



# Objektorientierte Systemanalyse

asystemanalyse

Teil 3: Aktivitätsdiagramme

**IT works.**

# Agenda



- ➔ Einführung
  - Basiskonzepte, UML
  - Fokus in der OOA
- ➔ Use Cases
  - Use Case Diagramme
  - Textschablone
- ➔ **Aktivitätsdiagramme**
  - Grundkonzepte
  - Erweiterungen
- ➔ Klassendiagramme
  - Klassendiagramme
  - Paketdiagramme, Komponentendiagramme
  - Objektmodelle, Szenarien, Zustandsautomaten
  - CRC-Karten
- ➔ UI-Design
  - Ergonomie, Usability, Gestaltung
  - Ableitung aus Klassendiagrammen
  - Beschreibung über Zustandsautomaten

# Aktivitätsdiagramme

- ➔ Aktivitätsdiagramme sind Hilfsmittel zur Illustration und Konkretisierung von Abläufen. Sie eignen sich daher innerhalb der objektorientierten Systemanalyse zur weiteren Detaillierung des Ablaufes der Use Cases.
- ➔ Die Modellierung des Ablaufs in grafischer Form ermöglicht eine standardisierte, normierte und formale Konkretisierung der bisherigen textuellen Use Case Beschreibungen.  
Zusätzlich:
  - Aktivitätsdiagramme können zur Verfeinerung beliebig geschachtelt werden.
  - Aktivitätsdiagramme bieten die Möglichkeit zur Darstellung nebenläufigen Verhaltens.
- ➔ Aktivitätsdiagramme können auch zur Geschäftsprozess-Modellierung verwendet werden. Inzwischen wird hierfür jedoch BPMN verwendet, welche den Aktivitätsdiagrammen stark ähnelt.
  - BPMN = Business Process Modeling Notation
  - Standardisiert durch die OMG seit 2006, aktuell BPMN 2.0

# Kernelemente in Aktivitätsdiagrammen



## ➔ Aktivität

Eine Aktivität umfasst einen gesamten Ablauf. In einem Aktivitätsdiagramm kann es mehrere Aktivitäten geben, meist wird jedoch pro Diagramm eine Aktivität beschrieben.

## ➔ Aktion

Eine Aktion bezeichnet einen einzelnen Schritt in einer Aktivität, den ein Ablauf unter Zeitaufwand durchschreitet.

- Hinweis: In der UML-Spezifikation 1.x wurde die Aktion als Aktivität bezeichnet, den Begriff der Aktion gab es nicht. Erst ab der UML 2.x wird der komplette Ablauf als Aktivität bezeichnet und der Einzelschritt als Aktion.

## ➔ Objektknoten

Ein Objektknoten repräsentiert im Aktivitätsdiagramm Daten, die am Ablauf in irgendeiner Weise beteiligt sind (z.B. Buchungsdaten, Rechnungen, Verträge). Ein Objektknoten kann als zusätzliche Information mit einem fachlichen Zustand versehen werden.

## ➔ Kontrollknoten

Sogenannte Kontrollknoten steuern den Fluss des Ablaufs in einer Aktivität. Hierzu gehören z.B. Start- und Endknoten, Verzweigungsknoten oder auch Parallelisierungs- oder Synchronisationsknoten.

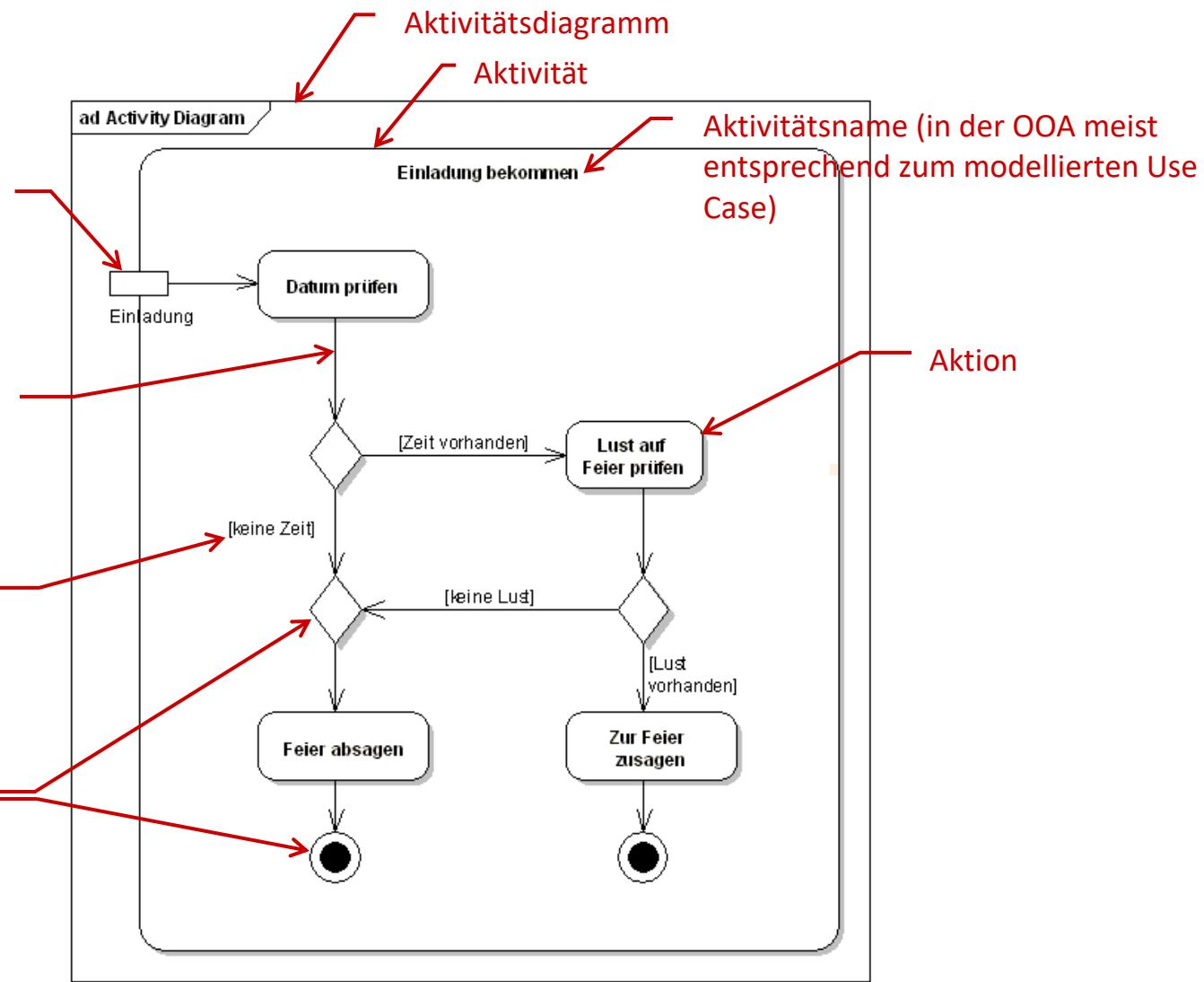
# Kernelemente in Aktivitätsdiagrammen – ein einfaches Beispiel

Objektknoten als Eingangs-  
Parameter, daher hier kein  
Startknoten

Kanten verbinden die  
einzelnen Aktionen und  
Kontrollknoten

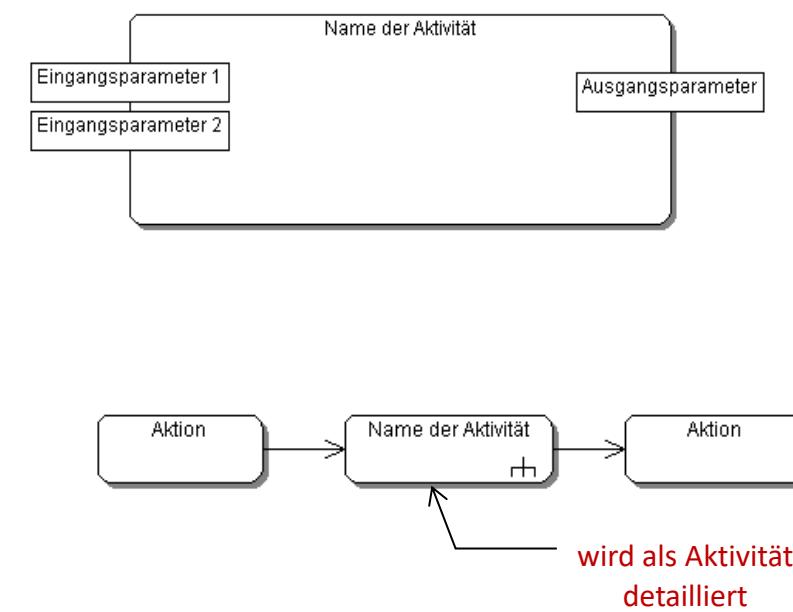
Bedingung

Kontrollknoten



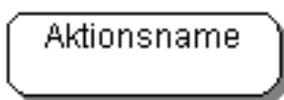
# Aktivität und ihre Eigenschaften

- Eine Aktivität umfasst ab der UML 2.0 einen gesamten Ablauf. Der Ablauf besteht aus einer Folge von Aktionen und weiteren Modellelementen.
- Eine Aktivität hat immer einen Namen. In der objektorientierten Analyse wird hier der Name des Use Cases übernommen, dessen Ablauf in der Aktivität modelliert wird.
- Eine Aktivität kann einen oder mehrere Eingangs- oder Ausgangsparameter besitzen. Dies ist optional und hängt vom modellierten Szenario ab. Viele Aktivitäten beginnen mit einem Start- und enden mit einem Endknoten. Ein- oder Ausgangsparameter werden als Objektknoten auf dem Rand der Aktivität modelliert. Die Position hängt dabei vom Ablauf ab und ist beliebig wählbar. Führt der Ablauf vom Objektknoten weg, handelt es sich um einen Eingangsparameter. Führt der Ablauf zum Objektknoten hin, dann handelt es sich um einen Ausgangsparameter.
- Aktivitäten können geschachtelt werden, d.h. statt einer Aktion (also einem Einzelschritt) wird auf eine weitere Aktivität verwiesen, die diese Aktion weiter detailliert. Ein Verweis auf eine Aktivität kann durch eine Markierung an der Aktion erkannt werden.



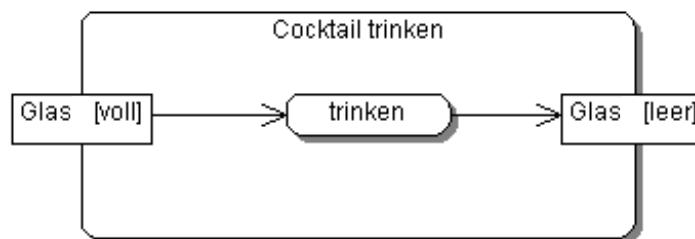
## Aktionen und ihre Eigenschaften

- Eine Aktion steht für den Aufruf eines Verhaltens oder die Bearbeitung von Daten. Die Aktion wird innerhalb einer Aktivität nicht weiter zerlegt.
- Aktionen sollten einen sprechenden kurzen Namen besitzen, welcher verdeutlicht, was in dieser Aktion durchgeführt wird.



- Wird vor und/oder nach einer Aktion ein Objektknoten modelliert, stellt dies die Verwendung von Daten in der Aktion oder ein Ergebnis der Aktion dar. Man spricht von einer parametrisierten Aktion.

Beispiel: Die Aktion „trinken“ wird mit je einem Ein- und Ausgangsparameter, einem Objektknoten mit dem Namen „Glas“, modelliert. Anhand des fachlichen Zustands am Objektknoten [voll] bzw. [leer] kann erkannt werden, wie sich der Zustand der Daten durch die Aktion verändert.



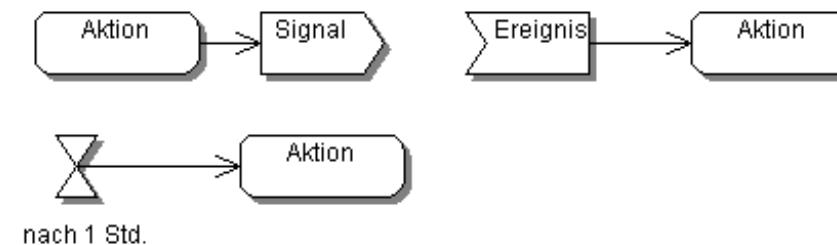
## Aktionen und ihre Eigenschaften (Forts.)

- Sende-Aktionen können Nachrichten verschicken. Der Versand erfolgt immer asynchron, d.h. der weitere Ablauf wird hierdurch nicht blockiert – es wird nicht auf ein Ergebnis gewartet. Für eine Sende-Aktion kann es einen oder auch mehrere Empfänger geben.
- Das Gegenstück hierzu wird durch die Ereignis-Aktion (Event/Accept Call-Action) modelliert. Hierdurch kann modelliert werden, dass eine Aktivität aufgrund eines dazu passenden Ereignisses (Events) beginnt bzw. fortgesetzt wird.
- Ein Sonderfall stellt in diesem Kontext der sogenannte Zeit-Trigger dar. Mit ihm kann dargestellt werden, dass ein Ablauf von Aktionen nach einer bestimmten Zeit, zu einem bestimmten Zeitpunkt oder in einem Zeitintervall angestoßen wird.

### Signale und Ereignisse



### Zeit-Trigger



# Der eigentliche Ablauf: Kontrollfluss

- Die Übergänge zwischen Knoten (Aktionen, Kontrollknoten, etc.) im Aktivitätsdiagramm werden Kanten genannt. Sie werden als Pfeil modelliert.

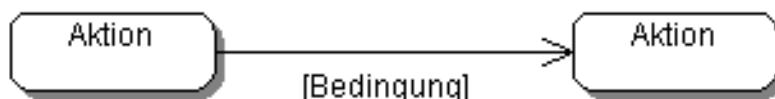
- Kontrollfluss**

Verbindet eine Kante zwei Aktionen oder eine Aktion und einen Kontrollknoten, dann wird diese Kante als Kontrollfluss bezeichnet.



- Bedingungen (Guards)**

Es ist möglich, Kanten mit Bedingungen (den sogenannten Guards) zu versehen. Diese werden in eckigen Klammern an der Kante notiert. Sie geben an, dass ein Übergang nur möglich ist, wenn die Bedingung zutrifft.



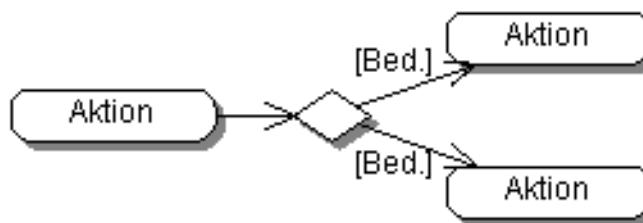
# Steuerung über Kontrollknoten

- Die Steuerung über den Ablauf erfolgt über die sogenannten Kontrollknoten. Die wesentlichen Kontrollknoten sind:

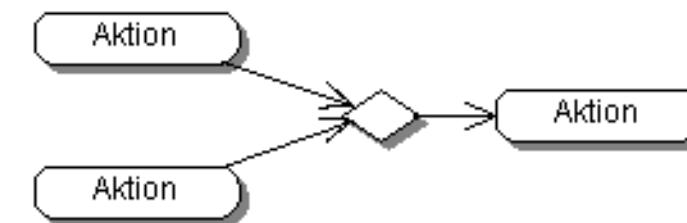
- Startknoten 
- Endknoten gibt es in zwei Varianten:
  - Endknoten, der die gesamte Aktivität beendet, wenn er erreicht wird 
  - Endknoten, der einen einzelnen Teil-Ablauf beendet, aber nicht die gesamte Aktivität 

## Verzweigungs- und Verbindungsknoten

- Ein Verzweigungsknoten ist nicht mit einer Entscheidung in einem Programmablaufplan zu verwechseln. Ein Verzweigungsknoten teilt den Ablauf in mehrere nachfolgende Teilabläufe. In diese wird verzweigt, wenn die notierte Bedingung zutrifft. Dies kann auch bedeuten, dass in mehrere Teilabläufe verzweigt wird, wenn die jeweiligen Bedingungen zutreffen.
- Über den Verbindungsknoten können Teilabläufe wieder zusammengeführt werden.



Verzweigungsknoten

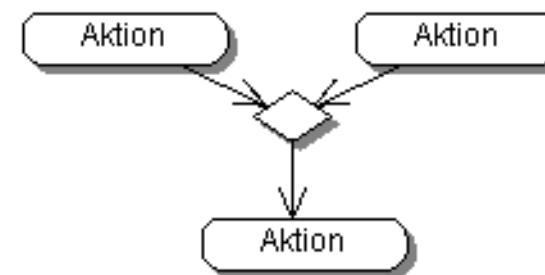
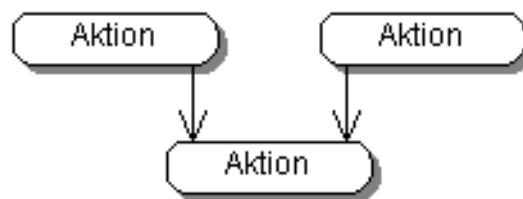


Verbindungsknoten

# Hinweis zu Kanten und der Bedeutung der Verbindungsknoten

## Achtung!

Es besteht eine unterschiedliche Semantik bei der Verbindung mehrerer Aktionen direkt über Kanten oder bei der Verwendung eines Verbindungsknotens.



### Implizites UND

Die nachfolgende Aktion wird nur dann ausgeführt, wenn der Ablauf an beiden eingehenden Kanten angelangt ist.

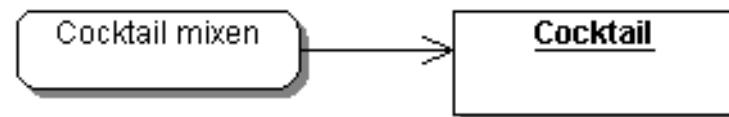
### Explizites ODER

Die nachfolgende Aktion wird ausgeführt, wenn der Ablauf an einer der beiden Kanten zum Verbindungsknoten angelangt ist.

**Hinweis:** In der UML 1.x bedeuteten beide Varianten ein explizites Oder. Daher ist es möglich, dass früher entstandene Modelle eigentlich nach der UML 2.x nicht mehr den korrekten Ablauf darstellen.

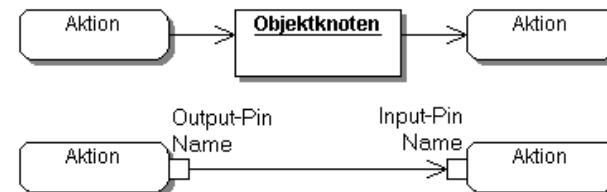
# Objektknoten

- Objektknoten repräsentieren Daten, die im Kontext eines Ablaufs benötigt, verarbeitet oder verändert werden.
  - Die Datenstruktur, die durch den Objektknoten repräsentiert wird, ist nicht zwangsläufig mit einem Objekt (Instanz einer Klasse) gleichzusetzen. Der Objektknoten kann durchaus auch eine komplexere Datenstruktur darstellen (z.B. ein Objektknoten „Bestellung“ bedeutet vielleicht die komplette Datenstruktur einer Bestellung mit ihren Bestellpositionen, die sich auf Warenartikel beziehen, Lieferadresse etc.)
  - Objektknoten können um einen fachlichen Zustand ergänzt werden. Dieser Zustand ist entweder eine konkrete Wertausprägung (z.B. [check=false]) oder beschreibt über den korrekten fachlichen Begriff den jeweiligen Zustand (z.B. [geprüft], [ungeprüft][storniert][freigegeben]). Letztere Variante ist für fachliche Modelle in der Analyse bevorzugt anzuwenden.
  - Objektknoten erläutern in Aktivitätsdiagrammen, zu welchen Zeitpunkten Daten benötigt, durch Aktionen verändert oder erzeugt werden. Sie verbessern damit das fachliche Verständnis für den modellierten Ablauf.



## Objektknoten (Forts.)

- Die so genannte PIN-Notation verdeutlicht den Objektknoten als Ein- oder Ausgabe-Parameter einer Aktion.
  - Häufig finden sich Objektknoten zwischen zwei Aktionen. Die Daten werden entweder als Ergebnis der vorausgehenden Aktion betrachtet oder sind Eingangsparameter für die nachfolgende Aktion, oder beides.
  - Die sogenannte PIN-Notation kann verdeutlichen, ob ein Objektknoten als Ausgangsparameter und/oder als Eingangsparameter modelliert ist. Der sogenannte Pin stellt hierbei den Objektknoten dar und kann mit einem Namen fachlich benannt werden. Der Pin wird hierbei jeweils an die Aktion angehängt, von der er benötigt/erzeugt wird. Entweder als Output-Pin, als Input-Pin oder beides.



### Beispiel

- Im Beispiel wird ein Output-Pin „Cocktail“ als Ergebnis der Aktion „Cocktail mixen“ modelliert. Die nachfolgende Aktion „trinken“ erhält einen Input-Pin mit der Bezeichnung „Getränk“. Hierdurch wird auch deutlich, dass diese Art der Modellierung wesentlich ausdrucksstärker ist. So kann auf diese Weise dargestellt werden, dass die Aktion „trinken“ nicht zwangsläufig einen „Cocktail“ als Eingangsparameter erwartet, sondern ganz allgemein ein „Getränk“.

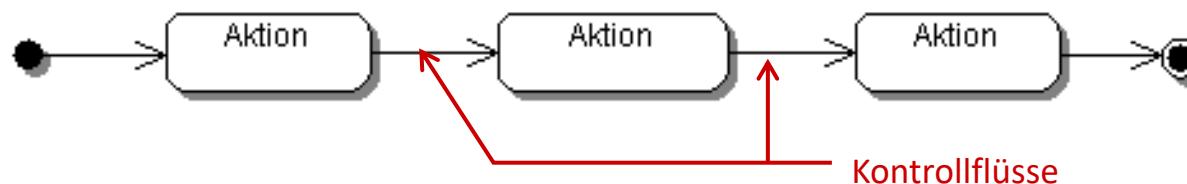


# Kanten zwischen Aktionen und Objekten: Kontrollfluss vs. Objektfluss

- Die Übergänge zwischen Knoten (Aktionen, Kontrollknoten, etc.) im Aktivitätsdiagramm werden Kanten genannt. Sie werden als Pfeil modelliert.

## ▪ Kontrollfluss

Verbindet eine Kante zwei Aktionen oder eine Aktion und einen Kontrollknoten, dann wird diese Kante als Kontrollfluss bezeichnet.



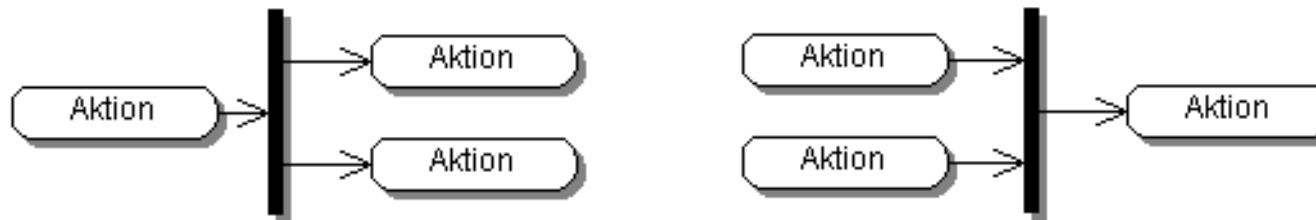
## ▪ Objektfluss

Verbindet eine Kante jedoch eine Aktion / einen Kontrollknoten mit einem Objektknoten bzw. einem Input- oder Output-Pin, dann wird die Kante als Objektfluss bezeichnet. An einem Objektfluss ist somit mindestens ein Objektknoten beteiligt.



# Weitere Elemente in Aktivitätsdiagrammen

## Parallelisierungs- und Synchronisationsknoten

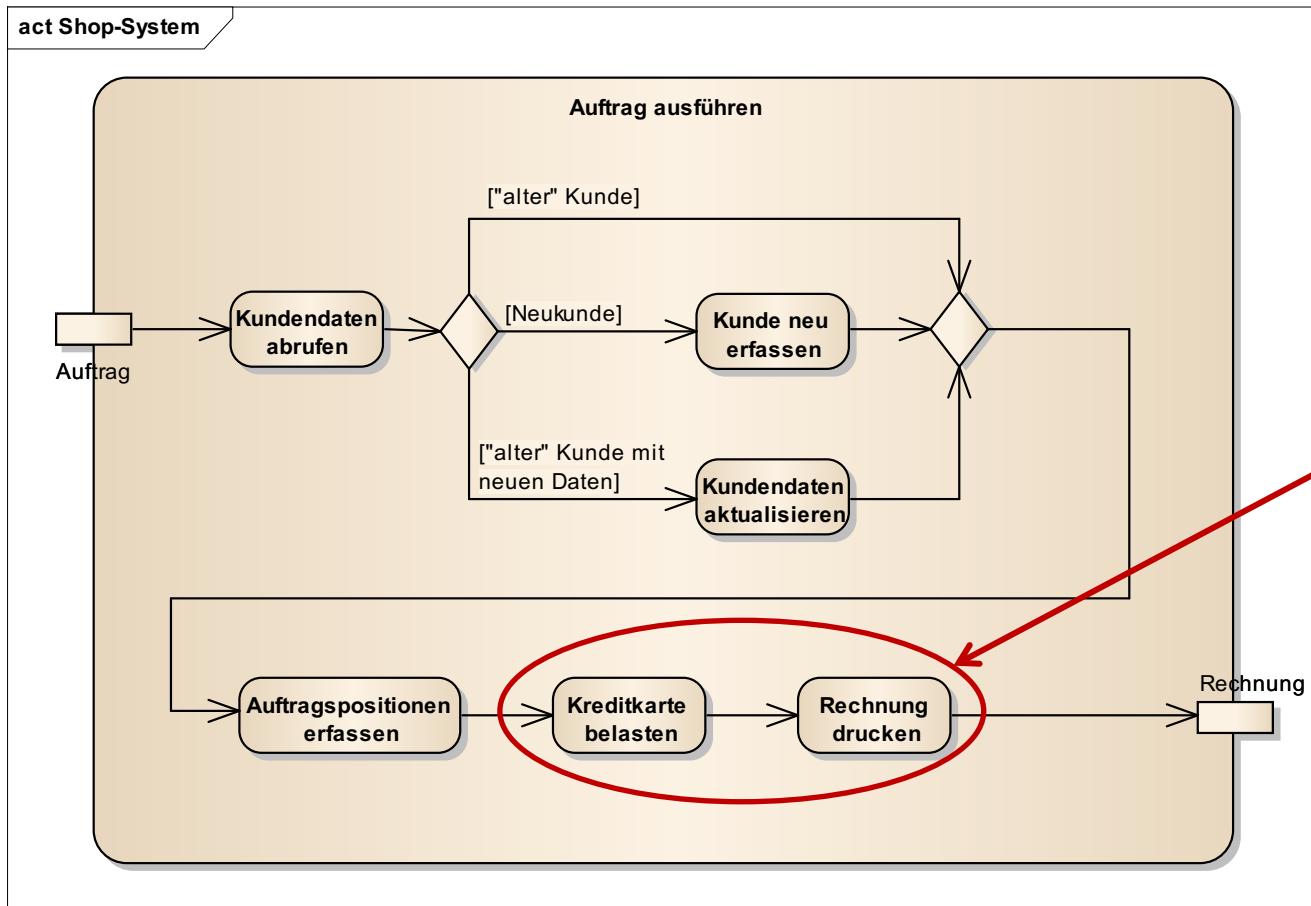


- Parallelisierungsknoten (Splitting) sind gleichbedeutend mit den Verzweigungsknoten, wenn an deren ausgehenden Kanten keine Bedingungen notiert sind. Durch einen Parallelisierungsknoten kann ein Ablauf auf mehrere nachfolgende, nebenläufige oder zeitlich unabhängige Teilabläufe, aufgeteilt werden.
- Synchronisationsknoten (Join): Der Synchronisationsknoten führt diese parallelisierten Teilabläufe wieder zusammen. Dabei gilt, dass die nachfolgende Aktion erst ausgeführt wird, wenn alle Teilabläufe durchlaufen sind (UND-Verknüpfung)
- Dies eignet sich besonders zur Modellierung von Aktionen/Teilabläufen, deren Reihenfolge egal ist oder von Aktionen, die zeitgleich durchgeführt werden sollen.
- Beispiel einer Reisebuchung: Die Buchung von Flug, Hotel und Mietwagen kann in beliebiger Reihenfolge durchgeführt werden. Die Reise wird aber erst final gebucht, wenn alle drei Teilabläufe durchgeführt sind.

## Strukturknoten

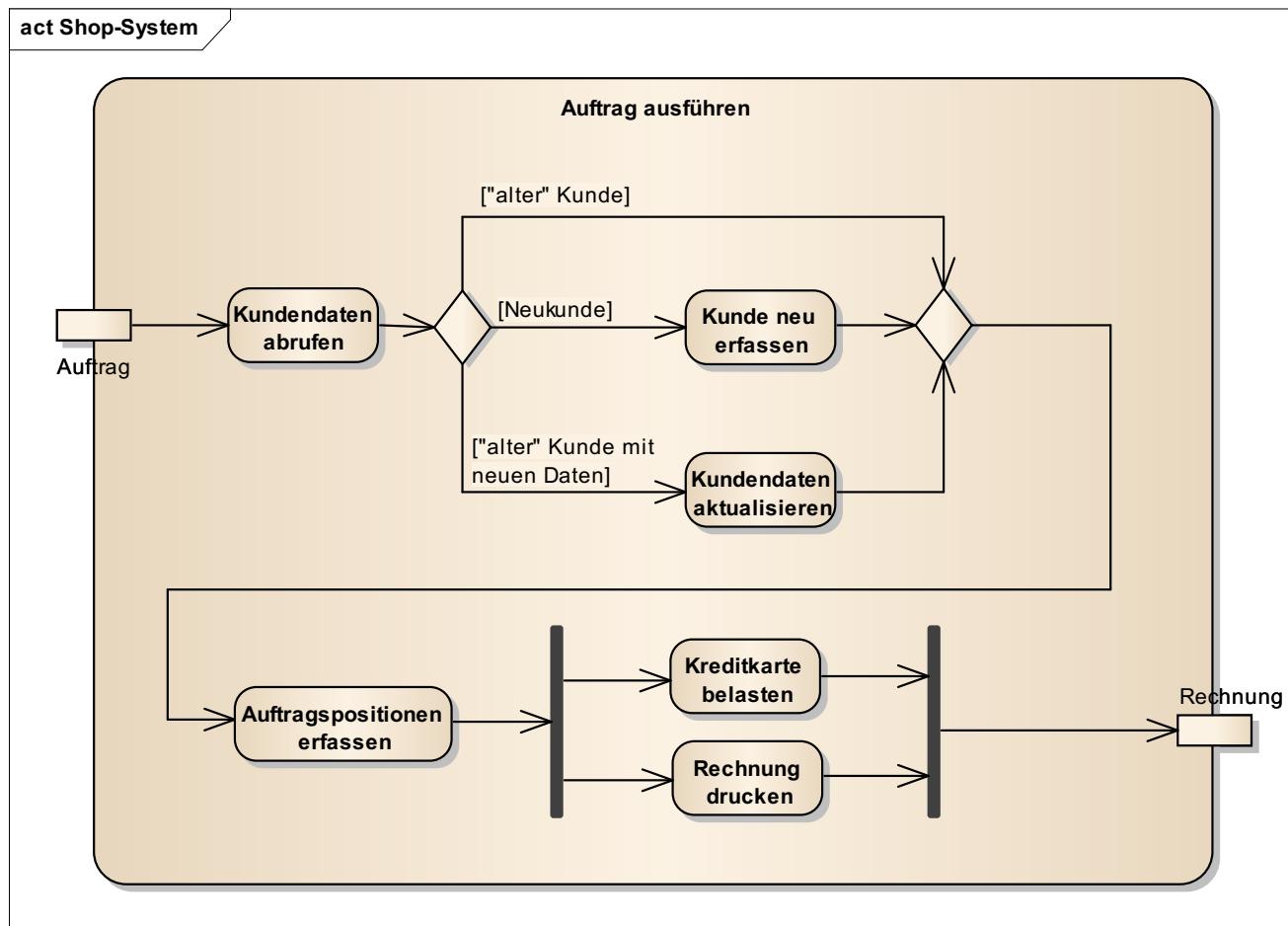
- Strukturknoten sind komplexe Modellelemente in Aktivitätsdiagrammen. Sie bieten die Möglichkeit, einen Bereich in einem Modell zu markieren, für den besondere Eigenschaften festgelegt werden können.
- Es gibt beispielsweise Strukturknoten für Wiederholungen (Schleifen) oder Unterbrechungen.

# Beispiel Parallelisierungs- und Synchronisationsknoten



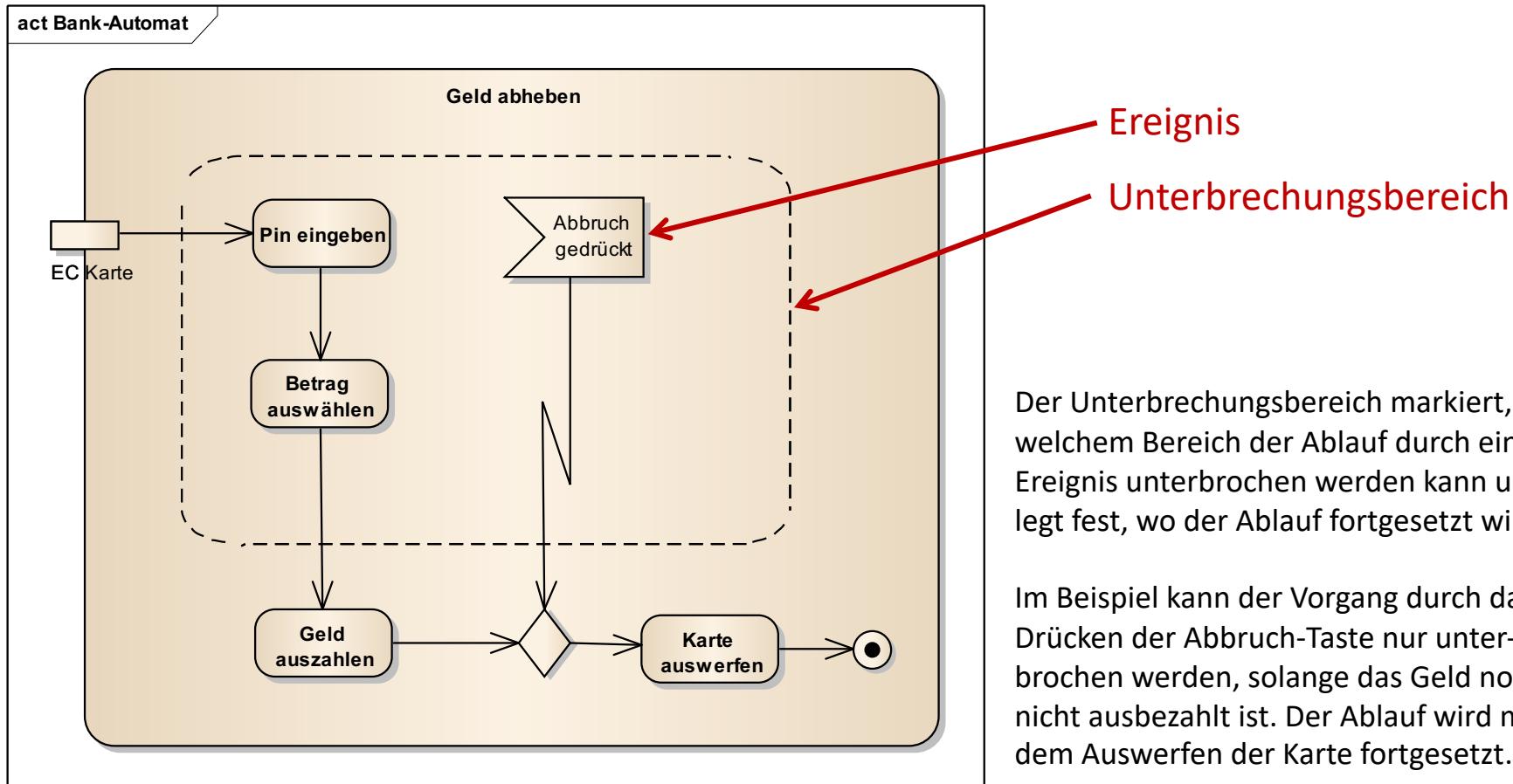
Reihenfolge der  
Aktionen ist in diesem  
Fall eigentlich egal.

# Beispiel Parallelisierungs- und Synchronisationsknoten



Darstellung mit  
Parallelisierungs- und  
Synchronisations-  
knoten.

# Beispiel für Strukturknoten: Unterbrechungsbereich



## Fragestellungen zur Erstellung von Aktivitätsdiagrammen



# Tipp

- ➔ Welches Ereignis oder welche Eingabedaten lösen die Verarbeitung aus?
- ➔ Wie sieht der Standardfall aus? Diesen zuerst modellieren.
- ➔ Welches Ziel soll im Erfolgsfall erreicht werden? Welches Ziel im Fehlerfall?
- ➔ In welcher Reihenfolge sollen die Aktionen ausgeführt werden?
- ➔ Gibt es Aktionen, die nur unter bestimmten Voraussetzungen ausgeführt werden?
- ➔ Können Aktionen parallel ausgeführt werden bzw. gibt es Aktionen, deren zeitliche Reihenfolge fachlich irrelevant ist oder nicht vorhergesagt werden kann?

# Übungen zu Aktivitätsdiagrammen



- ➔ Aufgabe 1: Einfaches Aktivitätsdiagramm zu einem Use Case erstellen.
- ➔ Aufgabe 2: Aktivitätsdiagramm um Objektknoten ergänzen.
- ➔ Aufgabe 3: Komplexeres Aktivitätsdiagramm zu einem Use Case erstellen
- ➔ Aufgabe 4: Use Case und Aktivitätsdiagramm erstellen

# Übungen zu Aktivitätsdiagrammen

## Ziel:

Neuen Leser im Bibliothekssystem aufnehmen

## Akteure:

Bibliotheksmitarbeiter an der Ausleihe

## Auslösendes Ereignis:

Ein neuer Leser möchte sich registrieren lassen.

## Vorbedingung

Keine.

## Nachbedingung

Der neue Leser ist mit seinen Daten im System erfasst und hat einen Bibliotheksausweis erhalten.

## Ablauf:

1. Der Leser weist sich gegenüber dem Mitarbeiter aus.
2. Der Mitarbeiter erfasst Name und Anschrift des neuen Lesers.
3. Das System prüft, ob der Leser bereits im System vorhanden ist. Wenn ja, wird darauf hingewiesen und die Daten zur Kontrolle angezeigt. Wenn nein, wird der Leser neu angelegt.
4. Das System druckt einen Bibliotheksausweis aus.

**Aufgabe 1: Erstellen Sie zum nebenstehenden Use Case „Leser registrieren“ ein einfaches Aktivitätsdiagramm.**

Verwenden Sie folgende Diagrammelemente:

- Start- und Endknoten
- Aktionen
- Kanten (ggf. mit Bedingungen) und
- Verzweigungs- und Verbindungsknoten

# Übungen zu Aktivitätsdiagrammen



## Aufgabe 2: Ergänzen Sie den Ablauf im Aktivitätsdiagramm aus Aufgabe 1 um Objektknoten.

- Welches Objekt/wessen Datenstruktur bietet sich als Objektknoten an?
- Finden Sie für das Objekt geeignete fachliche Zustände und zeichnen Sie die Objektknoten an den entsprechenden Stellen ins Diagramm ein.

# Übungen zu Aktivitätsdiagrammen

**Ziel:**

Ein zurückgegebenes Buch wird wieder zum Ausleihen zur Verfügung gestellt.

**Akteure:**

Bibliotheksmitarbeiter an der Ausleihe

**Auslösendes Ereignis:**

Ein Leser gibt ein Buchexemplar zurück.

**Vorbedingung**

Keine.

**Nachbedingung**

Das Buch ist wieder verfügbar oder ein Leser auf der Warteliste benachrichtigt.

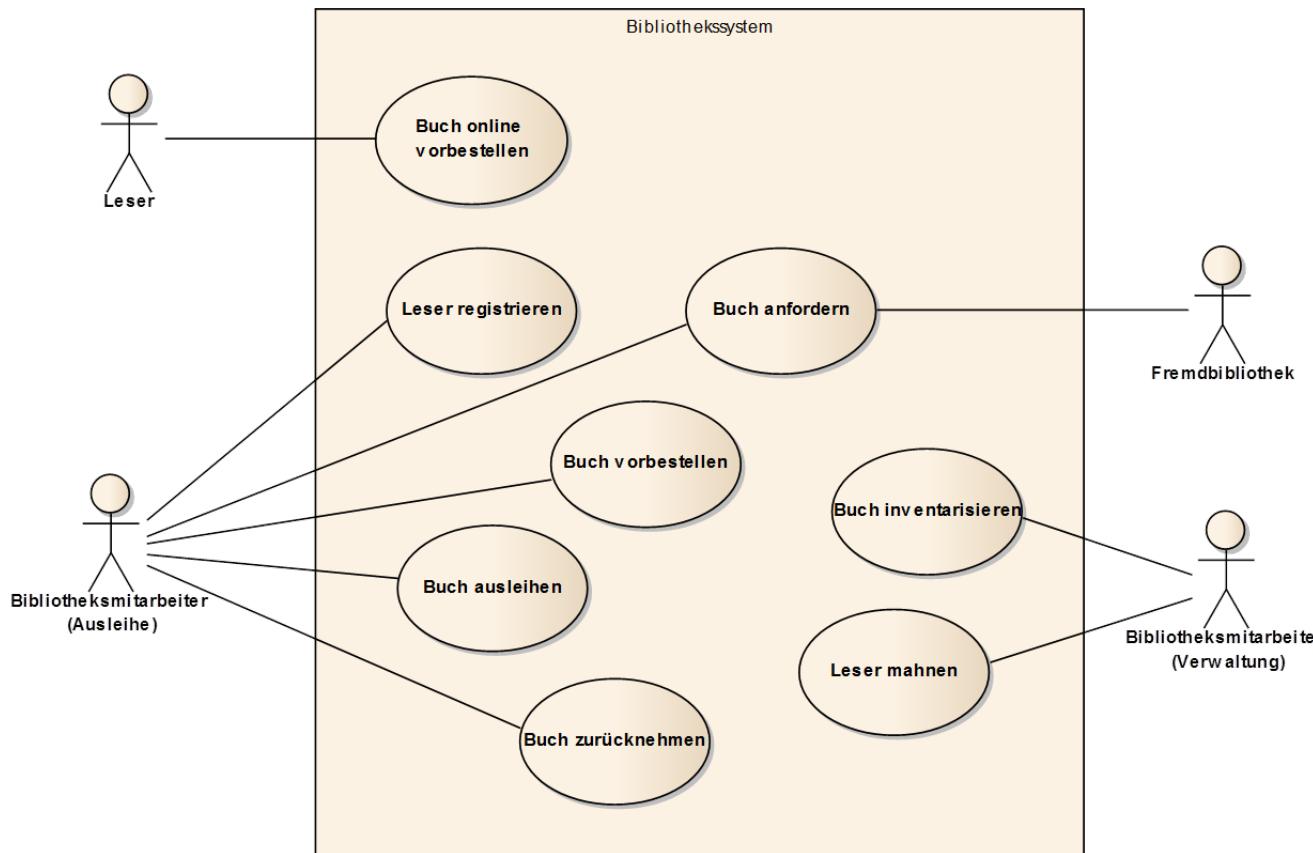
**Ablauf:**

1. Der Mitarbeiter nimmt das Buch entgegen, gibt die Id des Buchs ins System ein und vermerkt die Rückgabe.
2. Das System prüft, ob der Leser Mahngebühren zu bezahlen hat und weist den Mitarbeiter darauf hin. Dieser kassiert ggf. diese Mahngebühren.
3. Der Mitarbeiter prüft das Buch auf Beschädigungen. Ist es beschädigt, wird es zur Reparatur aussortiert.
4. Das System prüft, ob für dieses Buch eine Warteliste vorhanden ist. Wenn ja, wird der nächste Wartende benachrichtigt und das Buch zurückgelegt. Ansonsten wird das Exemplar als „verfügbar“ im System vermerkt und im Regal bereitgestellt.

**Aufgabe 3: Erstellen Sie zu dem nebenstehenden Use Case ein Aktivitätsdiagramm.****Hinweise:**

- Können bestimmte Aktionen in beliebiger Reihenfolge ausgeführt werden? Dann parallelisieren.
- Verwenden Sie auch Objektknoten mit fachlichen Zuständen.

# Übungen zu Aktivitätsdiagrammen



**Aufgabe 4: Wählen Sie einen der noch nicht beschriebenen Use Cases aus dem Use Case-Diagramm aus.**

Erstellen Sie hierzu die Use Cases-Beschreibung und das dazugehörige Aktivitätsdiagramm.

Anmerkung: Hierzu liegt keine Musterlösung vor.

Vielen Dank für Ihre Aufmerksamkeit. Haben Sie noch Fragen?



## Kontakt

Dr. Nikolaus Mairon

[klaus@mairon-online.de](mailto:klaus@mairon-online.de)  
Tel. +49 (0) 160 96678776  
Skype-ID: [klaus@mairon-online.de](mailto:klaus@mairon-online.de)

**msg nexinsure ag**

Geschäftsstelle St. Georgen  
Leopoldstr. 1  
78112 St. Georgen

[nikolaus.mairon@msg.group](mailto:nikolaus.mairon@msg.group)  
Tel. +49 (0) 89 96101-3004  
Mobil +49 (0) 171 9716462

**IT works.**