

Inject job



Inject job is a Switch default app that injects a job into the flow when triggered by an incoming job or by various time-based events. The injected job can be selected dynamically from a repository of jobs, it can be a fixed job, or it can be the incoming job itself.

The properties for each outgoing connection specify what needs to happen for that connection (independently of other connections). For example, to inject a job from a repository based on the metadata of some incoming job while keeping the original job around, setup one connection to receive the incoming job, and a second connection to receive the job from the repository.

Keywords

Keywords can be used with the search function above the Elements pane.

The keywords for the **Inject job** element are:

- trigger
- timer

Connections

Inject job supports an optional incoming connection and allows any number of outgoing connections. It offers additional properties on each of its outgoing connections.

Properties


Property	Description
Element type	The flow element type: Inject job. This property is useful to identify renamed flow elements. It cannot be changed.
Name	The name of the flow element displayed in the canvas.
Description	A description of the flow element displayed in the canvas. This description is also shown in the tooltip that appears when moving your cursor over the flow element.

Outgoing connection properties

The following properties are provided for each of the outgoing connections in addition to the basic connection properties

Property	Description
Action	Specifies the action for the connection: <ul style="list-style-type: none"> • Fail: the connection will only be used to move failed jobs • Inject: the connection will inject jobs
	<i>The following subordinate properties are present if Action is set to Inject.</i>
Trigger	Specifies the type of event that triggers injection for this connection: <ul style="list-style-type: none"> • Incoming job: an injection is triggered for each incoming job. • Time since previous job: an injection is triggered when the specified period of time passed since the last job arrived. • Time of day: an injection is triggered at the specified time(s) during the day.
	<i>The following subordinate properties are present if Trigger is set to Time since previous job.</i>
Unit	Selects the unit for the subsequent property: Minutes, Hours, Days.
Delay	The trigger delay in the units indicated by the previous property.
Repeat	If set to Yes, a new injection is triggered repeatedly at the specified interval as long as no jobs arrive; if set to no, an injection is triggered only once (until a new arrival resets the timer).
	<i>The following subordinate properties are present if Trigger is set to Time of day.</i>
Time (hh:mm)	The time of day when the first injection is triggered. The specified time value is used only for the first injection after a flow was activated. The subsequent injections are controlled by the <i>Repeat unit</i> and <i>Repeat delay</i> properties.
Repeat unit	Selects the unit for the subsequent property: Minutes, Hours.
Repeat delay	The repeat delay in the units indicated by the previous property; a value of zero means "do not repeat".
Inject	Specifies the scheme for determining the injected job for this connection: <ul style="list-style-type: none"> • Incoming job: inject the incoming job (or a copy, if more than one connection specify this scheme); if the injection is not triggered by an incoming job this behaves as if <i>Dummy job</i> was selected.

Property	Description
	<ul style="list-style-type: none"> • Dummy job: inject an empty file called <i>dummy.txt</i> (generated by the tool). • Specific job: inject the job specified in a subordinate property through its absolute file or folder path. • Job repository: inject the job specified in a subordinate property through an absolute folder path (the repository containing the jobs) and a job name (the name of the job residing in the repository).
	<i>The following subordinate properties are present if Inject is set to Specific job.</i>
<i>Job path</i>	The absolute file or folder path of the job to be injected (if a fixed file path is selected, the selected file is included in the exported flow).
<i>Delete after injection</i>	If set to Yes or if the condition is true, the original injection job is deleted after being injected in the outgoing connection. Otherwise, the original job is unaffected.
	<i>The following subordinate properties are present if Inject is set to Job repository.</i>
<i>Job repository</i>	The absolute folder path of the repository containing the jobs.
<i>Job name</i>	The file or folder name of the job residing in the repository (possibly including the filename extension, depending on the value of the subsequent property).
<i>Extension</i>	The filename extension for the job residing in the repository (without the period); an empty value means that the filename extension is included in the <i>Job name</i> property (or that there is no filename extension).
<i>Delete after injection</i>	If set to Yes or if the condition is true, the original injection job is deleted after being injected in the outgoing connection. Otherwise, the original job is unaffected.
	<i>The following subordinate property is present if Inject is set to anything but NOT Incoming job.</i>
Name job after	<p>Specifies how to determine the name of the jobs injected into this connection:</p> <ul style="list-style-type: none"> • Incoming job: use the name of the incoming job; if the injection is not triggered by an incoming job this behaves as if "Injection job" was selected. • Injection job: use the name of the job selected for injection.

Property	Description
	<ul style="list-style-type: none">• Custom: use a custom job name (optionally specified using variables and script expressions).
	<i>The following subordinate property is present if Name job after is set to Custom.</i>
Name	Specifies a custom name for the job injected into this connection  Note: Job variables (in text with variables) and objects (in script expressions) can only be used if the injection was triggered by an incoming job!