# Table of Contents

**JAMS 7.X Client Features**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jobs In JAMS</td>
<td>3</td>
</tr>
<tr>
<td>Working with Jobs</td>
<td>4</td>
</tr>
<tr>
<td>Execution Methods</td>
<td>5</td>
</tr>
<tr>
<td>Submitting Jobs Manually</td>
<td>19</td>
</tr>
<tr>
<td>Sequence Jobs</td>
<td>20</td>
</tr>
<tr>
<td>Workflow Jobs</td>
<td>21</td>
</tr>
<tr>
<td>Workflow Activities</td>
<td>22</td>
</tr>
<tr>
<td>Migrating JAMS Objects</td>
<td>23</td>
</tr>
<tr>
<td>File Transfer Features</td>
<td>24</td>
</tr>
<tr>
<td>Variables</td>
<td>25</td>
</tr>
<tr>
<td>Working with Variables</td>
<td>46</td>
</tr>
<tr>
<td>Elements</td>
<td>47</td>
</tr>
<tr>
<td>Documentation Elements</td>
<td>48</td>
</tr>
<tr>
<td>EventHandler Elements</td>
<td>49</td>
</tr>
<tr>
<td>Prerequisite Elements</td>
<td>50</td>
</tr>
<tr>
<td>Result Elements</td>
<td>51</td>
</tr>
<tr>
<td>Trigger Elements</td>
<td>52</td>
</tr>
<tr>
<td>Folders</td>
<td>53</td>
</tr>
<tr>
<td>Working with Folders</td>
<td>54</td>
</tr>
<tr>
<td>Dates and Times</td>
<td>55</td>
</tr>
<tr>
<td>Date Properties</td>
<td>56</td>
</tr>
<tr>
<td>Creating Special Date Definitions</td>
<td>57</td>
</tr>
<tr>
<td>Specifying Dates Using Natural Language</td>
<td>58</td>
</tr>
<tr>
<td>Named Times in JAMS</td>
<td>59</td>
</tr>
<tr>
<td>Dashboards and Reports</td>
<td>60</td>
</tr>
<tr>
<td>Custom Dashboards</td>
<td>61</td>
</tr>
<tr>
<td>Creating New Report Templates and Customizing Existing Reports</td>
<td>62</td>
</tr>
<tr>
<td>Manually Generating Reports</td>
<td>63</td>
</tr>
<tr>
<td>Integrating Reports with Automated JAMS Jobs</td>
<td>64</td>
</tr>
<tr>
<td>SQL Server Reporting Services (SSRS)</td>
<td>65</td>
</tr>
<tr>
<td>Monitors and Views</td>
<td>66</td>
</tr>
<tr>
<td>Using the Monitor</td>
<td>67</td>
</tr>
<tr>
<td>Working with Projected Schedules</td>
<td>68</td>
</tr>
<tr>
<td>Resources, and Queues</td>
<td>69</td>
</tr>
</tbody>
</table>

©2020 HelpSystems, LLC. All Rights Reserved.
JAMS 7.X Client Features
Jobs In JAMS
Working with Jobs

Essentially, every component in JAMS supports or controls how a Job is defined or run. A JAMS Job includes a number of Soft Properties and Elements that modify how and when it should run, as well as actions that should be taken when the Job finishes. A JAMS Job may be configured with hundreds of properties and elements to ensure that it executes at the right time and under the proper conditions.

Adding a New Job

1. To create a new Job, open the Definitions view from the Shortcut bar, then select the Job Definitions tab.
2. Click the Add button in the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, select the Folder where the Job should be created.
4. Define a Name for the new Job.
5. If desired, give the Job a Description.
6. Select an Execution Method for the Job from the drop-down list.
7. By default, the full Job Definition dialog will open when the Job is initially saved. Click OK.
8. The Job Definition dialog will open.
10. When the Job has been configured as desired, click **Save and Close**.

### Modifying an Existing Job

1. To make changes to an existing Job, open its **Job Definition** dialog from the Definitions View. To open the **Job Definition** dialog, users may **Double-Click** the Job, Select the Job and then click the **Properties** button in the **Control Bar**, or **Right-click** and select **Properties** from the **Job Options** menu.

2. The **Job Definition** dialog appears.


4. When the Job has been reconfigured as desired, click **Save and Close**.

### Deleting an Existing Job
1. Open the **Definitions** view from the Shortcut bar, then select the **Job Definitions** tab.
2. Select the desired **Job**, then click the **Delete** button in the **Control Bar**, or **Right-Click** the desired **Job**, then select **Delete** from the **Job Options** menu.
3. A dialog will appear to confirm the Job Deletion. If the desired Job was selected for deletion, click **Yes**. If the incorrect Job was selected, click **No** or **Cancel**.

### Job Definition Tabs

#### Job

This tab includes basic Job information:

- **Job Name** - This property incorporates the unique identifier for a JAMS Job and is analogous to the filename for a command file.
- **Description** - This optional property appears in menus, lists and reports and provides a more detailed explanation for the current Job.
- **Last Changed** - The Last Changed property displays the user (Credential) who last modified this Job along with the date and time of the modification.

#### Source

This tab is where you can view or edit the source for the current Job. The Job’s Execution Method defines what kinds of interface to display in this tab.

The Source interface of a Job in JAMS can be a source code editor containing any number of code languages, a fill-in-the-blank style form, a Sequence editor, or a Workflow editor.

The source for many Execution Methods will be a source code editor, which supports syntax highlighting and code coloring to enhance the readability of scripts.

Variable and Parameter references may be embedded in the Job’s command file. This variable data can come from the Job’s Parameters, JAMS Variables, or JAMS Predefined Symbols.

NOTE: The Source tab has an option to let you edit source code in the PowerShell ISE. If you make edits and close the PowerShell ISE, you need to save the Job, close it, and open it again to make additional edits to the source.
Keyboard shortcut commands for the Job Source Editor are shown in the table below. Use the header to expand the content.

### Source Code Editor Keyboard Shortcuts

<table>
<thead>
<tr>
<th>Command</th>
<th>Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Clipboard</strong></td>
<td></td>
</tr>
<tr>
<td>Copy</td>
<td>CTRL+C, CTRL+INSERT</td>
</tr>
<tr>
<td>Paste</td>
<td>CTRL+V, SHIFT+INSERT</td>
</tr>
<tr>
<td>Cut</td>
<td>CTRL+X, SHIFT+DEL</td>
</tr>
<tr>
<td>Select All</td>
<td>CTRL+A</td>
</tr>
<tr>
<td><strong>File Operations</strong></td>
<td></td>
</tr>
<tr>
<td>Save As</td>
<td>CTRL+S</td>
</tr>
<tr>
<td>Open</td>
<td>CTRL+O</td>
</tr>
<tr>
<td><strong>Printing</strong></td>
<td></td>
</tr>
<tr>
<td>Print</td>
<td>CTRL+P</td>
</tr>
<tr>
<td><strong>Positioning</strong></td>
<td></td>
</tr>
<tr>
<td>Go to line</td>
<td>CTRL+G</td>
</tr>
<tr>
<td>Go to start</td>
<td>CTRL+Home</td>
</tr>
<tr>
<td>Go to end</td>
<td>CTRL+End</td>
</tr>
<tr>
<td><strong>Search and Replace</strong></td>
<td></td>
</tr>
<tr>
<td>Find</td>
<td>CTRL+F</td>
</tr>
<tr>
<td>Find Next</td>
<td>F3</td>
</tr>
</tbody>
</table>
Parameters

A JAMS Parameter contains a single slice of data that (unlike a Variable) remains exclusive to the associated Job or Folder.

When JAMS interactively submits a Job with Parameters, the user is presented with a fill-in-the-blank form to define values for these parameters. When JAMS builds a Parameter form, the Parameters are placed in the order in which they are defined, unless the Parameters were given Sequence numbers.

If a Job uses a parsed Execution Method, you can embed Parameter names into the Job’s Source using the \<<ParameterName>> specification. When the Job runs, it uses the value of the Parameter in place of this specification.

A JAMS Job may contain zero or more Parameters.

Add a Parameter

1. In any Job Definition dialog, open the Parameters tab.
2. Click the Add button to open the Add a Parameter dialog.
3. In the dialog, define the Parameter Name, Data Type, and Default Value of the Parameter.
1. By default, the full Parameter Definition dialog will open when the Job is initially saved. Click OK.
2. The Parameter Definition dialog will open.
3. Define additional Parameter Properties as desired.
4. Save and Close the Parameter Definition dialog.
5. Save and Close the Job Definition dialog.

Parameter Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ParamName</td>
<td>This property defines the name of the Parameter. Each Parameter name must be unique within a particular Job.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataType</td>
<td>Use the dropdown to specify a data type for the Parameter.</td>
</tr>
<tr>
<td></td>
<td>• Text</td>
</tr>
<tr>
<td></td>
<td>• Integer</td>
</tr>
<tr>
<td></td>
<td>• Date</td>
</tr>
<tr>
<td></td>
<td>• Time</td>
</tr>
<tr>
<td></td>
<td>• DateTime</td>
</tr>
<tr>
<td></td>
<td>• Float</td>
</tr>
<tr>
<td></td>
<td>• Unknown</td>
</tr>
</tbody>
</table>
Boolean

Length
If the Parameter’s data type is set to Text or Integer, specify the maximum length within this property.

User Interface Description

Allow Entry
When this attribute is enabled, the user can make an entry to this Parameter when submitting the Job.

HelpText
The entered text displays when a user hovers over the Parameter entry field on manual submissions.

Hide
This attribute determines if the Parameter is displayed when this Job is submitted. If you have "Manage" access to the current Job, hold down the ALT key when submitting the Job and all hidden Parameters will be displayed.

Must Fill
When enabled, this attribute requires the user to completely fill this Parameter to its Maximum Length as set on the Data Type tab.

Prompt
When a Job is interactively submitted, a form is created which is used to prompt the user for the Job’s Parameter values. The entered text represents the prompt to the user submitting the Job.

Required
When checked, this attribute requires users to enter a value for this Parameter.

Sequence
An integer used to sort the parameters when presented to end-users.

Uppercase
When checked, this attribute converts all entered data to uppercase.

ValidationData
The value used when attempting to validate the Parameter value.

ValidationType
This property allows you to select the type of validation JAMS can perform on the Parameter value. The validation types are:

- **None**: No validation is performed.
- **Directory**: this validation is used on text parameters. A button is placed next to the parameter’s text box that opens a dialog allowing the user to browse the file system for a directory.
- **SaveFile**: this is used on text parameters. A button is placed next to the parameter’s textbox that opens a dialog allowing the user to browse for a file, which may or may not exist.
- **OpenFile**: this is used on text parameters. A button is placed next to the parameter’s textbox that opens a dialog allowing the user to browse the file system for a file, which must exist.
- **MaskedEdit**: this validation type uses a mask to distinguish between proper and improper user input. The mask is set in the Validation Data property.

Value Description

Default Format
The format string that defines how the Parameter will be formatted. Default formats vary depending on the Parameter’s Data Type.

Default Value
The property specifies the default value for this Parameter. The default value for dates may be entered as a specific date or you can use JAMS English language date text such as Today, Last Monday, First Monday of Month etc.

Variable
Optionally, you can acquire the Default Value for the Parameter using a JAMS Variable. Use the File Browser button to search for and select an existing variable.

Once the Parameter is configured in the Job definition it can be specified in the Source tab.

Elements

There are five Element categories in JAMS. These are Documentation, Event Handler, Prerequisite, Result, and
Trigger.

### Documentation Description

**Documentation**

- **Link**
  - Used to configure hyperlinks for the Documentation of a Job or Folder.

**Event Handler Description**

**Notification Job**
- Used to configure a notification Job to run based on the completion severity or status of a Job.

**Recovery Job**
- Configures a Recovery Job to run if the given Job completion does not exceed recovery severity.

**Repeat**
- Executes the same entry of the Job at a defined interval for a defined time window.

**Resubmit**
- Submits a new entry of the Job at a defined interval for a defined time window.

**Runaway**
- Determines when the Job becomes a Runaway Job, and what action is taken if the Job becomes Runaway.

**Send E-Mail**
- Used to an e-mail based on the completion severity or status of a Job.

**Short**
- Defines the minimum elapsed time for successful completion of a Job, and what action is taken if the Job does not meet that minimum time.

**Stalled**
- Defines the maximum elapsed time for the Job before it is considered stalled.

### Prerequisite Description

**Prerequisite**

**Date Specific Dependency**
- Requires that another Job completed on a natural-language specified date before this Job is allowed to start.

**File Dependency**
- A file that must be present, absent or available before this Job is allowed to start.

**Job Dependency**
- A Job that must satisfy the defined completion severity before this Job will be allowed to start.

**Precheck Job**
- A Job that must succeed before this Job will be allowed to start.

**Remote Job Dependency**
- A remote Job that must satisfy the defined completion severity before this Job will be allowed to start.

**Resource Requirement**
- A resource requirement that must be satisfied before this Job will be allowed to start.

**Variable Dependency**
- A variable dependency that must be satisfied before this Job will be allowed to start.

**Time Window**
- A time window where this Job will be allowed to start, and the action taken if the window is missed.

### Result Description

**Result**

**CR Job Report**
- Configures a CR Job Report to submit on completion of the Job.

**Email Report**
- Configures an Email Report to submit on completion of the Job.

**JDE Report**
- Configures a JD Edwards Print Report to submit on completion of the Job.

**MS Dynamics Email Report**
- Configures a Microsoft Dynamics E-mail Report to submit on completion of the Job.

**MS Dynamics File Report**
- Configures a Microsoft Dynamics File Report to submit on completion of the Job.

**MS Dynamics Print Report**
- Configures a Microsoft Dynamics Print Report to submit on completion of the Job.
<table>
<thead>
<tr>
<th>SSRS Print Report</th>
<th>Configures an SSRS Print Report to submit on completion of the Job.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Trigger</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>File Trigger</td>
<td>Triggers a Job to run based on the availability of a file or group of files. Users may define And Groups to require the presence of multiple files before the Job is triggered.</td>
</tr>
<tr>
<td>Job Completion Trigger</td>
<td>Triggers a Job to run based on the completion severity of a defined Job Completion Trigger.</td>
</tr>
<tr>
<td>Mail Trigger</td>
<td>Triggers a Job to run based on the receipt of a mail message.</td>
</tr>
<tr>
<td>Schedule Trigger</td>
<td>Uses natural date language and comma separated date lists to trigger a Job to run. Users may define multiple Schedule Triggers on a single object.</td>
</tr>
<tr>
<td>Variable Trigger</td>
<td>Triggers a Job to run when a variable is changed or set to a specific value.</td>
</tr>
</tbody>
</table>

**Security**

The Security tab acts as an access control list, capable of handling many Access Control Entries (ACE).

**NOTE:** In JAMS V7.0.1367 and later, removing all ACEs on an object behaves the same as Windows would. When all ACEs are removed from an object, only the GrantAdministratorsBypass group will have access to that object. Previously, removing all ACEs from an object would give all Authenticated Users access to that object.

While Jobs will inherit security from their Folder, a Job’s security setting may be configured to override any folder-level security.
Each ACE can specify the following rights on a Job:

- **Abort**: allows the user to access the JAMS Job Monitor to abort or restart an occurrence of this Job.
- **Change**: permits modification of the Job’s definition provided the user also has *Change* access to Job Definitions, which is located in the Access Control list on the Ribbon Bar.
- **Control**: permits modification of the current Job’s Access Control List.
- **Debug**: allows submission of this Job, but only if the `/DEBUG` qualifier is used on the JAMS SUBMIT command. This qualifier submits the Job under the submitter’s username rather than the username specified in the Job or Folder Definitions.
- **Delete**: permits deletion of the current Job definition provided the user also has *Delete* access to Job Definitions.
- **Inquire**: allows inquiry into the current Job definition provided that the user also has *Inquire* access to Job Definitions.
- **Manage**: permits the user to use the JAMS Job Monitor to hold, reschedule, release or delete an occurrence of the current Job.
- **Monitor**: allows the current Job to appear in the Job Monitor.
- **Submit**: grants the right to submit the current Job.

### History

This tab contains historical Job statistics and completion data broken down using a variety of performance measures.

---

**NOTE:** Using the Reset button in the History Tab will remove historical completion statistics and impact the percent completion calculations seen in the Monitor.
A list of Soft Properties are available below. These properties are available on any JAMS Job. Additional Soft Properties may be available depending on a Job's Execution Method.

Until a value is set for a property, it will not display in the Properties tab. To display a soft property, use the Add button in the Properties Tab, select the desired Property, then click OK in the Add Property dialog.

**Completion**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bad Regex Pattern</td>
<td>A regular expression pattern that indicates a failed execution.</td>
</tr>
<tr>
<td>Exit Code Handling</td>
<td>Specifies how JAMS should evaluate a Job's exit code.</td>
</tr>
<tr>
<td>Good Regex Pattern</td>
<td>A regular expression pattern that indicates a successful execution for this Job.</td>
</tr>
<tr>
<td>Minimum Severity</td>
<td>The minimum acceptable completion severity.</td>
</tr>
<tr>
<td>Retain Option</td>
<td>Selects how the Job will be displayed in the Monitor after it completes. If set to Always, completed Jobs will never leave the Monitor.</td>
</tr>
<tr>
<td>Retain Time</td>
<td>When the Retain Option is set to Timed, this property is used to specify (in minutes) the amount of time to display the completed Job in the Monitor.</td>
</tr>
<tr>
<td>Specific Informational</td>
<td>A comma separated list of integer values for Informational Job completion exit codes.</td>
</tr>
<tr>
<td>Specific Values</td>
<td>A comma separated list of integer for the Job completion exit codes.</td>
</tr>
<tr>
<td>Specific Warning</td>
<td>A comma separated list of integer for Warning Job completion exit codes.</td>
</tr>
</tbody>
</table>

**E-Mail**

| Property | Description |
|----------|-------------|-------------|

©2020 HelpSystems, LLC. All Rights Reserved.
<table>
<thead>
<tr>
<th><strong>CC Address</strong></th>
<th>A comma separated list of e-mail addresses that will be cc'd on the e-mail. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>From Address</strong></td>
<td>The from email address used when sending the e-mail from JAMS.</td>
</tr>
<tr>
<td><strong>Message Body</strong></td>
<td>The body of the e-mail message. Users may enter Markdown language in the Message Body.</td>
</tr>
<tr>
<td><strong>To Address</strong></td>
<td>A comma separated list of e-mail addresses that the e-mail will be sent to. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*.</td>
</tr>
</tbody>
</table>

**Execute**

<table>
<thead>
<tr>
<th><strong>Description</strong></th>
<th><strong>Execute As</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Select the credentials this Job should use when executing.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>SSRS Credentials</strong></th>
<th>The credentials that will be passed to the SSRS Server when executing.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application</strong></td>
<td>This value is passed to the Win32 Create Process function as the value for the Application Name argument.</td>
</tr>
<tr>
<td><strong>Command</strong></td>
<td>This value is passed to the Win32 Create Process function as the value for the Command Line argument.</td>
</tr>
<tr>
<td><strong>Home Directory</strong></td>
<td>Specifies the full path to the initial current directory for the Job.</td>
</tr>
</tbody>
</table>

**Options**

<table>
<thead>
<tr>
<th><strong>Description</strong></th>
<th><strong>Force 32 Bit</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>When set to true, this Job will run as a 32 bit process, even on 64 bit machines.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Force V2</strong></th>
<th>When set to true, the Job will run using V2.0 of the .NET Framework.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>No BOM</strong></td>
<td>When set to true, the Job's source file will not have a Unicode byte order mark.</td>
</tr>
<tr>
<td><strong>OSPromptPattern</strong></td>
<td>A regular expression that will match the operating system command prompt.</td>
</tr>
<tr>
<td><strong>Pass Parameters</strong></td>
<td>When set to true, Job parameters will be passed to a PowerShell Job as PowerShell parameters.</td>
</tr>
</tbody>
</table>

**Schedule**

<table>
<thead>
<tr>
<th><strong>Description</strong></th>
<th><strong>Agent</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Used to specify the Agent where the Job will execute.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Batch Queue</strong></th>
<th>Specifies the batch queue for a Job.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Submit on Hold</strong></td>
<td>When enabled, this Job will enter the schedule in an &quot;On Hold&quot; status, requiring user input.</td>
</tr>
<tr>
<td><strong>Submit Date</strong></td>
<td>The default Submit Date for this Job, when manually submitted.</td>
</tr>
<tr>
<td><strong>Job Concurrent Limit</strong></td>
<td>The maximum number of instances of this Job that may execute concurrently.</td>
</tr>
<tr>
<td><strong>Single Instance Action</strong></td>
<td>Defines the action to be taken if a Job tries to start while an instance of that Job is already running.</td>
</tr>
</tbody>
</table>

**Status**

<table>
<thead>
<tr>
<th><strong>Description</strong></th>
<th><strong>Enabled</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>This can be used to enable or disable a Job, Trigger or Event Handler.</td>
<td></td>
</tr>
</tbody>
</table>

**General**

<table>
<thead>
<tr>
<th><strong>Description</strong></th>
<th><strong>Debug</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Determines if Job(s) will run in Debug Mode. When a Job runs in Debug Mode, it will not satisfy Dependencies or Triggers.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Include MS Dynamics Log</strong></th>
<th>If the Job is a MS Dynamics Job, indicates whether the Job will include MS Dynamics Job Logs.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Credentials</strong></td>
<td>Select a set of credentials from the drop-down list for the given object.</td>
</tr>
<tr>
<td><strong>JDE Credential</strong></td>
<td>Credentials provided for JD Edwards Jobs within the Job or Folder. If this is not a JD Edwards Job, these credentials are not required.</td>
</tr>
<tr>
<td><strong>CompletionBearing</strong></td>
<td>Determines the acceptable Completion Severity configuration for a Job.</td>
</tr>
<tr>
<td><strong>Data Type</strong></td>
<td>Determines the Data Type used for this object.</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td><strong>Default Date</strong></td>
<td>The default submit date used when this Job is manually submitted.</td>
</tr>
<tr>
<td><strong>Default Submit Time</strong></td>
<td>The default submit time to use when this Job is manually submitted.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td></td>
</tr>
<tr>
<td><strong>File Transfer Agent</strong></td>
<td>The File Transfer Agent JAMS will connect to for this Job.</td>
</tr>
<tr>
<td><strong>From</strong></td>
<td>The origin or sender of this object.</td>
</tr>
<tr>
<td><strong>Interval</strong></td>
<td>The interval (in Delta Time) between checks for new mail on the defined mail server.</td>
</tr>
<tr>
<td><strong>Keep Logs</strong></td>
<td>Determines whether the batch log files are deleted after the current Job completes. (Only applies to OpenVMS platforms).</td>
</tr>
<tr>
<td><strong>LogFile Name</strong></td>
<td>Displays or sets the Log File Name for this Job.</td>
</tr>
<tr>
<td><strong>Log Location</strong></td>
<td>The default log location. This can include a filename and extension to set the default values.</td>
</tr>
<tr>
<td><strong>MS Dynamics Credential</strong></td>
<td>The credentials to use when running a MS Dynamics Job.</td>
</tr>
<tr>
<td><strong>Notify Severity</strong></td>
<td>The maximum completion severity required to send a notification.</td>
</tr>
<tr>
<td><strong>Notify User</strong></td>
<td>Include the user that submitted the Job when sending notifications?</td>
</tr>
<tr>
<td><strong>Options</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Port</strong></td>
<td>Specifies the Port JAMS will attempt to connect to. Zero (0) is the default.</td>
</tr>
<tr>
<td><strong>Print Location</strong></td>
<td>The directory where Print Files should be placed.</td>
</tr>
<tr>
<td><strong>Print Queue</strong></td>
<td>The Print Queue that will be used for Reports.</td>
</tr>
<tr>
<td><strong>Protocol</strong></td>
<td>The protocol to use when connecting to a mail server. Typical mail server protocols include IMAP and POP3.</td>
</tr>
<tr>
<td><strong>Remote Job Name</strong></td>
<td>The name of the Remote Job.</td>
</tr>
<tr>
<td><strong>Remote Scheduler</strong></td>
<td>The name of the Remote Machine where the Remote Job executes.</td>
</tr>
<tr>
<td><strong>Restartable</strong></td>
<td>Defines whether the Job can be restarted.</td>
</tr>
<tr>
<td><strong>Retry Count</strong></td>
<td>Defines the maximum number of automatic retry attempts if the Job fails.</td>
</tr>
<tr>
<td><strong>Retry Interval</strong></td>
<td>the interval, in Delta Time, between automatic retry attempts.</td>
</tr>
<tr>
<td><strong>Run Priority</strong></td>
<td>The execution priority for the Job. A Job's Priority is the sum of the Job's priority AND priority values on parent folders.</td>
</tr>
<tr>
<td><strong>Scheduling Priority</strong></td>
<td>The Scheduling Priority for the Job.</td>
</tr>
<tr>
<td><strong>Search Path</strong></td>
<td>A comma separated list of paths which are searched when trying to resolve relative references.</td>
</tr>
<tr>
<td><strong>SLA Time</strong></td>
<td>The time of day a Job must complete by to meet a Service Level Agreement.</td>
</tr>
<tr>
<td><strong>Submit Time</strong></td>
<td>The default Submit Time for the Job, when manually submitted.</td>
</tr>
<tr>
<td><strong>Suppress Menu Display</strong></td>
<td>Defines if this Job should be omitted from Submit Menus. Set to False (unchecked) to display this Job on Submit Menus.</td>
</tr>
<tr>
<td><strong>Template Library</strong></td>
<td>The full file specification that contains the JAMS Templates used when parsing the Job.</td>
</tr>
<tr>
<td><strong>Timestamp Logs</strong></td>
<td>If enabled, batch log files will be timestamped with the extension format &quot;LOG_yyyymmd_hhmmsscc&quot;.</td>
</tr>
<tr>
<td><strong>To</strong></td>
<td>The destination or recipient of this object.</td>
</tr>
<tr>
<td><strong>Host Key Checking</strong></td>
<td>Defines what JAMS should do if the SSH fingerprint does not match when connecting.</td>
</tr>
<tr>
<td><strong>Accept Host Key</strong></td>
<td>An indicator that a host key is accepted</td>
</tr>
<tr>
<td><strong>SSH Prompt</strong></td>
<td>A regular expression pattern to handle SSH prompts.</td>
</tr>
</tbody>
</table>
SSH Timeout  The SSH connection timeout, defined in seconds.
SSL Cipher Suite  Comma separated list of SSL Cipher Suites to allow.
SSL Version  Comma separated list of the SSL Versions to allow.
SSL Minimum Key Size  The minimum accepted key size. When set to zero (0), will default to 1024.
SSH Compression  The compression level to use with the SSH connection.
SSH Minimum Key Size  The minimum accepted decimal key size. When set to zero (0), will default to 1024.
SSH Mac Algorithm  The SSH Mac Algorithm used.
SSH Key Exchange  The SSH Key Exchange Algorithm used.
SSH Host Key  The SSH Host Key Algorithm used.
SSH Encryption  Comma separated list of SSH Encryption Algorithms used.

Documentation
The Documentation Tab displays the Documentation Elements assembled from the Job and any parent Folders.
Job Documentation may be edited from the Elements of the Job or Folder where the Documentation is inherited from.

My Sample Documentation Content
Put important Job information here.

Job Documentation can include basic formatting, like

**Bold Content**

*colored text*

Hyperlinks:

- Lists

and more.
Execution Methods

An Execution Method defines how a Job executes within the JAMS environment. A default Execution Method is assigned to every Job if one is not specifically assigned.

JAMS ships with dozens of Executions Methods to run Jobs on Windows, Unix, Linux, OpenVMS, or any operating system supporting connections using SSH.

In addition to the preconfigured Execution Methods that ship with JAMS, users can create and customize their own Execution Methods.

Creating a Custom an Execution Method entails a two-step process: creating the Execution Method, and then modifying its properties. Both are described in detail below.

Creating a New Execution Method

1. To create a new Execution Method, open the Execution Methods view from the Shortcut bar.
2. Click the Add button in the Control Bar to open the Add an Execution Method dialog.
3. Define a Name for the new Execution Method.
   NOTE: The name must be unique to existing Execution Methods.
4. If desired, give the Execution Method a Description.
5. Select a Base Execution Method from the drop-down list.
6. By default, the full Job Definition dialog will open when the Job is initially saved. Click OK.
7. The Job Definition dialog will open. Define the Execution Method, Template, Parameters, Properties, and Job Properties as desired.
8. When the Execution Method is configured as desired, Save and Close.

Modify an Existing Execution Method

1. To make changes to an existing Execution Method, open its Execution Method Definition dialog from the Execution Methods View.
   To open the Execution Method Definition dialog, users may Double-Click the execution method, Select the Execution Method and then click the Properties button in the Control Bar, or Right-click and select Properties from the Method Options menu.
2. The Execution Method Definition dialog appears.
3. Modify the Execution Method, Template, Parameters, Properties, and Job Properties as desired.
4. When the Execution Method has been reconfigured as desired, click Save and Close.

Delete an Existing Execution Method

1. Open the Execution Methods view from the Shortcut bar.
2. Select the desired Execution Method, then click the Delete button in the Control Bar, or Right-Click the desired Execution Method, then select Delete from the Method Options menu.
3. A dialog will appear to confirm the Execution Method deletion. If the desired Execution Method was selected for deletion, click Yes. If the incorrect Execution Method was selected, click No or Cancel.

Execution Method Definition Tabs

Execution Method

This tab contains the basic information about the Execution Method.

- Method Name - This is the unique identifier for the Execution Method.
- Description - This reference property is used to summarize the Execution Method definition.
- Last Changed - Indicates the date and time this Execution Method parameter was last modified.

Parameters
Execution Method parameters work the same way as Parameters within Job definitions and apply to all Jobs using this Execution Method. To insert a new Parameter, select the Add button on the Parameters tab to access the Add a Parameter dialog.

**NOTE:** if a Job has a Parameter with the same name as the Execution Method Parameter, the Job definition is used.

## Template

Templates provide an easy way to include standard pre and post processing for your Jobs. At submit time, JAMS parses a Job source using either the template defined on this tab or the module defined in the Job Module property for the Job’s source code. Use the built-in text editor to edit the template for this Execution Method.

The example below sets up a simple template that maps two network drives, inserts the Job’s source code and executes a LogJobComplete program:

```plaintext
LogJobComplete
NET USE E: \MyServer\EShare
NET USE F: \MyServer\FShare
<<JAMS.Current.Source>>
C:\LogJobComplete.exe <<JAMS.Job.JobName>>
```

## Properties Table

<table>
<thead>
<tr>
<th>Assemblies</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Assembly Name</td>
<td>The name of the assembly that contains the class used to edit the source for this execution method.</td>
</tr>
<tr>
<td>Edit Class Name</td>
<td>The name of the class that implements IViewExecutingJob and IViewJobHistory and will be used to view Jobs that use this execution method.</td>
</tr>
<tr>
<td>Host Assembly Name</td>
<td>The name of the assembly that contains the class used to host this execution method.</td>
</tr>
<tr>
<td>Host Class Name</td>
<td>The name of the class that implements IJAMSHost and will host this execution method.</td>
</tr>
<tr>
<td>View Assembly Name</td>
<td>The name of the class that implements IEditSource and will be used to edit the source for this Execution Method.</td>
</tr>
<tr>
<td>View Class Name</td>
<td>The name of the assembly that contains the class used to view Jobs that use this execution method.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Execute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution Type</td>
<td>This dropdown allows you to choose the Execution Method type. The supported types are:</td>
</tr>
<tr>
<td></td>
<td>- Agent runs a Job using a JAMS Agent.</td>
</tr>
<tr>
<td></td>
<td>- Batch runs a Job by creating a process with a Job or Command.</td>
</tr>
<tr>
<td></td>
<td>- Routine executes a Job using the specified class from a specified assembly using the IJAMSHost interface.</td>
</tr>
<tr>
<td></td>
<td>- SSH Agent executes a Job using a secure shell commands with JAMS AgentX.</td>
</tr>
<tr>
<td>Application</td>
<td>This value is passed to the Win32 Create Process function as the value for the Application Name argument.</td>
</tr>
</tbody>
</table>
Command

This value is passed to the Win32 Create Process function as the value for the Command Line argument.

### Options Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Force 32bit</td>
<td>When set to true, this Job will run as a 32 bit process, even on 64 bit machines.</td>
</tr>
<tr>
<td>Force V2</td>
<td>When set to true, this Job will run using V2.0 of the .NET Framework.</td>
</tr>
<tr>
<td>Host Locally</td>
<td>Indicates that the Job should only be hosted by the machine running the JAMS Scheduler. This setting is used for Execution Methods that need to manage connection to the Agent machine. An example would be the SSH Execution Method which executes locally and uses SSH to connect to the Agent in order to run the actual job.</td>
</tr>
<tr>
<td>Interactive</td>
<td>When enabled, this Execution Method can interact with the desktop. <strong>Caution:</strong> this setting is not recommended since it can lead to potential security issues. The Interactive property has been included here because some applications require access to the desktop in order run properly.</td>
</tr>
<tr>
<td>No BOM</td>
<td>When set to true, the Job's source file will not have a Unicode byte order mark.</td>
</tr>
<tr>
<td>Run As Job</td>
<td>When set to true, the Job will run as the user defined in the Job even when Host Locally is set to true.</td>
</tr>
</tbody>
</table>

### Source Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit After Start</td>
<td>When activated, this setting permits the source to be edited after a manually submitted Job has started.</td>
</tr>
<tr>
<td>Edit Source</td>
<td>When activated, this checkbox permits the source to be edited when manually submitting the Job</td>
</tr>
<tr>
<td>Extension</td>
<td>Specifies the default file extension for source code files</td>
</tr>
<tr>
<td>Job Module</td>
<td>This is the name of a text module in the template library which is used as the starting point for parsing the Job's command procedure. Entering a value in this property implies that this Job is parsed. The template library is a combination of the macros in the BaseMacros.xml file, the file specified in the &quot;DefaultMacroFile&quot; configuration setting and the macros in the xml file specified in the Job's Folder definition. The BaseMacros.xml file is replaced when installing a JAMS upgrade.</td>
</tr>
<tr>
<td>Snapshot Source</td>
<td>If enabled, the option allows JAMS to grab a copy of the source when submitting a Job.</td>
</tr>
</tbody>
</table>

### User Interface

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prompt</td>
<td>The prompt that is displayed to a user when the Job is manually submitted.</td>
</tr>
</tbody>
</table>

### Completion Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bad Regex Pattern</td>
<td>A regular expression pattern, matched in log file output, that indicates a failed execution.</td>
</tr>
<tr>
<td>Exit Code Handling</td>
<td>Specifies how JAMS should evaluate a Job's exit code.</td>
</tr>
<tr>
<td>Good Regex Pattern</td>
<td>A regular expression pattern, matched in log file outputs, that indicates a successful execution for this Job.</td>
</tr>
<tr>
<td>Specific Informational</td>
<td>A comma separated list of integer values for Informational Job completion exit codes.</td>
</tr>
<tr>
<td>Specific Values</td>
<td>A comma separated list of integer values for the Job completion exit codes.</td>
</tr>
</tbody>
</table>
based on the Exit Code Handling setting.

Specific Warning A comma separated list of integer values for Warning Job completion exit codes.

Predefined Execution Methods

JAMS ships with dozens of Predefined Execution Methods that can run a variety of Jobs on Windows, Linux, UNIX and other operating systems detailed below.

Predefined Execution Methods Table

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Banner</td>
<td>Executes a Banner job</td>
<td>Windows</td>
</tr>
<tr>
<td>BannerWin</td>
<td>Executes a Banner job on Windows</td>
<td>Windows</td>
</tr>
<tr>
<td>Command</td>
<td>Used to run a Windows batch procedure</td>
<td>Windows</td>
</tr>
<tr>
<td>CRJobV12</td>
<td>Crystal Reports Job using Crystal Reports version 12</td>
<td>Windows</td>
</tr>
<tr>
<td>CRJobV13</td>
<td>Crystal Reports Job using Crystal Reports version 13</td>
<td>Windows</td>
</tr>
<tr>
<td>File Transfer</td>
<td>Defines a file Transfer (FTP, SFTP, or FTP/SSL)</td>
<td>Windows</td>
</tr>
<tr>
<td>FileTransferV1</td>
<td>Defines a file Transfer (FTP, SFTP, or FTP/SSL) Replaces File Transfer from V6.4.51 and under</td>
<td>Windows</td>
</tr>
<tr>
<td>File Watch</td>
<td>Watches for a File</td>
<td>Multiple</td>
</tr>
<tr>
<td>InformaticaCloud</td>
<td>Executes an Informatica Cloud task.</td>
<td>Windows</td>
</tr>
<tr>
<td>JAMS Report</td>
<td>Generates a JAMS Report</td>
<td>Windows</td>
</tr>
<tr>
<td>JDEJob</td>
<td>Runs a J.D. Edwards job</td>
<td>Windows</td>
</tr>
<tr>
<td>JDEJobSsh</td>
<td>Runs a J.D. Edwards job on Linux via SSH</td>
<td>Windows</td>
</tr>
<tr>
<td>Mail Watch</td>
<td>Watches for an Email</td>
<td>Multiple</td>
</tr>
<tr>
<td>MF Job</td>
<td>Executes a Micro Focus Job</td>
<td>Windows</td>
</tr>
<tr>
<td>MSDAX2012Job</td>
<td>Executes a Microsoft Dynamics AX job</td>
<td>Windows</td>
</tr>
<tr>
<td>NeoBatchCatalog</td>
<td>Executes a NeoBatch Job from a NeoBatch Catalog</td>
<td>Windows</td>
</tr>
<tr>
<td>NeoBatchJcl</td>
<td>Executes NeoBatch Job from JCL stored in JAMS</td>
<td>Windows</td>
</tr>
<tr>
<td>NetezzaBackup</td>
<td>Runs a Netezza backup job using SSH</td>
<td>Unix, Linux</td>
</tr>
<tr>
<td>NetezzaLoad</td>
<td>Runs a Netezza Load job using SSH</td>
<td>Unix, Linux</td>
</tr>
<tr>
<td>NetezzaSQL</td>
<td>Executes a Netezza Query job using SSH</td>
<td>Unix, Linux</td>
</tr>
<tr>
<td>OdbcCommand</td>
<td>Executes ODBC Commands</td>
<td>Windows</td>
</tr>
<tr>
<td>OleDbCommand</td>
<td>Executes OLEDB Commands</td>
<td>Windows</td>
</tr>
<tr>
<td>OracleEBSConcurrent</td>
<td>Runs an Oracle EBS Concurrent process</td>
<td>AIX, HP-UX, Linux, Solaris</td>
</tr>
<tr>
<td>OracleEBSJobSsh</td>
<td>Runs an Oracle E-Business Suite Job via SSH</td>
<td>AIX, HP-UX, Linux, Solaris</td>
</tr>
</tbody>
</table>
| OracleEBSRequestSet   | Executes an Oracle ES Set job                                     | AIX, HP-
<table>
<thead>
<tr>
<th>Task Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Stored Proc</td>
<td>Executes an Oracle Stored Procedure Job</td>
</tr>
<tr>
<td>OS400</td>
<td>Runs IBM System i, OS/400 jobs</td>
</tr>
<tr>
<td>PeopleSoft Job Ssh</td>
<td>Executes a PeopleSoft job using SSH</td>
</tr>
<tr>
<td>PeopleSoft Job Windows</td>
<td>Runs a PeopleSoft job on Windows</td>
</tr>
<tr>
<td>PowerShell</td>
<td>Runs a PowerShell script on Windows</td>
</tr>
<tr>
<td>PowerShell32</td>
<td>Runs a PowerShell script (32 bit)</td>
</tr>
<tr>
<td>SAP Data Service</td>
<td>Executes a SAP Business Objects Data Services Job</td>
</tr>
<tr>
<td>SAP Job V2</td>
<td>Executes a SAP multi-step job</td>
</tr>
<tr>
<td>SAP Process Chain</td>
<td>Executes an SAP Process Chain</td>
</tr>
<tr>
<td>Sequence</td>
<td>A building-brick style interface to configure and run tasks such as Jobs and File Transfers in sequence or in parallel. Replaces V6.X Setups.</td>
</tr>
<tr>
<td>SQL Agent</td>
<td>Runs SQL Server Agent Job</td>
</tr>
<tr>
<td>SQL Command</td>
<td>Runs SQL commands</td>
</tr>
<tr>
<td>SQL Stored Proc</td>
<td>Executes an SQL Stored Procedure</td>
</tr>
<tr>
<td>SSH</td>
<td>Runs a Secure Shell</td>
</tr>
<tr>
<td>SSH Agent</td>
<td>Deploys JAMS AgentX using SSH</td>
</tr>
<tr>
<td>SSH Deploy</td>
<td>Deploys JAMS AgentX using AgentX</td>
</tr>
<tr>
<td>SSH Open VMS</td>
<td>Runs a Secure Shell with OpenVMS using prompt matching</td>
</tr>
<tr>
<td>SSH Prompt</td>
<td>Runs a Secure Shell using prompt matching</td>
</tr>
<tr>
<td>SSIS</td>
<td>Runs SQL Server Integration Services</td>
</tr>
<tr>
<td>SSIS Direct</td>
<td>Runs SQL Server Integration Services</td>
</tr>
<tr>
<td>SSRS</td>
<td>SQL Server Reporting Services</td>
</tr>
<tr>
<td>Unix Script</td>
<td>Run a Unix Script based on the shebang on the first line</td>
</tr>
<tr>
<td>Unix Shell</td>
<td>Run a shell script on a Unix Server.</td>
</tr>
<tr>
<td>Vbs</td>
<td>Executes a VB Script</td>
</tr>
<tr>
<td>VMS</td>
<td>Runs an OpenVMS DCL Command Procedure</td>
</tr>
<tr>
<td>VMS Parsed</td>
<td>Runs an OpenVMS DCL Command Parsed Procedure</td>
</tr>
<tr>
<td>Windows Deploy</td>
<td>Deploys a JAMS Agent to a Windows machine</td>
</tr>
<tr>
<td>Workflow</td>
<td>Runs a Windows Workflow job</td>
</tr>
<tr>
<td>z/OS</td>
<td>Executes an IBM z/OS job</td>
</tr>
</tbody>
</table>
Submitting Jobs Manually

The Submit Jobs View is a listing of all current JAMS Jobs that are available to run on demand. This View uses the same expanding folder hierarchy as displayed in the Definitions view.

With the built-in security model, JAMS users can submit Jobs as long as they have the proper access. As part of the submittal process, users can also modify Job output characteristics such as the print queue or number of copies to output a report.

Once the end-user submits a Job they may be presented with a fill-in-the-blank form to define the Job’s required parameters, which can then be edited and verified before the Job is submitted.

Submit a Job using the Submit view

1. Open the Submit Shortcut.
2. Expand the Folder and select the Job to be submitted.
3. Click the Submit button on the Control Bar to open the Submit dialog box. **NOTE:** Users may also open the Submit dialog by double-clicking the desired Job, or right-clicking the desired Job and selecting Submit from the drop-down menu.
4. Define information as required in the Parameters, Reports, and Schedule tabs of the Submit dialog.
NOTE: Depending on the Job Type, Elements, and Parameters, there may only be one or two tabs visible.

5. With the Job information defined, click **Submit Run Request**.
Sequence Jobs

The Sequence Execution Method is available in all JAMS 7.X installations. Sequences are commonly used to schedule a stream of Jobs that automatically execute at regular intervals. JAMS V6.X users will recognize Sequences as the successor to and replacement for JAMS Setups.

Sequences consist of Logical Control Flows and Tasks. Tasks may include File Transfer actions, Failure actions, and Job Submissions. Tasks reside in and are directed by Control Flow containers. For instance, any Sequence Source with more than one Task will be wrapped in a master Sequence container.

Creating a Sequence Job

1. Click Definitions from the Shortcut bar.
2. Select the Job Definitions tab.
3. Click the Add button in the Control Bar to open the Add a New JAMS Job Definition dialog. (Alternatively, use the Add a New JAMS Job Definition button from the Welcome screen.)
4. In the dialog, select the Folder where the Job should be created.
5. In the Name field, define a name for the new Job.
6. In the Description field, enter a description for the Job, if necessary.
7. In the Execution Method field, select Sequence Execution Method.
8. Click OK. By default, the full Job Definition dialog opens the Source tab when the Job is initially saved.
9. Use the Sequence Designer, Task Toolbar, and Properties to define the Sequence. (Task Toolbar, Sequence Designer, and Properties sections are described below.)
10. Define Elements and Properties on the Sequence Job as desired to control how the Sequence will run.
11. Save and Close the Sequence.

Sequence Source Layout

The object-oriented Sequence Source is made up of three sections - the Task Toolbox, the Design Area, and the Properties Panel.

The Task Toolbox (pinned left by default) contains a list of Tasks and Flow Controls that act as the building blocks of a Sequence. These tasks are organized into groupings based on their type and purpose, such as JAMS, File Transfer, and Control Flow.

The Design Area is where items from the Task Toolbox are assembled. Drag and drop tasks to the Design Area. Note that this Design Area will have a master Sequence container added automatically when multiple Tasks exist in the Design Area. All other tasks and conditional flows will be nested inside of this master Sequence container. The Tasks defined in the Sequence Source will run in sequential order from top to bottom by default.

**NOTE:** Deleting a container will also delete any objects (Tasks or Control Flows) within that container.

The Properties panel (pinned right by default) is used to populate or modify the tasks used in the Sequence. The Properties listed are responsive to the selected task or container inside of the Design Area. For example, selecting a Submit Job task will display the information relevant only to that Submit Job task.
Adding Tasks and Containers to a Sequence

1. Identify the Task or Container that should be added to the Sequence Source. Drag and drop the object into the Design area of the Sequence Source. A Sequence may be saved with a single Task in the Source.

2. Resolve any Validation Errors with the new Task or Container. Hovering over the Task will display the relevant issue.
3. Add additional Tasks and Containers as desired, ensuring that no outstanding errors exist on any object.

**NOTE:** By default, JAMS Sequences execute Tasks in order from top to bottom. Ensure Tasks are arranged accordingly.

4. If necessary, Tasks and Containers may be rearranged within the Design area using drag-and-drop functionality.

**NOTE:** Containers have a Parent-Child relationship. Removing or rearranging a container will remove or rearrange any child tasks or child containers.

5. When the Sequence Source is configured as desired, **Save** the Job.

**Using the Properties Panel**

Select any Task or Control Flow from the Designer area to view its specific properties. Use the Properties panel to populate or modify individual property fields for that object.
Resetting Values

If you specified an override value at the Task level and you want to revert the value back to the default, you can reset the value. Click the square next to the right of the value and select Reset. This works the same as resetting the property on the Job Properties tab.

Inherited Values

If you have values that are set within a Job and then add an override value within the Submit Job task for the Job, the Reset option uses the original value at the Job level and not the override value. Properties and Elements are inherited from the selected Job and are displayed in the Schedule section under Properties when you select a Submit Job Task.

Understanding Parameter Behavior in Sequences

JAMS identifies when a Sequence and child Job in that sequence each have a Parameter with the same name. By default, JAMS will override the child Job Parameter value with the value set on the Sequence itself for that matched Parameter.

How Sequence Parameters interact with child Job Parameters

If a Sequence and child Job contain Parameters with matching names, the default value set on the Sequence level Parameter will override the default value set on the child Job level Parameter. This happens because the Use Parameters field is checked by default.

To use the Parameter value set on the child Job definition rather than the Sequence level value, you must clear the Use Parameters field in the Properties Panel of the given Submit Job Task.
To override both the Sequence level Parameter value and the child Job level Parameter value, you would need to open the Job Parameters Collection and enter the override value in the Parameter Value field. This value overrides both Sequence and child Job level values for the given Parameter, regardless of the **Use Parameters** field state.
Parameter Value
The value of the Parameter.
Workflow Jobs

JAMS Workflow Jobs use the Windows Workflow Foundation that provides an extensible collection of Activities using a rich interface for designing Job execution flows. JAMS extends the base set for these Workflow Activities with a collection of its own Activities for interacting and performing tasks directly within the JAMS Client.

The JAMS Workflow execution method is designed for Job scenarios using decision logic and branching that require different execution paths. Users seeking similar function may wish to try using the Sequence Execution Method. Workflows are created and edited using the Workflow Designer which includes a menu of building blocks (Activities) for submitting Jobs, updating JAMS Variables, and many more JAMS specific tasks. To view a listing and description for all Workflow Activities, select the link.

Setting up the Workflow

1. To create a new Job, open the Definitions view from the Shortcut bar, then select the Job Definitions tab.
2. Click the Add button in the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, select the Folder where the Job should be created.
4. Define a Name for the new Job.
5. If desired, give the Job a Description.
6. Select the Workflow Execution Method for this Job.
7. By default, the full Job Definition dialog will open when the Job is initially saved. Click OK.
8. The Job Definition dialog will open.
9. Click on the Source tab to open the Workflow Designer Panel, explained below.

The Workflow Designer

The object-oriented workflow interface is made up of three panels. The left Toolbox panel contains a list of Activities that can be used as building blocks to customize a JAMS Job to most any specification.

The Design panel is where the workflow building blocks are assembled by dragging and dropping selected Activities from the Toolbox.

Finally, the Properties panel is used to populate or modify individual property fields related to individual Activities.
Each Activity is organized into a grouping (e.g., JAMS, SQL, PowerShell, etc.). If specific JAMS Add-ons have been installed, additional Activities/Grouping may also appear on the Toolbox panel.

**Defining a Workflow using a Sequence Container**

Workflow jobs that contain more than one Activity must be nested within a parent activity or container such as a Sequence or Flowchart. A Sequence Container is an object that can help create a more intricate workflow. Workflow Sequences are powerful as they can contain any activity including Flowcharts and decision structures.

**NOTE:** Workflow Sequence activities are not related to the Sequence Execution Method available in JAMS.

1. To include multiple Activities in a Job workflow, drag a Sequence Container to the Designer panel to configure the Workflow framework.
2. Drop other Activities from the Toolbar panel into that object. This action creates a parent-child relationship.
3. Drag and drop the next Activity to create a child activity within the parent Sequence.

More complex Jobs, such as those requiring branching, can be designed using nested Sequences.

**Flowchart Activity**

A Flowchart object executes Activities one after another, but it also provides controls to loop back to previous steps or can skip a step entirely. A Sequence activity can only move forward.

**Using the Properties Panel**
Select any Activity from the Designer panel to view its specific properties. Use the Properties panel to populate or modify individual property fields for that object.

Please review any task messages appearing on the Activity object. To view the message content hover the mouse over the object in the Designer panel.

**Note:** All errors must be resolved before a JAMS Job can be successfully submitted.

Select the **Save and Close** button to complete the JAMS Workflow job definition process.

**Note:** Some JAMS Activities, such as AskQuestion and ManageActivity, allow for runtime interaction. For more information on Workflow interaction, select the link: [Runtime Workflow Interaction](#).
Workflow Activities

Workflows are created and edited using the Workflow Designer which includes many building blocks (Activities) for submitting Jobs, updating JAMS Variables, and many more JAMS specific tasks.

Below is a listing and a brief description of Activities available in JAMS Workflow Designer. Each Activity is organized by grouping, with some containing links to their corresponding class and property descriptions in the Developers Guide. Note that some tasks listed here require JAMS Integration Packs.

**JAMS Grouping**

- **AskQuestion** ('AskQuestion Class' in the on-line documentation)
  
  Used to halt an executing workflow until user input is provided. This Activity sends the question to the JAMS Scheduler and displays the question in the Monitor View. Users with Manage access for the Job can respond to the question using the Monitor View detail window.

- **GetVariable** ('GetVariable<TResult> Class' in the on-line documentation)
  
  Gets the value of a JAMS variable.

- **ManageActivity** ('ManagedActivity Class' in the on-line documentation)
  
  Provides an execution scope in which Activities can be canceled or retried.

- **Repeat** ('Repeat Class' in the on-line documentation)
  
  Hosts one Activity that is repeated at an interval.

- **SetJAMSContext** ('SetJAMSContext Class' in the on-line documentation)
  
  This is necessary only if you are using JAMS Activities in a non-JAMS environment. When a JAMS workflow is executed, the JAMS Context is automatically established. If using Activities outside of JAMS, you can either set the JAMS Server in each Activity or add a SetJAMSContext activity to the workflow allowing all subsequent JAMS Activities to utilize the JAMS Server.

- **SetVariable** ('SetVariable Class' in the on-line documentation)
  
  Sets the value of a JAMS variable.

- **TimeLimit** ('TimeLimit Class' in the on-line documentation)
  
  Executes it’s child Activity until completed or if a set time limit is reached.

- **SetToday** ('SetToday Class' in the on-line documentation)
  
  Sets the value of “today” in the workflow.

- **Comment** ('Comment Class' in the on-line documentation)
  
  Allows its child Activity to be enabled or disabled.

- **SubmitEntry** ('SubmitEntry Class' in the on-line documentation)
  
  Submits another JAMS Job. The Activity can wait for the completion of the submitted Job and can take different actions depending on the success or failure of the entry.

**SQL Grouping**

- **OdbcScript** ('OdbcScript Class' in the on-line documentation)
  
  Runs SQL commands through an ODBC connection.
OleDbScript ('OleDbScript Class' in the on-line documentation) Runs SQL commands through an OLEDB connection.

SQLScript ('SQLScript Class' in the on-line documentation) Runs SQL commands

SQLStoredProc ('SQLStoredProc Class' in the on-line documentation) Executes a SQL stored procedure.

SQLQueryScalar<T> Runs a query on a SQL Server database and outputs the first results into a Workflow variable of the specific Type.

Oracle Grouping
OracleQueryScalar<T> Runs a query on a Oracle database and outputs the first results into a Workflow variable of the specific Type.

SQL*Plus Script Runs a SQL*Plus script on an Oracle database.

Execution Grouping
CMDScript ('CMDScript Class' in the on-line documentation) Runs a command script.

ExecuteCMD ('ExecuteCMD Class' in the on-line documentation) Executes a single command using CMD.EXE.

ExecuteProcess ('ExecuteProcess Class' in the on-line documentation) Executes any process or executable.

Cancelable ('Cancelable Class' in the on-line documentation) Includes an option to send a cancellation request to its body Activity.

PowerShell Grouping
PSWrapper ('PSWrapper Class' in the on-line documentation) Wrapper around a PowerShell function or command.

PSRunspace ('PSRunspace Class' in the on-line documentation) Defines the scope of a PowerShell runspace.
Class' in the on-line documentation)

PSScript ("PSScript Class' in the on-line documentation)

Runs a PowerShell script.

Communication Grouping

SendEmail ("SendEmail Class' in the on-line documentation)

Sends email using SMTP.

Files Grouping

ForEachFile ("ForEachFile Class' in the on-line documentation)

Finds the designated files within a directory matching a wildcard pattern.

UnzipFiles ("UnzipFiles Class' in the on-line documentation)


WaitforFile ("WaitForFile Class' in the on-line documentation)

Waits for events on a specified file.

ZipFiles ("ZipFiles Class' in the on-line documentation)


Files Transfer Grouping

FtpDownload ("FtpDownload Class' in the on-line documentation)

Downloads a file using FTP or SecureFTP (SFTP)

FtpUpload ("FtpUpload Class' in the on-line documentation)

Uploads a file using FTP or SFTP

SftpDownload ("SftpDownload Class' in the on-line documentation)

Downloads a file using Secure FTP (SFTP)

SftpUpload ("SftpUpload Class' in the on-line documentation)

Uploads a file using Secure FTP (SFTP)

ScpDownload ("ScpDownload Class' in the on-line documentation)

Downloads a file using Secure Copy Protocol (SCP)
ScpUpload
('ScpUpload Class' in the on-line
documentation)

Uploads a file using Secure Copy Protocol (SCP)

S3Download
('S3Download Class' in the on-line
documentation)

Downloads a file using Amazon's Simple Storage Service (S3)

S3Upload
('S3Upload Class' in
the on-line
documentation)

Uploads a file using Amazon's Simple Storage Service (S3)

File Transfer Session Grouping

SFTPSession ('SFTPSession Class' in the
on-line documentation)

Hosts an SFTP connection for other FileTransferSession Activities.

FTPSession ('FTPSession Class' in the
on-line documentation)

Hosts an FTP connection for other FileTransferSession Activities.

ScpSession ('SCPSession Class' in the
on-line documentation)

Hosts an SCP connection for other FileTransferSession Activities.

S3Session ('S3Session Class' in the on-line
documentation)

Hosts an Amazon S3 connection for other FileTransferSession Activities.

FileTransferSessionUpload
('FileTransferSessionUpload Class' in the
on-line documentation)

Uploads a file within the current file transfer session.

FileTransferSessionDownload
('FileTransferSessionDownload Class' in the
on-line documentation)

Downloads a file within the current file transfer session

FileTransferSessionDelete
('FileTransferSessionDelete Class' in the
on-line documentation)

Deletes a file within the current transfer session.

FileTransferSessionRename
('FileTransferSessionRename Class' in the
on-line documentation)

Renames a file within the current transfer session.

FileTransferSessionChangeDirectory
('FileTransferSessionCreateDirectory
Class' in the on-line documentation)

Changes the working directory in the current file transfer session.

FileTransferSessionGetCurrentDirectory
('FileTransferSessionGetCurrentDirectory
Class' in the on-line documentation)

Gets the working directory in the current file transfer session.

FileTransferSessionGetFileList
('FileTransferSessionGetFileList Class' in the
on-line documentation)

Gets the listing of files in a directory for the current file
transfer session.

FileTransferSessionGetFileCreationDate
('FileTransferSessionGetFileCreationDate
Class' in the on-line documentation)

Gets the files creation date for the current file transfer session.

FileTransferSessionGetFileLength
('FileTransferSessionGetFileLength Class'
in the on-line documentation)

Gets the files size for the current file transfer session.
### Mail Server Session Grouping

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IMAPSession</strong></td>
<td>Creates an IMAP Session that host a mail server connection.</td>
</tr>
<tr>
<td><strong>POPSession</strong></td>
<td>Creates an POP Session that host a mail server connection.</td>
</tr>
<tr>
<td><strong>DeleteMailMessage</strong></td>
<td>Deletes a mail message on a mail server with a corresponding header.</td>
</tr>
<tr>
<td><strong>GetMailHeaderList</strong></td>
<td>Gets a collection of mail headers on a mail server that matches the specified mask. The returned collection is a list of JAMSMailMessageInfo objects.</td>
</tr>
<tr>
<td><strong>GetMailMessage</strong></td>
<td>Gets a JAMSMailMessage object for the specified header on a mail server.</td>
</tr>
<tr>
<td><strong>GetMailServerFolder</strong></td>
<td>Gets the current folder on a mail server.</td>
</tr>
<tr>
<td><strong>ProcessEmails</strong></td>
<td>Looks for one or more emails matching a specified mask and invokes the body Activity for all matching email.</td>
</tr>
<tr>
<td><strong>SaveMessageAttachments</strong></td>
<td>Saves attachments on a mail message to the file system.</td>
</tr>
<tr>
<td><strong>SetMailServerFolder</strong></td>
<td>Sets the current folder on the mail server.</td>
</tr>
<tr>
<td><strong>ForEachMailMessage</strong></td>
<td>Looks for one or email messages matching a mail specification and invokes the body Activity once for each matching message.</td>
</tr>
<tr>
<td><strong>ForEachAttachment</strong></td>
<td>Looks for one or more mail messages matching a mail specification, then downloads the attachments to the specified working directory and invokes the body Activity for each matching file.</td>
</tr>
</tbody>
</table>

### Coordinators Grouping

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ClearEvent</strong></td>
<td>Sets the specified event to false.</td>
</tr>
<tr>
<td><strong>SetEvent</strong></td>
<td>Sets the event time.</td>
</tr>
</tbody>
</table>
WaitForEvent ('WaitForEvent Class' in the on-line documentation)

Waits for the specified event to be set by a SetEvent Activity. If the event has already been set, the WaitForEvent Activity completes immediately.

Control Flow Grouping

DoWhile
Executes the Activity contained in its body at least once until a specified condition evaluates to false.

ForEach<T>
Executes an Activity action once for each value provided in the values collection.

If
Provides a conditional if-then-else condition.

Parallel
Is a container object that executes multiple child Activities at the same time. This class cannot be inherited.

ParallelForEach<T>
Enumerates a collection and executes an Activity for each element of the collection in parallel.

Pick
Contains a collection of PickBranch Activities where each PickBranch is a pairing between a Trigger and an Action Activity. At execution time the triggers for all branches are executed in parallel. When one trigger completes its corresponding action is executed and all other triggers are canceled.

PickBranch
Each PickBranch is contained within a branch of the Pick Activity and can be executed based on an incoming event that serves as a trigger.

Sequence
Is a container object that executes Activities one after another. Sequences can incorporate more than one child Activity. Sequences can only execute forward, not backward.

Switch<T>
Evaluates a specified expression and executes using a collection of Activities whose associated key matches the value obtained from the evaluation.

Flowchart Grouping

Flowchart
Is a container object that executes multiple child activities one after another. Similar to a Sequence, a Flowchart is more flexible allowing control to return to an earlier step. Sequences can only execute forward, not backward.

FlowDecision
Is a conditional node that provides a branch for the flow of control into one of two alternatives based on whether a specified condition is satisfied. If the flow requires more than two branches, use FlowSwitch instead.

FlowSwitch<T>
Is a conditional node that provides branching for the flow of control based on matching criterion when more than two alternative branches are required. If the flow branching requires only two paths, use the FlowDecision activity instead.

Messaging Grouping

CorrelationScope
Provides implicit CorrelationHandle management for child messaging Activities.

InitializeCorrelation
Initializes correlation without sending or receiving a message.

Receive
Receives a message.
### ReceiveAndSendReply
Receives a message as part of a request/reply message exchange pattern.

### Send
Sends a message to a service.

### SendAndReceiveReply
Sends a message as part of a request/reply message exchange pattern.

### TransactedReceiveScope
Scopes the lifetime of a transaction which is initiated by a received message. The transaction may be flowed into the workflow on the initiating message or created by the dispatcher when the message is received.

## RunTime Grouping

### Persist
Saves a workflow to disk, if possible. This Activity cannot be executed in a non-persistence zone, for example, within a TransactionScope Activity.

### TerminateWorkflow
Terminates the execution of a workflow.

## Primitives Grouping

### Assign
Assigns a value to a variable at the current scope.

### Delay
Puts one path of execution into an idle state, possibly allowing the workflow to be unloaded.

### InvokeMethod
Executes a public method of a CLR object.

### WriteLine
Writes a specified string to the console or a specified TextWriter object.

## Transactions Grouping

### CancellationScope
Specifies an Activity for execution and cancellation logic for that Activity.

### CompensableActivity
Supports compensation of its child activities.

### Compensate
Used to explicitly invoke the compensation handler of a CompensableActivity.

### Confirm
Invokes the confirmation handler of a CompensableActivity.

### TransactionScope
Demarcates a transaction boundary.

## Collection Grouping

### AddToCollection<T>
Adds an item to a specified collection.

### ClearCollection<T>
Clears a specified collection of all items.

### ExistsInCollection<T>
Determines whether a specified item exists in a particular collection.

### RemoveFromCollection<T>
Removes an item from a specified collection.

## Error Handling Grouping

### Rethrow
Throws a previously thrown exception. This Activity can only be used in a Catch handler in the TryCatch Activity.

### Throw
Throws an exception

### TryCatch
Contains Activities to be executed by the workflow runtime in an exception handling block.

## SYM (Symitar) Grouping

### DataFileToPC ('DataFileToPC Class' in the
FTP download from the "DATAFILES" folder on the Symitar server.

©2020 HelpSystems, LLC. All Rights Reserved.
<table>
<thead>
<tr>
<th>Class Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeleteLetterFile</td>
<td>Removes a letter file from the “LETTERSPECS” folder on the Symitar server.</td>
</tr>
<tr>
<td>EmailReports</td>
<td>Attaches one or more reports to an email message and sends it to the designated recipient(s).</td>
</tr>
<tr>
<td>FileToPC</td>
<td>Generates an FTP download from any Symitar directory to a local PC.</td>
</tr>
<tr>
<td>LetterFileToPC</td>
<td>FTP download from the “LETTERSPECS” folder on the Symitar server.</td>
</tr>
<tr>
<td>PCToDataFile</td>
<td>FTP upload to the “DATAFILES” folder on the Symitar server.</td>
</tr>
<tr>
<td>PCToEditFile</td>
<td>Initiates an FTP from a PC to the Symitar edit file.</td>
</tr>
<tr>
<td>PCToFile</td>
<td>Creates a generic upload to any directory on the Symitar system.</td>
</tr>
<tr>
<td>PCToLetterFile</td>
<td>FTP upload to the “LETTERSPECS” folder on the Symitar server.</td>
</tr>
<tr>
<td>ReportsFileToPC</td>
<td>FTP download from the “REPORT” folder on the Symitar server.</td>
</tr>
<tr>
<td>RunJobFile</td>
<td>Runs a job file in the specified SYM (e.g. Sym222).</td>
</tr>
<tr>
<td>SymConfig</td>
<td>The parent activity that allows users to set the configuration settings (e.g., Server Name, SymUserId, Symitar logon credentials) for all Symitar activities contained within it. Each child activity can override the settings in the SymConfig.</td>
</tr>
</tbody>
</table>
Migrating JAMS Objects

Exporting and Importing JAMS Objects

Many Definitions in JAMS may be exported or imported using JAMS PowerShell cmdlets, or from within the JAMS Client. These Definitions include:

- Agent Definitions
- Execution Methods
- Folder Definitions
- Job Definitions (Including Samples)
- Menu Definitions
- Named Time Definitions
- Queue Definitions
- Resource Definitions

To Export any existing Definition(s) from JAMS, simply select one or more objects from the relevant view in the GUI, then use the Export button on the Control Bar to open the Save as XML dialog. Give the XML Export a Name, set a location, and save the export.

Using the Copy To Function

The Copy To function allows users to rename and copy selected JAMS objects to another local or remote location. This feature is most often used when mirroring the Jobs within a folder on the same server, or to promote Jobs from one server to another. E.g. From Development to QA.

Copy one or more Objects to another location

1. Open the Definitions Shortcut
2. Select the Job(s) or Variable(s) to Copy.
   
   **NOTE:** Users may select multiple objects by using CTRL+A to select all, CTRL+Left-Click to select multiple non-contiguous objects, or SHIFT+Left-Click to select contiguous objects.
3. Right-Click the selected object(s), then select Copy To... from the dropdown menu.
4. The Copy To dialog will open. Define the Destination Server, Folder, and object Name.
   
   **NOTE:** When copying multiple Jobs, the copies will be created using the Source Job Names.
5. When Finished, click OK to start the copy function. Depending on the size of the copy, a progress dialog may appear.

**Copy To Properties**

<table>
<thead>
<tr>
<th>Copy To Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job/Variable Name</td>
<td>This property displays the original name of the selected Job or Variable to be copied.</td>
</tr>
<tr>
<td>Folder Name</td>
<td>To choose a destination folder location click the file browser button to navigate to a folder within the chosen server.</td>
</tr>
<tr>
<td>Job/Variable Name</td>
<td>JAMS uses the original name of the selected object for the copy by default unless a new name is entered here.</td>
</tr>
<tr>
<td>Server</td>
<td>Use the dropdown menu to select an available destination server for the object.</td>
</tr>
</tbody>
</table>
File Transfer Features

JAMS provides built-in support for all standard file transfer protocols including:

- **FTP**: Classic (unencrypted) FTP support that conforms to RFC 959 and 1123.
- **FTPS**: FTP over TLS/SSL. Provides encrypted password and data encryption. Defined by RFC 959, 1123, 4217, and 2228.
- **SFTP**: SSH File Transfer. Includes encrypted password and data encryption. Considered a more secure protocol over FTPS.
- **SCP**: Secure CP. A variant of BSD rcp that transfers files over an SSH session.

The File Transfer Execution Method

The File Transfer Execution Method works with all the above protocols using a “fill in the blanks” source. In addition, this method supports automatic retries and wildcards, and includes complete integration with JAMS user security.

Since file transfers execute as standard JAMS Jobs, users have the following capabilities:

- Automatic notification for failed or stalled transfers or runaway processes.
- Dependency support before and after the file transfer.
- File event triggering for file transfers.
- Direct integration into a JAMS Sequence for process sequencing and flow control.

Defining a New Job using the File Transfer Execution Method

1. To create a new Job, open the Definitions view from the Shortcut bar, then select the Job Definitions tab.
2. Click the Add button in the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, select the Folder where the Job should be created.
4. Define a Name for the new Job.
5. If desired, give the Job a Description.
6. Select the File Transfer Execution Method for the Job from the drop-down list.
7. By default, the full Job Definition dialog will open when the Job is initially saved. Click OK.
8. The Job Definition dialog will open.
9. Open the Source tab to define the File Transfer Options. First, select a Transfer Type.
10. The File Transfer fill-in-the-blank options will reflect the selected File Transfer Type. Define additional information in the Files, Security, Zip, Retry, and Options sub-tabs as desired.
11. Define additional Elements, Parameters, Security, Properties, and Documentation on the Job as desired.
When the Job has been configured as desired, click **Save and Close**.

**File Transfer Source Options**

The table below contains some of the options available on the Source sub-tabs for File Transfer Jobs in JAMS.

<table>
<thead>
<tr>
<th><strong>File tab</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Remote Server Name</td>
<td>Enter the DNS name or IP address of the remote server.</td>
</tr>
<tr>
<td>Remote Directory</td>
<td>Provide the default working directory on the selected remote server.</td>
</tr>
<tr>
<td>Remote File Specification</td>
<td>Insert the remote file specification. When left black, the filename and extension of the local file is substituted. This field also allows wildcards.</td>
</tr>
<tr>
<td>Local File Specification</td>
<td>Enter or navigate to the file specification for the local file. This field also allows wildcards.</td>
</tr>
<tr>
<td>Recursive</td>
<td>When checked, file transfers are performed recursively.</td>
</tr>
<tr>
<td>Delete Source Files After Transfer</td>
<td>When checked, the source files will be deleted once transfer is complete.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Security</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential</td>
<td>Select a predefined JAMS User Credential object to access the remote server.</td>
</tr>
<tr>
<td>Username and Password</td>
<td>Enter a user name and password to access the remote server.</td>
</tr>
</tbody>
</table>

©2020 HelpSystems, LLC. All Rights Reserved.
Zip Source Files
When checked, the source files will be zipped before upload.

Unzip After Download
When checked, JAMS will unzip the source files after download.

Delete Zip File after Extraction/Upload
When checked, JAMS will delete the source files after transfer.

Zip File Directory
Specifies the location of the zipped source files.

Zip File Name
Specifies the name of the zipped source files.

Credential and Password for Archiver
Specifies the credentials to use when zipping or unzipping the files.

**Retry**

**Description**
Include the maximum number of retry attempts if the transfer fails.

Maximum Retries
This property defines the delay (in minutes) between retries.

Retry Delay

**Options**

**Description**

Port
Enter the TCP/IP port the selected server is using.

Timeout
The maximum number of seconds to wait before timing out during a file transfer.

Passive
When checked, passive mode is enabled. In this mode, TCP/IP connections are always made from the client to the server, which makes it easier to support NAT and firewalls.

Binary
When enabled, a binary transfer is performed.

Keep Alive
When checked, JAMS continues to ping in order to maintain the connection.

Keep Alive Interval
This property defines the time between JAMS ping messaging.

Server Type
The server type used for this file transfer.

Minimum Key Size
The minimum key size for SSH negotiations (use 0 for default).

Require Secure Ciphers
Determines if a secure cipher is required.

Accept All Certificates
When enabled, all certificates are accepted for the Job.

Client Certificate
The client certificate to be used with the file transfer, defined as the location of the file that contains client certificate.

Use Implicit Security
When set to true, implicit SSL is used.

Allow SSL 3.0
Enable to allow SSL 3.0

Allow TLS 1.0
Enable to allow Transport Layer Security (TLS) 1.0

Allow TLS 1.1
Enable to allow Transport Layer Security (TLS) 1.1

Allow TLS 1.2
Enable to allow Transport Layer Security (TLS) 1.2

**Modifying an Existing File Transfer Job**

1. To make changes to any existing Job, open its *Job Definition* dialog from the Definitions View.
   To open the *Job Definition* dialog, users may **Double-Click** the Job, **Select the Job** and then click the **Properties** button in the **Control Bar**, or **Right-click** and select **Properties** from the **Job Options** menu.
2. The **Job Definition** dialog appears.
4. When the Job has been reconfigured as desired, click **Save and Close**.
Variables
Working with Variables

A Variable contains a single slice of data that can be globally defined for all JAMS Jobs. Variables are a powerful tool that can make batch processing easier and more effective. Once a Variable is defined it can be used in a variety of ways, including:

- **As a default value for a Job parameter.** When working with multiple Jobs that require the same parameter value, creating a default value within a Variable can greatly increase efficiencies, allowing you to modify a value in one place to make changes to all Jobs that use the Variable.

- **Directly within the Job Source.** Variables can be specified in the Job Source using `<<VariableName>>`. When the Job runs, it replaces `<<VariableName>>` with the value of the Variable.

- **As an undefined parameter in a parsed Job.** Parsed Jobs can use the value of a Variable. In fact, the Variable does not need to be associated with the Job, other than to include the Variable within the command file (source) where you need the Variable's value to appear.

- **In Job Dependencies.** One type of Job and Sequence Dependency relies on the value of a Variable. You can quickly define Dependency Elements that require a specific Variable condition to execute a Job.

- **In Trigger Elements.** Trigger Elements support both event and calendar scheduling simultaneously. You can design a Trigger Element to use the value of a Variable to fire Trigger actions.

- **In your application programs and/or DCL command procedures.** You can retrieve or set the value of Variables using command procedures. This can be very useful for Jobs that do not use a parsed execution method, but still need to use the value of the Variable for their process. Whether parsed or not, Jobs that have access to the JAMS command-line interface can set the value of Variables.

### Defining a New Variable

1. To create a new Variable, open the Definitions view from the Shortcut bar, then select the Variable Definitions tab.
2. Click the Add button in the Control Bar to open the Add a New Variable Definition dialog.
3. In the dialog, select the Folder where the Variable should be created.
4. Define a unique Name for the new Variable. Note that this variable name will be used when referencing the Variable elsewhere in JAMS.
5. If desired, give the Variable a Description.
6. Select a Data Type for the Variable.
7. Set a Default Value for the Variable.
8. By default, the full Variable Definition dialog will open when the Variable is initially saved. Click OK.
9. The Variable Definition dialog will open.
10. Define the Variable’s basic information, Value, and add Security to the Variable as desired.

Variable Data Types

- Text
- Integer
- Date
- Time
- DateTime (a date and time of day, e.g., 12/31/2015 2:00:00 PM)
- Float (a floating point number, e.g., 3.0, -122.5)
- Unknown (any data type)
- Boolean

Variable Definition Dialog

You can modify an existing Variable by opening its Variable Definition dialog. Double-click an item in the Variable Definitions View to open the dialog.

<table>
<thead>
<tr>
<th>Variable Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Name</td>
<td>Includes the unique name for the Variable.</td>
</tr>
<tr>
<td>Description</td>
<td>This optional property can provide a more complete explanation of the Variable.</td>
</tr>
<tr>
<td>Last Changed</td>
<td>Includes the username who last modified this Variable and the date and time when this change occurred.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Type</td>
<td>Each Variable must be assigned a Data Type as described in the previous section. Use the dropdown list to make a selection.</td>
</tr>
<tr>
<td>Value</td>
<td>Enter the value of the Variable that corresponds with the selected Data Type.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Security Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Control Entries</td>
<td>The ACE(s) set on the Variable. Each ACE is configured with a set of rights, specified below.</td>
</tr>
<tr>
<td>Security Rights</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>permits modification of the Variable definition provided the user also has Change access to Variable Definitions.</td>
</tr>
<tr>
<td>Control</td>
<td>allows modification of the Variable’s Access Control List.</td>
</tr>
<tr>
<td>Delete</td>
<td>allows deletion of the Variable’s definition, provided the user also has Delete access to Variable Definitions.</td>
</tr>
<tr>
<td>Inquire</td>
<td>allows inquiry into the Variable definition, provided the user also has Inquire access to Variable Definitions.</td>
</tr>
</tbody>
</table>
Elements
Documentation Elements

Documentation Elements are user-defined sections of content included in a Job’s notification, or in the Documentation tab of a Job or Folder.

Adding a Documentation Element to a Job or Folder

1. Open the Definition dialog for the Job or Folder where the Documentation Element will be defined.
   NOTE: Users may right-click either a Folder or Job and select Properties from the drop-down list to open the Definition dialog.
2. Select the Elements tab, then click the Add… button. The Add Element dialog will appear.
3. Expand the Documentation category to reveal the Documentation Elements.
4. Select the relevant Element and click Next.
5. Define the available properties for the given Documentation Element, then click Finish.
6. Save and Close the Definition dialog.

Modify an existing Documentation Element

1. Open the Definition dialog for the desired Job or Folder where the Documentation Element resides.
   NOTE: Users may right-click either a Folder or Job and select Properties from the drop-down list to open the Definition dialog.
2. Select the Elements tab, then double-click the desired Documentation Element to open the Documentation Properties dialog.
3. Modify the properties as desired, then click Save and Close on the Documentation Properties dialog.
4. Save and Close the Definition dialog.

Documentation Element Properties

Each documentation element has a list of associated properties used to define how the documentation is handled. Reference the tables below for a description of the properties available on each documentation element.

### Documentation

<table>
<thead>
<tr>
<th>Documentation Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Documentation Type</td>
<td>Is this for Documentation, Notifications, All, or None?</td>
</tr>
<tr>
<td>Section</td>
<td>When documentation is assembled from Jobs and Folders, the parts of each section are merged in order.</td>
</tr>
<tr>
<td>Content</td>
<td>The actual Documentation content. Documentation content supports Markdown language.</td>
</tr>
</tbody>
</table>

### Link

<table>
<thead>
<tr>
<th>Documentation Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Documentation Type</td>
<td>Is this for Documentation, Notifications, All, or None?</td>
</tr>
<tr>
<td>Section</td>
<td>When documentation is assembled from Jobs and Folders, the parts of each section are merged in order.</td>
</tr>
<tr>
<td>Link Target</td>
<td>The actual URL of the hyperlink.</td>
</tr>
<tr>
<td>Link Text</td>
<td>The text to display for the hyperlink</td>
</tr>
</tbody>
</table>
Link Template

Template used for displaying the link. By default, {0} will be replaced with the defined link target and {1} will be replaced by the defined Link Text.
EventHandler Elements

EventHandler Elements are used to respond to executing or completed JAMS Jobs. Event Handlers may be configured to respond to Jobs running too long, completing too quickly, or to send notifications when Jobs fail or complete with warning severity. Multiple event handlers may be used for a single Job. Event Handler Elements added to a Folder will be inherited by that Folder’s child objects.

Adding an Event Handler Element to a Job or Folder

1. Open the Definition dialog for the Job or Folder where the Event Handler Element will be defined.  
   NOTE: Users may right-click either a Folder or Job and select Properties from the drop-down list to open the Definition dialog.
2. Select the Elements tab, then click the Add... button. The Add Element dialog will appear.
3. Expand the Event Handler category to reveal the Event Handler Elements.
4. Select the relevant Element and click Next.
5. Define the available properties for the given Event Handler Element, then click Finish.
6. Save and Close the Definition dialog.

Modify an existing Event Handler Element

1. Open the Definition dialog for the desired Job or Folder where the Event Handler Element resides.  
   NOTE: Users may right-click either a Folder or Job and select Properties from the drop-down list to open the Definition dialog.
2. Select the Elements tab, then double-click the desired Event Handler Element to open the Event Handler Properties dialog.
3. Modify the properties as desired, then click Save and Close on the Event Handler Properties dialog.
4. Save and Close the Definition dialog.

Event Handler Element Properties

Each Event Handler element has a list of associated properties used to define how the event will be handled. Reference the tables below for a description of the properties available on each event handler.

Notification Job

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>This can be used to enable or disable a Job, Element, or Event Handler.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Job Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notification Job</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Events</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry Success</td>
<td>Should the event be triggered on a successful completion?</td>
</tr>
<tr>
<td>Entry Informational</td>
<td>Should the event be triggered on an informational completion?</td>
</tr>
<tr>
<td>Entry Warning</td>
<td>Should the event be triggered on a warning completion?</td>
</tr>
<tr>
<td>Entry Error</td>
<td>Should the event be triggered on an error completion?</td>
</tr>
<tr>
<td>Entry Fatal</td>
<td>Should the event be triggered on a fatal completion?</td>
</tr>
<tr>
<td>Entry Started</td>
<td>Defines whether the event should be triggered when the Job has started executing.</td>
</tr>
<tr>
<td>Normal Event</td>
<td>Defines whether the event should be triggered when a normal event occurs.</td>
</tr>
</tbody>
</table>
Low Event
Defines whether the event should be triggered when a low event occurs.

Moderate Event
Defines whether the event should be triggered when a moderate event occurs.

High Event
Defines whether the event should be triggered when a high event occurs.

Urgent Event
Defines whether the event should be triggered when an urgent event occurs.

Critical Event
Defines whether the event should be triggered when a critical event occurs.

**Paladin Event**

**Paladin**

**Description**
A comma separated list of Paladin Monitor IDs that should receive the event.

**Monitor ID**
The severity of this event

**Severity**
A summary of the event.

**Summary**
A detailed description of the event

**Detail**
Should JAMS ask for a response from Paladin and take action based on the response?

**Ask for a response**
Should the job’s log file be uploaded to Paladin?

**Upload the log file**

**Events**

**Description**
Should the event be triggered on a successful completion?

**Entry Success**
Should the event be triggered on an informational completion?

**Entry Informational**
Should the event be triggered on a warning completion?

**Entry Warning**
Should the event be triggered on an error completion?

**Entry Error**
Should the event be triggered on a fatal completion?

**Entry Fatal**
Defines whether the event should be triggered when the Job has started executing.

**Entry Started**
 Defines whether the event should be triggered when a normal event occurs.

**Normal Event**
 Defines whether the event should be triggered when a low event occurs.

**Low Event**
 Defines whether the event should be triggered when a moderate event occurs.

**Moderate Event**
 Defines whether the event should be triggered when a high event occurs.

**High Event**
 Defines whether the event should be triggered when an urgent event occurs.

**Urgent Event**
 Defines whether the event should be triggered when a critical event occurs.

**Critical Event**

**Status**

**Description**
This can be used to enable or disable a Job, Trigger, or Event Handler

**Enabled**

**Recovery Job**

**General**

**Description**
The JAMS Job that should run when the current Job does not exceed the configured Recovery Severity.

**Recovery Job**
The maximum completion severity required to trigger the Recovery Job.

**Recovery Severity**

**Repeat**

**Status**

**Description**
| **Enabled** | This can be used to enable or disable a Job or Element. |
| **Started** | The date and time when this series starts repeating. |
| **Will Stop Repeating** | The date and time this series will stop repeating. |

**Repeat**

| **Description** | **Description** |
| **Delay** | What is the delay between repetitions? |
| **End Time** | What time should the repetitions stop? |
| **Base Time** | When this Job repeats, which time should be used to calculate the next repetition? Options include End Time, Scheduled Time, and Start Time. |
| **Repeat on Error** | Should the Job continue to Repeat even after it fails? |

**Resubmit**

| **Status** | **Description** |
| **Enabled** | This can be used to enable or disable a Job or Element. |
| **Started** | The date and time when this series starts resubmitting. |
| **Will Stop Repeating** | The date and time this series will stop resubmitting. |

**Resubmit**

| **Description** | **Description** |
| **Delay** | What is the delay between resubmissions? |
| **End Time** | What time should the resubmissions stop? |
| **Base Time** | When this Job resubmits, which time should be used to calculate the next resubmission? Options include End Time, Scheduled Time, and Start Time. |
| **Repeat on Error** | Should the Job continue to Resubmit even after it fails? |

**Runaway**

| **Runaway** | **Description** |
| **Runaway Elapsed** | Determines, in Delta Time, how long the Job can run before it is considered a runaway Job. |
| **Runaway Elapsed Percent** | Determines how long the Job can run before it is considered a runaway Job, set as a percentage of the Job’s average elapsed time. |
| **Runaway Action** | Determines the action taken on the Job if it becomes a runaway Job. |

**Notify**

| **Notify** | **Description** |
| **Event Class** | Select the event class that this event will generate when it occurs. |
| **Level** | This level can be used by event handlers to classify events. |
| **Message** | A message that will be passed to event handlers when this event is generated. |

**Send E-Mail**

| **Status** | **Description** |
Enabled  This can be used to enable or disable a Job, Element, or Event Handler.

**Events**  
**Description**  
Entry  Should the event be triggered on a successful completion?
Success  Should the event be triggered on an informational completion?
Entry  Should the event be triggered on a warning completion?
Warning  Should the event be triggered on an error completion?
Entry Error  Should the event be triggered on a fatal completion?
Entry Started  Defines whether the event should be triggered when the Job has started executing.
Normal Event  Defines whether the event should be triggered when a normal event occurs.
Low Event  Defines whether the event should be triggered when a low event occurs.
Moderate Event  Defines whether the event should be triggered when a moderate event occurs.
High Event  Defines whether the event should be triggered when a high event occurs.
Urgent Event  Defines whether the event should be triggered when an urgent event occurs.
Critical Event  Defines whether the event should be triggered when a critical event occurs.

**E-Mail**  
**Description**  
CC Address  A comma separated list of e-mail addresses that will be cc’d on the e-mail. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*
From Address  The from email address used when sending the e-mail from JAMS.
Message Body  The body of the e-mail message. Users may enter Markdown language in the Message Body.
Send Plain Text  When set to true, e-mail is sent using only plain text. When false, html formatting is used. Note that if Send Plain Text is true, Markdown Language in the Message Body will not format properly.
Subject  The Subject of the e-mail.
To Address  A comma separated list of e-mail addresses that the e-mail will be sent to. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*

**Short**  
**Description**  
Short Elapsed  The minimum elapsed time for successful completion of the Job. If the Job completes successfully without exceeding the short time, configured notifications and completion severity changes will execute.
Short Elapsed Percent  The minimum elapsed time for a successful completion of the Job, set as a percentage of the Job's average elapsed time. If the Job completes successfully without exceeding the short elapsed percent, configured notifications and completion severity changes will execute.
Short Severity  If the Job meets Short Job criteria, what should the Job's completion severity be changed to?

**Notify**  
**Description**  

<table>
<thead>
<tr>
<th><strong>Notify Event Class</strong></th>
<th>Select the event class that this event will generate when it occurs.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Level</strong></td>
<td>This level can be used by event handlers to classify events.</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>A message that will be passed to event handlers when this event is generated.</td>
</tr>
</tbody>
</table>

**Stalled**

**Stalled Description**

- **Stalled Time**
  - The maximum elapsed time for a Job. A Job execution that exceeds the stalled time is considered stalled, and will cause any configured notifications to be sent.

**Notify Description**

- **Notify Event Class**
  - Select the event class that this event will generate when it occurs.

- **Level**
  - This level can be used by event handlers to classify events.

- **Message**
  - A message that will be passed to event handlers when this event is generated.
Prerequisite Elements

Prerequisite Elements must be completed before a Job can execute. A Job or folder can have an unlimited number of Prerequisite Elements. JAMS Jobs on any server in a given environment can serve as a prerequisite for other Jobs in JAMS by using the Job Dependency or Remote Job Dependency Elements. Any elements set on a Folder will be inherited by that folder's child objects.

Users may also create their own user-defined Dependencies through the use of Pre-Check Jobs. User-defined Dependencies add the ability to handle any type of pre-processing needs.

Adding a Prerequisite Element to a Job or Folder

1. Open the Definition dialog for the Job or Folder where the Prerequisite Element will be defined. NOTE: Users may right-click either a Folder or Job and select Properties from the drop-down list to open the Definition dialog.
2. Select the Elements tab, then click the Add... button. The Add Element dialog will appear.
3. Expand the Prerequisite category to reveal the Prerequisite Elements.
4. Select the relevant Element and click Next.
5. Define the available properties for the given Prerequisite Element, then click Finish.
6. Save and Close the Definition dialog.

Modify an existing Prerequisite Element

1. Open the Definition dialog for the desired Job or Folder where the Prerequisite Element resides. NOTE: Users may right-click either a Folder or Job and select Properties from the drop-down list to open the Definition dialog.
2. Select the Elements tab, then double-click the desired Prerequisite Element to open the Prerequisite Properties dialog.
3. Modify the properties as desired, then click Save and Close on the Prerequisite Properties dialog.
4. Save and close the Definition dialog.

Prerequisite Element Properties

Each prerequisite element has a list of associated properties used to define how the prerequisite will be satisfied. Reference the tables below for a description of the properties available on each prerequisite.

### Date Specific Dependency

<table>
<thead>
<tr>
<th>Depends On</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depends on Job</td>
<td>The Job that this element depends on.</td>
</tr>
<tr>
<td>Wait For All</td>
<td>If there is more than one instance of the Depends On Job, should we wait for all of them to complete before satisfying the dependency?</td>
</tr>
<tr>
<td>Completion Severity</td>
<td>The Depends on Job must meet or exceed this completion severity to satisfy the dependency.</td>
</tr>
<tr>
<td>Depend on Date</td>
<td>A natural language date specification for the date that the depends-on job must have been scheduled to run.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>This can be used to enable or disable a Job, Element, or Event Handler.</td>
</tr>
</tbody>
</table>
**File Dependency**

**File Watch**

- **Credentials**
  - Description: Select a set of Credentials for this element. The credentials must have access to the file location.
- **File Name**
  - Description: The file specification to watch for. Wildcards are allowed.
- **File Presence Option**
  - Minimum Size:
    - Description: Used to specify a minimum acceptable size for the file.
  - Agent:
    - Description: Used to specify the Agent where the file watch will be performed.

**Status**

- **Enabled**
  - Description: This can be used to enable or disable a Job, Element, or Event Handler.

**Job Dependency**

**Depends On**

- **Depends on Job**
  - Description: The Job that this element depends on.
- **Wait For All**
  - Description: If there is more than one instance of the Depends On Job, should we wait for all of them to complete before satisfying the dependency?
- **Completion Severity**
  - Description: The Depends on Job must meet or exceed this completion severity to satisfy the dependency.

**Status**

- **Enabled**
  - Description: This can be used to enable or disable a Job, Element, or Event Handler.

**Since**

- **Since Job**
  - Description: The Since Job must have last completed with a severity that meets or exceeds the severity set here to satisfy the dependency.
- **Completion of the Depends On Job**
  - Description: The completion of this Job defines how far back we look for the completion of the Depends On Job.
- **Within Time**
  - Description: Users may enter a delta time to define how recently the Depends On Job must have completed in order to satisfy the dependency.

**Precheck Job**

**Prerequisite**

- **Precheck Job**
  - Description: Specifies a Job which is executed before this Job is allowed to start.
- **Precheck Interval**
  - Description: The delay between executions of the Precheck Job

**Status**

- **Enabled**
  - Description: This can be used to enable or disable a Job, Element, or Event Handler.
Remote Job Dependency

**Depends On**

<table>
<thead>
<tr>
<th>Completion Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The Depends on Job must meet or exceed this completion severity to satisfy the dependency.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Remote Job Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The name of the Remote Job</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Remote Scheduler</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The name of the Remote Scheduler where the Job resides.</td>
</tr>
</tbody>
</table>

**Status**

<table>
<thead>
<tr>
<th>Enabled</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This can be used to enable or disable a Job, Element, or Event Handler.</td>
</tr>
</tbody>
</table>

**Since**

<table>
<thead>
<tr>
<th>Since Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The Since Job must have last completed with a severity that meets or exceeds the severity set here to satisfy the dependency.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Since Job</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The completion of this Job defines how far back we look for the completion of the Depends On Job.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Within Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Users may enter a delta time to define how recently the Depends On Job must have completed in order to satisfy the dependency.</td>
</tr>
</tbody>
</table>

Resource Requirement

**General**

<table>
<thead>
<tr>
<th>Quantity Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The resource quantity required to satisfy the dependency.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The resource associated with this definition</td>
</tr>
</tbody>
</table>

Variable Dependency

**Condition**

<table>
<thead>
<tr>
<th>Compare Condition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Specifies the logical comparison that should be performed.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compare Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Specifies the value used in the condition comparison.</td>
</tr>
</tbody>
</table>

**Status**

<table>
<thead>
<tr>
<th>Enabled</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This can be used to enable or disable a Job, Element, or Event Handler.</td>
</tr>
</tbody>
</table>

Value

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A JAMS Variable that contains the default value.</td>
</tr>
</tbody>
</table>

Time Window

**General**

<table>
<thead>
<tr>
<th>Missed Window Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The action taken when the Schedule Window is missed.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Schedule From Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The start of the Job's scheduled time window.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Schedule To Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The end of the Job's scheduled time window.</td>
</tr>
</tbody>
</table>
Schedule Window  | A flexible window of time where Jobs may be scheduled to run.
--- | ---
**Notify**  | **Description**
Notify Event Class  | Select the event class that this event will generate when it occurs.
Notify Of Missed Window  | Determines whether a notification will be performed if this Job misses its schedule window.
Level  | This level can be used by event handlers to classify events.
Message  | A message that will be passed to event handlers when this event is generated.
Result Elements

Result Elements are used to create reports from JAMS Jobs or JAMS Integration Pack applications. Result Elements added to a Folder will be inherited by that Folder’s child objects.

Adding a Result Element to a Job or Folder

1. Open the Definition dialog for the Job or Folder where the Result Element will be defined.
   NOTE: Users may right-click either a Folder or Job and select Properties from the drop-down list to open the Definition dialog.
2. Select the Elements tab, then click the Add... button. The Add Element dialog will appear.
3. Expand the Result category to reveal the Result Elements.
4. Select the relevant Element and click Next.
5. Define the available properties for the given Result Element, then click Finish.
6. Save and Close the Definition dialog.

Modify an existing Result Element

1. Open the Definition dialog for the desired Job or Folder where the Result Element resides.
   NOTE: Users may right-click either a Folder or Job and select Properties from the drop-down list to open the Definition dialog.
2. Select the Elements tab, then double-click the desired Result Element to open the Result Properties dialog.
3. Modify the properties as desired, then click Save and Close on the Result Properties dialog.
4. Save and Close the Definition dialog.

Result Element Properties

Each Result element has a list of associated properties used to define how the event will be handled. Reference the tables below for a description of the properties available on each result element.

CR Job Report

<table>
<thead>
<tr>
<th>Execute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Printer Duplex</td>
<td>Specifies if page is printed on both sides. Options include Simplex (single sided), Vertical, or Horizontal.</td>
</tr>
<tr>
<td>Paper Orientation</td>
<td>The orientation of the report. Options include Portrait and Landscape.</td>
</tr>
<tr>
<td>Margin Left</td>
<td>The left margin of the report. Unit of measure determined by regional settings.</td>
</tr>
<tr>
<td>Margin Right</td>
<td>The right margin of the report. Unit of measure determined by regional settings.</td>
</tr>
<tr>
<td>Margin Top</td>
<td>The top margin of the report. Unit of measure determined by regional settings.</td>
</tr>
<tr>
<td>Margin Bottom</td>
<td>The bottom margin of the report. Unit of measure determined by regional settings.</td>
</tr>
<tr>
<td>Start Page</td>
<td>The starting page of the report.</td>
</tr>
<tr>
<td>End Page</td>
<td>The ending page of the report.</td>
</tr>
<tr>
<td>Paper Size</td>
<td>The size of the paper report is printed on.</td>
</tr>
<tr>
<td>Paper Source</td>
<td>Defines which printer tray to print from.</td>
</tr>
<tr>
<td>Printer Name</td>
<td>The name of the printer used.</td>
</tr>
<tr>
<td>Collate</td>
<td>Specifies if prints are collated.</td>
</tr>
</tbody>
</table>
### Email Report

<table>
<thead>
<tr>
<th><strong>E-Mail</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>CC Address</td>
<td>A comma separated list of e-mail addresses that will be cc'd on the e-mail. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*.</td>
</tr>
<tr>
<td>From Address</td>
<td>The from email address used when sending the e-mail from JAMS.</td>
</tr>
<tr>
<td>Message Body</td>
<td>The body of the e-mail message. Users may enter Markdown language in the Message Body.</td>
</tr>
<tr>
<td>Send Plain Text</td>
<td>When set to true, e-mail is sent using only plain text. When false, html formatting is used. Note that if Send Plain Text is true, Markdown Language in the Message Body will not format properly.</td>
</tr>
<tr>
<td>Subject</td>
<td>The Subject of the e-mail.</td>
</tr>
<tr>
<td>To Address</td>
<td>A comma separated list of e-mail addresses that the e-mail will be sent to. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*.</td>
</tr>
</tbody>
</table>

### JDE Report

<table>
<thead>
<tr>
<th><strong>Execute</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Printer Name</td>
<td>The name of the printer used.</td>
</tr>
<tr>
<td>Copies</td>
<td>The number of copies to be printed.</td>
</tr>
<tr>
<td>Conversion Filter</td>
<td>If a conversion filter should be applied to this report before it is sent to the printer, enter the conversion filter's name here.</td>
</tr>
</tbody>
</table>

### MS Dynamics Email Report

<table>
<thead>
<tr>
<th><strong>Execute</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Page</td>
<td>The starting page of the report.</td>
</tr>
<tr>
<td>End Page</td>
<td>The ending page of the report.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>E-Mail</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>CC Address</td>
<td>A comma separated list of e-mail addresses that will be cc'd on the e-mail. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*.</td>
</tr>
<tr>
<td>Subject</td>
<td>The Subject of the e-mail.</td>
</tr>
<tr>
<td>To Address</td>
<td>A comma separated list of e-mail addresses that the e-mail will be sent to. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*.</td>
</tr>
</tbody>
</table>
### MS Dynamics File Report

<table>
<thead>
<tr>
<th>Execute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Page</td>
<td>The starting page of the report.</td>
</tr>
<tr>
<td>End Page</td>
<td>The ending page of the report.</td>
</tr>
<tr>
<td>File Format</td>
<td>Specifies the file extension of the file.</td>
</tr>
</tbody>
</table>

### File Watch

<table>
<thead>
<tr>
<th>File Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The file specification to watch for. Wildcards are allowed.</td>
</tr>
</tbody>
</table>

### MS Dynamics Print Report

<table>
<thead>
<tr>
<th>Execute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Page</td>
<td>The starting page of the report.</td>
</tr>
<tr>
<td>End Page</td>
<td>The ending page of the report.</td>
</tr>
<tr>
<td>Printer Name</td>
<td>The name of the printer used.</td>
</tr>
<tr>
<td>Copies</td>
<td>The number of copies to be printed.</td>
</tr>
</tbody>
</table>

### SSRS Print Report

<table>
<thead>
<tr>
<th>Execute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Page</td>
<td>The starting page of the report.</td>
</tr>
<tr>
<td>End Page</td>
<td>The ending page of the report.</td>
</tr>
<tr>
<td>Printer Name</td>
<td>The name of the printer used.</td>
</tr>
<tr>
<td>Page Width</td>
<td>The width of the printed page.</td>
</tr>
<tr>
<td>Page Height</td>
<td>The height of the printed page.</td>
</tr>
<tr>
<td>Print DPI X</td>
<td>Determines the Horizontal resolution of the printout in Dots Per Inch.</td>
</tr>
<tr>
<td>Print DPI Y</td>
<td>Determines the Vertical resolution of the printout in Dots Per Inch.</td>
</tr>
</tbody>
</table>
Trigger Elements

Trigger Elements are used to define schedules in JAMS. They may be event-based or defined using a date schedule. Trigger Elements include the following:

- File Trigger
- Interval Trigger
- Job Completion Trigger
- Mail Trigger
- Schedule Trigger
- Variable Trigger

Trigger Elements in the same And Group must all be satisfied before the relevant Job is started.

- Trigger Elements may be added to a Job or Folder.
- Any Trigger Elements set on a Folder will be inherited by that folder's child objects.

Adding a Schedule Object Trigger Element to a Job or Folder

1. Open the Definition dialog for the Job or Folder where the Trigger Element will be defined.

   **NOTE:** You can right-click either a Folder or a Job and select Properties from the drop-down list to open the Definition dialog.

2. Select the Schedule tab, and click Add.
3. Click the Run this job button.
4. Choose the Trigger needed to run the Job. The Trigger window will launch to display the Trigger Elements.
5. Define the available properties for the given Trigger Element.
6. Click Finish.
7. Save and Close the Definition dialog.

Modifying an Existing Schedule Object Trigger Element

1. Open the Definition dialog for the Job or Folder where the Trigger Element will be defined.

   **NOTE:** You can right-click either a Folder or a Job and select Properties from the drop-down list to open the Definition dialog.

2. Select the Schedule tab, and click Add.
3. Click the Run this job button.
4. Choose the Trigger needed to run the Job. The Trigger window will launch to display the Trigger Elements.
5. Define the available properties for the given Trigger Element.
6. Click Finish.
7. Save and Close the Definition dialog.

Trigger Schedule Object Properties

Each trigger element has a list of associated properties used to define how it will behave. Reference the tables below for a description of the properties available on each trigger.

File Trigger

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>This can be used to enable or disable a Job, Trigger, or Event Handler. When disabled, this Trigger Element will not cause the Job to run.</td>
</tr>
</tbody>
</table>
### Schedule

**Schedule for Date**
The date the Job is scheduled to run.

If you do not specify a date, the Job will run today at the time specified in the **Scheduled Time** property.

**NOTE:** If neither property is set, the Job will run at the default date and time that is specified in the Job.

**Scheduled Time**
The time of day when the Job is scheduled to run.

If you do not specify a time, the Job will run at the current time and on the day specified in the **Schedule for Date** property.

**NOTE:** If neither property is set, the Job will run at the default date and time that is specified in the Job.

### Trigger

**And Group**
All of the triggers in the same "And Group" must fire before the Job is triggered.

### File Watch

**Credentials**
Select a set of Credentials for this trigger. The credentials must have access to the file location.

**File Name**
The file specification to watch for. Wildcards are allowed.

**File Presence Option**
Specifies if the file must be present, absent, or available.

**Minimum Size**
Used to specify a minimum acceptable size for the file.

**Recursive**
Specifies whether to perform this action recursively (used for directories).

**Agent**
Used to specify the Agent where the file watch will be performed.

### Override

**Execute As**
Used to specify the set of credentials which the Job will execute as when submitted by this Trigger.

**Agent**
Used to set a specific Agent where the Job will execute when the Job is scheduled by this Trigger.

**Batch Queue**
Specifies the batch queue when this Job is scheduled by this Trigger.

**Submit On Hold**
Submits the Job on Hold when it is scheduled by this Trigger.

**Comment**
A comment that will be added to the entry when it is submitted by this Trigger.

### Interval Trigger

**Interval**
The Job will be scheduled on this interval.

**Next Scheduled Time UTC**
The date and time when this Job is next scheduled to run.

**Always Resubmit?**
When checked, a new instance is submitted for each interval. When unchecked, the existing instance is repeated if the interval is shorter than the retain time.
**Base Time**
When this Job repeats, which time should be used to calculate the next repetition? Options include EndTime, ScheduledTime, and StartTime.

**Status**

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>This can be used to enable or disable a Job, Trigger, or Event Handler. When disabled, this Trigger Element will not cause the Job to run.</td>
</tr>
</tbody>
</table>

**Override**

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>Used to set a specific Agent where the Job will execute when the Job is scheduled by this Trigger.</td>
</tr>
<tr>
<td>Batch Queue</td>
<td>Specifies the batch queue when this Job is scheduled by this Trigger.</td>
</tr>
<tr>
<td>Submit On Hold</td>
<td>Submits the Job on Hold when it is scheduled by this Trigger.</td>
</tr>
<tr>
<td>Execute As</td>
<td>Used to specify the set of credentials which the Job will execute as when submitted by this Trigger.</td>
</tr>
<tr>
<td>Comment</td>
<td>A comment that will be added to the entry when it is submitted by this Trigger.</td>
</tr>
</tbody>
</table>

**Job Completion Trigger**

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>This can be used to enable or disable a Job, Trigger, or Event Handler. When disabled, this Trigger Element will not cause the Job to run.</td>
</tr>
</tbody>
</table>

**Schedule**

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schedule for Date</td>
<td>The date the Job is scheduled to run.</td>
</tr>
</tbody>
</table>

If you do not specify a date, the Job will run today at the time specified in the Scheduled Time property.

**NOTE:** If neither property is set, the Job will run at the default date and time that is specified in the Job.

| Schedule Time | The time of day when the Job is scheduled to run. |

If you do not specify a time, the Job will run at the current time and on the day specified in the Schedule for Date property.

**NOTE:** If neither property is set, the Job will run at the default date and time that is specified in the Job.

**Trigger**

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigger Job</td>
<td>The Job which the trigger should watch for.</td>
</tr>
<tr>
<td>Success or Failure</td>
<td>Should this Trigger watch for a successful or failed completion of the Trigger Job? If Success and Failure conditions have not been set on the Trigger Job, Success will be satisfied by Informational or better completion status, and Failure will be satisfied by Warning or worse completion status.</td>
</tr>
<tr>
<td>And Group</td>
<td>All of the triggers in the same &quot;And Group&quot; must fire before the Job is triggered.</td>
</tr>
</tbody>
</table>

**Override**

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execute As</td>
<td>Used to specify the set of credentials which the Job will execute as when submitted by this Trigger.</td>
</tr>
</tbody>
</table>
Agent
Used to set a specific Agent where the Job will execute when the Job is scheduled by this Trigger.

Batch Queue
Specifies the batch queue when this Job is scheduled by this Trigger.

Submit On Hold
Submits the Job on Hold when it is scheduled by this Trigger.

Comment
A comment that will be added to the entry when it is submitted by this Trigger.

Mail Trigger

Status
Enabled
This can be used to enable or disable a Job, Trigger, or Event Handler. When disabled, this Trigger Element will not cause the Job to run.

Schedule
Schedule for Date
The date the Job is scheduled to run.
If you do not specify a date, the Job will run today at the time specified in the Scheduled Time property.

NOTE: If neither property is set, the Job will run at the default date and time that is specified in the Job.

Scheduled Time
The time of day when the Job is scheduled to run.
If you do not specify a time, the Job will run at the current time and on the day specified in the Schedule for Date property.

NOTE: If neither property is set, the Job will run at the default date and time that is specified in the Job.

Trigger
And Group
All of the triggers in the same "And Group" must fire before the Job is triggered.

Mail Server
Mail Server
The mail server to use when sending e-mail.

Mail Protocol
The mail protocol to use when connecting to the e-mail server.

Mail Credentials
The credentials used to access the mail server.

Mail Check Interval
The interval (in Delta Time) between checks for new mail on the defined mail server.

Use SSL
When true, SSL is used to secure the connection to the mail server.

Implicit SSL
When true, implicit SSL is used to secure the connection to the mail server.

Port
The TCP/IP port to connect to on the mail server. Zero (0) is the default.

Disposition
Mark as Read
When set to true, the e-mail will be marked as read when the event fires.

Delete E-Mail
When set to true, the e-mail will be deleted from the mail server when the event fires.

Mail Selection
Subject
The Subject of the e-mail.
From Address
The from address that will be used in e-mails sent by JAMS.

To Address
The address the e-mail will be sent to.

Override Description
Used to specify the set of credentials which the Job will execute as when submitted by this Trigger.

Agent
Used to set a specific Agent where the Job will execute when the Job is scheduled by this Trigger.

Batch Queue
Specifies the batch queue when this Job is scheduled by this Trigger.

Submit On Hold
Submits the Job on Hold when it is scheduled by this Trigger.

Comment
A comment that will be added to the entry when it is submitted by this Trigger.

Schedule Trigger

Status
Enabled
This can be used to enable or disable a Job, Trigger, or Event Handler. When disabled, this Trigger Element will not cause the Job to run.

Schedule
Time Zone
The JAMS Time Zone used for this Element

Scheduled Date
A natural language date specification for this Schedule. A comma separated list may be used.

Scheduled Time
The time of day when the Job should start

Exceptions
Description
Except For Date
Determines the Dates or Date Specifications when this Job will not run.

Start Date
If defined, the Job will not schedule to run until this date.

End Date
If defined, the Job will not schedule to run after this date.

Non Workday Scheduling
Determines if this Job should be scheduled on non-workdays.

Override
Agent
Used to set a specific Agent where the Job will execute when the Job is scheduled by this Trigger.

Batch Queue
Specifies the batch queue when this Job is scheduled by this Trigger.

Submit On Hold
Submits the Job on Hold when it is scheduled by this Trigger.

Execute As
Used to specify the set of credentials which the Job will execute as when submitted by this Trigger.

Comment
A comment that will be added to the entry when it is submitted by this Trigger.

Variable Trigger

 NOTE: A Variable Trigger is evaluated and can run only when the value of the variable has changed, even if the Trigger conditions are currently true. Changing the Job will not cause a Trigger to be evaluated.
<table>
<thead>
<tr>
<th><strong>Status</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>This can be used to enable or disable a Job, Trigger, or Event Handler.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Schedule</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Schedule For Date</td>
<td>The date the Job is scheduled to run.</td>
</tr>
<tr>
<td></td>
<td>If you do not specify a date, the Job will run today at the time specified in the</td>
</tr>
<tr>
<td></td>
<td><strong>Scheduled Time</strong> property.</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> If neither property is set, the Job will run at the default date and</td>
</tr>
<tr>
<td></td>
<td>time that is specified in the Job.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Value</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>A JAMS Variable that contains the default value.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Condition</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Compare Condition</td>
<td>Specifies the logical comparison that should be performed.</td>
</tr>
<tr>
<td>Compare Value</td>
<td>Specifies the value used in the condition comparison</td>
</tr>
</tbody>
</table>
Folders
Working with Folders

A JAMS Folder contains a user defined grouping of Jobs that appear on the Shortcut Bar. Folders can be defined by department (e.g., Accounting, IS, Sales), by function (e.g., EDI, Web, Security, Audit), or any other way that is meaningful to your organization.

Add a New Folder Definition

A Folder contains a logical grouping of Jobs. Folders can be organized by department (e.g., Accounting, IS, Sales) by function (e.g., EDI, Web, Security, Audit) or any other way that is meaningful to your organization.

1. To create a new Folder, Right-Click the desired parent folder in the Folder Navigator and choose the Add Folder command from the popup menu.

2. This action opens the Add a New Folder dialog. Give the new Folder a Name, then click OK to create the Folder.

   NOTE: To quickly open the full Folder Definition dialog, check Edit the new Folder’s properties after adding before clicking OK.
3. **Right-click** on any existing folder and select **Properties** from the drop-down list to open the full *Folder Definition* dialog. In this full definition dialog, users may define a number of Properties, Elements, Parameters, and Security controls on the Folder.

4. **Save and Close** the *Folder Definition* dialog.

**Modifying an Existing Folder**

1. To make changes to an existing Job, open its *Folder Definition* dialog from the Definitions View.

2. To open the *Folder Definition* dialog, **Right-click** the desired Folder and select **Properties** from the *FolderOptions* menu.

3. The *Folder Definition* dialog appears.


5. When the Folder has been reconfigured as desired, click **Save and Close**.

**Deleting an Existing Folder**

1. Open the *Definitions* view from the Shortcut bar.

2. **Right-Click** the desired Folder, then select **Delete Folder** from the drop-down menu.

3. A dialog will appear to confirm the Folder Deletion. If the desired Folder was selected for deletion, click **Yes**. If the incorrect Folder was selected, click **No** or **Cancel**.

**Folder Definition Tabs**

**Folder**

This tab contains basic Folder Information:

- **Folder Name** - This property incorporates the unique identifier for the Folder.
- **Description** - Entries made in the Description property appear in menus, lists and reports and provides a more detailed explanation for the Folder.
- **Last Changed** - The Last Changed property displays the user who last modified this Folder including the date and time of the modification.
- **Reset Statistics** - When a Job or Sequence executes it creates statistics based on the most recent run as well as the average of previous runs. The **Reset** button can be used when moving from a development to a production cycle in order to clear erroneous Job and Sequence statistics and begin with a clean slate.
Properties

The Soft Properties listed below are available for any JAMS Folder (or Job).

Until a value is set for a property, it will not display in the Properties tab. To display a soft property, use the Add button in the Properties Tab, select the desired Property, then click OK in the Add Property dialog.

**Completion**

**Bad Regex Pattern**
A regular expression pattern that indicates a failed execution.

**Exit Code Handling**
Specifies how JAMS should evaluate a Job’s exit code.

**Good Regex Pattern**
A regular expression pattern that indicates a successful execution for this Job.

**Minimum Severity**
The minimum acceptable completion severity.

**Retain Option**
Selects how the Job will be displayed in the Monitor after it completes. If set to Always, completed Jobs will never leave the Monitor.

**Retain Time**
When the Retain Option is set to Timed, this property is used to specify (in minutes) the amount of time to display the completed Job in the Monitor.

**Specific Informational**
A comma separated list of integer values for Informational Job completion exit codes.

**Specific Values**
A comma separated list of integer for the Job completion exit codes.

**Specific Warning**
A comma separated list of integer for Warning Job completion exit codes.

**Condition**

**CompareCondition**
Specifies the logical comparison that should be performed.

**CompareValue**
Specifies the value used in the condition comparison.

**E-Mail**

**CC Address**
A comma separated list of e-mail addresses that will be cc’d on the e-mail. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*.

**From Address**
The from email address used when sending the e-mail from JAMS.

**Message Body**
The body of the e-mail message. Users may enter Markdown language in the Message Body.

**To Address**
A comma separated list of e-mail addresses that the e-mail will be sent to. You can remove inherited addresses by prefixing the address with a minus sign (-). You can remove all inherited addresses by entering -*.

**Execute**

**Execute As**
Select the credentials this Job should use when executing.

**SSRS Credentials**
The credentials that will be passed to the SSRS Server when executing.

**Application**
This value is passed to the Win32 Create Process function as the value for the Application Name argument.

**Command**
This value is passed to the Win32 Create Process function as the value for the Command Line argument.

**Home Directory**
Specifies the full path to the initial current directory for the Job.

**Options**

**Force 32 Bit**
When set to true, this Job will run as a 32 bit process, even on 64 bit machines.

**Force V2**
When set to true, the Job will run using V2.0 of the .NET Framework.

**No BOM**
When set to true, the Job's source file will not have a Unicode byte order mark.
OSPromptPattern
A regular expression that will match the operating system command prompt.

Pass Parameters
When set to true, Job parameters will be passed to a PowerShell Job as PowerShell parameters.

Schedule

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>Used to specify the Agent where the Job will execute.</td>
</tr>
<tr>
<td>Batch Queue</td>
<td>Specifies the batch queue for a Job.</td>
</tr>
<tr>
<td>Submit on Hold</td>
<td>When enabled, this Job will enter the schedule in an &quot;On Hold&quot; status, requiring user input.</td>
</tr>
<tr>
<td>Submit Date</td>
<td>The default Submit Date for this Job, when manually submitted.</td>
</tr>
<tr>
<td>Job Concurrent Limit</td>
<td>The maximum number of instances of this Job that may execute concurrently.</td>
</tr>
<tr>
<td>Single Instance Action</td>
<td>Defines the action to be taken if a Job tries to start while an instance of that Job is already running.</td>
</tr>
</tbody>
</table>

Status

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>This can be used to enable or disable a Job, Trigger or Event Handler.</td>
</tr>
</tbody>
</table>

General

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debug</td>
<td>Determines if Job(s) will run in Debug Mode. When a Job runs in Debug Mode, it will not satisfy Dependencies or Triggers.</td>
</tr>
<tr>
<td>Include MS Dynamics Log</td>
<td>If the Job is a MS Dynamics Job, indicates whether the Job will include MS Dynamics Job Logs.</td>
</tr>
<tr>
<td>Credentials</td>
<td>Select a set of credentials from the drop-down list for the given object.</td>
</tr>
<tr>
<td>JDE Credential</td>
<td>Credentials provided for JD Edwards Jobs within the Job or Folder. If this is not a JD Edwards Job, these credentials are not required.</td>
</tr>
<tr>
<td>CompletionBearing</td>
<td>Determines the acceptable Completion Severity configuration for a Job.</td>
</tr>
<tr>
<td>DataType</td>
<td>Determines the Data Type used for this object.</td>
</tr>
<tr>
<td>Default Date</td>
<td>The default submit date used when this Job is manually submitted.</td>
</tr>
<tr>
<td>KeepLogs</td>
<td>Determines whether the batch log files are deleted after the current Job completes. (Only applies to OpenVMS platforms).</td>
</tr>
<tr>
<td>Log File Name</td>
<td>Displays or sets the Log File Name for this Job.</td>
</tr>
<tr>
<td>Log Location</td>
<td>The default log location. This can include a filename and extension to set the default values.</td>
</tr>
<tr>
<td>MS Dynamics Credential</td>
<td>The credentials to use when running a MS Dynamics Job.</td>
</tr>
<tr>
<td>Notify Severity</td>
<td>The maximum completion severity required to send a notification.</td>
</tr>
<tr>
<td>Notify User</td>
<td>Include the user that submitted the Job when sending notifications?</td>
</tr>
<tr>
<td>Options</td>
<td></td>
</tr>
<tr>
<td>Port</td>
<td>Specifies the Port JAMS will attempt to connect to. Zero (0) is the default.</td>
</tr>
<tr>
<td>Configuration</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>PrintLocation</td>
<td>The directory where Print Files should be placed.</td>
</tr>
<tr>
<td>PrintQueue</td>
<td>The Print Queue that will be used for Reports.</td>
</tr>
<tr>
<td>Protocol</td>
<td>The protocol to use when connecting to a mail server. Typical mail server protocols include IMAP and POP3.</td>
</tr>
<tr>
<td>RemoteJobName</td>
<td>The name of the Remote Job.</td>
</tr>
<tr>
<td>RemoteScheduler</td>
<td>The name of the Remote Machine where the Remote Job executes.</td>
</tr>
<tr>
<td>Restartable</td>
<td>Defines whether the Job can be restarted.</td>
</tr>
<tr>
<td>Retry Count</td>
<td>Defines the maximum number of automatic retry attempts if the Job fails.</td>
</tr>
<tr>
<td>Retry Interval</td>
<td>the interval, in Delta Time, between automatic retry attempts.</td>
</tr>
<tr>
<td>Run Priority</td>
<td>The execution priority for the Job. A Job's Priority is the sum of the Job's priority AND priority values on parent folders.</td>
</tr>
<tr>
<td>Scheduling Priority</td>
<td>The Scheduling Priority for the Job.</td>
</tr>
<tr>
<td>Search Path</td>
<td>A comma separated list of paths which are searched when trying to resolve relative references.</td>
</tr>
<tr>
<td>SLA Time</td>
<td>The time of day a Job must complete by to meet a Service Level Agreement.</td>
</tr>
<tr>
<td>Submit Time</td>
<td>The default Submit Time for the Job, when manually submitted.</td>
</tr>
<tr>
<td>Suppress Menu Display</td>
<td>Defines if this Job should be omitted from Submit Menus. Set to False (unchecked) to display this Job on Submit Menus.</td>
</tr>
<tr>
<td>Template Library</td>
<td>The full file specification that contains the JAMS Templates used when parsing the Job.</td>
</tr>
<tr>
<td>Timestamp Logs</td>
<td>If enabled, batch log files will be timestamped with the extension format &quot;.LOG_yyyymmdd_hhmmsscc&quot;.</td>
</tr>
<tr>
<td>To</td>
<td>The destination or recipient of this object.</td>
</tr>
<tr>
<td>Host Key Checking</td>
<td>Defines what JAMS should do if the SSH fingerprint does not match when connecting.</td>
</tr>
<tr>
<td>Accept Host Key</td>
<td>An indicator that a host key is accepted</td>
</tr>
<tr>
<td>SSH Prompt</td>
<td>A regular expression pattern to handle SSH prompts.</td>
</tr>
<tr>
<td>SSH Timeout</td>
<td>The SSH connection timeout, defined in seconds.</td>
</tr>
<tr>
<td>SSL Cipher Suite</td>
<td>Comma separated list of SSL Cipher Suites to allow.</td>
</tr>
<tr>
<td>SSL Version</td>
<td>Comma separated list of the SSL Versions to allow.</td>
</tr>
<tr>
<td>SSL Minimum Key Size</td>
<td>The minimum accepted key size. When set to zero (0), will default to 1024.</td>
</tr>
<tr>
<td>SSH Compression</td>
<td>The compression level to use with the SSH connection.</td>
</tr>
<tr>
<td>SSH Minimum Key Size</td>
<td>The minimum accepted decimal key size. When set to zero (0), will default to 1024.</td>
</tr>
<tr>
<td>SSH Mac Algorithm</td>
<td>The SSH Mac Algorithm used.</td>
</tr>
<tr>
<td>SSH Key Exchange</td>
<td>The SSH Key Exchange Algorithm used.</td>
</tr>
<tr>
<td>SSH Host Key</td>
<td>The SSH Host Key Algorithm used.</td>
</tr>
<tr>
<td>SSH Encryption</td>
<td>Comma separated list of SSH Encryption Algorithms used.</td>
</tr>
</tbody>
</table>

**Elements**

There are five Element categories in JAMS. These are Documentation, Event Handler, Prerequisite, Result,
<table>
<thead>
<tr>
<th><strong>Documentation</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Documentation</td>
<td>Used to configure Documentation elements shown on Jobs. Documentation supports Markdown.</td>
</tr>
<tr>
<td>Link</td>
<td>Used to configure hyperlinks for the Documentation of a Job or Folder.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Event Handler</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Notification Job</td>
<td>Used to configure a notification Job to run based on the completion severity or status of a Job.</td>
</tr>
<tr>
<td>Recovery Job</td>
<td>Configures a Recovery Job to run if the given Job completion does not exceed recovery severity.</td>
</tr>
<tr>
<td>Repeat</td>
<td>Executes the same entry of the Job at a defined interval for a defined time window.</td>
</tr>
<tr>
<td>Resubmit</td>
<td>Submits a new entry of the Job at a defined interval for a defined time window.</td>
</tr>
<tr>
<td>Runaway</td>
<td>Determines when the Job becomes a Runaway Job, and what action is taken if the Job becomes Runaway.</td>
</tr>
<tr>
<td>Send E-Mail</td>
<td>Used to an e-mail based on the completion severity or status of a Job.</td>
</tr>
<tr>
<td>Short</td>
<td>Defines the minimum elapsed time for successful completion of a Job, and what action is taken if the Job does not meet that minimum time.</td>
</tr>
<tr>
<td>Stalled</td>
<td>Defines the maximum elapsed time for the Job before it is considered stalled.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Prerequisite</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Date Specific</td>
<td>Requires that another Job completed on a natural-language specified date before this Job is allowed to start.</td>
</tr>
<tr>
<td>Dependency</td>
<td>A file that must be present, absent or available before this Job is allowed to start.</td>
</tr>
<tr>
<td>File Dependency</td>
<td>A Job that must satisfy the defined completion severity before this Job will be allowed to start.</td>
</tr>
<tr>
<td>Job Dependency</td>
<td>A Job that must succeed before this Job will be allowed to start.</td>
</tr>
<tr>
<td>Precheck Job</td>
<td>A remote Job that must satisfy the defined completion severity before this Job will be allowed to start.</td>
</tr>
<tr>
<td>Remote Job</td>
<td>A resource requirement that must be satisfied before this Job will be allowed to start.</td>
</tr>
<tr>
<td>Dependency</td>
<td>A variable dependency that must be satisfied before this Job will be allowed to start.</td>
</tr>
<tr>
<td>Resource Requirement</td>
<td>A time window where this Job will be allowed to start, and the action taken if the window is missed.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Result</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>CR Job Report</td>
<td>Configures a CR Job Report to submit on completion of the Job.</td>
</tr>
<tr>
<td>Email Report</td>
<td>Configures an Email Report to submit on completion of the Job.</td>
</tr>
<tr>
<td>JDE Report</td>
<td>Configures a JD Edwards Print Report to submit on completion of the Job.</td>
</tr>
<tr>
<td>MS Dynamics Email Report</td>
<td>Configures a Microsoft Dynamics E-mail Report to submit on completion of the Job.</td>
</tr>
<tr>
<td>MS Dynamics File Report</td>
<td>Configures a Microsoft Dynamics File Report to submit on completion of the Job.</td>
</tr>
</tbody>
</table>
MS Dynamics Print Report  Configures a Microsoft Dynamics Print Report to submit on completion of the Job.

SSRS Print Report  Configures an SSRS Print Report to submit on completion of the Job.

**Trigger**

**Description**

**File Trigger**  Triggers a Job to run based on the availability of a file or group of files. Users may define And Groups to require the presence of multiple files before the Job is triggered.

**Job Completion Trigger**  Triggers a Job to run based on the completion severity of a defined Job Completion Trigger.

**Mail Trigger**  Triggers a Job to run based on the receipt of a mail message.

**Schedule Trigger**  Uses natural date language and comma separated date lists to trigger a Job to run. Users may define multiple Schedule Triggers on a single object.

**Parameters**

A JAMS Parameter contains a single slice of data that (unlike a Variable) remains exclusive to the associated Job or Folder. Parameters defined on a Folder are inherited by all of that Folder’s child objects.

When JAMS interactively submits a Job with Parameters, the user is presented with a fill-in-the-blank form to define values for these parameters. When JAMS builds a Parameter form, the Parameters are placed in the order in which they are defined, unless the Parameters were given Sequence numbers.

If a Job uses a parsed Execution Method, you can embed Parameter names into the Job’s Source using the `<<ParameterName>>` specification. When the Job runs, it uses the value of the Parameter in place of this specification.

A JAMS Folder may contain zero or more Parameters.

**Add a Parameter**

1. In any *Folder Definition* dialog, open the *Parameters* tab.
2. Click the *Add* button to open the *Add a Parameter* dialog.
3. In the dialog, define the *Parameter Name*, *Data Type*, and *Default Value* of the Parameter.
4. By default, the full Parameter Definition dialog will open when the Job is initially saved. Click OK.
5. The Parameter Definition dialog will open.
6. Define additional Parameter Properties as desired.
7. Save and Close the Parameter Definition dialog.
8. Save and Close the Folder Definition dialog.

Parameter Properties

**Name**

**ParamName**

This property defines the name of the Parameter. Each Parameter name must be unique within a particular Job.

**Data Type**

**DataType**

Use the dropdown to specify a data type for the Parameter.

- Text
- Integer
- Date
- Time
- DateTime
- Float
Length
If the Parameter’s data type is set to Text or Integer, specify the maximum length within this property.

**User Interface Description**

**Allow Entry**
When this attribute is enabled, the user can make an entry to this Parameter when submitting the Job.

**HelpText**
The entered text displays when a user hovers over the Parameter entry field on manual submissions.

**Hide**
This attribute determines if the Parameter is displayed when this Job is submitted. If you have "Manage" access to the current Job, hold down the ALT key when submitting the Job and all hidden Parameters will be displayed.

**Must Fill**
When enabled, this attribute requires the user to completely fill this Parameter to its Maximum Length as set on the Data Type tab.

**Prompt**
When a Job is interactively submitted, a form is created which is used to prompt the user for the Job’s Parameter values. The entered text represents the prompt to the user submitting the Job.

**Required**
When checked, this attribute requires users to enter a value for this Parameter.

**Sequence**
An integer used to sort the parameters when presented to end-users.

**Uppercase**
When checked, this attribute converts all entered data to uppercase.

**ValidationData**
The value used when attempting to validate the Parameter value.

**ValidationType**
This property allows you to select the type of validation JAMS can perform on the Parameter value. The validation types are:

- **None**: No validation is performed.
- **Directory**: this validation is used on text parameters. A button is placed next to the parameter’s text box that opens a dialog allowing the user to browse the file system for a directory.
- **SaveFile**: this is used on text parameters. A button is placed next to the parameter’s textbox that opens a dialog allowing the user to browse for a file, which may or may not exist.
- **OpenFile**: this is used on text parameters. A button is placed next to the parameter’s textbox that opens a dialog allowing the user to browse the file system for a file, which must exist.
- **MaskedEdit**: this validation type uses a mask to distinguish between proper and improper user input. The mask is set in the Validation Data property.

**Value Description**

**Default Format**
The format string that defines how the Parameter will be formatted. Default formats vary depending on the Parameter's Data Type.

**Default Value**
The property specifies the default value for this Parameter. The default value for dates may be entered as a specific date or you can use JAMS English language date text such as Today, Last Monday, First Monday of Month etc.

**Variable**
Optionally, you can acquire the Default Value for the Parameter using a JAMS Variable. Use the File Browser button to search for and select an existing variable.
Security

The properties on this tab essentially act as an access control list with one to many Access Control Entries (ACE) for the current Folder as well as for any Job or Sequence contained within the Folder.

NOTE: In JAMS V7.0.1367 and later, removing all ACEs on an object behaves the same as Windows would. When all ACEs are removed from an object, only the GrantAdministratorsBypass group will have access to the object. Previously, removing all ACEs from an object would give all Authenticated Users access to that object.

Each ACE can specify the following rights:

- **Abort**: allows the user to access the JAMS Job Monitor to abort or restart an occurrence of a Job or Sequence within the Folder.
- **Add Jobs**: allows the addition of new Job and Sequence definitions to this Folder.
- **Change**: permits modification of the Folder definition provided the user also has Change access to Folder definitions.
- **Change Jobs**: Allows the modification of existing Job and Sequence definitions in this Folder.
- **Control**: permits modification of the current Folder’s Access Control List.
- **Debug**: allows submission of Jobs within this Folder, but only if the /DEBUG qualifier is used on the JAMS SUBMIT command. This qualifier submits the Job or Sequence under the submitter’s username rather than the username specified in the Folder Definition.
- **Delete**: permits deletion of the current Folder definition provided the user also has DELETE access to Folder definitions.
- **Delete Jobs**: allows the deletion of existing Job and Sequence definitions within the Folder.
- **Inquire**: permits inquiry into the current Folder definition provided that the user also has INQUIRE access to Folder definitions.
- **Inquire Jobs**: permits inquiry into the current Folder definition as well as all contained Jobs and Variables.
- **Manage**: allows the user to control the JAMS Job Monitor to hold, reschedule, release or delete an occurrence of a Job or Sequence in this folder.
- **Monitor**: allows Jobs to appear in the Job Monitor.
- **Submit**: grants the right to submit a Job or Sequence within the Folder

Enable Inheritance

This dropdown property controls how the Folder inherits settings from its subfolders and Jobs. The inheritance options include:

- This folder only
- This folder, subfolders and jobs
- This folder and subfolders
- This folder and jobs
- Subfolders and jobs only
- Subfolders only
- Jobs only
Dates and Times
Date Properties

Date properties are used to define a single date that has been marked with a specific Date Type. Date Types are high level definitions for special dates, such as holidays or fiscal periods. For more information, go to the Date Type Properties topic.

The Date properties can be accessed by selecting the Dates Shortcut and double-clicking on a marked day, as shown in the example below.

The Date properties can be accessed by selecting the Dates Shortcut and double-clicking on a marked day, as shown in the example below.

This action opens a dialog that includes the following properties:

**Specific Type**

This property represents the specific name of the date you are defining. This is not required. However, if you specify a value, it must be one of the values defined in the Date Type definition.

This property is significant when referring to a specific instance of a Date Type. For example, if you schedule a Job to run on Christmas, you could define a Date Type of HOLIDAY with a Specific Date Type of CHRISTMAS.

To add a new Specific Type open the Dates Shortcut and select the Edit Date Types button.
From the Date Types listing either create a new Date Type or double-click on an existing entry to open its properties window.

**Description**

This property is for reference purposes only and is used to quickly summarize the date definition.

**Workday**

There are three options for this property: True (checked), False (null), or Maybe (shaded).

When JAMS checks if a date is a workday, it initially looks at the date definitions for the date in question. If it finds one with either a True or False in its property, it stops checking. However, if none of the date definitions for this date specify True or False (or there are no definitions for this date), JAMS then checks the configuration to see if the day of the week on which this date falls is normally a workday.

Maybe is generally the right setting unless one of the reasons for creating the date definition is to change the workday status.

⚠️ **Caution**: If you create two definitions for the same date, one that says this is a workday and one that says this is not a workday, JAMS will stop checking when it finds the first definition.

**Date Type**

Identifies the Date Type of the date. Every Date definition must be associated with a Date Type, but the same date may be associated with more than one Date Type.

**Last Change**

Indicates the date and time this parameter was last modified.
Creating Special Date Definitions

In order to build a custom calendar in JAMS you must first set up special date definitions that are meaningful to your organization. Date Types and Specific [Date] Types are used to define various classes for these special dates, such as holidays or fiscal periods. See the Date Properties topic for more information.

**Note:** In order for JAMS to evaluate a calendar, it must always have at least one date defined in the past and at least one date defined in the future.

For example, evaluating a July 4th holiday would be satisfied if a May 31st holiday and December 25th holiday are also defined. It is important to have calendars consistently updated so that jobs can properly evaluate the dates within the calendar.

When choosing Date Types and Specific Date Types identifiers for your organizations, make sure they are readable names as they will be used in JAMS English language specifications. In addition, JAMS recognizes month names before checking for Date Types, so it is best to avoid using Date Type definitions to override calendar months. For example, JAMS converts "FIRST DAY OF APRIL" to April 1st for the current calendar year. If your accounting department prefers to tag another date/year for the first day of April, a workaround can be made using Date Types and Dates. In this example, using an identifier such as FISCAL as the Date Type with the month name as the Specific Date Type your organization’s accountants can now use the "FIRST DAY OF APRIL" to specify the first day of April’s fiscal period. Or, as an alternative, they can use "FIRST DAY OF FISCAL" to mean the first day of the current fiscal period.

**Note:** there is nothing special about the name FISCAL. It was chosen to make the date text more readable and could just as easily been substituted with other identifiers such as PERIOD or FP.

You can also use unique names for the Specific Date Types in a Date Type definition. In the previous example, we could have used FP_JAN, FP_FEB etc. for the specific date type names. Then you could express the first day of fiscal April as "FIRST DAY OF FP_APR".

Adding or Modifying Date Types

When setting up a custom calendar you must first define or modify a Date Type. To get started, open the Dates Shortcut in the Management Group and select the File Browser button adjacent to the Date Type dropdown.

This action opens a pick list displaying existing Date Types. To add a new Date Type click on the blue Add document button located on the top, middle left of the dialog as shown below. This action opens the JAMS Date Type Definition Wizard to guide you through the steps in setting up a new Date Type definition.
To modify an existing Date Type, highlight the listed entry, then select the Properties button (highlighted above) or double-click an existing Date Type listing to access its Properties display.

Finally, to delete a Date Type, highlight the item and then choose the Delete button.

**Date Type tab**

Date Type properties are organized under two tabs (Date Type and Specific Types).
**Date Type**

This initial property is the unique identifier for the particular Date Type. Every date definition must be associated with a Date Type. It should be noted that the same date can be associated with more than one Date Type.

**Description**

This optional property is used in menus, lists and reports to provide a more expanded Date Type description.

**Continuous Data Type?**

Enabling this checkbox property indicates whether or not this Date Type is continuously occurring. A continuous Date Type is one which spans a number of consecutive dates, such as a fiscal period. In this case, to define your organization’s fiscal periods create a Date Type, e.g., FISCAL, and then set up the start date for each fiscal period by marking the date in the Special Date Definitions calendar window. Please note, you may need additional continuous Date Types for periods such as fiscal quarters, pay periods in order to define an ending date for the preceding period.

Disabling the checkbox defines a non-continuous Date Type. The most common example of a non-continuous Date Type are holidays. Most likely, your environment includes other situations where a non-continuous Date Type is useful. One example might be if your organization takes physical inventories on a scheduled basis. If you have special batch process that runs on a day when a physical inventory is taken, you could create a Date Type called PHYSICAL and then define a Job that is scheduled to run on PHYSICAL. When a physical inventory is scheduled,
you simply add the specific dates to correctly schedule the Jobs.
You could also use the PHYSICAL Date Type to obtain default values for a Job’s parameters by specifying "LAST PHYSICAL" as the default value for a date parameter.

**Last Changed**

This property indicates the date and time the Date Type definition was last modified.

**Specific Types tab**

**Specific Types**

This property lists the specific occurrences for this Date Type. The order of these names does not matter except for the first name. The first Specific Type must identify the date or period which occurs first in any given year.

The concept of "Year" is user defined. You can specify the first date/period to occur in a calendar year or a fiscal year.

To define a Specific Type select the **New** button on the top right of the window and enter the Specific Type identifier in the blank listing. To modify an existing Specific Type listing, just highlight and retype.

To delete a Specific Type entry, highlight the item and then choose the **Delete** button.
Specifying Dates Using Natural Language

As a scheduling application JAMS uses date specifications for many of its features and functions. Sometimes using exact date specification is sufficient for scheduling a Job. However, most of the time a generic date (e.g., using Workdays or First Friday of the Month) can provide additional flexibility. JAMS has been designed to support natural language and generic date entries throughout the client interface.

There are three key scenarios for JAMS to accept generic date specifications:

- **Scheduled dates**: generic dates can be used when scheduling a Job in JAMS.
- **Default parameters**: if a Job parameter’s data type is DATE, you can specify a default value using generic date specifications.

**Note**: When a user selects a Job, they are presented with a form to set values for the Job's parameters. Parameters with a data type of DATE are presented using the standard MM/DD/YY date field, but JAMS uses the generic date specification to determine the default date.

**Note**: When JAMS automatically submits Jobs, prompting for the parameter values is not possible. Instead, the generic date specification is evaluated to obtain the value for the parameter.

Simple Date Specifications

Simple date specifications specify a date relative to the current date. The format and description for a simple date specification is as follows:

- **Day-of-Week**: can be the full name of a weekday or the first three letters of a weekday name. When specifying a day of the week, e.g., MONDAY, JAMS interprets this as “Monday of this week.” In this context, a week begins on Monday and ends on Sunday.

  The days of the week can be preceded by a modifier such as LAST, THIS, or NEXT. Inserting a THIS modifier has no effect and is used only for readability.

  When using a modifier, such as LAST, JAMS interprets this to mean “the weekday from last week.”

- **Workdays**: If you specify "WORKDAY" or "WORKDAYS", JAMS determines if a day is a workday using a two-step process.
  1. JAMS looks at the WORK_ddd configuration setting, where ddd indicates the day of the week (e.g. Wed). This sets up the initial true or false indication that the day is indeed a workday.
  2. For the date in question, JAMS also searches for special dates. To access, select the Date shortcut and double-click the desired date.
    - If the Workday check box is enabled, the date is a workday.
    - If the Workday box is unchecked, the date is not a workday.
    - However, if the check box is filled, it indicates that the special date should have no effect on the workday status.

- **Weekdays**: specifies the first weekday after the current date.
- **Today**: identifies the current date.
• **Tomorrow**: displays the current date plus one day.
• **Yesterday**: displays the current date minus one day.
• **Daily**: is synonymous with **TODAY** and used primarily for readability.
• **+/- number of days**: is a simple date specification that can have positive or negative integers appended, adding or subtracting the indicated number of days.
• **Every Other (day)**: JAMS will use the next available occurrence of the specified date type, repeating on each alternate available date. Days can be set using **Day-of-Week** specifications, **Weekday**, or **Workday**.
  
  E.g. *Job A* is created on Saturday and scheduled to run "Every Other Workday". The job will run on Monday, Wednesday, and Friday the first week, then Tuesday and Thursday the second week, and so on.

### Valid Simple Date Specifications

Included below are some examples of valid, simple-date specifications:

- **TODAY + 5**
- **NEXT MONDAY**
- **LAST TUESDAY**
- **WORKDAYS**
- **Every Other Friday**

### Complex Date Specifications

Complex date specifications are used to stipulate a date relative to an arbitrary period of time (e.g., describing a month or fiscal period).

A complex date specification may be thought of as two components: the day specification and the period specification. For example, in the text "1st WORKDAY of NEXT MONTH," the day specification is 1st WORKDAY and the period specification is NEXT MONTH.

The general format of a complex date specification is as follows: `[day-of-period OF] period [+/- number of days]` The syntax for the day-of-period specification can take one of two forms. Both forms are comparable. The two forms for the day-of-period specification are:

#### Form 1

- **FIRST**
- **LAST**
- 1[st]
- 2[nd]
- 3[rd]
- integer [st, nd, th]
  
  - **Day-of-week**
  - **WORKDAY**
  - **WEEKDAY**
  - **DAY**
  - **MONTH**

Or . . .

#### Form 2

- **Day-of-week**
- **WORKDAY**
- **WEEKDAY**
- **WEEK**
- **DAY**
- **MONTH**
The syntax for the period specification is:

- **THIS**
- **NEXT**
- **LAST**
  - **YEAR**
  - **MONTH**
  - **Month-name**
  - **Date-Type**
  - **Specific-Date-Type**
  - **Date-Type Specific-Date-Type**

**Note:** While this syntax specification may seem overly complex it actually mimics how people commonly specify dates.

### User Defined Period

When specifying a period with user defined Date Types, you may need to supply both specific and generic Date Types. For example, if you defined the Date Type `FISCAL` with specific Date Types of `PERIOD_01` through `PERIOD_12`, you can specify the second period as: `FISCAL PERIOD_02` or `PERIOD_02`.

If you defined two Date Types that both use the specific Date Type `PERIOD_02`, then you must specify `FISCAL PERIOD_02`.

### Specifying Weeks

When used in the context of a time period, a week is defined as starting on the first day of the period and continuing for seven days.

**Valid Complex-Date Specifications**

The following examples of valid complex-date specifications are based on the current date being January 1st, 2014.

<table>
<thead>
<tr>
<th>Specification</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIRST MONDAY OF NEXT MONTH</td>
<td>03-FEB-2014</td>
</tr>
<tr>
<td>2ND MONDAY OF THIS MONTH</td>
<td>13-JAN-2014</td>
</tr>
<tr>
<td>FIRST MONDAY OF LAST JANUARY</td>
<td>07-JAN-2013</td>
</tr>
<tr>
<td>6TH WEEKDAY OF NEXT MONTH</td>
<td>10-FEB-2014</td>
</tr>
<tr>
<td>6TH WEEKDAY OF MONTH</td>
<td>08-JAN-2014</td>
</tr>
<tr>
<td>LAST DAY OF LAST MONTH</td>
<td>31-DEC-2013</td>
</tr>
<tr>
<td>2nd DAY OF THIS WEEK</td>
<td>31-DEC-2013</td>
</tr>
<tr>
<td>3RD DAY OF WEEK</td>
<td>3RD DAY OF EACH WEEK</td>
</tr>
</tbody>
</table>
Named Times in JAMS

Named Times in JAMS offer users a unique way to schedule Jobs. With Named Times, users may define flexible windows of time when Jobs may be scheduled.

By leveraging Named Times, JAMS can prevent Jobs from executing until their scheduled window is enabled and can also take various actions when a Job’s window closes before the Job completes. Named Times can also be used to limit when manually submitted Jobs are allowed to run. Users could leverage this function to prevent manual Job submissions from running during periods when other Jobs use significant system resources. An overview of the Missed Windows Actions is available below.

Adding a New Named Time Definition

1. To create a new Named Time, open the Times view from the Shortcut bar.
2. Click the Add button in the Control Bar to open the Add a Named Time dialog.
3. Define a Name for the Named Time.
4. Define the other properties of the Named Time as desired.

   **NOTE:** An overview of the available properties is available below.

5. Click OK. The Time Name dialog will open.
6. Modify the fields as necessary.
7. With all properties defined as desired, click Save and Close.

Named Time Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Named Time Name</td>
<td>The name of this new Named Time.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional. A brief description of the Named Time.</td>
</tr>
<tr>
<td>Start Time</td>
<td>The starting time of this Named Time.</td>
</tr>
<tr>
<td>Automatically Enable</td>
<td>When checked, this Named Time is automatically enabled at the defined start time.</td>
</tr>
<tr>
<td>End Time</td>
<td>The ending time of this named time.</td>
</tr>
<tr>
<td>Automatically Disable</td>
<td>When checked, this Named Time is automatically disabled at the end time.</td>
</tr>
<tr>
<td>Edit the Named Time Definition after Adding</td>
<td>When checked, the full named time definition is opened for edit after the named time is added.</td>
</tr>
</tbody>
</table>

Defining or Modifying Schedule Windows to Jobs

Adding a Named Time Window to Jobs or Folders

1. Right-click the desired Job or Folder and then select Properties to open the Definition Dialog.
2. Select the Schedule tab.
3. Click the Add Item button.
4. Select This Job depends on a > time window.
5. Select a Missed Window Action, and then select the Schedule Window from the drop-down list.

   **NOTE:** When using a Named Time on a Time Window element, leave Schedule From Time and Schedule to Time blank.

6. Click Finish.
7. Save and Close the Job or Folder definition.

Missed Window Actions

To understand how Jobs will behave when using a Time Window, read the explanations of each Missed Window Action.
Action below.

- **No Action**: If you manually or automatically submit a Job that has No Action selected, the Job will start based on the submitted time. If a Schedule window has been set and you manually submit the Job, a message will be displayed on the Submit dialog to display the current Schedule window time range. A No Action Job will run to completion, even if the Schedule window closes.
- **Continue**: The Job does not start until the Schedule Window opens and continues running to completion, even if the Schedule Window closes. There is no rescheduling option.
- **Abort or Delete**: The Job does not start until the Schedule Window opens and is deleted or aborted if the window closes before the Job completes.
- **Reschedule or Continue**: The Job does not start until the Schedule Window opens. If the Job does not start by the time the window closes, it is rescheduled for the next time the window opens. If the Job does start, this action allows the Job to continue to completion even if the window closes.
- **Restart or Reschedule**: The Job does not start until the Schedule Window opens. If the Job has not completed by the time the window closes, the Job is forced to abort, and is rescheduled to run the next time the window opens.

**Named Time Definition Tabs: Properties**

The Named Time Properties are divided into two tabs: Named Time and Times.

**Named Time tab**

![Named Time tab screenshot]

**Name**

This property includes the unique identifier for the Named Time.

**Description**

This optional property appears in menus, lists and reports and provides a more detailed explanation for the Named Time.
**Last Changed**

The Last Changed property displays the user (Username) who last modified the Named Time along with the date and time of the modification.

**Times tab**

Start Time

Enter the time of day the Named Time is scheduled to begin.

**Automatically enable**

If checked (true), JAMS automatically enables the Named Time at its specified Start Time. When unchecked (false), the Named Time remains disabled until it is manually enabled.

End Time

Enter the time of day the Named Time is scheduled to end.

**Automatically disable**
If checked (true), JAMS automatically disables the Named Time at its specified End Time. When unchecked (false), the Named Time remains enabled until it is manually disabled.

**Last Time Window was Automatically Enabled or Disabled**

This property displays the last date and time when the Named Time was automatically enabled or disabled.

**Currently Enabled?**

If true, use the **Disable** button to manually disable the Named Time. If false, use the **Enable** button to manually enable the Named Time.
Dashboards and Reports
Custom Dashboards

JAMS includes a variety of preinstalled Dashboards that are intended to meet most user needs. Users are recommended to use preinstalled Dashboards before attempting to create custom dashboards, in order to gain an understanding of what kinds of data are important and what could potentially be added to enhance a dashboard for the given organization.

JAMS comes with powerful design features to modify or completely alter a Dashboard to fit the way you use JAMS. Custom Dashboards can incorporate a variety of data sources including: JAMS specific data, PowerShell scripts, and data stored in external databases - such as Microsoft SQL Server, MySQL, and Microsoft Access.

There are three key steps to building a custom Dashboard:

1. Defining a Data Source (using JAMS or external data sources).
2. Setting up a Dashboard Item (determining the Dashboard’s functionality and formatting).
3. Adding or modifying Parameters, setting up drill downs, filtering functionality.

Follow the steps below to build your own custom Dashboard using JAMS' powerful Dashboard Designer feature.

Defining a Data Source

The process of setting up a custom Dashboard begins by opening the Dashboard Designer and locating and defining a Data Source. Custom Dashboards can incorporate a variety of data sources that can be used to build simple or complex Dashboards.

1. Select the Dashboard Designer shortcut from the Menu. The Dashboard Designer window will open.
2. Select the Data Source tab, then select New JAMS Data Source. The SaveDashboard File dialog will open.
3. Enter a name for the Dashboard file (.jdb) and click Save. The Add a Dashboard Data Source Wizard will open.
4. Enter a Data Source Name and use the dropdown to select a Data Source Type.

<table>
<thead>
<tr>
<th>Data Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>History</td>
<td>Provides JAMS history query that returns all JAMS Job Properties matching a specific criteria and time frame.</td>
</tr>
<tr>
<td>Completions by Severity</td>
<td>Includes an optimized version of the History data source that only returns completion count data. This is the preferred choice when retrieving historical data for a large number of jobs.</td>
</tr>
<tr>
<td>Queue Name</td>
<td>Contains all the properties from defined JAMS Queues.</td>
</tr>
<tr>
<td>Resource Name</td>
<td>Incorporates comprehensive information about each JAMS resource, including usage data.</td>
</tr>
</tbody>
</table>
Agent Name  Provides comprehensive information for all installed JAMS Agents.

PowerShell  Runs a PowerShell script. The returned PowerShell objects are accessible to all Dashboard items.

5. Click Next. The Parameters Wizard opens.
6. Enter parameters as desired. Note that parameter options are dependent upon the data source selected. Wildcards may be used in the parameter options.

7. With the parameters defined as desired, click the Finish button.

Using External Data Sources

Users may utilize non-JAMS data sources for dashboards, such as a database or XML files. To add an external data source, follow these steps:

1. On the Dashboard Designer, select the Data Source tab.
2. Select the New External Data Source.
   NOTE: is different than the New JAMS Data Source button used in the previous section.
3. The Create Data Source Wizard opens.
4. Select a Data Source Type from the available options, then click Next. (Database, Olap, CSV, Data extract)
5. Based on the Data Source Type, either Define the Database Connection, Define the OLAP cube connection, Select the CSV file, or Select the Data Extract, then click Next.

   ![Create Data Source Wizard](image)

   **Provider:** Microsoft SQL Server
   **Server name:** localhost
   **Authentication type:** Windows authentication
   **User name:**
   **Password:**
   **Database:**
   **Connection name:** localhost_Connection

   NOTE: Additional configuration settings may be required, based on the Data Source Type.
7. With all settings configured, click Finish to display the external data source in the dashboard designer.

Setting up a Dashboard Item
Once a Data Source has been defined, the next step is to set up the Dashboard item(s).

1. Begin this step by opening the Designer’s **Home** tab and select a Dashboard Item type on the Ribbon.

2. This action opens the Dashboard Elements page made up of three contiguous sections: **Data Source Browser**, **Data Items Pane** and **Preview Pane**. This is the display where you link (or bind) the data source created in the first step with the new Dashboard Item.

3. On the top of the Data Source Browser select an existing Data Source from the drop down menu.

4. As shown below, the selected Data Source opens displaying all associated **Data Fields**.

5. Drag the desired **Data Field** from the Data Source Browser and drop it onto the appropriate **container** field on the Data Items pane.

   ![Dashboard Elements Page](image)

   **Note**: you can also remove the data item by dragging it outside the Data Items pane.

6. Use the **Preview Pane** to confirm each selection.

   ![Preview Pane](image)

   **Note**: you can quickly change the Dashboard type by right-clicking in the Preview pane. This action opens the Dashboard context menu. Select the **Convert To** command submenu to view a listing of Dashboard types (e.g., Pivot, Grid, Chart, Pies, etc.)

7. You can insert additional Dashboard items by selecting another Dashboard type on the Designer’s **Home** tab ribbon.

   ![Dashboard Items Pane](image)

   **Note**: you can only use one Data Source per Dashboard Item.
8. As shown in step 3, open an existing Data Source from the Data Source Dropdown menu and drag and drop the desired Data Fields to the appropriate section on the Data Item Pane.

9. From the Design tab use the available tools to customize the new Dashboard item.

10. Click Save to complete the design process.

**Modifying Parameters**

Parameters pass information to Dashboard Items, allowing you to alter what is displayed.

**Changing an Existing Parameter**

To access the parameters settings click the Parameters button (icon with document and gears located on the top right portion of the Dashboard Design pane). This action opens the Parameters dialog.

**Creating New Parameters**

You can define new parameters within a Dashboard to further customize its display.

To add a parameter(s) to a Dashboard, open the Dashboard Designer.

1. On the Dashboard Designer’s Home tab click Open and navigate to the desired Dashboard (.jdb) file, typically located in the JAMS Installation directory (MVPSI/JAMS/Client).
2. On the Home tab, click the Parameters button. (located in the Dashboard section of the ribbon bar).
3. The Parameters Properties dialog opens. On the left panel select an existing parameter to modify its properties, or click the Add button to define a new parameter.
4. In the right panel, define the properties for the selected parameter.

5. Click the OK button when completed.

**Using the New Parameter**

The value of Dashboard parameters can be used for any JAMS Data Source Parameter. This is done using the syntax `$parameterName`. For example, if you had a parameter called `OffsetTime` you would reference the Dashboard parameter in a JAMS Completion by Severity data source by entering `$OffsetTime` into the wizard instead of hard coding a number into the Time Offset field.

```
Time Offset:

$OffsetTime

This is the offset from the current time to the desired time in the past from which to return data for the previous 24 hours. Leave the value at zero to start from the current time.
```

**Setting up a Drill-Down Display**

The Dashboard Designer includes a drill down function to help you create a more dynamic display with multiple levels of information.

The example below shows how selecting a single node on 24 hour line graph zooms the information view to a particular hour.
To create a drill down effect you must define multiple related data sets in the Arguments container field located on the Data Items pane. The Dashboard’s highest level data is entered on the first Data Item Container field and the “drill down” data is defined on the second.

**Note**: If three or more drill-down levels are needed, additional containers fields will appear on the Data Items pane.

### A Drill Down Example in 3 Steps

Use the following example to create a pie chart that displays JAMS Jobs that have executed over the past 24 hours by severity levels (e.g., Success, Warning, Error, Fatal and Informational).

**Step 1: Set up the Custom Dashboard Item**

To get started with this example, first create a Data Source.

1. From the Menu, open the **Dashboard Designer**.
2. On the **Data Source** tab select the **New JAMS Data Source** button located on the left end of the Ribbon.
3. The Add Dashboard File dialog opens.
4. On the first wizard page enter a Data Source Name. For this example, name it **JAMS History** and use the pull down menu to select a Data Source Type. Choose the **History** option, which tells JAMS to provide information on previously run Jobs. Select the **Next** button.
5. On the Parameters page, keep the default settings and click the **Finish** button.

**Step 2: Add a Custom Dashboard Item**

6. From the Dashboard Designer, select the **Home** tab. On the Ribbon, choose the **Pies** option.

7. This action opens the Dashboard Elements page. This is where you bind specific Data Items to create the pie chart.
8. From the Data Source Browser, use the pull-down menu to select the **JAMS History** Data Source you defined in step 4. This action displays all Data Fields associated with the data source.
9. On the Data Source Browser drag and drop the specific Data Fields to the appropriate Data Item Containers on the Data Items pane, as described below.
   - Find and then drag the FinalSeverity Data Field to the first Values Data Item Container.
   - Second, drag the same FinalSeverity item again, this time to the first Arguments container field. These data items will appear in the top level display as shown in the preview window below.
   - To add the data item for the drill down view, drag the JobName data item to the second Argument container.

Step 3: Define and configure the Drill down Property

10. On the Dashboard Designer, select the Data tab.
11. Activate the Arguments button. By enabling the Arguments option the Dashboard item treats the two arguments as distinct, but related data sets.
12. Click to activate (highlight) the Drill Down button. This allow the Dashboard Designer to interpret different sets of data between the two Arguments data item containers.

13. On the Preview pane, test the drill down by clicking on the Success portion of the pie chart. This actions should now detail all successful Jobs executed during the past 24 hours.
14. To move back to the first level pie chart display, right-click on the chart to display the **Drill up** context menu or alternatively choose the curved arrow **Drill up** icon.

15. Select the other elements of the pie chart to test the drill down function.

16. Optionally, open the **Design** tab to take advantage of the built-in formatting tools.

17. Choose the **Save** button to save the designed custom Dashboard.

### Using the Master Filter

The Master Filter is a Dashboard Designer feature that allows you to choose what data is displayed on a Dashboard Item. For example, when Master Filtering is enabled you can click a specific Dashboard Item to trigger updates to other items (chart, pies, gauges, etc.).

The Dashboard example below uses the Master Filter to control what data is displayed in the donut graph in the Preview panel. In this case, the donut graph is linked to the Agent Selection and Date Range items. Making changes to either element (e.g., highlighting one or more agents or changing the date range slider), immediately updates the graph to reflect the changing input data.

### Using Master Filter Modes

For each Dashboard Item, the Master Filter supports two selection modes: Multiple and Single.

#### Multiple Master Filter Mode

This mode allows you to select multiple elements within a Dashboard subpanel. For example, when choosing the
Agent Selection subpanel with the Multiple Master Filter enabled you can Control + click to highlight multiple agents as shown in the two screenshots below.

![Multiple Master Filter](image)

**Single Master Filter Mode**

Unlike the Multiple Master Filter, the Single Master Filter mode only allows you to select one element at a time within a selected Dashboard subpanel.

**Ignoring the Master Filter**

Dashboard items can be set to ignore the Master Filter entirely. To do this, choose a Dashboard Item and select the **Ignore Master Filters** button located on the Dashboard Designer’s Data tab.

For example, the Agent Selection sub panel, shown above, is set to ignore the master filter because it is the primary setting for that subpanel. In other words, you input an agent selection in order to update the donut graph, not the other way around.

**Enabling the Master Filter**

As described above, there are several Master Filter modes and settings, but there is no one place where you can actually view all these settings. Instead, each individual Dashboard Item must be selected within the Designer to view the Master Filter settings.

However, you can view the Master Filter state by hovering over the filter icon adjacent to the Dashboard title. This filter icon only appears when there is more than one criteria affecting the filter.

![Master Filter Icon](image)

The Dashboard Title is turned off by default on the primary Dashboard. To change this settings open the Designer and select the **Home** tab. On the far right side of the Ribbon select the **Title** button. On the Dashboard Title dialog, activate the **Visible** and **Show Master Filter state** checkboxes.
Finally, to make changes to a Dashboard’s Master Filter settings, open the Dashboard Designer.

1. On the Designer’s Home tab click the Open button.
2. Navigate to and select the desired Dashboard (.jdb) file.
3. This opens the multi-panel Dashboard Elements page.
4. Select the Data tab to view what, if any, Master Filter buttons are activated.
5. On the Preview pane, select a subpanel. Notice how the Data Item Container fields are updated to reflect the subpanel’s settings. This is where you define or modify how each Dashboard Item affects the Master Filter.

![Dashboard Designer Interface](image-url)
Creating New Report Templates and Customizing Existing Reports

You can modify existing reports or build reports from scratch using the report designer. Both options are described in more detail in the following subsections.

Once a new or customized report is created you can save the report definition to the JAMS system or your local machine. Either way, the report can be loaded later or viewed manually or automatically as part of any JAMS Job.

Creating a New Report Template from Scratch

1. From the JAMS Shortcuts menu, click Report Designer.

2. From the JAMS Shortcuts menu, click Report Composer.

3. From the Select Report Data Source dialog, select a data source.

4. Click OK. Each data source comes with pre-defined parameters with values that can be changed to produce a different report output.

5. Drag and drop control objects (field, label, table and shape, etc.) from the Tool Box to one of the Report Layout “bands” (top margin, detail and bottom margin).

6. Right-click a “band” to view its context menu. The context menu lets you modify, reorder, and define each “band” from its Properties panel.
7. The Report Explorer panel displays the current report structure using a tree format.

8. Use the following sections to make additional updates to the report:
   - The Field List panel displays a listing of the report’s data sources, allowing you to add new bound controls and binding existing controls. To add a new bound report control, click on the desired item in the Field List window and then drag and drop the item onto a report band on the report layout panel.
     
     **NOTE**: You can also right-click a Field List item and drag and drop it onto the report's Layout Panel. This action opens its context menu. From the menu list, select the field type desired.

   - The Property Grid panel provides a variety of options for formatting existing field controls.
   - The Group and Sort panel lets you create and define grouping fields instead of manually inserting Group Header and Footer bands.
   - Additional formatting tools are available on the Report Designer Ribbon Bar.

9. To output a partial or complete report, click the Print Preview or the HTML View tabs.

10. From the Ribbon Bar click Save or Save As to create a standalone report (.pmx) onto your local hard drive. All standalone report files can be accessed by clicking Open and selecting File from the Open JAMS Report dialog.

**Customizing an Existing Report**

The Report Designer also provides tools for modifying existing reports.

To access these tools:

1. In the Report Designer, click Open.
2. From the Open JAMS Report dialog, expand a category and select an existing report to modify.
3. Click OK. The Report Designer Window opens with the report’s layout displayed in the report layout panel.
4. Drag and drop a control object (field, label, table, and shape, etc.) from the Tool Box to one of the Report Layout pane’s component “bands”.
5. Refer to steps 4 - 10 in the previous subsection to complete the existing report design.
Manually Generating Reports

JAMS makes it easy to add reporting capabilities for either existing or new Jobs. The following topic summarizes how to create and generate Reports both manually and automatically.

Users may view all existing Reports from the JAMS Client by selecting the Report Viewer shortcut from the Menu.

To generate a JAMS Report:

1. From the JAMS Shortcuts menu, click Report Viewer.
2. In the Report Viewer, click Open to view any existing Reports. The Open JAMS Report window appears with a listing of categories containing Reports already built into JAMS.
3. Expand a category, and highlight one of the Reports.
4. Click OK. You may also open any stand-alone Reports by clicking the File button.
5. On the Parameters dialog, enter the properties for the specific Report type.
6. Click Submit to query the JAMS system and create the Report based on the entered parameters in the previous step.
7. After running the Report, you can export the Report in a variety of file formats. Click the Export/Export To command located on the right of the Report Viewer Ribbon Bar.

Using the Report Viewer is an easy way to manually run reports against the JAMS system. It is also possible to run the same Reports automatically. See the topic: Integrating Reports with Automated JAMS Jobs for additional details.
Integrating Reports with Automated JAMS Jobs

JAMS includes a JAMSReport Execution Method to streamline reporting by allowing users to create scheduled Jobs to send out their reports.

Setting Up a JAMS Report Job

To get started, select a default Report Job and open and modify its properties.

1. Select the Definitions shortcut from the menu and select the folder where the new Report Job should reside.
2. Click the Add button from the Control Bar to open the Add a New JAMS Job Definition dialog.
3. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select JAMSReport to create a JAMS Report Job.
4. By default, the full Job Definition dialog will open when the Job is initially saved. Click OK.
5. The Job Definition dialog will open.
6. Select the Source tab. The source tab for JAMSReport Jobs contains a variety of configurable properties to control the Report Output. Define the report properties as desired for the report this Job should generate.

<table>
<thead>
<tr>
<th>Report Job Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report Name</td>
<td>Select a predefined JAMS Report.</td>
</tr>
<tr>
<td>Export Path</td>
<td>Enter the location where the report should be saved. If the print queue property is defined as an email address, an export path is not required.</td>
</tr>
<tr>
<td>File Name</td>
<td>Enter the name for the report file.</td>
</tr>
<tr>
<td>Print Queue</td>
<td>Enter either a printer name to output the report, or a list of email addresses to send the report. Use commas or semi-colons to separate multiple email recipients.</td>
</tr>
<tr>
<td>Export Format</td>
<td>Use the dropdown to define a file format for the report. The supported formats include: PDF, HTML, MHT, CSV, RTF, Text, XLSX (Excel) or Print. Note that using the Print option will send the report output directly to the default printer.</td>
</tr>
<tr>
<td>Number of Copies</td>
<td>Set the number of copies to print.</td>
</tr>
<tr>
<td>Margins</td>
<td>Defines the margins of a report page (measured in hundredths of an inch). When setting this property, the value should be entered as &quot;Left,Right,Top,Bottom&quot;. For example, a 1&quot; top and bottom with .75&quot; side margins would be entered as &quot;75,75,100,100&quot;.</td>
</tr>
<tr>
<td>Paper Height</td>
<td>Define the height of the report page, measured in hundredths of an inch. This property can be enabled only if Paper Kind is set to Custom.</td>
</tr>
<tr>
<td>Paper Width</td>
<td>Define the width of the report page, measured in hundredths of an inch. This property can be enabled only if Paper Kind is set to Custom.</td>
</tr>
<tr>
<td>Orientation</td>
<td>Sets a value for the page orientation - landscape or portrait.</td>
</tr>
<tr>
<td>Paper Kind</td>
<td>Determines the type of paper for the report. If this is set to Custom, the printer paper is selected according to the property’s value, and its size is set to the Page Height and Page Width Paper Name property values.</td>
</tr>
<tr>
<td>Paper Name</td>
<td>Sets the name of the custom paper which is used in the output printer. This can only be set if the Paper Kind is set to Custom.</td>
</tr>
<tr>
<td>Printer Name</td>
<td>Defines the printer to use for outputting the report.</td>
</tr>
</tbody>
</table>
7. Define additional Elements, Parameters, Security, Properties, and Documentation as desired.
   
   **NOTE**: To schedule the report job, add a *Schedule Trigger Element* to the Job.

8. **Save and Close** the Job Definition.

   **NOTE**: If both the File Name and Print Queue properties are populated, the generated report is saved and forwarded to the designated recipient or group of recipients. However, if either field is left blank the default printer is used.

   **NOTE**: A Report Job may also be manually submitted at any time.
SQL Server Reporting Services (SSRS)

SQL Server Reporting Services (SSRS) is Microsoft’s server-based reporting platform that includes a variety of tools to help you create, manage and deliver reports throughout your organization. With Reporting Services you can define interactive, tabular, or free-form reports from relational, multidimensional or XML-based data sources. In addition, you can publish, access and schedule reports on-demand.

Reporting Services tools work hand-in-hand within the Microsoft Visual Studio environment and are fully integrated with SQL Server tools.

These reporting services also enable developers to integrate or extend data and report processing using custom applications, such as JAMS.

Scheduling SSRS within JAMS

SQL Server Reporting Services can be natively scheduled within JAMS by using the built-in SSRS Execution Method.

**NOTE:** When using the SSRS Execution Method, the **Source** of the Job is used to save reports to a physical location. An **Email Report** element may be used to Email the saved Report. In order to Email the Report, users must have an SMTP Server configured, and the user running the SSRS Job must have access to the network location where they want to save the report.

**NOTE:** SQL Reporting Services in JAMS works with SQL Server 2012 and later.

Creating a JAMS Job using the SSRS Execution Method

Users must first define a Job that can work with SSRS. This is done by creating a JAMS Job using the SSRS Execution Method, then setting up a JAMS Report for that Job.

1. In the JAMS Client, navigate to the **Definitions** section.
2. Select the desired folder for the SSRS Job, then ensure the Job Definitions tab is selected.
3. Click the **Add** button from the Control Bar to open the **Add a New JAMS Job Definition** dialog.
4. In the dialog, give the new Job a Name, Description (optional), and Execution Method. In this case, select **SSRS** to create an SSRS Job.
5. By default, the full Job Definition dialog will open when the Job is initially saved. Click **OK**.

6. The Job Definition dialog will open. Select the **Source** tab.
7. Enter the address for the SQL reporting server in the **Report Server** field. Once the Report Server is
properly specified, all accessible reports will display in the Report Path dropdown property. Select the desired report from the drop-down list.

8. With the Report Server and Path specified, set the Authentication Method JAMS will use when logging onto the server and, if applicable, the logon User.

9. Select a Report Format from the drop-down list.

10. Specify the Report Location (where a copy of the report will be saved).

11. Enter the Title for the saved report.

12. If the saved SSRS report should be e-mailed, navigate to the Elements tab and click the Add... button.

13. Expand the Result category, then select Email Report and click Next.

14. Define the Email Report Properties as desired for the Report. Multiple e-mail addresses may be used in the CC Address and To Address fields by sepearating them with a comma or semicolon.

15. When the Email Report element is defined as desired, click Finish.

16. Configure additional Elements, Properties, and Documentation as desired.

17. Save and Close the Job.
Monitors and Views
Using the Monitor

The Monitor enables you to view current Jobs in the Scheduler and provides access to comprehensive Job management tools. Use the following steps to open and manage current Jobs.

**Accessing the Monitor**

To get started, select the **Monitor** shortcut to open the Monitor Current Jobs view.

![Monitor Current Jobs view](image)

**Customizing the Monitor View Using the Column Chooser**

With the Column Chooser tool, you can add, remove, and move columns in the Monitor View, as well as the History and Folder Views. To use this tool, follow the steps below:

1. Right-click on any column header within a View to open its context menu.
2. Select the **Column Chooser** command to display all column titles available in the Customization window.

![Column Chooser](image)
3. To add a column to the current View, drag and drop a column title to the left or right of an existing column.

**NOTE:** Double-clicking a column title appends the new column to the right within the current View.

4. To remove an existing column, drag-and-drop its header out of position until an “X” appears and then release the mouse button.

5. To move any column, drag-and-drop the column header into the desired position within the current View.

**Monitor View Options**

With the Monitor view, you can view and manage Jobs, Sequences, and Workflows. The Monitor view provides buttons in the Control Bar at the top of the screen to do the following:

- Cancel - Cancel the selected Entry.
- Release - Release an Entry from one or more requirements that are preventing it from running.
- Hold - Put a manual hold on an Entry. The Entry will not start until the hold is manually released.
- Reschedule - Specify a new date and time to submit the selected Entry.
- Submit - Submit the selected Entry.
- Properties - View the Properties for the selected Entry.
- Show Detail - View the Monitor Detail dialog.
- Query - Select options to update the type of Entries that are displayed in the Monitor.
- Refresh - Update the list of entries that are displayed.

You can also right-click a Job, Sequence, or Workflow to access a menu that displays similar options as the buttons at the top of the screen. The displayed options will vary based on the type of Entry selected. For example, Sequences and Workflows have additional options for Show Sequence or Show Workflow to let you view the Sequence or Workflow tab within the Monitor Detail dialog. This allows you to easily view the selected Sequence or Workflow.

**Querying the Monitor**

Within the Monitor View, you can query and filter the Entries that are displayed, allowing you to customize the view.

1. Open the Monitor view.
2. Click the **Query** Button.
3. On the **Filters** tab, select one or more of the following options:
   - Automatically Add/Remove Entries
   - Show Related Entries
   - Show Executing
   - Show Completed
   - Show Pending
   - Show Timed
   - Show Only My Entries
4. Click **OK**. The Monitor view is updated.

**NOTE:** If you clear the "Show Only My Entries" option, you will see all Entries in the Monitor. An error message will be displayed if you try to view an Entry that you cannot access.

**Monitor View Icons**

Every Job contained in the Monitor view includes an icon showing the current state of each entry as listed in the table below.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Job Status Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="icon" /></td>
<td>Waiting for a dependency to complete</td>
</tr>
</tbody>
</table>
Currently executing

Halted, usually due to a failed Sequence Job

On hold by the user

Pending—waiting for a Sequence Job step to begin executing

Status flagged as containing an error

Status flagged as fatal

Status flagged as "informational", based on specific source code values or the Execution Method

Status flagged as successful

Status flagged as containing a warning

Timed or scheduled

Managing a Job

To manage a Job currently in the Schedule, open the Monitor Detail dialog. To open this dialog, you can do one of the following:
  - Right-click on a Job from the Monitor Current Jobs view and select Show Detail.
  - Double-click the Job name.
  - Select the Job and then click the Show Detail button in the Control Bar.
Each Job listed in the Scheduler displays the Properties as shown in the above screenshot and detailed in the table below.

### Job Properties

<table>
<thead>
<tr>
<th>Job Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td>Displays Folder, Submit, and Job status information.</td>
</tr>
<tr>
<td>Properties</td>
<td>The Properties of the executing Job.</td>
</tr>
<tr>
<td>Statistics</td>
<td>Shows times, execution statistics, and related charts.</td>
</tr>
<tr>
<td>Log File</td>
<td>Lists all OS generated logs.</td>
</tr>
<tr>
<td>Schedule</td>
<td>The configured Items of the Executing Jobs.</td>
</tr>
<tr>
<td>Parameters</td>
<td>Details parameter names and values associated with the Job.</td>
</tr>
<tr>
<td>Audit Trail</td>
<td>Lists audit trail items for the Job, including message, comment, user name, local and UTC audit times.</td>
</tr>
<tr>
<td>History</td>
<td>Displays historical Job Execution information for the selected Job.</td>
</tr>
<tr>
<td>Documentation</td>
<td>The assembled documentation for the given Job and any Documentation inherited from parent folders.</td>
</tr>
</tbody>
</table>
Working with Projected Schedules

The Projected Schedule displays a graphical view of pending JAMS Jobs and Sequences. Users can view all scheduled Jobs or filter a subset of Jobs over a specified time range.

To generate a projected schedule for all scheduled Jobs, click on the Projected Schedule Shortcut in the Shortcut Bar. The resulting display defaults to the current 24 hour timeline view.

Filtering the Projected Schedule Results

To filter a specific group of Jobs and Sequences, use the Query button located in the Control Bar.

1. In the Projected Schedule view, select the Query button from the Control Bar.
2. The JAMS Projected Schedule Query dialog opens. Select a Folder Name or enter a Job Name.
   NOTE: Entering multiple values is not allowed, but users may use wildcards.
3. Enter a Time Range using the data and time selector.
4. Activate the appropriate checkboxes to refine your query.
   Use Current Schedule: when enabled, pulls in the current schedule from the Monitor View.
   Include Planned: filters all scheduled Jobs including those with the “Automatically Submit?” option unchecked.
   Include Repeated: when checked, this option includes all Job repetitions in the schedule.
5. When the query criteria is defined as desired, click the Ok button to rebuild the projected schedule based on the query.

Modifying the Timeframe Display

After generating a Projected Schedule, users can modify their results by selecting different views: Day, Work Week, Week, Month and Timeline.

- To change the timeframe, right-click within the Projected Schedule display to view the context menu.
- Make a selection from the Change View To submenu or use the Go to Today or Go To Date options to modify the timeframe display. Depending on the selected view, you can also adjust the Time Scales and Captions.

Note: to zoom in and out of any Projected Schedule display, hold down Ctrl key while scrolling the mouse wheel.

Color Flags

The Projected Schedule display uses specific color flags when a Job’s execution time deviates from its set parameters as detailed below:

- Items (Jobs or Sequences) in red are expected to run beyond their designated SLA (Service Level Agreement) time.
- Jobs and Sequences displaying orange have run past their predefined end time.
- Items in yellow have run beyond their original schedule time.

Note: double-clicking or right-clicking a Job within the display will open its settings window.

Installing the Projected Schedule Shortcut after Upgrading JAMS

JAMS V6.1.281 added the Projected Schedule function to the Shortcut Bar. For new JAMS installations, the Projected Schedule Shortcut appears by default. However, if you are running an earlier version of JAMS you must first upgrade to JAMS 6.1 or later and then add the Projected Schedule Shortcut using the manual process described below.
1. Right-click the Shortcut Bar and select **Add Shortcut** command from the context menu.
2. In the Add a Shortcut wizard, define the Name and Title (e.g., Projected Schedule) and choose an icon. Click the **Next** button.
3. On the View page, choose the Projected Schedule option from the list and click **Next**.
4. On the following page, define the Properties for the Projected Schedule view. For each view you can set different parameters for displaying the schedule (e.g., by Folder or Job name). Select the JAMS Server from the dropdown list, or choose the **Active Server** option to automatically generate a Projected Schedule using any accessible JAMS server.
   - **Use Current Schedule**: pulls in the current schedule from the Monitor View which is controlled from the “ScheduleAdvance” option in Configuration Settings.
   - **Include Planned**: filters all scheduled Jobs including those with the “Automatically Submit?” option unchecked.
   - **Include Repeated**: allows the Projected Schedule to include Jobs submitted based on their recurrence schedule.
5. On the last page of the wizard, confirm your choices. At this point you can create additional Views or change the View properties by selecting the Query button from the Ribbon Bar. Click the **Finish** button to close the wizard.
6. The Projected Schedule Shortcut should now appear in the Shortcut list.
Resources, and Queues
Using Resources

JAMS Resources are powerful tools that can help you allocate how batch Jobs are run. Resources let you define the necessary requirements for all Jobs and Sequences, allowing JAMS to determine which Jobs can run concurrently.

Resources can be assigned to Folders, Jobs, and Sequences. The Resource itself is defined by setting a specific quantity. Jobs and Sequences typically utilize a portion of this quantity when they run.

Resource Allocation—Automatic and Manual Configuration

JAMS can automatically control how each Job or Sequence acquires and releases the Resources assigned to them. The Resource quantity available can be manually increased or decreased at any time. You make the modifications in one place and it is available to all Jobs and Sequences that need it.

As your batch processes move from one operating system to another JAMS continues to enforce the assigned Resources requirements.

Resource Behavior

Resources have some unique characteristics that are listed below:

- Jobs and Sequences inherit the Resources that have been defined in their respective Folders.
- Jobs and Sequences can actually reduce or increase the inherited Resource requirements.
- Jobs and Sequences can add their own Resource requirements.
- Sufficient Resources must be available for a Job to run.
- During the Sequence execution process you can hold Resources across for all component Jobs.
- Resources can be specific to a node or server.
- Resources can be used on any JAMS supported operating system.
- The available Resource quantity can be increased or decreased at any time.

Adding a New Resource Definition

1. To define a Resource quantity, select the Resources shortcut located on the Configuration group.

2. The Resources Definitions View opens.

3. From the Ribbon, choose the Add button to access the Resource Definition wizard.

4. On the first page of the wizard enter a name and optional description for the new Resource. Choose the Next button.

5. On the following page set the Resource’s Quantity Available value. Click the Next button and then choose Finish to complete the Resource definition process. The new Resource now appears in the Resource Definitions View.

Modifying an Existing Resource Definition

1. Open the Resources shortcut.

2. From the Resource Definitions View, double-click the selected Resource to open its definitions window.

3. Use the Available tab to change the Quantity Available setting.

4. Click the Security tab to choose the access level for the Resource. See the following subsection (Setting Resource Security) for a description of each access level.

5. Click the Close button to save your changes.

For a complete listing and description of each Resource property, refer to the Resource Properties topic.

Determining How Much Resource to Assign a Job or Sequence

When manually modifying Resources, as described above, how do you know how much or how little to assign to a particular Job or Sequence?

JAMS provides several ways to determine if a Resource is working above or below its effective allocation.
Use the Dashboard

Both the Advanced and Current Load Balancing Dashboards incorporate a graphical summary of information and trends within the JAMS environment. Specifically, both include a Resources subpanel to determine the load availability for each available Resource.

Insert a Quantity in Use Column

To see what Resources are currently active, use the Column Chooser to insert a Quantity in Use column in the Resource Definitions View.

Access the Monitor

Open the Monitor to view the status of all Jobs and Sequences. If a Job or Sequence has stalled or flags a ‘Waiting for Resources’ description, consider increasing the Resource allocation.

Resources in Use

You can query the JAMS SQL database to determine what Resource definitions are currently in use.

```
SELECT M.resource_name,
       M.qty_available,
       IsNull((select SUM(U.qty_in_use)
               from dbo.ResourceInUse as U
               where U.resource_id = M.resource_id), 0) as InUse
FROM dbo.ResourceM as M
```

Adding an Existing Resource to a Job, Sequence or Folder

1. Open a Job or Sequence definition by double-clicking the selecting the item.
2. Select the Elements tab. Open the Resources subsection and choose the Add button.
3. This actions opens the Resource Search window. Double-click an existing Resource from the Resource Search list to associate it to the current Job.
4. Save the Job by clicking the Close button at the upper right of the definitions window.

Deleting a Resource

To delete a Resource, open the Resource Definitions View and highlight the Resource to be removed. From the Ribbon Bar choose the Delete button.

Note: you must have the proper level of access in order add or delete a Resource.

Inheriting a Resource Definition via Folders

Jobs and Sequences can inherit the Resources defined within their Folder. This inheritance feature simplifies the assignment of Resources to any Job or Sequence.

To assign a new Resource definition: open the Folder’s definition by right-clicking and choosing Properties > Resources. Follow the steps described in the previous section: Adding an Existing Resource to a Job, Sequence, or Folder.
Resource Properties

Resources let you allocate the necessary requirements to help run your Jobs and Sequences more efficiently. A Resource is defined by setting a specific numeric quantity. Jobs and Sequences typically utilize a portion of this quantity when they execute. JAMS automatically allocates the total quantity of resources assigned to a given Job or Sequence. In addition, your command procedures can also acquire and release quantities of Resources enabling you to define more flexible and complex scheduling scenarios.

Existing Resources contain several properties that can be modified when necessary. Resource properties are organized in three tabs: **Resource**, **Available** and **Security** which are detailed in the sections below.

To access a Resource definition:

1. Select the **Resources** shortcut located on the **Configuration** group.
2. The **Resources Definitions View** opens.
3. From the **Resource Definitions View**, double-click the selected Resource to open its definitions window.

**Resource tab**

**Resource Name**
This property includes the unique identifier for the Resource.

**Description**
The Description property can provide a more detailed explanation for the Resource.

**Last Change**
The Last Change property displays the date and time the Resource was last modified.

**Available tab**

**Quantity Available**
This property shows the available quantity for the current Resource.

**Node Specific**
If enabled, this property measures the Resource by node. If unchecked, the Resource is defined by the original quantity available.

**Node Name**
This property displays the name of the node or nodes.

**Quantity Available**
This displays the available Resource quantity for the specific node.

**Security tab**
The Resources Security tab defines the level of access for this Resource. Essentially, this is an Access Control List with one to many Access Control Entries (ACE). Each ACE can specify the following rights:

- **Acquire**: allows the user to obtain a portion or all of the Resource's quantity allocation.
- **Change**: permits the modification of this Resource definition provided the user has Change access in Resource Definitions.
- **Control**: allows users to modify the Resource’s Access Control List.
- **Delete**: permits deletion of the Resource definition provided that the user also has Delete access for Resource Definitions.
- **Inquire**: users have the ability to inquire into the Resource definition as long as they have Inquire access to Resource Definitions.
Using Queues

A JAMS Queue provides a temporary holding place for pending Jobs and Sequences on a first-in-first-out basis. Queues can be assigned to Jobs, Sequences, and Folders.

Queues are either in a started or stopped state and can be set to run multiple jobs simultaneously. To enhance load balancing capabilities you can also specify what node a Queue is started on.

**Note:** Defining and using Queues in JAMS is optional.

Defining a New Queue

To define a new Queue, follow the steps below:

1. On the JAMS Client select the **Queues** shortcut.
2. This action opens the Queues Definitions View.
3. Choose the green **Add** button on the JAMS Ribbon to open the JAMS Queue Definition Wizard.
4. On the Queue Name page, enter a unique name and description for the Queue. Click the **Next** button.
5. Choose **Finish** to complete the Queue definition process.

Modifying Queue Properties

Queues can be modified by accessing its definitions window.

1. To open a Queue's property definition, double-click a Queue from the Definitions View list.
2. A Queue definition is organized into two tabs. The **Name** tab displays a Queue’s basic **Name**, **Description** and **Last Changed** properties. The **Status** tab allows you to adjust the Queue’s start/stop status and Job Limit setting.
   - On the Status tab the **Status** checkbox indicates if the Queue has currently started. If true, Jobs submitted to the Queue are available to run.
   - Use the **Job Limit** adjustment to define the maximum number of Jobs that can run concurrently.
   - From the **Started On** section, you can add, delete or modify the Agent nodes where the Queue originates. This can include a single node or a list of nodes.

**Note:** An Agent node name in a Job definition overrides an Agent node name in a Queue definition. However, the Job still runs on the assigned Queue and adheres to any Job Limit settings defined within the Queue.

Manually Stopping and Starting a Queue

When a Job is waiting for a Queue Job limit slot it cannot execute. However, you can manually start a Queue by right-clicking the Queue’s context menu from the Queue Definitions View and choosing the **Start** command. Conversely, you can select the **Stop** command to manually halt the Queue.

Queues and Resources

While there are similarities between Batch Queues and Resources there are some significant differences that are spelled out below.

A Batch Queue is defined by the number of concurrent Job executions and Job limitation settings.

When submitting a Job, a Batch Queue can be adjusted, but a Resource cannot.

A Resource definition is based on the total number of unit quantities available. As these Jobs run they consume the available units that are released upon completion.

Resource requirements can be defined at the System, Folder and the Job level. These values are then added together to define the total units required for the Job to begin executing.
JAMS Scheduler Features
JAMS Scheduler Services

The JAMS Scheduler is essentially the heart of the JAMS system and is responsible for keeping track of the status of all Jobs, firing triggers, checking dependencies, and communicating that status back to the JAMS Client.

While you interact with the JAMS Scheduler through the JAMS Client, the JAMS Scheduler can be configured to operate in a centralized or decentralized fashion. In other words, you can choose to run a JAMS Scheduler on many different machines to spread the management tasks and eliminate single points of failure.

When first installing JAMS, the Scheduler creates three Windows Services that are detailed below.

- JAMS Scheduler Service
- JAMS Executor Service (JAMS Agent Service)
- JAMS Server Service

JAMS Scheduler Service

This key component provides background Services to allow for scheduling, maintaining and executing JAMS Jobs, Sequences, Tasks and Scripts on all types of systems available on your network.

The Scheduler Service prepares Scripts to run, takes action if required during Job execution or when a Job completes. A more detailed explanation of the Scheduler Service is described below.

Before a Job can run, the Scheduler Service determines if . . .

- A Job's Dependencies are satisfied.
- It can execute the Job's Pre-Check Job, if one is specified.
- The Job's Queue is started and there is a slot available.
- The Job's Resources are available.
- The Job's Scheduling Window is open.
- The Job requires parsing (based on its Execution Method). If so, then the Job's source is parsed and substitutions occurs for Parameter and Variable values.

While a Job is running, the Scheduler Service. . .

- Reports the Job's execution details to all JAMS Clients monitoring the Schedule.
- Performs notification if the Job runs longer than its set max limit.
- Uses the Job's defined action if the Scheduling Window closes during execution.

Once a Job completes, the Scheduler Service . . .

- Reports the Job's completion to all JAMS Clients monitoring the schedule.
- Records the Job's run instance in JAMS History, which is available for query from the JAMS Client.
- Performs notification, if required.
- Sends reports, if required.
- Determines if the Job's completion satisfies Dependencies for other Jobs in the schedule.
- Determines if the Job's completion satisfies Trigger Elements so that Triggers can fire.
- Makes sure the Job's Resources are released.

Executor Service

As its name implies, the JAMS Executor Service is responsible for the actual execution of any given Job.

Jobs are prepared to run by the JAMS Scheduler. Once the Scheduler Service determines the Job is ready to run, the Executor . . .

1. Is notified and begins executing the Job.
2. Then writes output generated from the Script to the Job's log file.
3. And notifies the Scheduler when the Job has completed.

If the Job is set to run on a different machine, JAMS Agent Services (an extension of the Executor Service) contacts the Agent on the remote machine and directs it to run the Job.

The JAMS Scheduler cannot function without access to the JAMS Database. However, JAMS is designed to be resilient. All Job execution functions are handled by the JAMS Executor Service; so if the JAMS Scheduler Service fails, all Job execution information remains secure.

Server Service

The JAMS Server Service provides middle-tier services to the JAMS Client. This includes all client components including: GUI, Powershell, .NET Class Library, and Web Services.

The JAMS Server Service is primarily responsible for maintaining database access for one to many JAMS Clients.

While the Server Service is not involved in the execution of Jobs, many Jobs may utilize the JAMS Powershell client, which requires this Service.

Troubleshooting JAMS Services

Reviewing the Event Log and .log Files

All the described JAMS Services creates a log file in the JAMS installation directory (Program Files/MVPSI/JAMS). This log file is named ServiceName.log (i.e., JAMSScheduler.log) and is reset every Sunday. The previous week's log files are then renamed to ServiceNameArchive.log (i.e., JAMSSchedulerArchive.log).

Each Service writes serious errors to the Windows Event log. When troubleshooting JAMS, you should check both the event log and the aforementioned .log files.

Stopping and Restarting a Service

If you suspect there is a problem with JAMS and want to restart the JAMS Services, begin with the JAMS Scheduler Service. The JAMS Scheduler Service does the most work and restarting it is the least disruptive to the system.

Restarting the JAMS Scheduler Service will not cause any Jobs to fail or any Job completion information to be lost. However, when the JAMS Scheduler Service is stopped, new Jobs cannot execute.

As a next step, you can stop the JAMS Server Service. This can be done without losing any Job execution information. However, JAMS Clients cannot function while the JAMS Server Service is down.

As a last resort, users can try stopping the JAMS Executor Service. Stopping the Executor Service is not recommended, as it doesn't usually resolve most problems. When putting this Service on hold, all completion information for executing Jobs will be lost, likely causing some Jobs to fail.

When doing SQL Service maintenance, there is no need to stop the JAMS Executor Service, since it does not access the JAMS Database.

Service Accounts

JAMS Services are set to run using the LocalSystem account. However, if this needs to change this to a Windows Domain based account, JAMS Support recommends leaving the Executor and JAMS Agent Services running under LocalSystem as these Services do not need to access the database or network, but they do require privileges associated with the LocalSystem account.

You can use the Service Control application to change the account that the JAMS Scheduler and JAMS Server services run under as this could be important when controlling network and database access.

When changing the account you may also need to adjust the security settings on:

- C:\Program Files\MVPSI\JAMS\Scheduler folder
- C:\Program Files\MVPSI\JAMS\Scheduler\JAMSScheduler.log
- MSMQ JAMSRequests Private queue
- SQL Server
- JAMS Database

For the MSMQ JAMSRequests private queue, you must make sure to modify the security on the queue to grant the domain account full access to the queue. This may require you to "Take Ownership" of the MSMQ queue.
The following Local Security access should also be granted for the domain based account:

- Log on as a Batch job.
- Log on as a Service.
- Adjust memory quotas for a process.
- Bypass traverse checking.
- Replace a process level token.

If the domain based user account is not in the administrators group, create an Active Directory Group and add the user to the group, and then include the following:

```xml
<add key="AuthorizedGroup" value="Domain\YourGroup" />
```

in the Common.config file located in the Program Files\MVPSI\JAMS\Scheduler directory.
Configuring the JAMS Scheduler

There are two ways to configure the JAMS Scheduler. The first is to modify the Configuration Settings using the JAMS Client. The other is to edit the .exe.config file. Both methods are described below.

Configuration Setting on the JAMS Client

Configuration Settings define and maintain many of JAMS' system-wide options. Generally, users define these options after installing JAMS, but usually don't update them on a frequent basis after that.

JAMS comes installed with several Scheduler-specific configuration settings. These allow users to define how far a schedule can extend into the future or the Scheduler's maximum downtime hours. Other installed Configuration Settings include specific organizational scheduling definitions, such as “Is Monday usually a workday?”.

To view, create or modify these settings select the Configuration Shortcut. The Configuration Settings View opens listing the available Configurations.

1. To modify a Configuration Setting, double-click an item from the Configuration Settings View.

2. The Setting properties Name tab window opens, which includes three parameters: Name, Description and Last Changed. On this tab, only the Description parameter can be modified.

   Note: Only on the Name tab can the Description parameter can be modified. To modify these parameters you must define a new Configuration Setting, as described in the previous subsection.

3. The Value tab allows users to change the Configuration’s Value settings. In this example (“Is Monday usually a holiday?”), users with the proper security access can modify the value from “True” to “False”. This modifies Monday’s typical status to a non-workday. The Data Type parameter, located just above, cannot be modified except during the definition process, as described in the previous section.

   Note: When a Configuration setting is modified it usually take affect immediately. However, in some cases you may need to restart the JAMS Scheduler Service before the change takes effect.

Editing .exe config

Configuration settings are also found in one of several .config files located in the JAMS installation directory, Program Files\MVPSI\JAMS\Scheduler. The filename of each configuration file is:

- Common.config
- JAMS Scheduler.exe.config
- JAMS Executor.exe.config
- JAMS Server.exe.config

Common.config is shared by the other .config files and contains all user-specific settings.

Note: You should only change one of the other .config files under the direction of JAMS Technical Support unless you understand the implications of altering these files.

The Common.config file is preserved when upgrading JAMS; the other files are overwritten each time JAMS is upgraded to a new version.

You can change settings in these configuration files at any time; however, in most cases, these settings are only loaded when the service starts.
Logon as Batch

JAMS Jobs run as batch processes using Windows. As a requirement, these batch jobs must have the Log on as Batch user rights for the Windows account that runs them. If running the Job on an agent, users must also have Log on as a Batch privileges within the local security of the agent machine.

Keep in mind that the agent Job must have appropriate access to perform whatever operation the particular Job is doing; for example, if a copy process to a specific folder is included, you must have access to that folder.

Granting Rights using Windows 2008/2012

From Windows 2008/2012, you can grant the Log on as a Batch Job rights at the domain or the local level. To manage rights under Windows 2008/2012, use the Local Security Policy or Domain Security Policy Administrator utilities.

Managing Rights on Member Servers and Workstations

To manage user rights on member servers and workstations, choose the Select Domain... option from the User menu. Enter `\\machinename` in the Domain field and click OK to be connected to the specific machine. From here you can manage user rights on that machine.
Upgrading JAMS
Upgrading JAMS Using a Primary-Failover Environment

Use the steps below to upgrade JAMS if you have a Primary-Failover Server Environment.

1. Upgrade the Secondary (Passive) JAMS Server. This will upgrade the JAMS Server as well as the JAMS SQL Database.

2. After the Secondary JAMS Server is successfully upgraded, open the PowerShell command line and use `Set-JAMSFailoverStatus` to set the Secondary JAMS Server as the active server. The Secondary JAMS Server takes over scheduling and executing Jobs from the Primary JAMS Server.

   **NOTE:** Replace instances of `MyPrimaryServerName` and `MySecondaryServerName` with the actual names of the Primary and Secondary Servers in the environment.

   ```powershell
   Import-Module JAMS
   ### List current status of the Secondary
   Get-JAMSFailoverStatus -Server MySecondaryServerName
   ### Set the Secondary to running mode
   Set-JAMSFailoverStatus -Server MySecondaryServerName -Active
   ### Double check the Primary status - should read: "Backup"
   ```

3. Wait for all of the Jobs on the Primary (local) JAMS Server to complete.

   **NOTE:** In a JAMS HA Configuration, the best practice is to have all Jobs execute on an Agent to prevent Job loss during a failover.

4. Upgrade the Primary JAMS Server that is now in passive mode.

5. After the Primary JAMS Server is successfully upgraded, open the PowerShell command line and use `Set-JAMSFailoverStatus` to set the Primary JAMS Server as the active server.

   **NOTE:** Replace instances of `MyPrimaryServerName` and `MySecondaryServerName` with the actual names of the Primary and Secondary Servers in the environment.

   ```powershell
   Import-Module JAMS
   ### List current status of the Primary
   Get-JAMSFailoverStatus -Server MyPrimaryServerName
   ### Set the Primary to running mode
   Set-JAMSFailoverStatus -Server MyPrimaryServerName -Active
   ### Double check the Secondary status - should read: "Backup"
   ```

6. Verify the Primary JAMS Server is now active and the Secondary JAMS Server is now passive by doing the following:

   a. From the JAMS Client, click the **About** icon and select **JAMS Servers**.

   b. Click the **Status** tab.

   c. Ensure the server is listed in the correct mode (Running or Backup Mode). You may need to scroll down to see all of the information.
Upgrading from JAMS 7.x to 7.x

JAMS can be upgraded directly over an existing installation of an earlier 7.X JAMS Client and/or JAMS Scheduler. Specifically, users may run a new 7.X installation executable to update JAMS 7.X using the current database without disturbing existing Properties, Configuration settings, or Jobs.

Use the steps below to prepare and complete a JAMS 7.X upgrade.

**NOTE:** JAMS V7.1.557 includes enhancements to the Submit Job Task that have limited functionality when you edit a Sequence with an older JAMS Client. For example, some properties of the new Submit Job Task cannot be seen or modified with an older JAMS Client. You can continue to use older JAMS Clients for other features of JAMS, but we recommend upgrading to V7.1.557 or higher to edit Submit Job Tasks in a Sequence.

**NOTE:** When you upgrade to V7.1.57X or higher, you need to update the JAMS Agents to V7.1.57X or higher if you are using a Sequence with any of the following:
- An override property.
- A binding on an inherited property.
- A binding on a Credential.

**NOTE:** JAMS Agents must be running JAMS V7.0.1441 or later to be fully compatible with later JAMS releases. Ensure all Agents are upgraded to V7.0.1441 or later before upgrading the JAMS Scheduler.

Scheduling the Upgrade

- Schedule a Maintenance Window, and then wait for any executing Jobs to finish before starting an upgrade. Maintenance Windows may be created using the Resource definition method. No Jobs should be running when the upgrade process starts.
- Make a backup of your JAMS database just before the upgrade. Typically, there should be no issues with the upgrade, but it is always good to have the backup if needed.

Downloading the Latest JAMS Release

Download the latest JAMS release from Product [Download Links](#) page.

**NOTE:** To access the Product Download Links page, you must be registered for a JAMS account and logged in to [JAMS Support](#).

Upgrading JAMS 7.x

During the upgrade, the JAMS installer stops the appropriate JAMS services and then restarts them after the upgrade is complete.

1. Right-click the JAMS installer that you downloaded and select Run as administrator.
2. On the Welcome page, click Next.
3. On the License Agreement page, review the license agreement.
4. Select the checkbox next to I accept this license agreement.
5. Click Next.
6. On the Core Components page, keep the items that are already selected. **Clearing any of these items will uninstall them.** You can also select any additional Components that are needed.

7. Click Next.
8. On the Integration Packs page, keep any Integration Packs that are already selected. Clearing any of these items will uninstall them. You can also select any additional Integrations that are needed.

9. Click Next.
10. If displayed, review the values on the **Web Client Details** page. This screen is displayed only if the Web Client is being upgraded.

11. Click **Next**.
12. On the Install page, review the summary of what will be installed.
13. Click Install.
14. Click Close when the Installation Complete message is displayed.
Make sure the user that you are logged in as on the JAMS Scheduler machine has full admin rights to the JAMS SQL database. This is important to allow the database schema to be properly updated. Typically, giving the user dbcreator or sysadmin rights takes care of all necessary permission issues. These permissions are only needed during the upgrade and can be removed after the upgrade.

If the JAMS Services were configured to run as a Domain account before the install took place, the JAMS Server and JAMS Scheduler service must be set back to run as the domain account. Windows will reset the JAMS Services to run as Local System when the services are updated.

Once the JAMS Scheduler Server has been upgraded, the same installation executable can be used to upgrade any remote JAMS Client installation. To upgrade the JAMS Client, select only the JAMS Client option on the Core Components page of the install wizard. Note that having only the JAMS Client option selected while upgrading a remote client will not uninstall other options listed on the Core Components page from the Schedule Server.
If you are leveraging only the Web Client, you are not required to perform any installation or upgrade after a Server upgrade. The Web Client view will update if the Web Client was previously installed or selected in step 6.

Redeploying JAMS Agents

JAMS Agents can be upgraded by redeploying the Agents from the JAMS Client's Agent Definitions Shortcut. JAMS Agents can be automatically deployed or manually deployed.
NOTE: JAMS users attempting to upgrade from a previous major version to JAMS 7.X should contact the JAMS Support Team.