

Prozesse mit BPMN managen und gestalten Praxisbuch mit Lexikon, Aufgaben und Lösungen

» Hier geht's
direkt
zum Buch

DIE LESEPROBE

A

Adhoc-Unterprozess

Ein Adhoc-Unterprozess enthält wie der [eingebettete Unterprozess](#) einen zwar eigenständigen, aber vom aufrufenden Prozess abhängigen Prozess. Beim Adhoc-Unterprozess ist die Reihenfolge der [Aktivitäten](#) nicht oder nur teilweise definiert. Er wird durch die Symbole in [Bild 1](#) dargestellt.

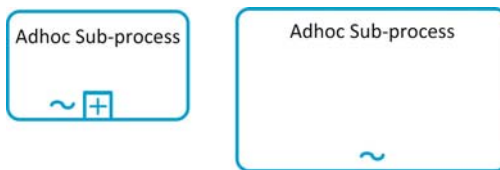


Bild 1 Symbole für den Adhoc-Unterprozess reduziert und erweitert

Der Adhoc-Unterprozess erhält im Symbol zur Kennzeichnung eine Tilde. Es stehen wie bei allen Unterprozessen die reduzierte (collapsed) und erweiterte (expanded) Form (siehe [Reduzierter/Erweiterter Unterprozess](#)) zur Verfügung. Es ist nur ein eingeschränkter Satz von Elementen erlaubt:

- In jedem Fall: Aktivitäten
- Bedarfsweise: Gateways, Zwischenereignisse, Datenobjekte, Sequenzflüsse, Verbindungen, Datenverbindungen, Nachrichtenflüsse
- Nicht erlaubt: Startereignis, Endereignis, Konversationen, Choreografie-Aktivitäten

Die Aktivitäten werden nicht durch Sequenzflüsse miteinander verbunden, wenn die Reihenfolge der Bearbeitung beliebig ist. Wenn sie verbunden sind, muss die Folgeaktivität in jedem Fall, aber nicht zwingend unmittelbar, ausgeführt werden. Die Endebedingung des Unterprozesses kann nicht durch einen Sequenzfluss definiert werden, sondern durch das Attribut *completionCondition*. Dieses Attribut sollte im Diagramm angezeigt werden.

1. Beispiel: Design Thinking

Design Thinking ist ein kreativer Prozess, der aus den nachfolgenden Aufgaben besteht:

- Verstehen
- Beobachten
- Sichtweise definieren
- Ideen finden

- Prototypen entwickeln
- Testen

Die Aufgaben werden ohne bestimmte Reihenfolge ausgeführt. Nur die Aufgaben „Prototypen entwickeln“ und „Testen“ müssen nacheinander ausgeführt werden. Daraus ergibt sich das in [Bild 2](#) gezeigte Modell.

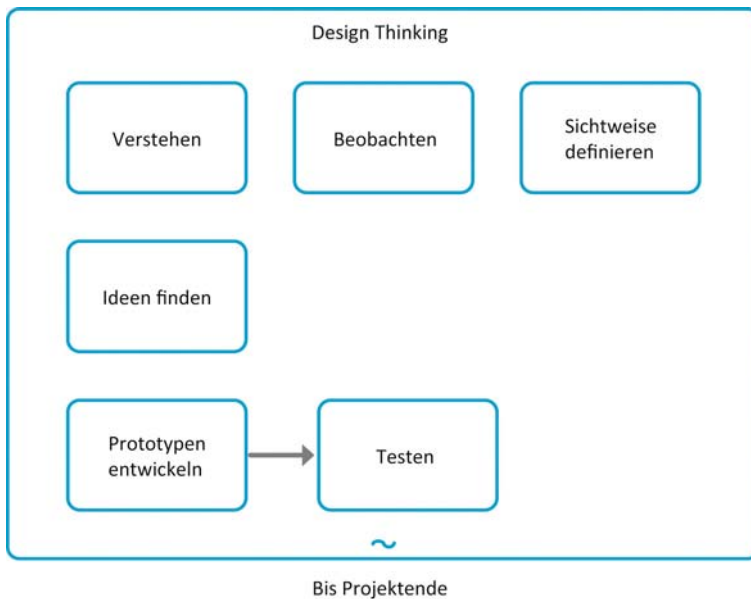


Bild 2 Design-Thinking-Prozess modelliert als Adhoc-Unterprozess

2. Beispiel: Hotelsuche

Das Vorgehen bei der Hotelsuche in einem Internet-Reiseportal ist häufig je nach Zweck der Reise unterschiedlich. Es ist nicht festgelegt, ob die Hotelauswahl zunächst über den Termin eingeschränkt wird, oder zunächst das Hotel nach Preis, Ort und Ausstattung gesucht wird. Der Unterprozess wird verlassen, wenn ein Hotel gefunden wurde (Endebedingung „Bis Hotel gefunden“) oder über das angeheftete Ereignis „Nichts Passendes gefunden“ ([Bild 3](#)).

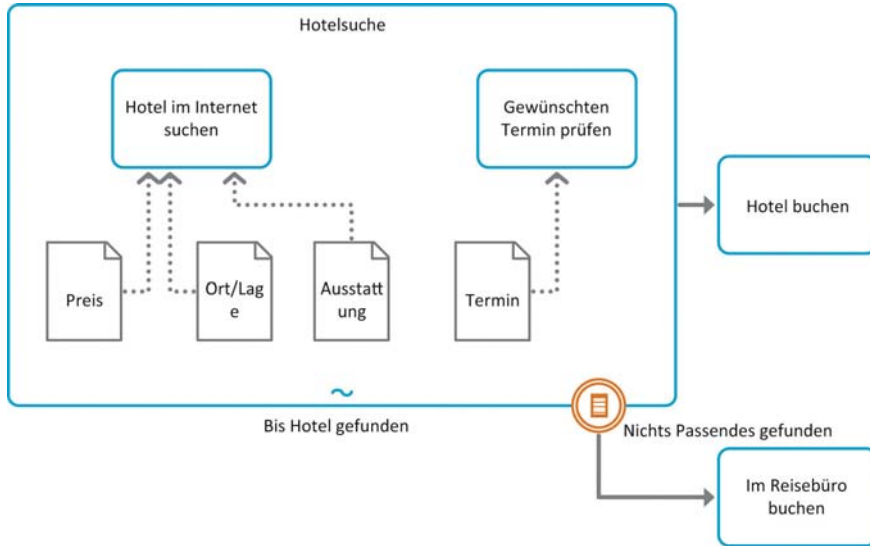


Bild 3 Hotelsuche im Internet als Adhoc-Unterprozess mit teilweise nicht festgelegter Reihenfolge der Aufgaben

Aktivität

Eine Aktivität (*activity*) ist ein allgemeiner Begriff für die Arbeit, die innerhalb eines Prozesses verrichtet wird. Zu den Aktivitäten gehört die [Aufgabe \(task\)](#), der [Unterprozess \(sub-process\)](#) und die [Aufrufaktivität \(call activity\)](#). Die Aktivität gehört zu den [BPMN-Elementen](#) der Kategorie [Flussobjekte](#), wie [Ereignisse](#) und [Gateways](#). Die Symbole sind in [Bild 4](#) dargestellt.

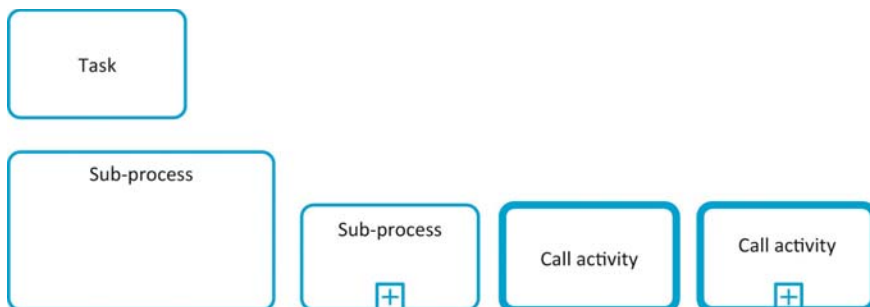


Bild 4 Verschiedene Symbole für die Aktivität

Unterprozesse und Aufrufaktivitäten stehen in der erweiterten Form und in der reduzierten Form zur Verfügung. Das Rechteck der erweiterten Form kann beliebig vergrößert werden, um den Unterprozess oder den von der Aufrufaktivität aufgerufenen Prozess innerhalb (*inline*) des Symbols darzustellen.

Angeheftetes Ereignis

Angeheftete Ereignisse sind [Zwischenereignisse](#), die am Rand von [Aktivitäten](#) (Aufgaben und Unterprozessen) „angeheftet“ (*attached to boundary*) werden. Sie ermöglichen die Modellierung von Ausnahmebedingungen der Aufgaben oder reagieren auf Ereignisse innerhalb des Unterprozesses. Sie sind immer „empfangend“. Es wird zwischen unterbrechenden (*interrupting*) und nicht unterbrechenden (*non interrupting*) Zwischenereignissen unterschieden. Bei unterbrechenden Zwischenereignissen wird die Ausführung der Aktivität unterbrochen, sobald das Ereignis aktiv wird und der Prozess über den vom Zwischenereignis ausgehenden [Ausnahmesequenzfluss](#) fortgesetzt wird. Im nicht unterbrechenden Fall wird die Ausführung fortgesetzt, und es wird ein weiterer parallel laufender Ausnahmesequenzfluss ausgelöst (ein weiterer [Token](#) generiert). Es können mehrere Zwischenereignisse an eine Aktivität angeheftet werden.

Angeheftete Zwischenereignisse können wie folgt typisiert werden:

- Nachricht
- Timer
- Bedingung
- Signal
- Fehler (immer unterbrechend)
- Eskalation
- Kompensation (immer unterbrechend)
- Mehrfach
- Parallel-Mehrfach

In [Bild 5](#) sind alle möglichen Fälle dargestellt. Die Typen werden unter dem Stichwort [Typen von Zwischenereignissen](#) im Einzelnen beschrieben. Dort finden sich neben den Prozessen in [Bild 6](#) und [Bild 7](#) auch weitere Beispiele.

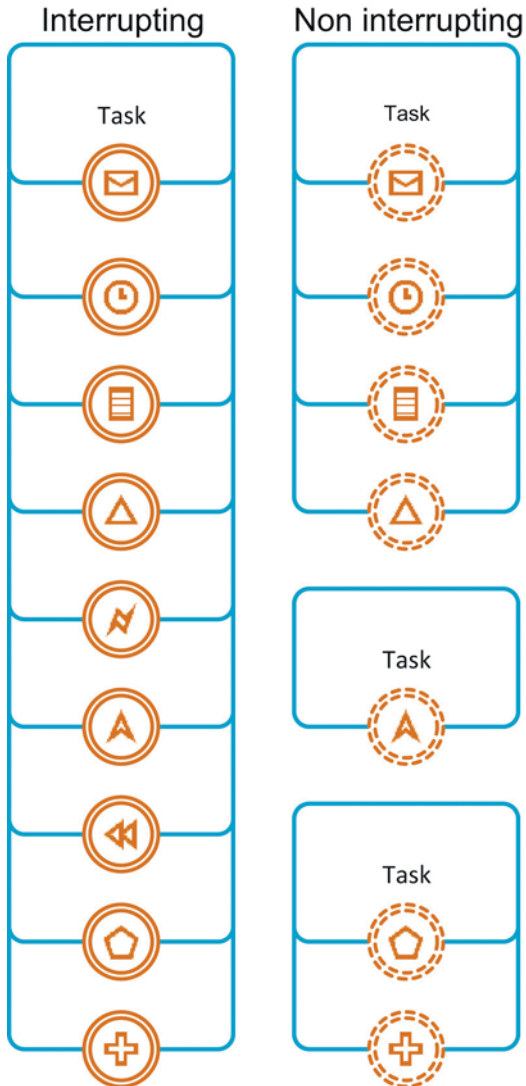


Bild 5 Angeheftete Ereignisse (unterbrechend/nicht unterbrechend)

1. Beispiel Rechnungsprüfung

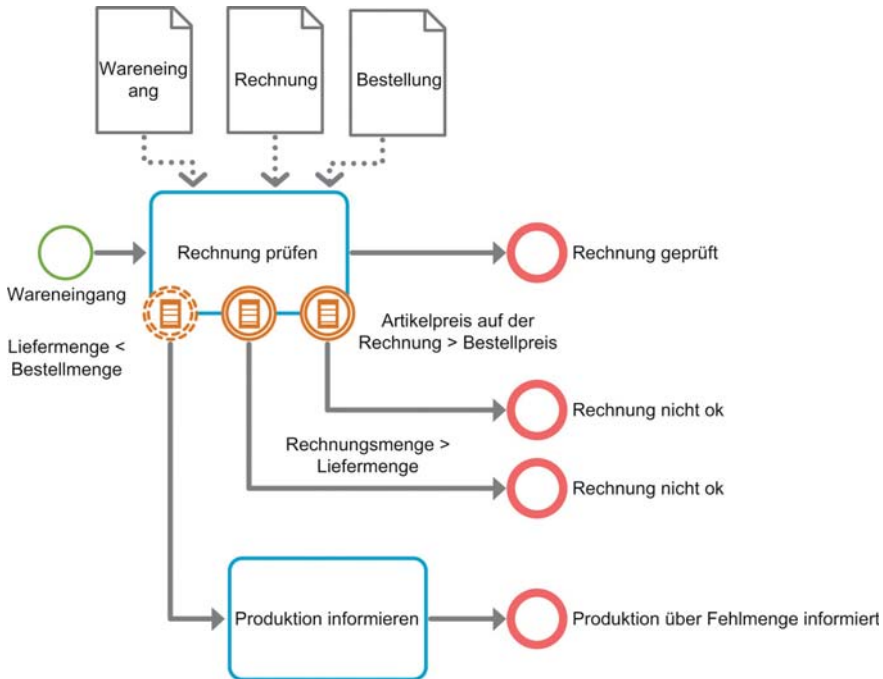


Bild 6 Rechnungsprüfungsprozess mit angehefteten Ereignissen

Der in [Bild 6](#) gezeigte Rechnungsprüfungsprozess beinhaltet eine zentrale Aufgabe „Rechnung prüfen“, in der die Rechnung gegen Wareneingang und Bestellung geprüft wird. Die verschiedenen möglichen Fälle werden durch angeheftete Zwischenereignisse modelliert. Wenn der Artikelpreis auf der Rechnung größer als der Bestellpreis ist oder die in der Rechnung ausgewiesene Menge größer als die Liefermenge ist, wird die Aufgabe durch Bedingungs-Zwischenereignisse abgebrochen und der Prozess endet mit dem Endereignis „Rechnung nicht ok“. Wenn die Liefermenge kleiner als die Bestellmenge ist, wird durch ein nicht unterbrechendes Bedingungs-Zwischenereignis die Produktion über die Fehlmenge informiert, die Rechnungsprüfung läuft aber weiter.

2. Beispiel Fehlerhandling

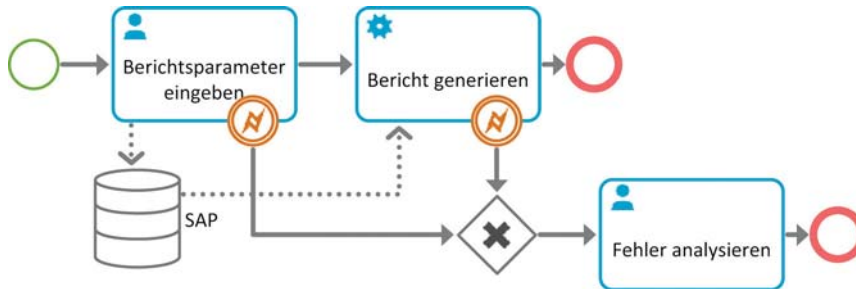


Bild 7 Fehlerhandling bei der Berichtsgenerierung

Es wird ein Bericht automatisch durch die Service-Aufgabe „Bericht generieren“ generiert (Bild 7). Die Berichtsparameter werden vorher in der Aufgabe „Berichtsparameter eingeben“ erfasst. Beide Aufgaben besitzen ein angeheftetes Fehler-Zwischenereignis, das im Fehlerfall die Aufgabe abbricht und eine Fehleranalyse veranlasst.

Artefakte

Artefakte (*artifacts*) wird eine Kategorie der Elemente bezeichnet, durch die der Modelltyp [Kollaborationsdiagramm](#) beschrieben wird. Der Standard kennt die Artefakte [Gruppe](#) (*group*) und [Textzusatz](#) (*text annotation*), deren Symbole in [Bild 8](#) zu sehen sind.

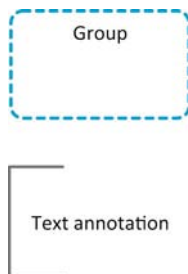


Bild 8 Symbole für die Artefakte „Gruppe“ und „Textzusatz“

Durch diese Element können zusätzliche Informationen zum Prozess hinzugefügt werden, die den eigentlichen Ablauf mit der [Instanziierung](#) nicht betreffen. Die [BPMN-Spezifikation](#) erlaubt, zusätzliche Artefakte über [Erweiterungen](#) (*extensibility*) zu definieren.

Attribute

Attribute (*attributes*) sind Daten, die neben der grafischen Darstellung des Symbols weitere Eigenschaften (*properties*) von [Elementen](#) beschreiben, wie z. B. der Name eines Gateways. Sie definieren neben dem grafischen Symbol die [Semantik](#) der Elemente. Sie werden teilweise auch zur Automatisierung der Prozesse benötigt. Attribute können bezogen auf die [BPMN-Konformität](#) zwingend (*required*), obligatorisch (*mandatory*) oder optional sein und sind Bestandteil des Datenmodells, das hinter der grafischen Darstellung als Diagramm liegt.

Zwingend heißt, dass in jedem Fall ein Wert erfasst oder vorbelegt oder generiert wird.

Das Attribut „ID“ ist zwingend und muss zusätzlich ein eindeutiger Schlüssel sein, über den das Element oder der Prozess eindeutig identifiziert wird. Es wird vom Modellierungstool generiert. Eindeutigkeit bezieht sich mindestens auf alle Modelle der Datenbank eines Tools, damit die Elemente modellübergreifend referenziert werden können. Um die [Austauschbarkeit](#) der Modelle toolübergreifend zu garantieren, verwenden in der Zwischenzeit nahezu alle Tools die sogenannte GUID (*globally unique ID*), die aufgrund des Generierungsalgorithmus praktisch global eindeutig ist (theoretisch könnte es Dubletten geben).

In [Bild 9](#) ist die GUID im Format 8-4-4-4-12 in der XML-Notation des Austauschformats zu sehen.

Das Attribut „name“ gehört ebenfalls zu den zwingenden, allerdings nicht notwendigerweise eindeutigen Attributen. Der Name wird in der Regel mit Defaultwerten vorbelegt. Gateways zum Beispiel können den Standardnamen „Gateway“ haben. Arbeitet das Tool mit einem Verzeichnis aller Elemente (Objekte), auch [Repository](#) genannt, sollten Defaultnamen durch sprechende Namen ersetzt werden, um die wiederverwendbaren Elemente eindeutig zu identifizieren und die Wiederverwendbarkeit zu erleichtern. In [Bild 9](#) ist das Attribut „name“, hier speziell „task name“, mit dem Wert „System nutzen“ belegt.

Obligatorisch heißt, dass das Tool, wenn es eine konforme BPMN-Implementierung garantiert, das Attribut als Datenfeld zur Verfügung stellen muss mit entsprechenden Ein-/Ausgabemechanismen. Ein Beispiel ist das Attribut „documentation“.

Optional heißt, dass zur Herstellung der Konformität das Attribut nicht unterstützt werden muss. Wenn es aber bereitgestellt wird, muss dies entsprechend den Regeln der Spezifikation erfolgen.

Daneben gibt es noch zusätzlich Attribute, die in der [BPMN-Spezifikation](#) nicht definiert sind, aber als [Erweiterung](#) toolspezifisch hinzugefügt werden.