

Unified Modeling Language (UML)

- ☐ Was ist ein Modell?
 - ☐ Warum Modellieren?
 - ☐ Warum UML?
 - ☐ Viele, viele Diagramme
-
- ☐ Einführung in Use-Case-Diagramme
-

Was ist ein Modell?

- **Ein Modell:**

ist eine abstrakte Repräsentation eines Systems, bzw.
ist eine Vereinfachung der Realität.

- stellt die Kernpunkte eines Systems dar.

- Abstrakt bedeutet nicht kompliziert, sondern auf das Wesentliche reduziert!

- Durch Abstraktion verhindern wir, uns beim Entwurf eines Systems in Details zu verlieren!

Warum Modellieren?

- ❑ Teamarbeit und Kommunikation ist notwendig
 - ❑ Nicht nur Programmierer, auch Manager, Vertrieb und Kunden sind in den Entwicklungsprozess involviert
-> gemeinsame, verständliche Sprache ist notwendig
 - ❑ Das Rad muss nicht jedes mal neu erfunden werden
-> Dokumentation
 - ❑ natürliche Sprache ist mehrdeutig und unpräzise
Programmiersprachen sind zu präzise und nicht für alle (z.B. Manager) verständlich ->
graphische Modellierung mit UML füllt diese Lücke aus
-

Warum Modellieren?

Software sollte:

- ☐ nützlich und nutzbar
- ☐ zuverlässig
- ☐ flexibel
- ☐ kostengünstig und
- ☐ verfügbar sein.

Negative Gegenbeispiele:

- ☐ Ariane 5
 - ☐ Therac 25
 - ☐ ...
-

Warum UML?

- ❑ **Unified Modeling Language (UML)** derzeit in Version 2.0
 - ❑ 1995, erfunden von Grady Booch, Jim Rumbaugh und Ivar Jacobson
 - ❑ mittlerweile de facto Standard
 - ❑ graphisch orientierte Modellierungssprache
 - ❑ objektorientiert
 - ❑ unabhängig von Entwicklungsprozessen und -methoden
 - ❑ neun verschiedene Diagramme, um die verschiedenen Aspekte von Softwaresystemen widerzuspiegeln
-

Viele, viele Diagramme

- Sinn und Zweck eines Systems -> Use-Case-Diagramme (Anwendungsfall-Diagramme)
 - Struktur eines Systems -> Klassendiagramme, Objektdiagramme
 - Verhalten eines Systems – Interaktion zwischen Objekten -> Sequenzdiagramme, Kommunikationsdiagramme
-

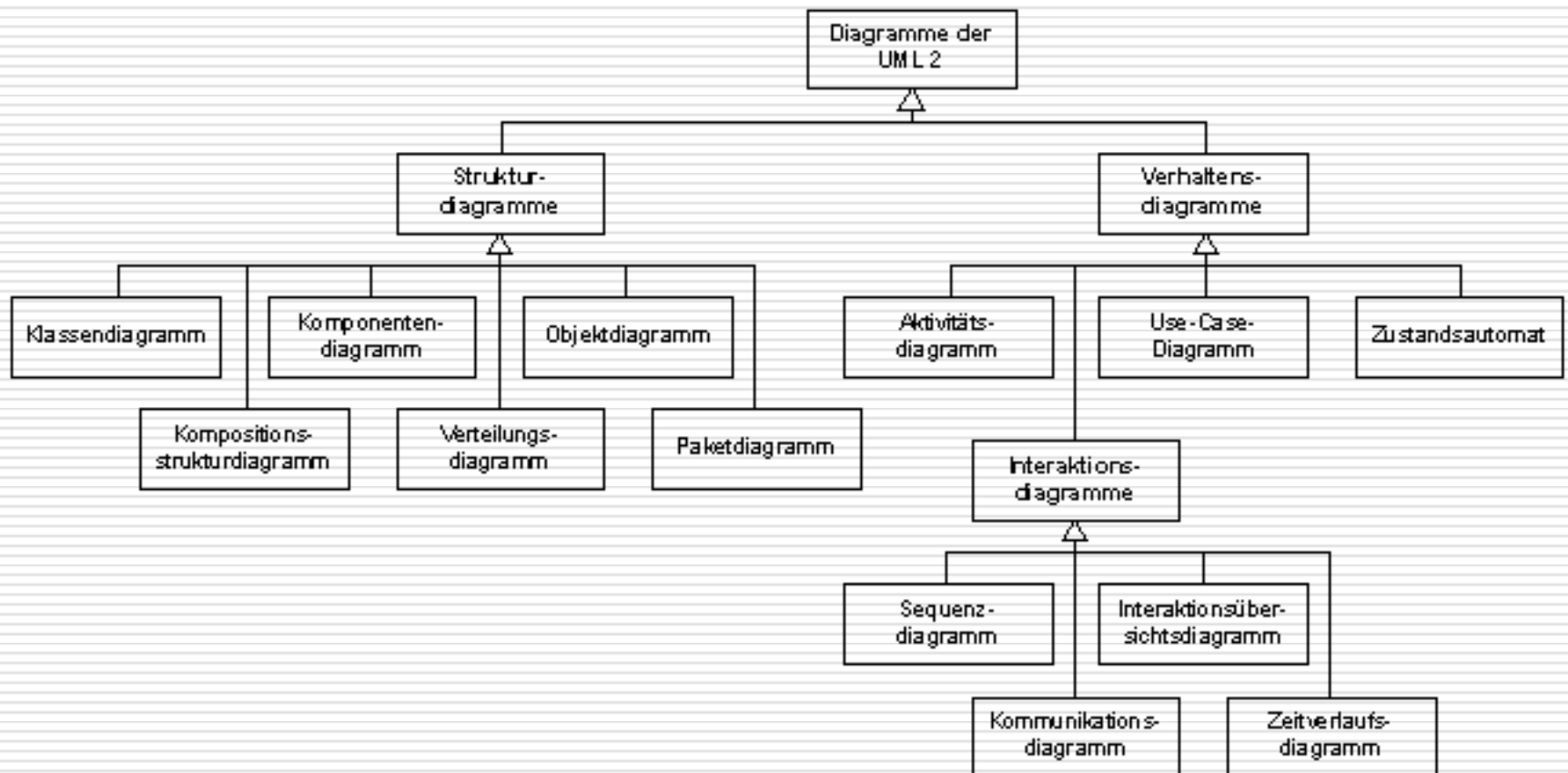
Viele, viele Diagramme

- ☐ Verhalten eines Systems – Verhalten eines Objektes -> Zustandsdiagramme
 - ☐ Verhalten eines Systems – unabhängig von Objekten -> Aktivitätsdiagramme
 - ☐ Physikalische Struktur eines Systems – Softwarekomponenten -> Komponentendiagramme
-

Viele, viele Diagramme

- Physikalische Struktur eines Systems –
Software und Hardware ->
Verteilungsdiagramme
 -
-

Viele, viele Diagramme



Use-Case-Diagramme

Use Case Diagramme

- Erster Schritt der Problemanalyse
 - Festlegung des Funktionsumfangs des Systems

 - Ziel
 - Vollständige Problemspezifikation durch Darstellung der Gesamtheit von Anwendungsfällen
 - Planung von Testfällen von Beginn an

 - Beschreibung
 - Informale Spezifikation (keine formale Vollständigkeitsanalyse)
-

Use Case Diagramme

Motivation: Use Case Diagramme

- ❑ ... erfassen
Systemanforderungen und spezifizieren damit die Funktionalität, die das System erfüllen soll
 - ❑ ... bestehen aus
einzelnen Anwendungsfällen(use-cases)
Aktoren(actors) sowie deren Beziehungen
 - ❑ ...dienen
auch als Kommunikationsgrundlage mit dem Auftraggeber
 - ❑ ... verknüpfen
einzelne Anwendungsfälle u. beziehen sowohl menschliche Benutzer als auch externe Systeme mit ein
-

Use Case Diagramme

Definition:

Ein Use-Case-Diagramm zeigt die Beziehungen zwischen Akteuren und Anwendungsfällen.

Beschreibung:

- Beschreibt die Zusammenhänge zwischen einer Menge von Anwendungsfällen und den daran beteiligten Akteuren.
 - Hilfsmittel zur Anforderungsanalyse
 - Anwendungsfälle beschreiben das gewünschte externe Systemverhalten aus Sicht des Anwenders und somit Anforderungen, die das System erfüllen soll.
 - **Was** soll das System leisten aber nicht **wie** es dies leisten soll.
-

Use Case Diagramme

☐ Anwendungsfälle:

Definition:

- ☐ Ein Anwendungsfall beschreibt eine Menge von Aktivitäten eines Systems aus der Sicht seiner Akteure
 - ☐ Ein Anwendungsfall wird stets durch einen Akteur initiiert.
 - ☐ Ein Anwendungsfall ist eine komplette, unteilbare Beschreibung.
-

Use Case Diagramme

Beschreibung:

- ☐ Ein Anwendungsfall kann verschiedene Ablauf-varianten umfassen.
 - ☐ Eine einzelne spezielle Ausprägung eines Anwendungsfalls wird Szenario genannt.
 - ☐ Ein Anwendungsfall beschreibt eine Menge von möglichen Szenarien.
 - ☐ An jedem Anwendungsfall ist mindestens ein Akteur beteiligt.
 - ☐ Jeder Anwendungsfall hat einen fachlichen Auslöser.
 - ☐ Jeder Anwendungsfall produziert ein für die Akteure relevantes fachliches Ergebnis.
-

Use Case Diagramme

□ Akteure

Definition:

Ein Akteur ist eine außerhalb des zu realisierenden Systems liegende Einheit, die an der in einem Anwendungsfall beschriebenen Interaktion mit dem System beteiligt ist (z.B. ein Mensch, ein anderes technisches System, das Betriebssystem u.ä.).

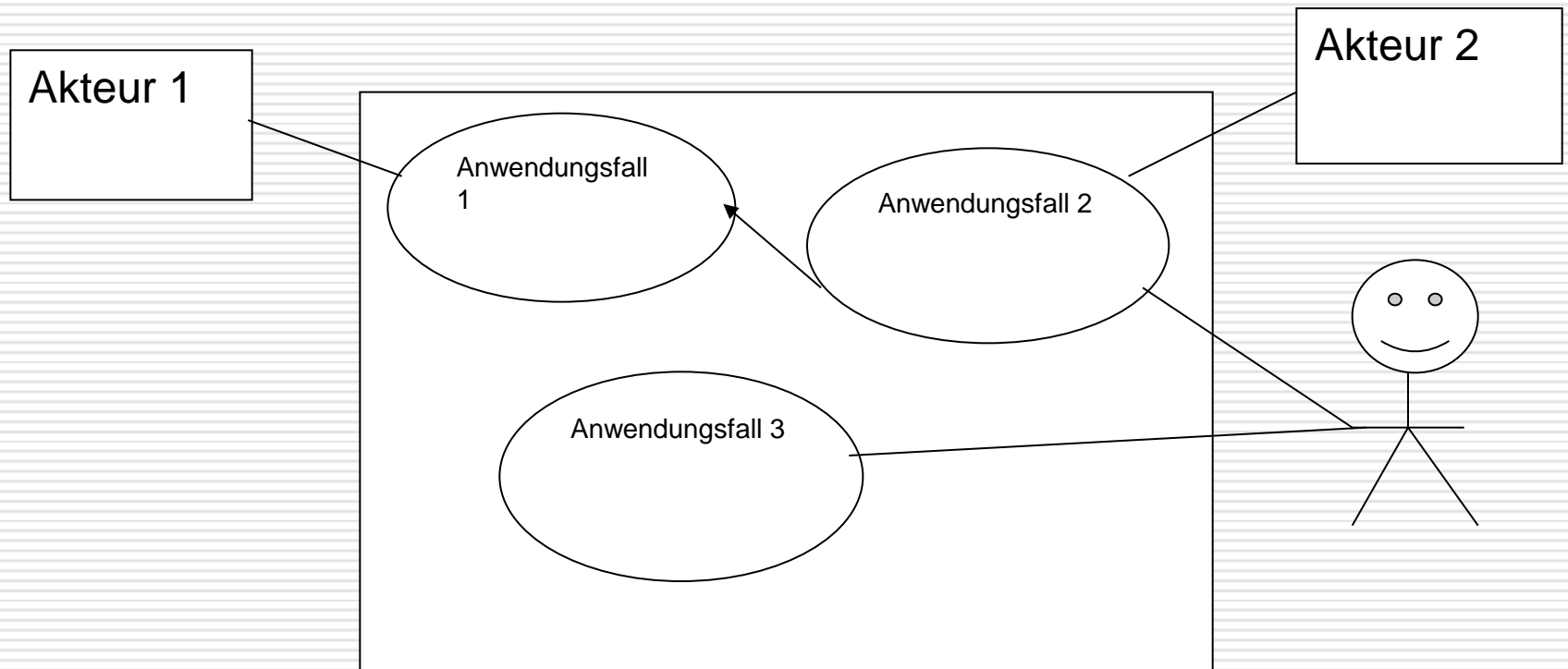
Use Case Diagramme

Beschreibung:

- ☐ Akteure sind beispielsweise die Anwender des Systems.
 - ☐ Bei den Akteuren werden nicht die beteiligten Personen unterschieden, sondern ihre Rollen, die sie im Kontext des Anwenderfalls einnehmen (Bsp. Kundenberatung und Antragsannahmen durch eine Person = mehrere Rollen = mehrere Akteure)
 - ☐ Akteure besitzen Assoziationen zu Anwendungsfällen, wenn sie an den dort beschriebenen Abläufen beteiligt sind.
 - ☐ Akteure können auch untereinander Assoziationsbeziehungen haben
-

Use Case Diagramme

Notation:



Use Case Diagramme

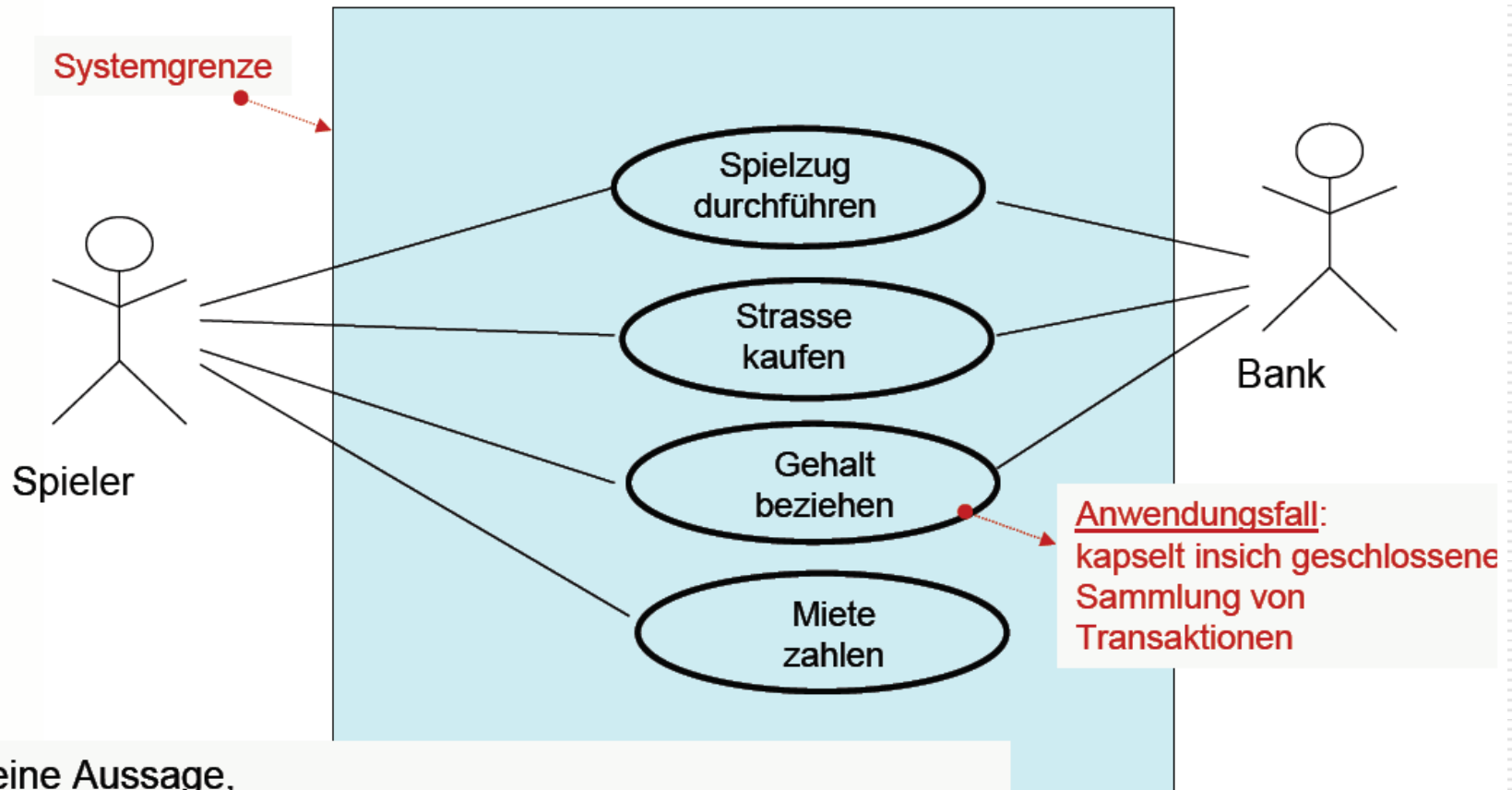
Beispiel Monopoly (internet basiert):

- ☐ Ein Spieler würfelt
- ☐ Ein Spieler kauft eine Straße
- ☐ Ein Spieler führt einen Spielzug aus
- ☐ Ein Spieler zahlt Miete
- ☐ Die Bank zahlt Geld an einen Spieler
- ☐ Der Spieler errichtet Häuser und Hotels
- ☐ Der Spieler nimmt eine Hypothek auf
- ☐ Ein Spieler beendet das Spiel

(weitere Spielzüge siehe Spielanleitung)

Use Case Diagramme Monopoly

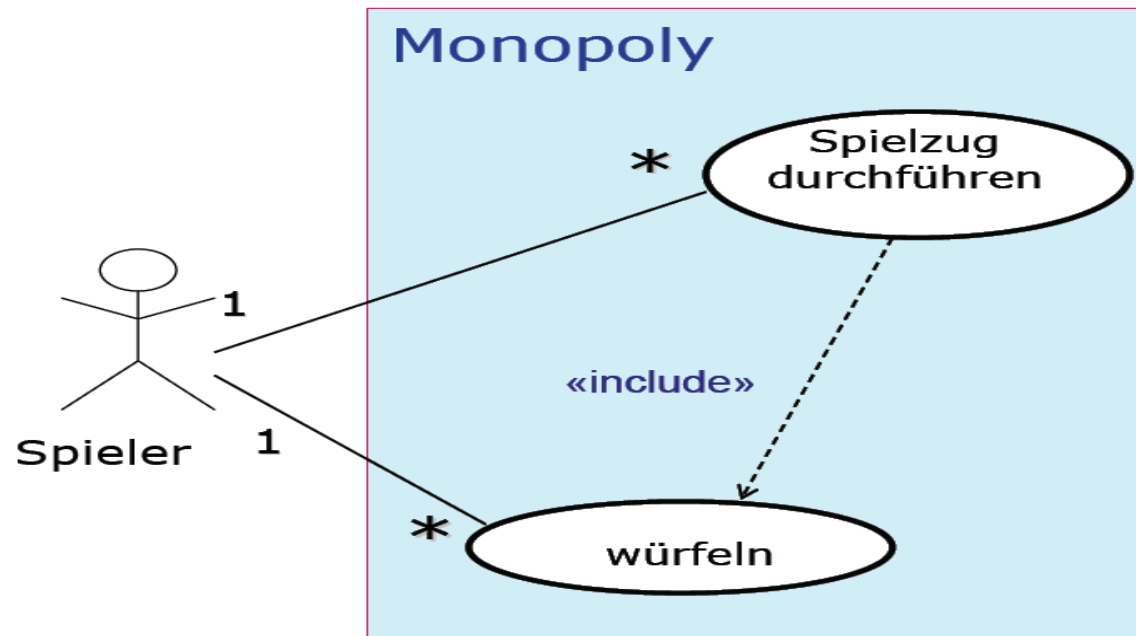
Akteur (als Rollenmuster): Beschreibung der Personen (aus der Systemumgebung)



keine Aussage,
ob Spieler- und Bank-Rolle von **einer** oder **verschiedenen**
Personen wahrgenommen werden

Use Case Diagramme

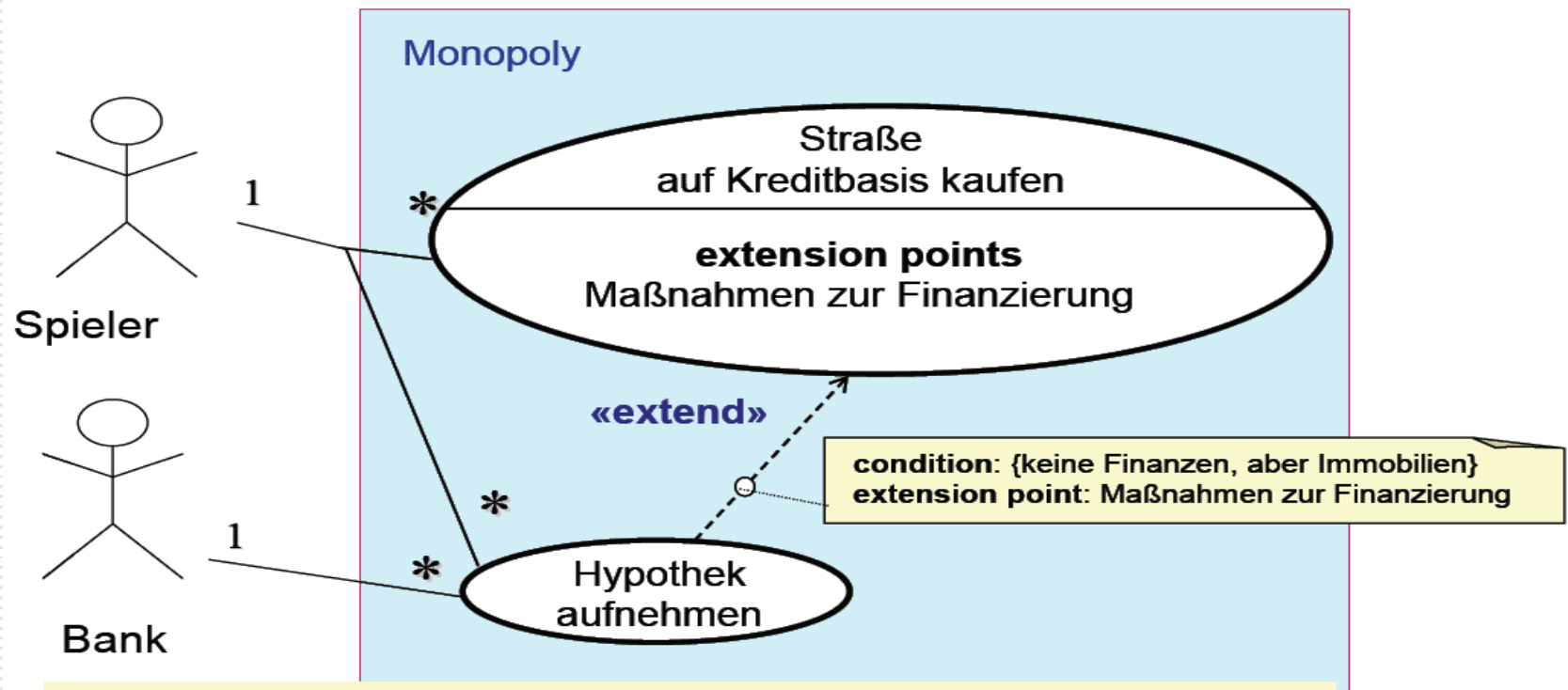
Game-Beispiel: *includes*



- Spielzug von einem Spieler durchführen zu lassen, erfordert **stets** ein Würfeln
- noch nicht spezifiziert ist, dass auch derjenige würfeln muss, der den Spielzug durchführt

Use Case Diagramme

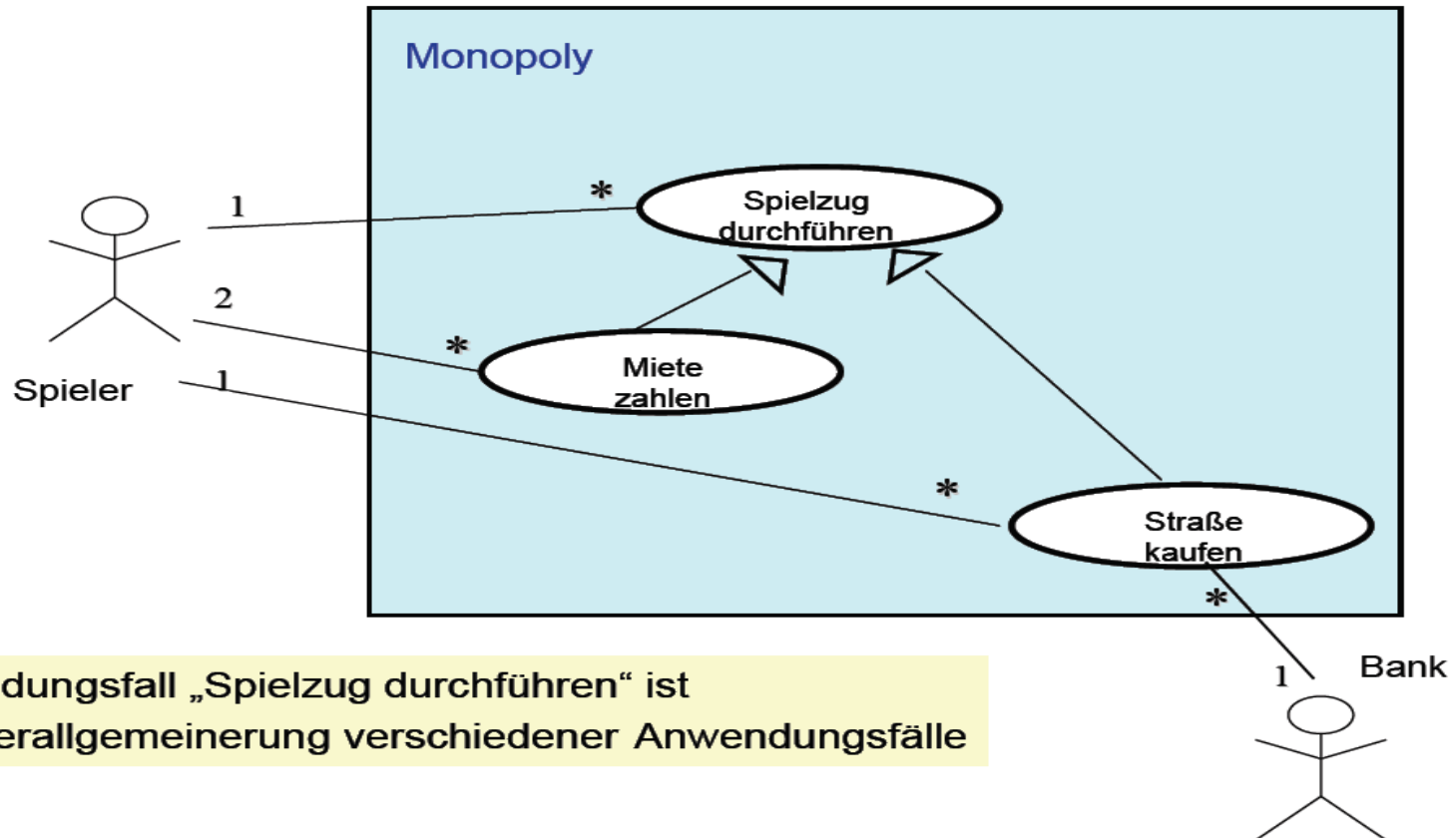
Game-Beispiel: *Extends*



Erweiterungsfall wird nur unter bestimmten Bedingungen ausgeführt.
(Verfeinerung eines Anwendungsfalles)

Use Case Diagramme

Game-Beispiel: Use-Case-Generalisierung



Anwendungsfall „Spielzug durchführen“ ist eine Verallgemeinerung verschiedener Anwendungsfälle