Process.

**NOTE** The following settings are available for the Hub element:

- File inputs processing:
  - **Group** — all input files are processed in a group
  - **Individually (Export Only)** — each input file is processed separately by the following element types: Report Export, Table Export, Summary Export, DataPrep Export, Verify Model, and Verify Workspace.
  - **Individually** — each input file is processed by all elements after the Hub separately
- Order in which to process multiple files:
  - **No order** — input files are processed in random order
  - **Alphabetical** — input files are processed alphabetically according to their names
  - **Old First** — input files are processed in chronological order, i.e., from oldest files to newest ones
  - **New First** — input files are processed in reverse chronological order, i.e., from newest files to oldest ones

Click **Apply** and then click **Ok.**

6. Add the **Verify Workspace** element and connect to it the Hub element.




In the Property panel, apply the necessary settings (i.e., Name, Load Plans to be verified, Verification options, and If verification fails settings).

**NOTE** The following settings are available for the Verify Workspace element:

- **Name** — name of the graphic element.
- **Load Plans to be verified** — select the load plan/s for verification.

You can select several load plans by ticking their corresponding checkboxes. To export all load plans, select the **All Load Plans** checkbox.

**Verification options** — choose the verification to be done:

- **Verify DataSource availability** – select this option to check whether the data source specified is available.
- **Check that number of rows is** – select this option to check if the number of rows meets the specified value:
  - **Greater than**: set the minimum number of rows for verification
  - **Equals to zero**: set the number of rows for verification to zero

**If verification fails**

- **Continue**. Automator will continue processing the associated actions for the workspace even if a verification failure of the specified kind is encountered.
- **Continue with verified only**. Automator will continue processing the associated actions for the workspace only if it was verified successfully.
- **Stop**. The first failed verification causes the Verify Workspace to

immediately fail.

- **Stop without error notifications**. The job is stopped, but no email alert is sent

For this Tutorial, let's try the following settings:

- **Load Plans to be verified**: Set to **All Load Plans.**
- **Verification options:** Set to **Check that number of rows is** and **greater than**, and then input "150."
- **If verification fails:** Set to **Stop**.

Click **Apply** and then click **Ok.**

7. Add a Distribution element to the workspace, and connect the Verify Workspace element to it. As we have two input elements in our example, make two connecting lines. Apply the necessary settings in the Property panel as usual.

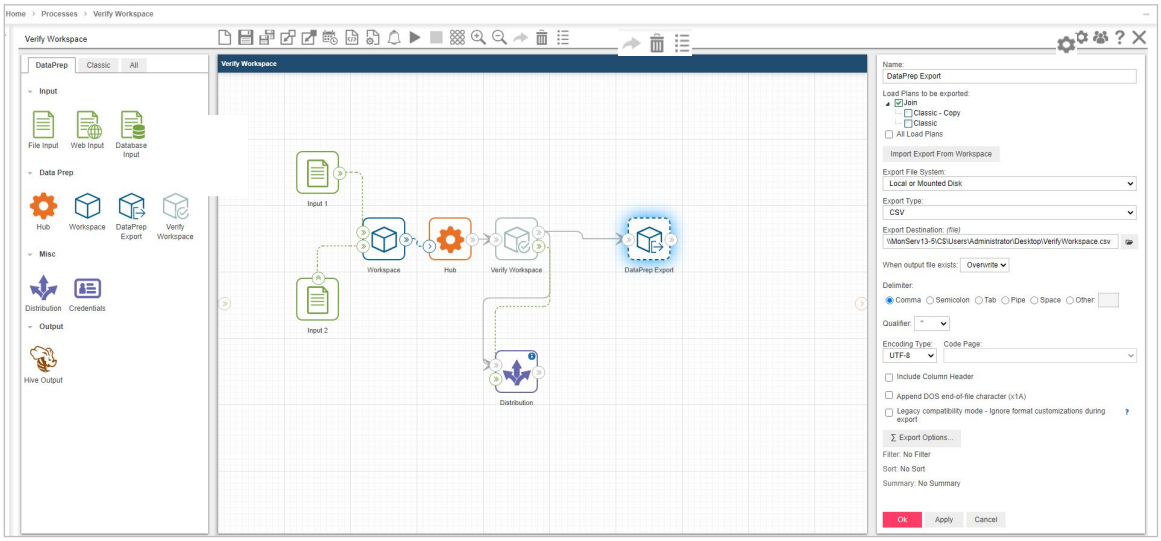
**NOTE** The Connected Sources field should show the Verify Workspace element.

The Available Sources field in the Property panel should show the input files you used.

Let's try selecting the **Log** option to see how our workspace is verified after the process is run.

Click **Apply** and then click **Ok.**

8. To export our workspace should the verification be successful, let's add a DataPrep Export element, and connect to it the Verify Workspace element. Make the necessary changes in the Property panel.

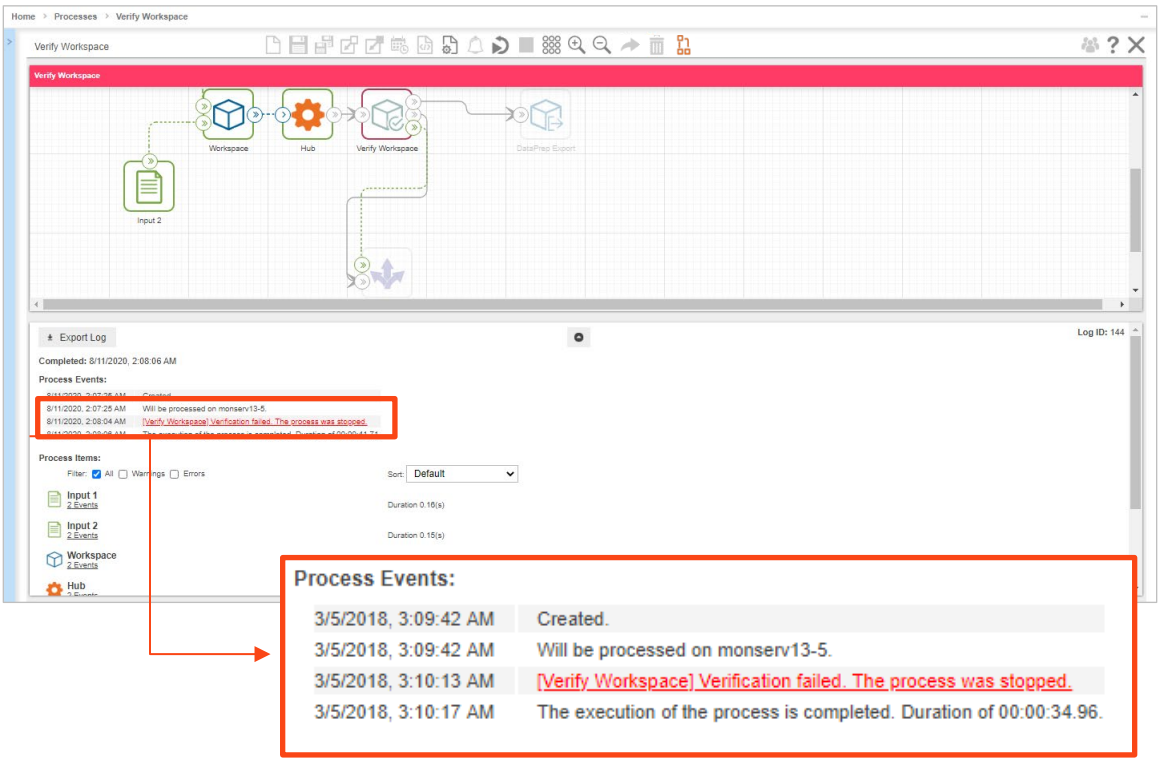



Click **Apply** and then click **Ok.**

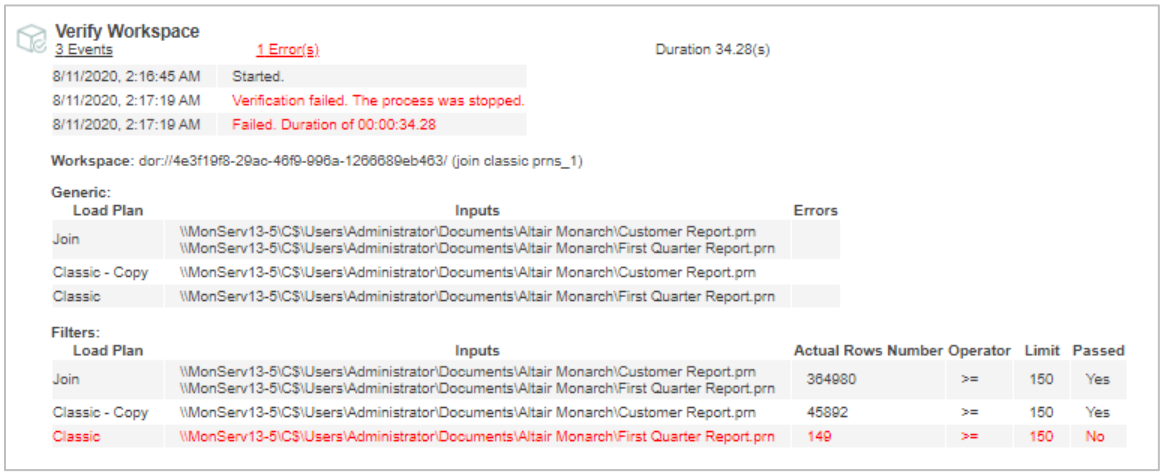
9. At this point, we can save the process and then run it.

In Step 6, we modified the settings of the Verify Workspace element. In this scenario, all load plans, i.e., the workspace and the input files, will be checked as to whether the number of rows exceed 150, which is the number we assigned in the Property panel.

- If the number of rows in any of the load plans is less than 150, the process stops, i.e., the Verify Workspace fails, because we set the **If verification fails** option to **Stop.** See the Log Mode below.




A closer look at the Log Mode shows this:

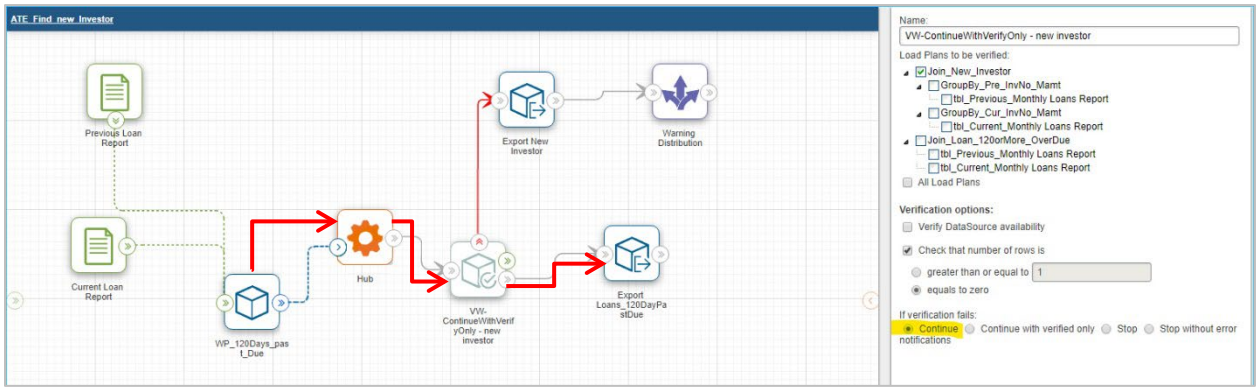


The verification process failed because one of the input files had less than 150 rows, which was the number we set in Step 6.

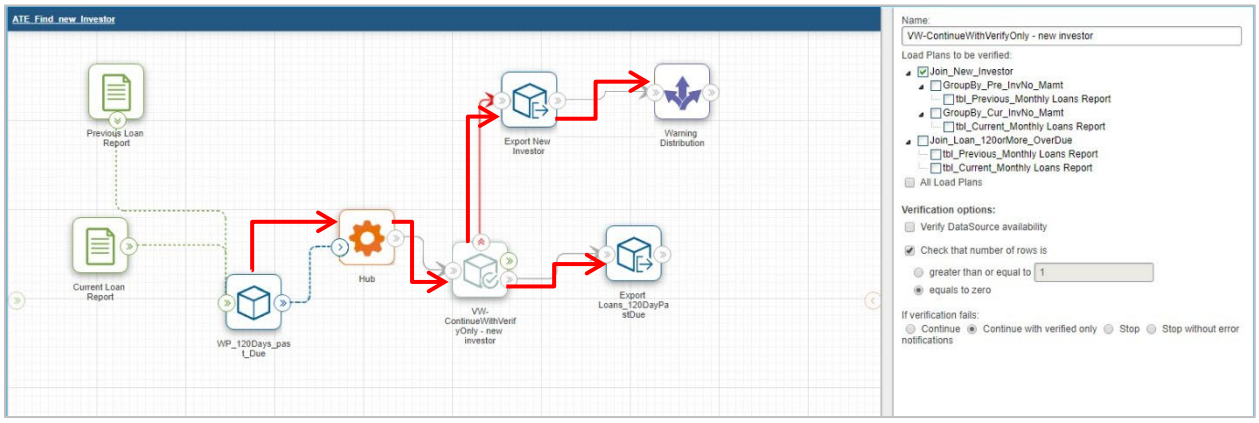
In another case in which the conditions are satisfied, the process should run successfully without errors.

Below is a set of examples showing the flow of a Visual Process in which the selected load plan generates one or more rows and the number of rows for verification is set to 0. Take note of the red arrows in each image.

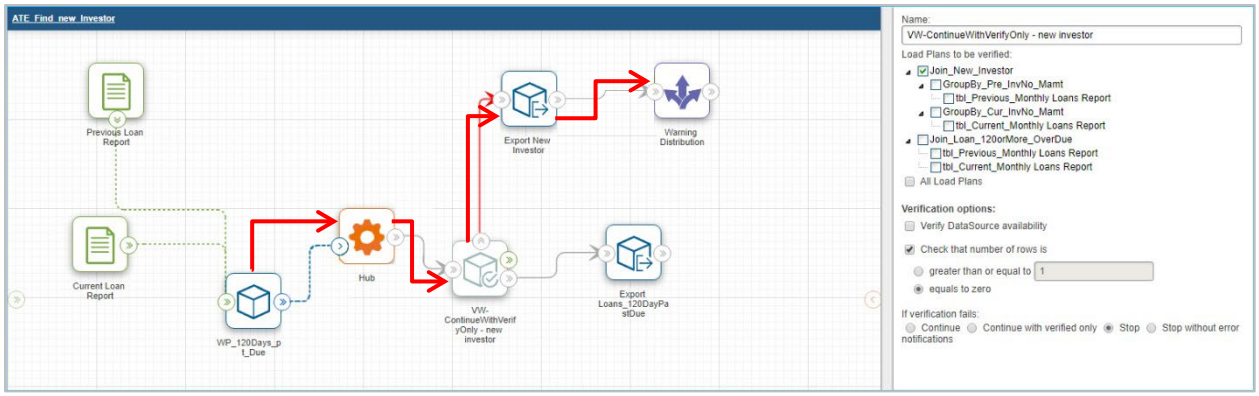
- **If verification fails, continue:** The process continues even if the workspace fails to meet the verification setting.



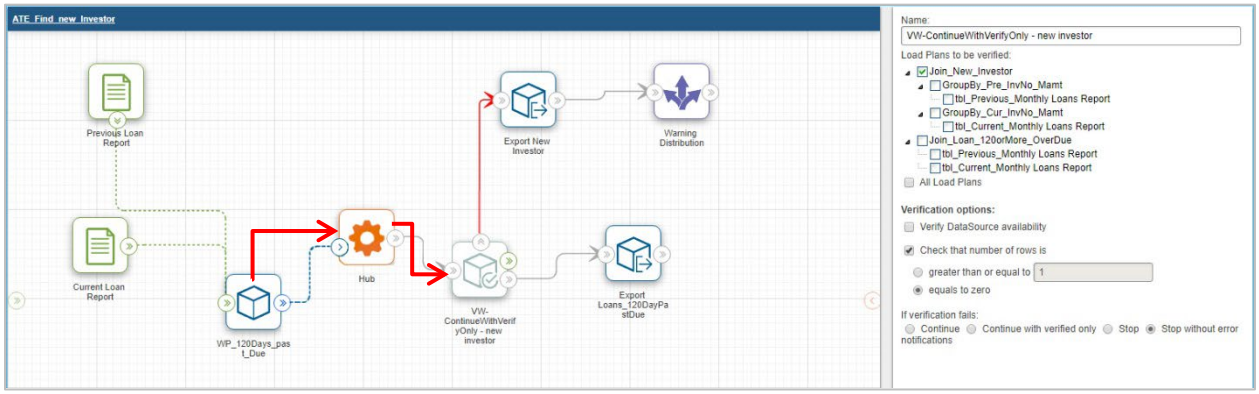

- **If verification fails, continue with verified only:** The process continues with the two export actions, but a warning indicates that some inputs were not verified.




- **If verification fails, stop:** The process continues for one export but stops for the other that involves unverified inputs. For the former, the export proceeds, but a warning indicates that some inputs were not verified.




- **If verification fails, stop without error notifications:** The process stops, and no warning is issued.




# WORKING WITH THE HUB ELEMENT

As we mentioned in the previous lesson, the Hub element is an export element in Visual Process Designer that allows you to define how to group multiple input files and the order in which they are processed. In this lesson, we will look more closely at the Hub element and see how it handles multiple input files.

**NOTE** Before we proceed, make sure that you have a couple of input files to work on, as well as a model. We will use them in the sample Visual Process below.

In the **File inputs processing** setting of the Hub element, you have three options, as we described previously.

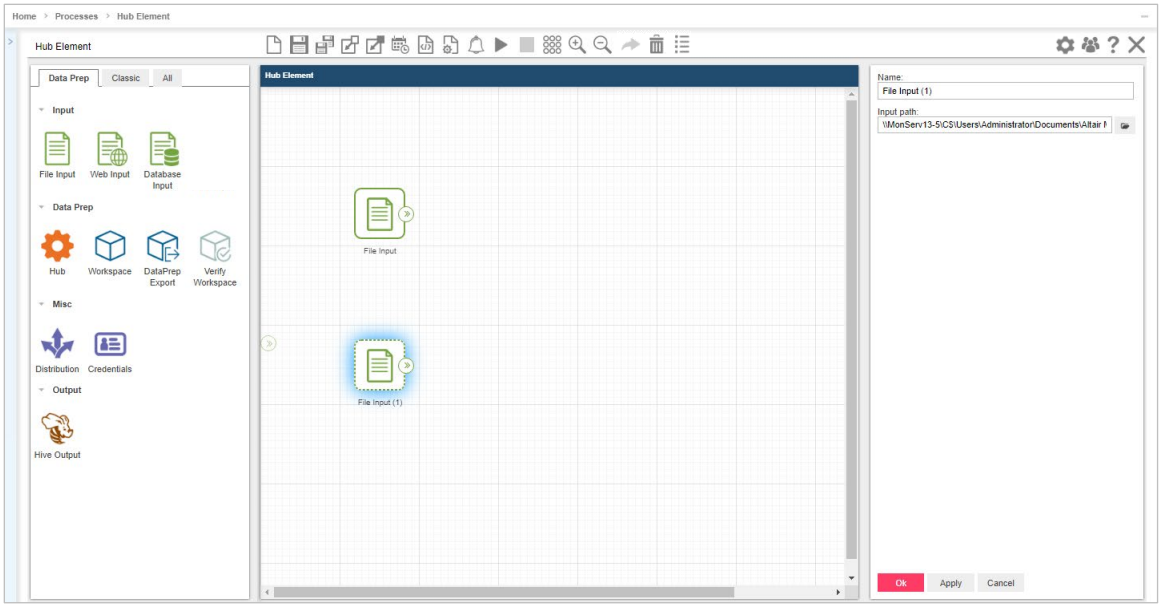
Let's see how these options differ by creating a new Visual Process.

**Steps**

1. Create a new process as you've learned in the beginning of Lesson 6. On the **Navigation** Tree, click **Automator**, click **Visual Processes**, and select **Processes**. Click the **Add** icon , and then choose **Blank Process**.

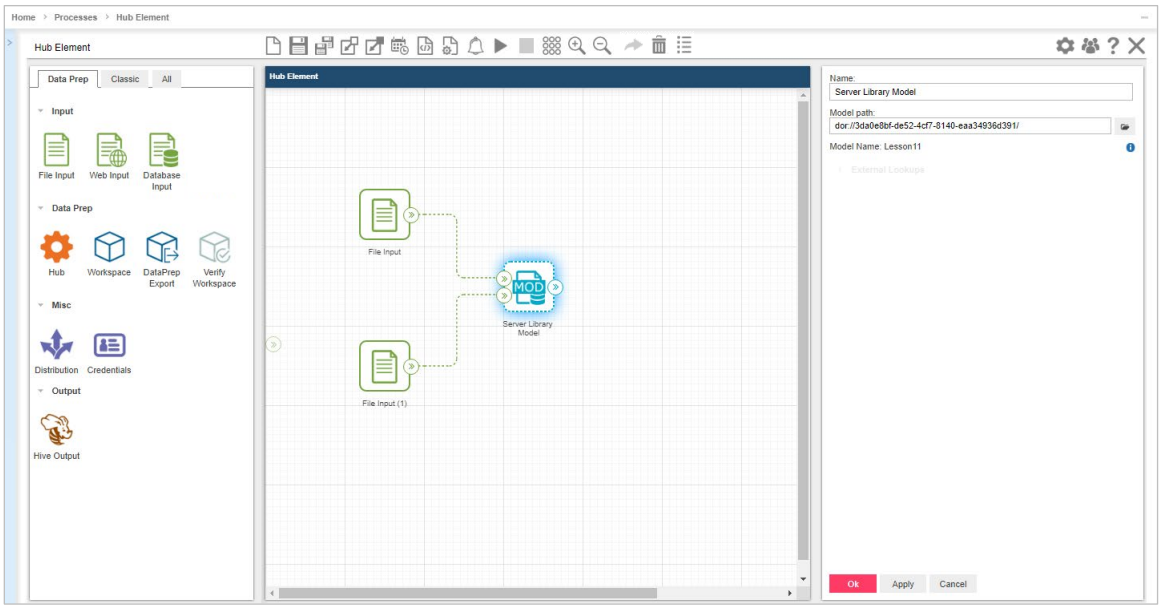
2. Rename the workspace accordingly (e.g., *Hub Element*), and then add two input files (e.g., *File Input* and *File Input (1))*. Apply the necessary changes in the Property panel.

   Click **Apply**.

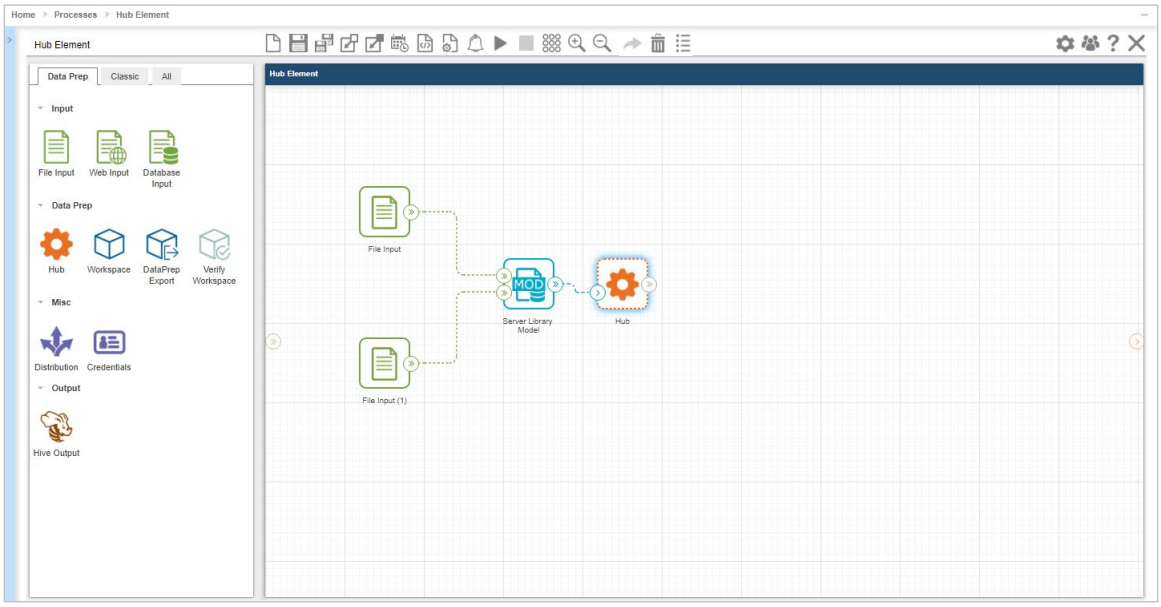



3. Add your desired model to the workspace by dragging the **Server Library Model** element found in the Classic tab of the Process Designer Library. Connect the two **File Input** elements to it, and modify the settings in the Property panel.

   Click **Apply**.



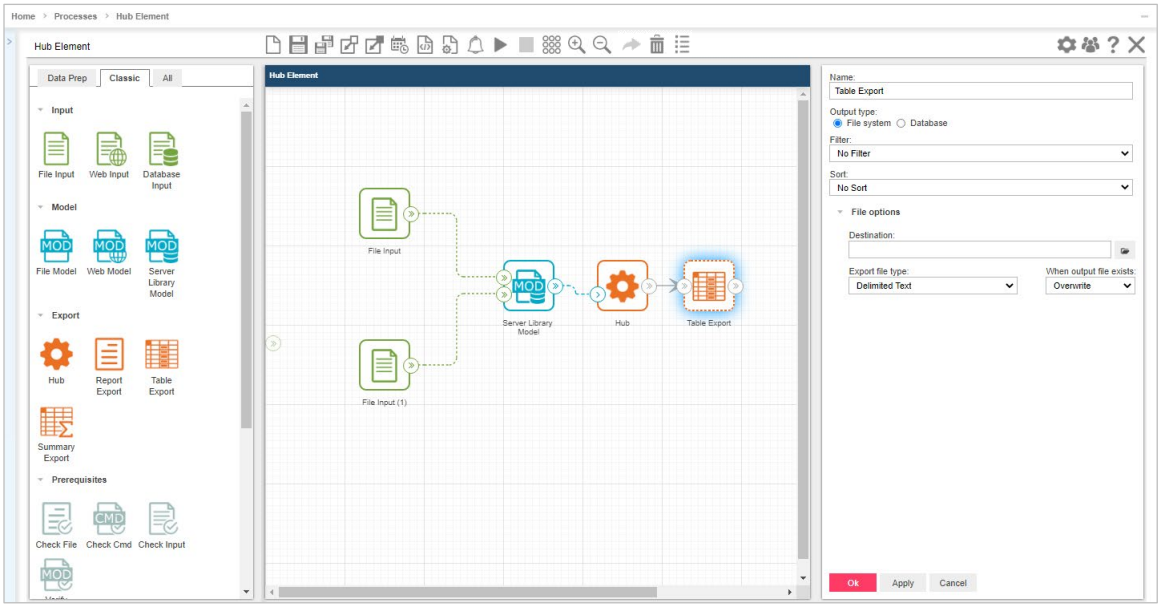
4. Add a Hub element to the workspace, and connect to it the Server Library Model element.




Before we proceed with modifying the properties of the Hub element, let's connect it first to a Table Export element to see how it treats an export.

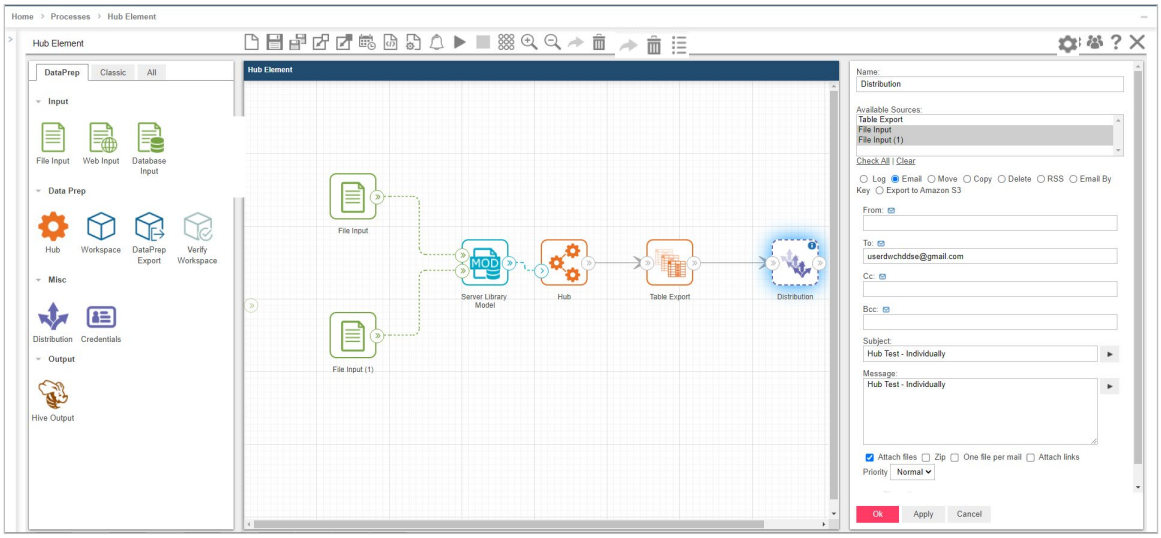
5. Add a Table Export element, and connect to it the Hub element. As usual, modify the settings in the Property panel.

   Click **Ok**.



Let's now add a Distribution element to clearly observe how the Hub element treats distributions in the Visual Process.

6. Add a Distribution element, and connect to it the Table Export element.




In the Property panel, modify the following settings:

- Under **Available Sources**, choose the two file inputs we used (i.e., *File Input* and *File Input (1)* in the screenshot).
- Choose **Email** distribution, and fill in the details.
- Choose **Attach files** so we can later observe how the Hub element treats the input files we used.

Click **Ok.**

Now we are ready to try out the file inputs processing of the Hub element.

### File inputs processing: Group

The Hub element can process file inputs by group. Let's go back to the Property panel of the Hub element and modify the settings.

a. In the Property panel, give the Hub element a name.

b. Under File inputs processing, select Group.

c. In the Order in which to process multiple files, retain the default option (i.e., No order).

Click **Ok**.

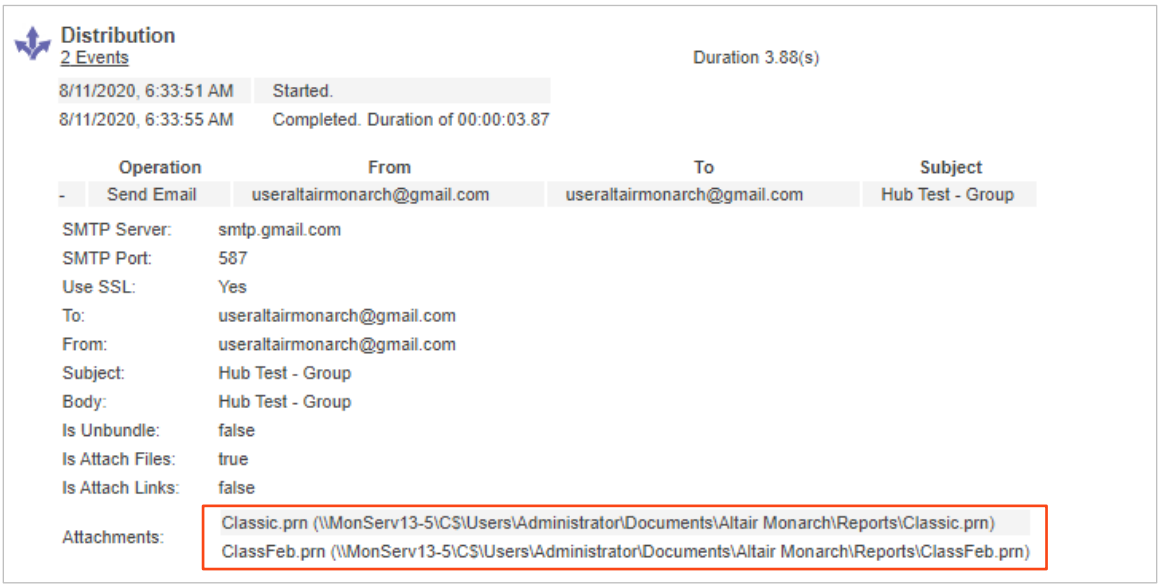
d. With a complete Visual Process, let's try running it by clicking the **Save and Run** button .

**Result – File inputs processing: Group**

As we mentioned previously, the **Group** setting allows the Hub element to process the file inputs by group. Let's look at how the Hub element treated the Table Export. In the Log Mode, the result should be something like this.

Here, the Hub element exported the file inputs as **one file** (i.e., *Classic.pdf*), as shown in the **Export Location**.

As for the email distribution, the result is as follows.




As we selected the two input files in the Distribution Property panel, **both input files were sent as attachments in one email**.

**File inputs processing: Individually (Export Only)**

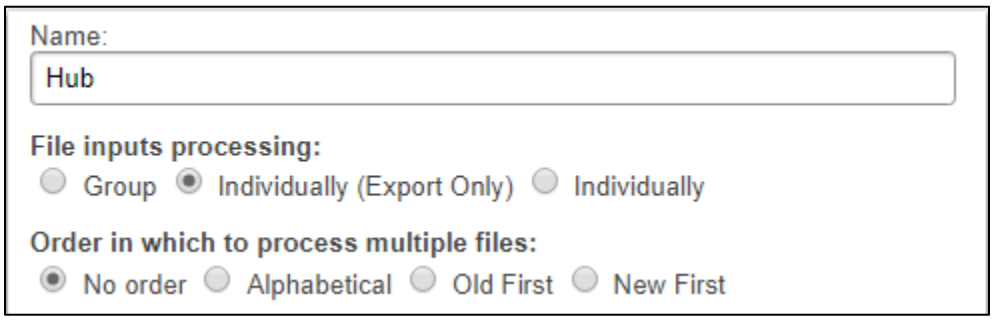
The Hub element can process file inputs separately for export actions. Let's go back to the Property panel of the Hub element and modify the settings.

a. In the Property panel, give the Hub element a name.

b. Under **File inputs processing**, select **Individual (Export Only)**.

c. In the **Order in which to process multiple files**, retain the default option (i.e., **No order**).

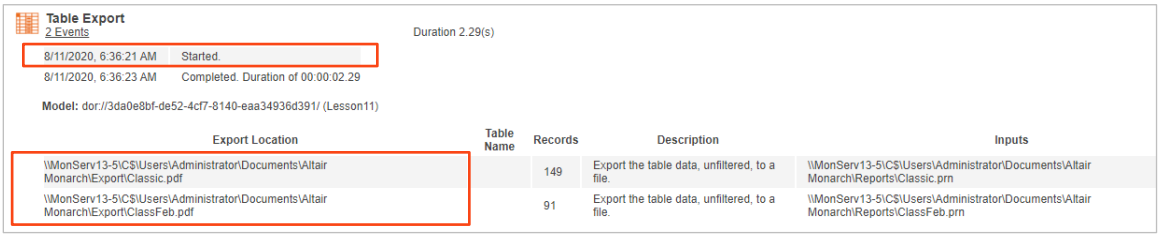
Click **Ok**.




d. With a complete Visual Process, let's try running it by clicking the **Save and Run** button .

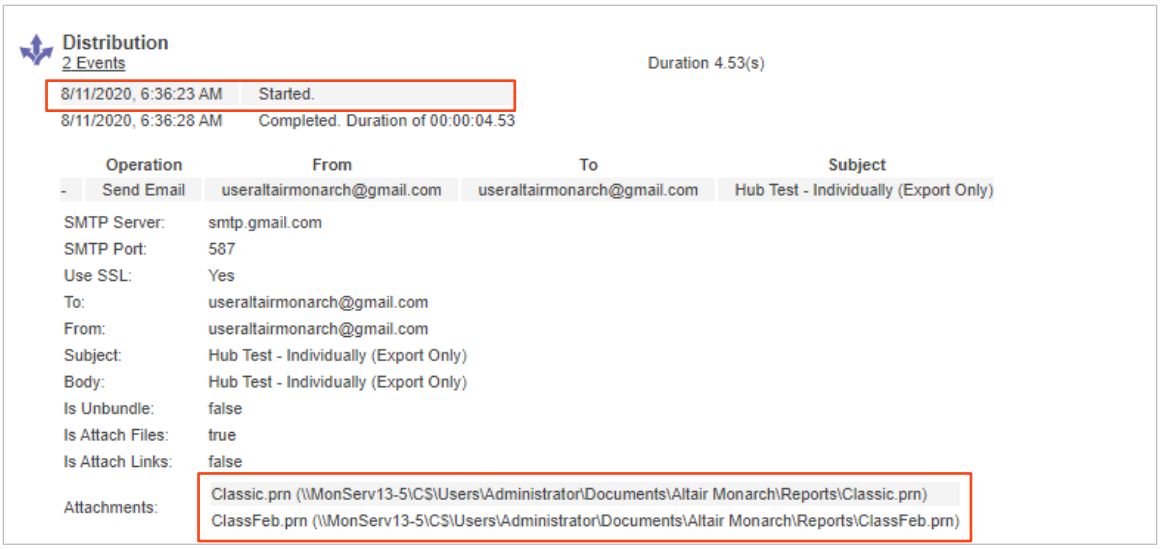
**Result – File inputs processing: Individually (Export Only)**

As we mentioned previously, the **Individually (Export Only)** setting allows the Hub element to process the file inputs separately during export. Let's look at how the Hub element treated the Table Export. In the Log Mode, the result should be something like this.




Here, the Hub element exported the file inputs into **two separate files simultaneously** (i.e., *Classic.pdf*), as shown in the **Export Location**.

As for the email distribution, the result is as follows.




As we selected the two input files in the Distribution Property panel, **both input files were sent as attachments in one email,** similar to that under the **Group** setting. You will notice that **the distributions of the two files were performed simultaneously.**

**File inputs processing: Individually**

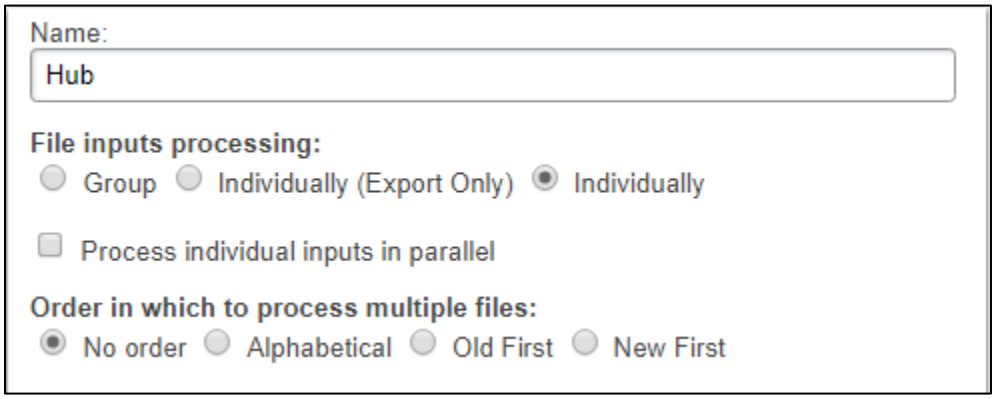
The Hub element can process file inputs separately for actions apart from export. Let's go back to the Property panel of the Hub element and modify the settings.

a. In the Property panel, give the Hub element a name.

b. Under **File inputs processing**, select **Individually**. Leave the checkbox (**Process individual inputs in parallel**) unmarked.

c. In the **Order in which to process multiple files**, retain the default option (i.e., **No order**).

Click **Ok**.




d. With a complete Visual Process, let's try running it by clicking the **Save and Run** button .

**Result – File inputs processing: Individually**

As we mentioned previously, the **Individually** setting allows the Hub element to process the file inputs separately for all actions following the Hub element. Let's look at how the Hub element treated the Table Export. In the Log Mode, the result should be something like this.

**Table Export**
4 Events — Duration 2.49(s)

| | |
|---|---|
| 8/11/2020, 6:38:43 AM | Started. |
| 8/11/2020, 6:38:45 AM | Completed. Duration of 00:00:01.47 |
| 8/11/2020, 6:38:48 AM | Started. |
| 8/11/2020, 6:38:49 AM | Completed. Duration of 00:00:01.01 |

**Model:** dor://3da0e8bf-de52-4cf7-8140-eaa34936d391/ (Lesson11)

| Export Location | Table Name | Records | Description | Inputs |
|---|---|---|---|---|
| \\MonServ13-5\C$\Users\Administrator\Documents\Altair Monarch\Export\Classic.pdf | | 149 | Export the table data, unfiltered, to a file. | \\MonServ13-5\C$\Users\Administrator\Documents\Altair Monarch\Reports\Classic.prn |
| \\MonServ13-5\C$\Users\Administrator\Documents\Altair Monarch\Export\ClassFeb.pdf | | 91 | Export the table data, unfiltered, to a file. | \\MonServ13-5\C$\Users\Administrator\Documents\Altair Monarch\Reports\ClassFeb.prn |

Here, the Hub element exported the file inputs into **two separate files**, as shown in the **Export Location**. But note that the export was performed **consecutively** and not simultaneously, as indicated by the pink boxes above. In sum, the export action for the first file input is started and completed first before the next export action for the second file input is started and completed.

As for the email distribution, the result is as follows.

**Distribution**
4 Events — Duration 6.83(s)

| | |
|---|---|
| 8/11/2020, 6:38:45 AM | Started. |
| 8/11/2020, 6:38:48 AM | Completed. Duration of 00:00:03.32 |
| 8/11/2020, 6:38:49 AM | Started. |
| 8/11/2020, 6:38:53 AM | Completed. Duration of 00:00:03.50 |

| | Operation | From | To | Subject |
|---|---|---|---|---|
| - | Send Email | useraltairmonarch@gmail.com | useraltairmonarch@gmail.com | Hub Test - Individually |

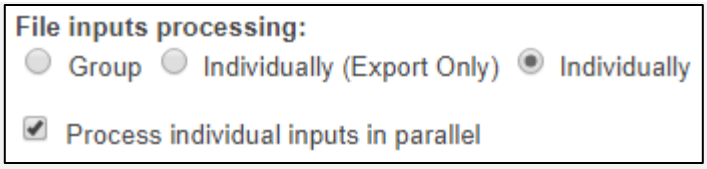
| | |
|---|---|
| SMTP Server: | smtp.gmail.com |
| SMTP Port: | 587 |
| Use SSL: | Yes |
| To: | useraltairmonarch@gmail.com |
| From: | useraltairmonarch@gmail.com |
| Subject: | Hub Test - Individually |
| Body: | Hub Test - Individually |
| Is Unbundle: | false |
| Is Attach Files: | true |
| Is Attach Links: | false |
| Attachments: | Classic.prn (\\MonServ13-5\C$\Users\Administrator\Documents\Altair Monarch\Reports\Classic.prn) |

| | Operation | From | To | Subject |
|---|---|---|---|---|
| - | Send Email | useraltairmonarch@gmail.com | useraltairmonarch@gmail.com | Hub Test - Individually |

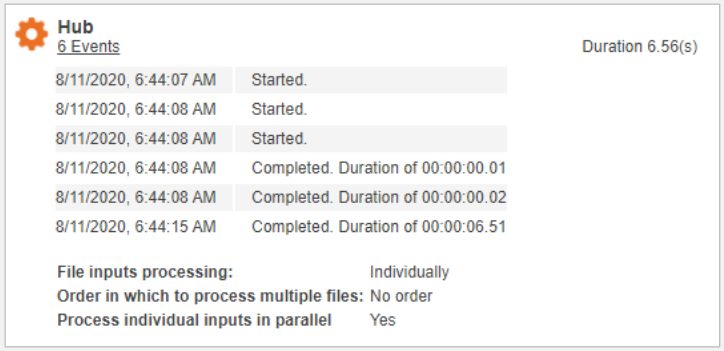
| | |
|---|---|
| SMTP Server: | smtp.gmail.com |
| SMTP Port: | 587 |
| Use SSL: | Yes |
| To: | useraltairmonarch@gmail.com |
| From: | useraltairmonarch@gmail.com |
| Subject: | Hub Test - Individually |
| Body: | Hub Test - Individually |
| Is Unbundle: | false |
| Is Attach Files: | true |
| Is Attach Links: | false |
| Attachments: | ClassFeb.prn (\\MonServ13-5\C$\Users\Administrator\Documents\Altair Monarch\Reports\ClassFeb.prn) |

As we selected the two input files in the Distribution Property panel, **both input files were sent as attachments in two separate emails**. Similar to the Table Export, the email action for the first file input is started and completed first before the email action for the second file input is started and completed. This is because the Hub element processed the input files separately and consecutively.

**NOTE** When we select the **Individually** option in the Hub element, a checkbox is shown, i.e., **Process individual inputs in parallel**.




Once this checkbox is selected, the icon of the Hub element changes from to . This is because all actions after the Hub element are performed in parallel, meaning all actions are started consecutively and completed consecutively, as shown below.




The same change can be seen for the export elements when the **Individually (Export Only)** option is selected. For example, becomes .

As for the **Individually** option, the same change in the appearance of icons is observed not only in the export elements but also in the distribution and script elements.

Note that only 10 inputs can be processed in parallel under the **Individually** setting because the default Max Threads setting in the Automator Dispatcher Service Host is 10. This value can be changed in Configurator. For more details about the Max Threads setting, refer to Monarch Server Configurator Guide (see Configuring Servers).

Go ahead and try out the Hub element using several file inputs.

# WORKING WITH SCRIPTS

Automator supports the use of Visual Basic.NET script in global environment scripts and process scripts.

**Global Environment Scripts**

Global scripts allow users to view functions, classes, and variables that are available or will execute over all processes. It also allows the setting of global assembly imports and references, which are valid over Automator process boundaries.

**Process Scripts**

Process scripts are used for process-specific scripting. They enable you to specify pre-process and post-process operations that execute before or after any operations in the process itself.

> **NOTE** For more details about scripts in Automator, refer to the Automator Administrator's Guide.
>
> Note that scripts from Standard Processes will not work in Visual Processes (e.g., after migrating Standard Processes to Visual Processes).

In this lesson, let's discover how global environment and process scripts work in a Visual Process.

### Example of Scripts in Visual Processes

As we described previously, global environment scripts apply to all processes. In Visual Processes, any script provided under **Global Declarations** in the Script Editor will apply to all scripts used in any Visual Process.

Take for example this script:

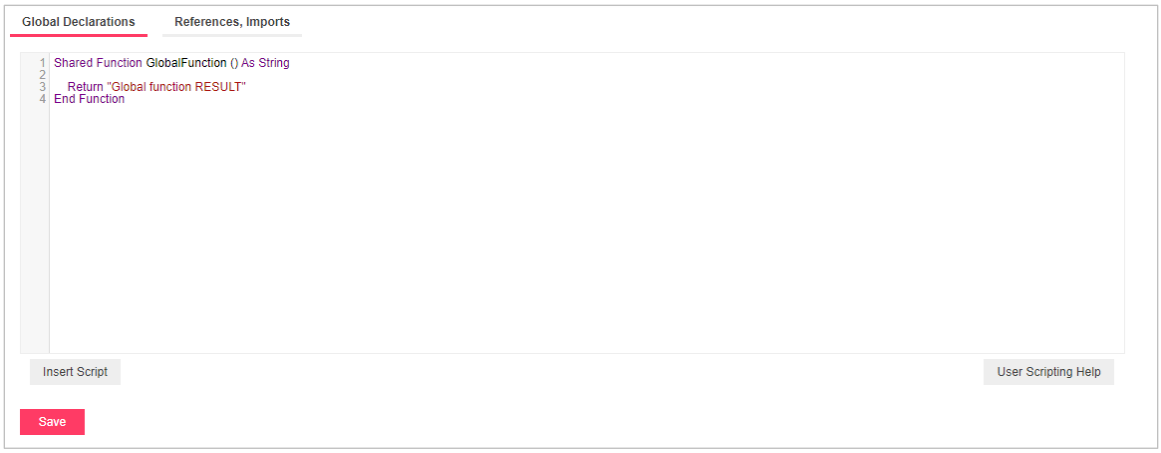
```
Shared Function GlobalFunction () As String


    Return "Global function RESULT"

End Function
```

Let's try it in Automator.

### Steps

1. In the Navigation Tree, click **Automator**. Under **Visual Processes,** click **Scripts.**
2. Under **Global Declarations**, add the sample script above. This script shall apply to all Visual Processes.

> **NOTE** Click the **User Scripting Help** button to access the help file on user scripting.



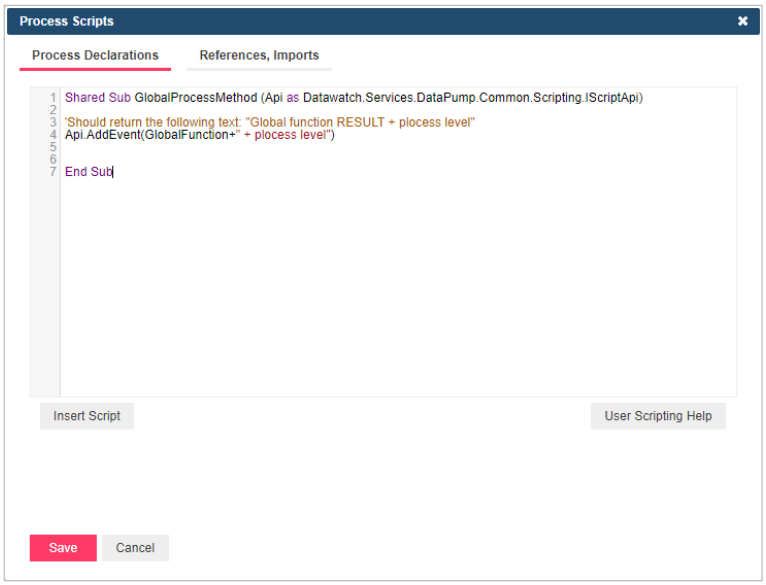

Click **Save**.

3. Create a new Visual Process. In the Navigation Tree, click **Automator**, click **Visual Processes**, and select **Processes**. Click the **Add** icon .
4. Provide a name for the Visual Process (e.g., *Script*).

   In the Visual Process Designer toolbar, click on the **Scripts** icon .

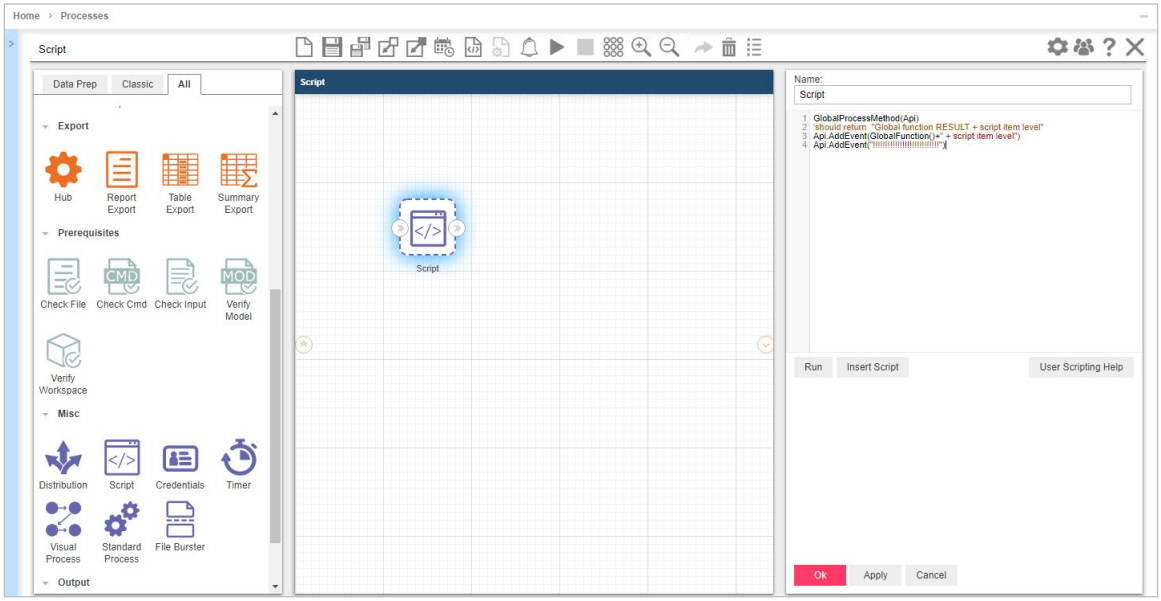
   The Process Scripts window displays.
5. Under Process Declarations, use this sample script:

```
Shared Sub GlobalProcessMethod (Api as Datawatch.Services.DataPump.Common.Scripting.IScriptApi)


'Should return the following text: "Global function RESULT + plocess level"

Api.AddEvent(GlobalFunction+" + plocess level")



End Sub
```




Click **Save**.

6. Now let's try adding a script to the workspace. Add a Script element, which is found under **Misc** in the **All** tab of the Library.



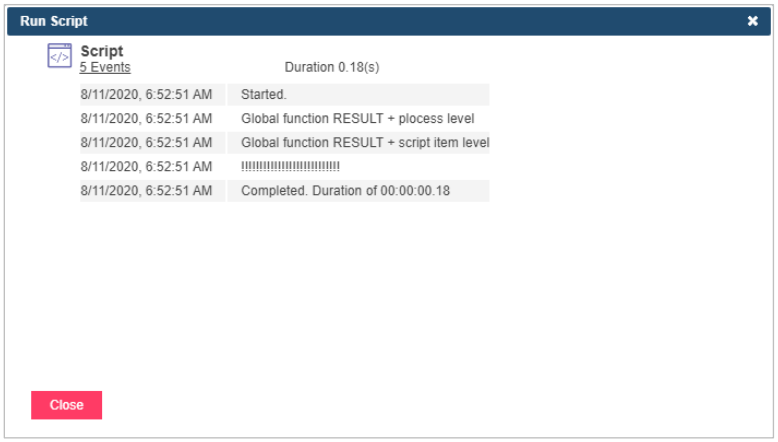

7. In the Property panel, provide a name for the element, and then input the sample process script below. Take note of the AddEvent APIs used.

```
GlobalProcessMethod(Api)
'should return  "Global function RESULT + script item level"
Api.AddEvent(GlobalFunction()+" + script item level")
Api.AddEvent("!!!!!!!!!!!!!!!!!!!!!!!!!!!!")
```

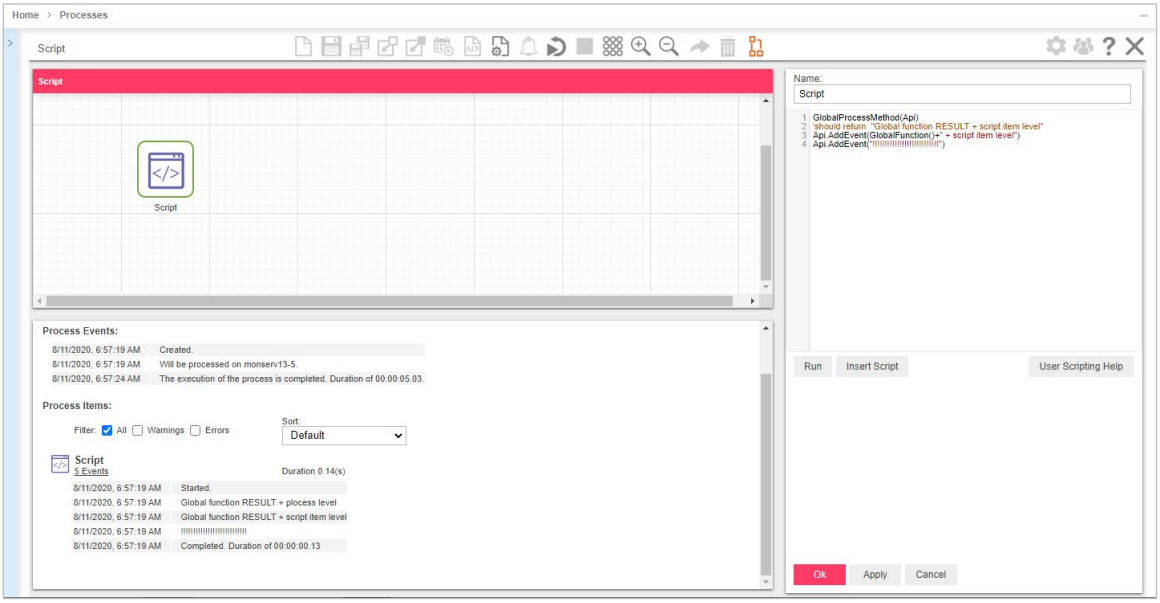
Click **Apply.**

8. Let's try running our script. In the Property panel, click the **Run** button.

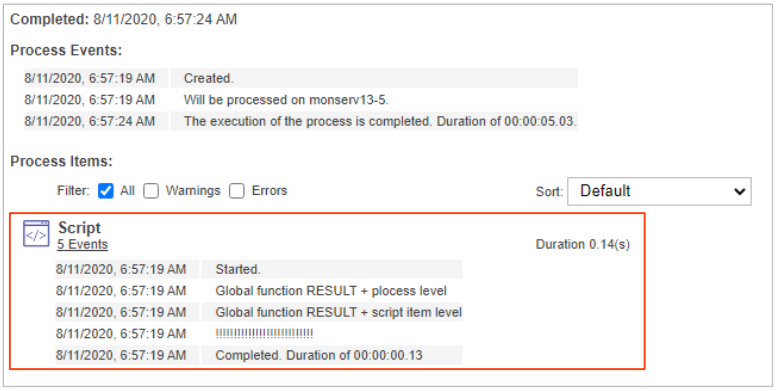
   The results are shown in the Run Script window.




When we run our script as part of a process by clicking the **Save and Run** button in the Visual Process Designer toolbar, the Log Mode view should display as follows.



You will notice that the results of the script we ran in the previous window are shown in the log details.




**NOTE** The results of running the script from the Property panel may not always be the same as those of running the full Visual Process.

You can now try out other scripts for your Visual Processes. Click the User Scripting Help button to find more examples of scripts you can use.

# ADDITIONAL RESOURCES

This completes the Automator tutorial.

For additional information not covered in this tutorial, you can do the following:

- Thoroughly explore the web help.
- Ask for technical assistance from the Technical Support department (for all registered Automator owners). For more information, refer to the "Technical Support" topic in the web help.

## CONTACT US

### GET IN TOUCH

We'd love to hear from you. Here's how you can reach us.

### SALES CONTACT INFORMATION

**Portal**: Contact Altair

**US:** + 1.800 445 3311

**International:** + 1 978 441 2200

### TECHNICAL SUPPORT

**Portal:** Customer Support | Altair