

Requirements and Constraints on BPMN

BPMN Requirements and Supported BPMN Elements

The BPMN diagrams should comply to the following rules:

- Each Activity (Tasks, Events) must belong to exactly one process and at max to one pool. This implies that if it belongs to a pool it must not belong to a process.
- Transitions must not cross pools (BPMN rule).

All BPMN elements supported by the import are listed below.

BPMN Element	Comment
Pools & Lanes	Pools are supported and mapped to UML State Machines communicating via messages.
Tasks	All tasks are mapped to UML. It is important to distinguish between User Tasks and all other tasks that can be executed automatically (Script Task, Service Task, Receive Task, etc.).
Start / End Events	All start/end events are mapped to UML.
Intermediate Events	All intermediate messages are mapped to UML events.
Sequence Flows	All sequence flows are mapped to UML transitions.
Message Flows	Can be used in the diagram but are ignored when being imported.
Gateways	Or-Gateways are not supported yet, all other gateways are mapped to UML state machine elements.
Subprocesses	Subprocesses are mapped to separate UML State Machines.
Compensation	Limited support: Non-nested compensation associations are supported.
Associations and Data Objects	Can be used in the diagram but are ignored when being imported.
Process properties	Can be used in the diagram but are ignored when being imported.

On this Page:

- [BPMN Requirements and Supported BPMN Elements](#)
- [Constraints on BPMN when Modeled in MagicDraw](#)
- [Role of BPMN, Prerequisites for BPMN Modeling](#)

Constraints on BPMN when Modeled in MagicDraw

- Not all feasible BPMN diagrams are executable, thus two stereotypes were introduced to mark executable BPMN fragments:
 - All executable processes must get the stereotype `<<BPMNExecutableProcess>>`.
 - If the process contains pools, then all executable pools must have the stereotype `<<BPMNExecutablePool>>`. In this case, the stereotype `<<BPMNExecutableProcess>>` is optional.
 - If the process contains pools, tasks must be contained either in an executable- or in a non-executable pool otherwise the importer will complain.
- All BPMN model elements require a name - with exception of flows and start events. We recommend to give short but meaningful names.

Role of BPMN, Prerequisites for BPMN Modeling

There is some severe critic on the purpose of BPMN ([Reviewing BPMN 2.0](#), [The seven fallacies of BPMN](#)). Nevertheless, more and more people are using BPMN from sketching business processes to defining process architectures. It is even possible to see it as software specification language for architects.

However, the prerequisite that BPMN diagrams are usable as base (constraint) for executable UML models is:

BPMN pools *must* correspond to systems and/or human roles. One or more of these systems must be the BPMN platform (i.e. the E2E Bridge).

This requirement is crucial, because it defines what the BPMN engine can control and what it cannot. For example, many real live business processes involve tasks distributed over different organizations (even companies) and systems. A central BPMN- or workflow engine cannot generally control all involved parties. For instance, it is not possible to control a sequence of tasks executed in a backend system (e.g. SAP) or tasks solely executed by humans.