mx (https://sprintr.home.mendix.com)

Buzz (https://sprintr.home.mendix.com/link/home)Apps (https://sprintr.home.mendix.com/link/myprojects)People (https://developer.mendixcloud.com/link/people)

Community (https://developers.mendix.com/)   App Store (https://appstore.home.mendix.com)Academy (https://gettingstarted.mendixcloud.com)

Blogs (https://developers.mendix.com/spotlight/)

Jobs (https://developers.mendixcloud.com/jobs/)Docs (https://docs.mendix.com)

Model Share (https://modelshare.mendix.com/)

MVP Program (https://developer.mendixcloud.com/link/mvp)

(https://sprintr.home.mendix.com/link/not

Leaderboards (https://developer.mendixcloud.com/link/leaderboards)

Our Partners (https://developer.mendixcloud.com/link/partneroverview)

Docs (/) **/** Studio Pro 8 Guide (/refguide/) **/** App Modeling (/refguide/modeling) **/** Application Logic (/refguide/application-logic)

Mendix Shop (https://shop.mendix.com)

Search documentation

Microflows

Last update: Oct 22, 2019

✎ Edit (https://github.com/mendix/docs/blob/development/content/refguide/microflows.md)

1 Introduction

> This page is an overview of all the elements that can be used in a microflow. For the properties of the microflow itself, see Microflow Properties (microflow).

Microflows allow you to express the logic of your application. A microflow can perform actions such as creating and updating objects, showing pages and making choices. It is a visual way of expressing what traditionally ends up in textual program code.

2 Keyboard Support

The microflow editor offers keyboard support for navigating and manipulating microflows. The following table shows the keys that can be used.

Key	Effect
Arrow Keys	Select nearby element (activity, event, loop or parameter) in the direction of the arrow.

Key	Effect
Enter	Edit the properties of the selected element.
F2	Rename the item returned by the selected element.
Shift+F2 or just start typing	Edit the caption of the selected element.
Ctrl+Arrow Keys	Move the selected element in the direction of the arrow.
Tab	If a loop is selected, the first element inside the loop will be selected.
Shift+Tab	If an element inside a loop is selected, the loop itself will be selected.
Home	Select the start event.
End	Cycle through the end events.
Context-menu key or Shift-F10	Open the context-menu for the currently selected element.

3 Notation

The graphical notation of microflows is based on the Business Process Model and Notation (https://en.wikipedia.org/wiki/Business_Process_Model_and_Notation) (BPMN). BPMN is a standardized graphical notation for drawing business processes in a workflow.

A microflow is composed of elements. Below is a categorized overview of all elements. The following categories are used:

- Events represent start and endpoints of a microflow and special operations in a loop.
- Flows form the connection between elements.
- Decisions deal with making choices and merging different paths again.
- Activities are the actions that are executed in a microflow.
- Artifacts provide the microflow with input and allow comments to be made.
- Error handlers can be set on an activity, decision, or loop to define how to handle an error.

4 Events

Events represent start and endpoints of a microflow and special operations in a loop.

Graphic	Name	Description
⬤ (start-event)	Start Event (start-event)	A start event is the starting point of the microflow. A microflow can only have one start event.
⬤ (end-event)	End Event (end-event)	An end event defines the location where the microflow will stop. Depending on the return type of the microflow in some cases a value must be specified. There can be more than one end event.
⬤ (attachments/16713739/16843951.png)	Error Event (error-event)	An error event defines a location where the microflow will stop and throw an error that occurred earlier. If you call a microflow, you may want to know whether any errors occurred within the microflow or not.

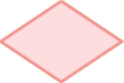| Graphic	Name	Description
(continue-event)	Continue Event (continue-event)	A continue event is used to stop the current iteration of a loop and continue with the next iteration. Please note that continue events can only be used inside a Loop (loop).
(break-event)	Break Event (break-event)	A break event is used to stop iterating over the list of objects and continue with the rest of the flow after the loop. Please note that break events can only be used inside a Loop (loop).

5 Flows

Flows form the connection between elements.

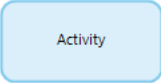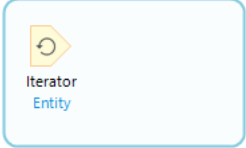| Graphic	Name	Description
(sequence-flow)	Sequence Flow (sequence-flow)	A sequence flow is an arrow that links events, activities, decisions, and merges with each other. Together they define the order of execution within a microflow.
(annotation#annotation-flow)	Annotation flow (annotation#annotation-flow)	An association is a connection that can be used to connect an annotation to another element.

6 Decisions

Decisions deal with making choices and merging different paths again..

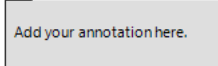| Graphic	Name	Description
(decision)	Decision (decision)	A decision makes a decision based on a condition and follows one and only one of the outgoing flows.
Note: There is no parallel execution in microflows.		
(object-type-decision)	Object Type Decision (object-type-decision)	An object type decision is an element that makes a choice based on the specialization (entities) of the selected object. You can give the specialized object a name using a Cast Object (cast-object) action.
(merge)	Merge (merge)	A merge can be used to combine multiple sequence flows into one. If a choice is made in a microflow and afterwards some common work needs to be done, you can combine the two (or more) paths using a merge.

7 Activities

Activities are the actions that are executed in a microflow.

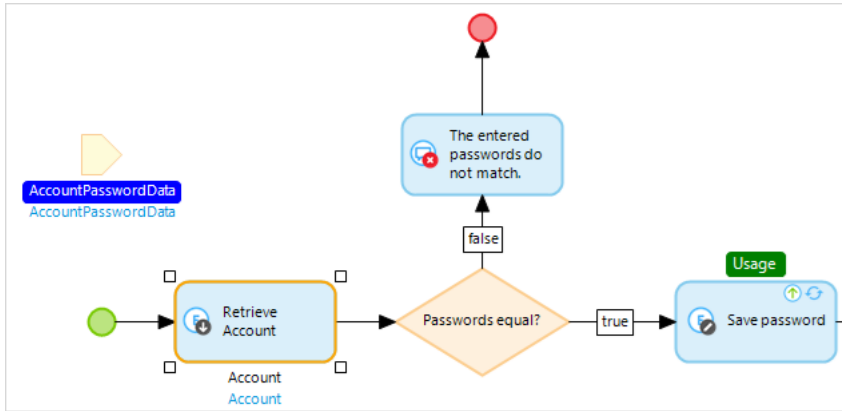| Graphic	Name	Description
Activity (activities)	Activity (activities)	An activity can be used to execute a specific action in a microflow.
Iterator Entity (loop)	Loop (loop)	A looped activity is used to iterate over a list of objects. For every object the flow inside the looped activity is executed. A looped activity can contain all elements used in microflows, with the exception of start and stop events. The flow starts at the first element with no incoming flows.

8 Artifacts

Artifacts provide the microflow with input and allow comments to be made.

Graphic	Name	Description
EntityList List of Entity (parameter)	Parameter (parameter)	A parameter is data that serves as input for the microflow. Parameters are filled at the location from where the microflow is triggered.
Add your annotation here. (annotation)	Annotation (annotation)	An annotation is an element that can be used to put comments in a microflow.

9 Item Usages

Studio Pro visualizes which items are used by the selected element(s). It does this by showing the used items in white text on a blue background. Conversely, elements that use the item(s) returned by the selected element(s) are marked with the word 'Usage' in white text on a green background.

In the example below, the parameter **AccountPasswordData** is highlighted because it is used in the selected activity (**Retrieve Account**). And the activity **Save password** has a **Usage** label because it uses the object returned by **Retrieve Account**.

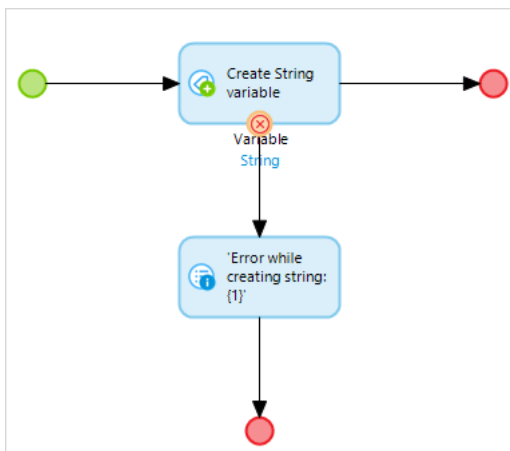(attachments/16713739/16843950.png)

10 Errors

When an error occurs in a microflow, all changes that have been made to objects are rolled back and the microflow is aborted. Optionally, you can handle errors (/howto/logic-business-rules/set-up-error-handling) in the microflow itself by configuring different error handling settings. You can even inspect the details of the error by looking at the predefined objects `$latestError` and `$latestSoapFault`.

10.1 Error Handlers

An error handler can be set on an activity, decision, or loop. On an activity or decision, it gives you three options:

- Rollback (default)
- Custom with rollback
- Custom without rollback

For the latter two options you can draw an additional flow from the block and mark this flow as the error handler flow. When selecting 'Custom with rollback' it will trigger this path when the error occurs and still rollback your objects afterwards. The 'Custom without rollback' option does not rollback the objects. After you selected a flow as the error handler it will show this as in the following image.
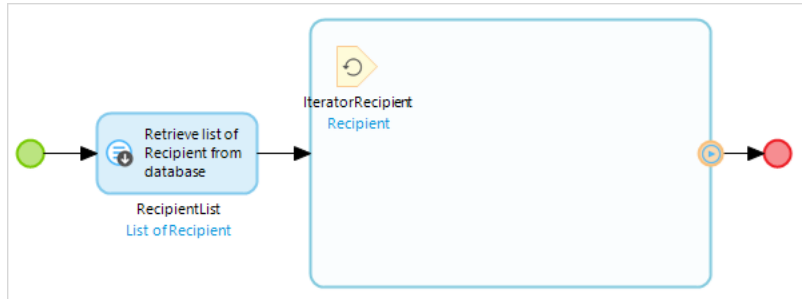

(attachments/819203/918247.png)

On a loop you get two options:

- Rollback (default)
- Continue

The continue option means that when an error occurs, the loop will simply continue to the next iteration. It will show as a continue icon on the exit flow of the loop.

(attachments/819203/918246.png)

10.2 Inspecting Errors

When an error occurs inside a microflow, under the hood a Java exception is raised that contains information about the error that occurred. Inside a custom error handler (as in, after an error handling flow), you can inspect the type of this Java exception as well as several other properties. Every microflow contains two predefined error objects, `$latestError` and `$latestSoapFault` . `$latestError` is an object of entity System.Error, while `$latestSoapFault` is an object of entity System.SoapFault, which is a specialization of System.Error.

In a custom error handler that is executed after an error occurs, `$latestError` is set to an object containing information about the error that occurred. If the error is a SOAP fault (an error that occurs as a result of a web service call), `$latestSoapFault` is set to an object that contains more specific information about the SOAP fault. Otherwise, `$latestSoapFault` is `empty` .

> You can determine whether an error was a SOAP fault by checking `$latestSoapFault` for `empty` .

The following table shows the attributes of System.Error and System.SoapFault.

Entity	Attribute	Type	Description
System.Error	ErrorType	String	The Java exception type of the error that occurred.
System.Error	Message	String	The message of the Java exception.
System.Error	Stacktrace	String	The stacktrace of the Java exception.
System.SoapFault	Code	String	The Code element of the SOAP fault.
System.SoapFault	Reason	String	The Reason element of the SOAP fault.
System.SoapFault	Node	String	The Node element of the SOAP fault.
System.SoapFault	Role	String	The Role element of the SOAP fault.
System.SoapFault	Detail	String	The Detail element of the SOAP fault.

Click here (http://www.w3.org/TR/soap12-part1/#soapfault) for more information on SOAP faults.

> In microflows that apply entity access, it is not possible to inspect the attributes of error objects for security reasons. You can pass the error object to a sub-microflow that does not apply entity access and inspect the attributes there.

11 Microflow Debugging

If you want to see what happens while a microflow is executing, you can use the microflow debugger. See the following how-tos:

- Debugging Microflows (/howto/monitoring-troubleshooting/debug-microflows)
- Debugging Microflows Remotely (/howto/monitoring-troubleshooting/debug-microflows-remotely)

Want to contribute to our documentation? Start here! (/developerportal/community-tools/contribute-to-the-mendix-documentation)

Table of contents

Buzz (https://sprintr.home.mendix.com/link/home)

Apps (https://sprintr.home.mendix.com/link/myprojects)

People (https://developer.mendixcloud.com/link/people)

App Store (https://appstore.home.mendix.com/)

Community

Blogs (https://developers.mendix.com/spotlight/)

Jobs (https://developers.mendix.com/jobs/)

Model Share (https://modelshare.mendix.com/)

MVP Program
(https://developer.mendixcloud.com/link/mvp)

Leaderboards
(https://developer.mendixcloud.com/link/leaderboards)

Our Partners
(https://developer.mendixcloud.com/link/partneroverview)

Mendix Shop (https://shop.mendix.com)

Academy (https://gettingstarted.mendixcloud.com)

Learning Paths
(https://gettingstarted.mendixcloud.com/link/path)

Modules
(https://gettingstarted.mendixcloud.com/link/module)

Webinars
(https://gettingstarted.mendixcloud.com/link/webinar)

Classroom
(https://gettingstarted.mendixcloud.com/link/classroom)

Certifications
(https://gettingstarted.mendixcloud.com/link/certification)

Forum (https://forum.mendixcloud.com)

Forum Questions
(https://forum.mendixcloud.com/link/questions)

Forum Ideas
(https://forum.mendixcloud.com/link/ideas)

Docs

Refe

How-

Relea

(http

API 8

(http

mxsc

Low-

(http

guide