

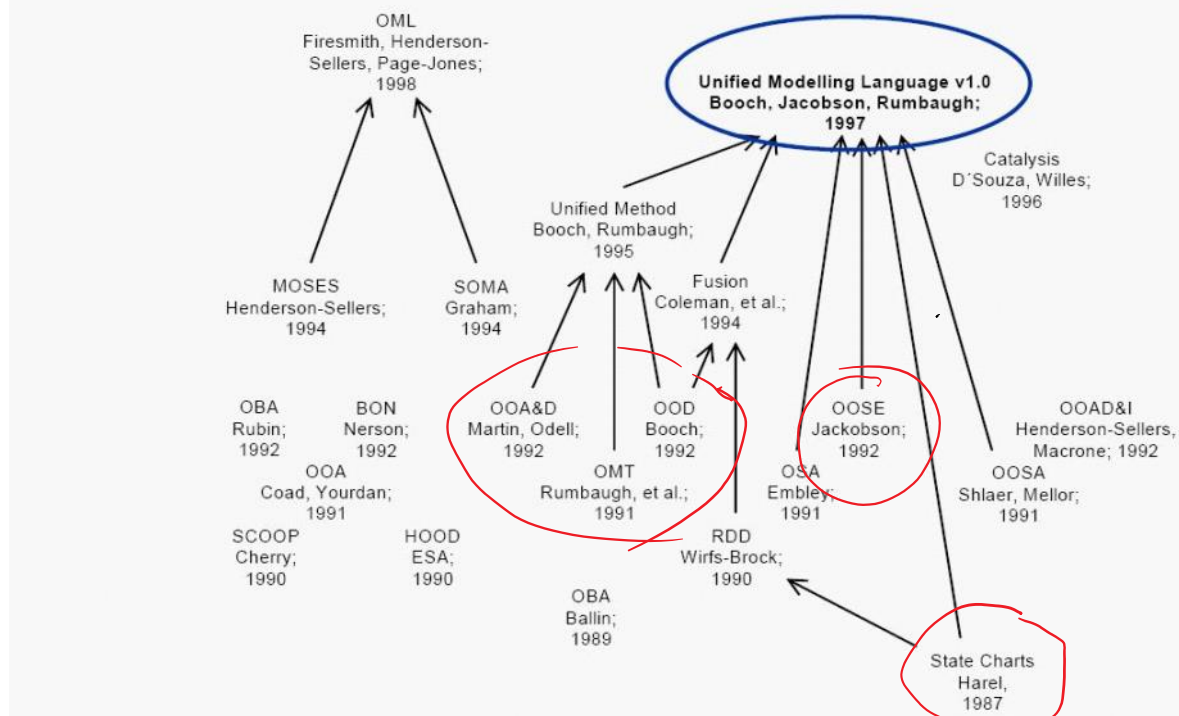
Woche 1

Sonntag, 15. September 2019

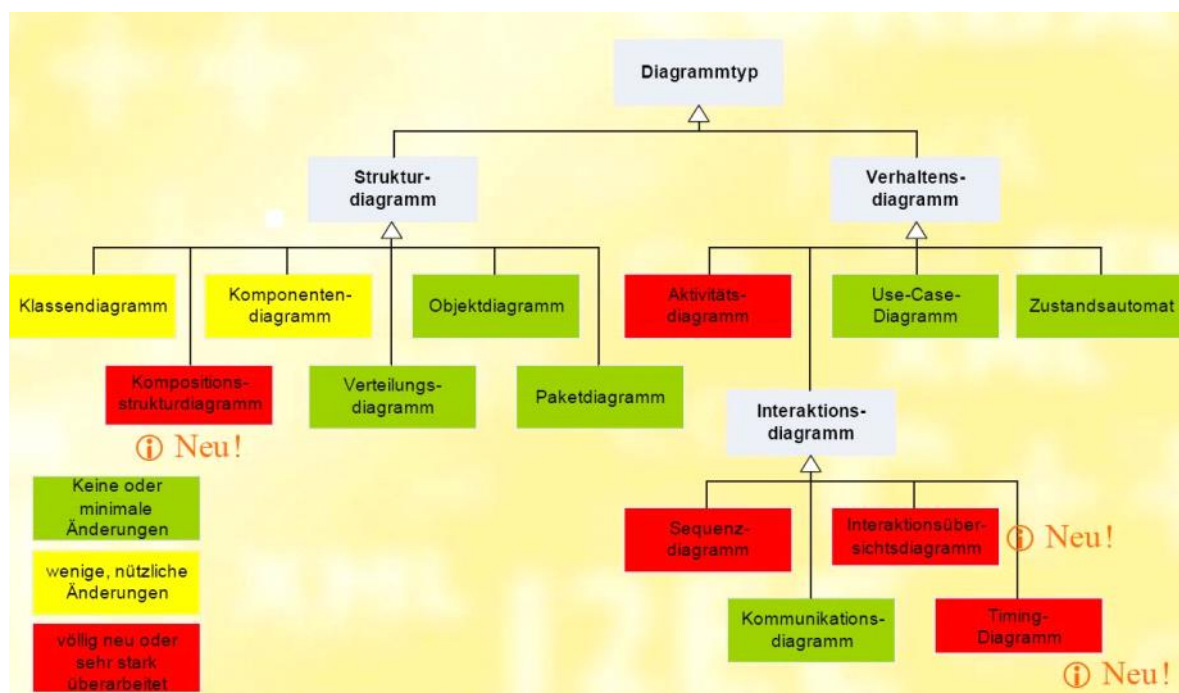
10:29

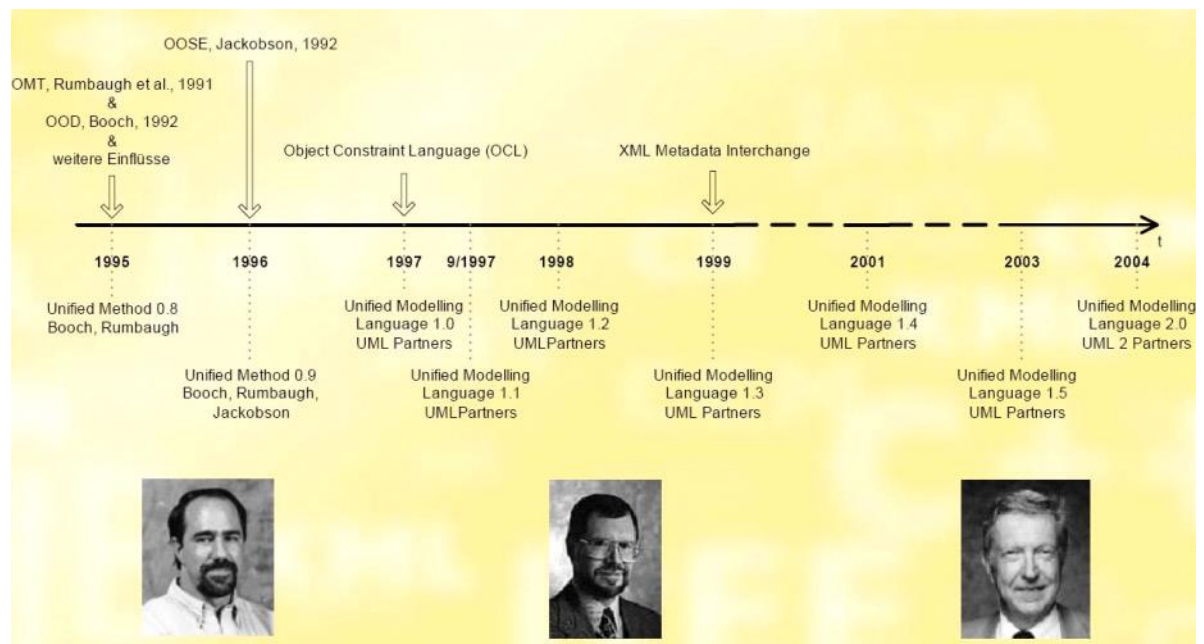
UML - Überblick

„Die Methodenkriege der Softwareentwicklung“



Methodenkrieg





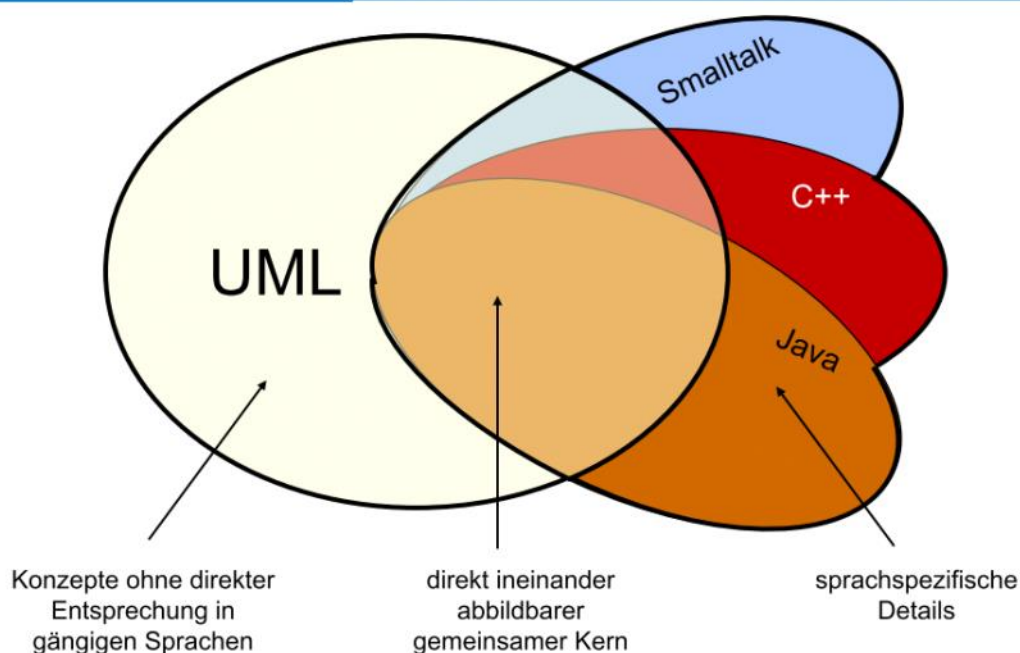
3 Amigos

UML - Verständnis

Software ist schon eine Abstraktion, also warum noch modellieren?

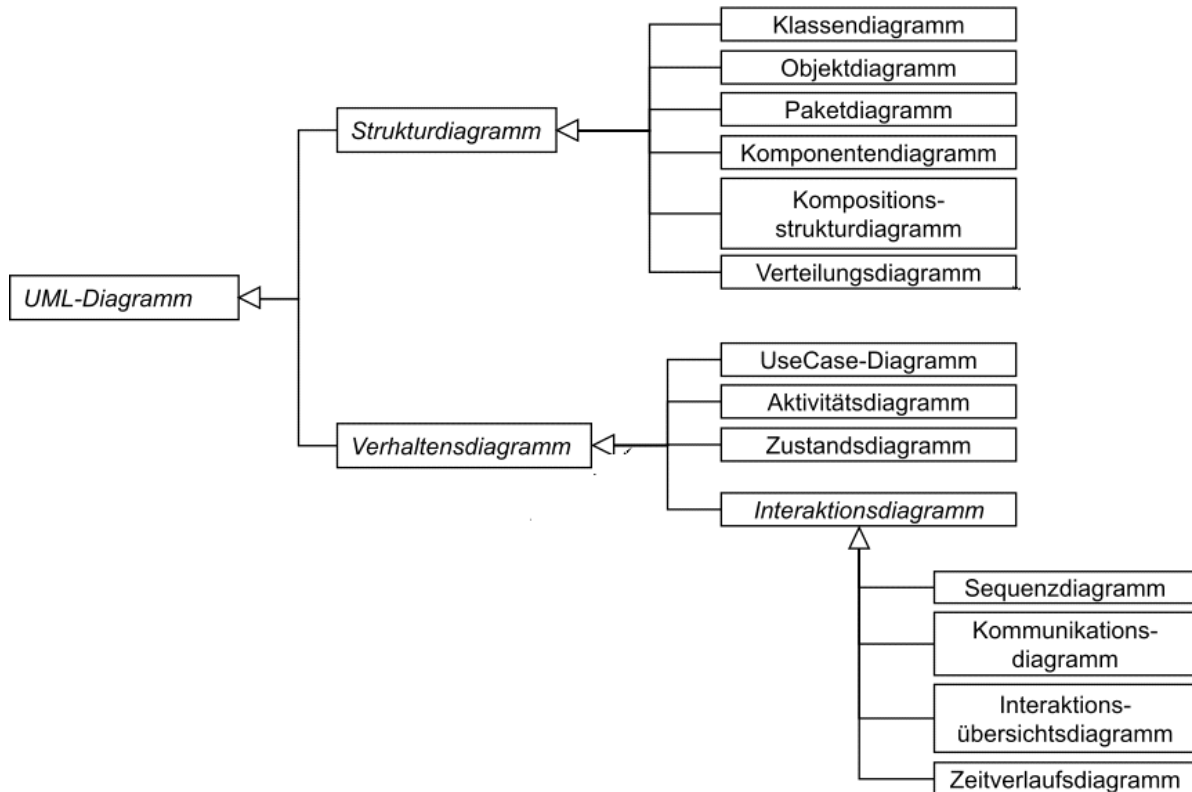
- Software wird umfangreicher, nicht kleiner
 - ◆ Windows NT 5.0 ~ 40 Millionen Zeilen Code
 - ◆ Kein Programmierer kann diese Menge Code alleine bewältigen.
- Code ist oft von Entwicklern, die an der Entwicklung nicht mitgewirkt haben, nicht direkt zu verstehen
- Wir brauchen einfachere Repräsentationen für komplexe Systeme
 - ◆ Modellierung ist ein Mittel um mit Komplexität umzugehen
- Standardisierte graphische Notation für alle Aktivitäten der Softwareentwicklung
 - ◆ Nutzen für Anforderungserhebung, Entwurf, Implementierung, Einsatz („deployment“), ...
 - ◆ Besondere Unterstützung für objektorientierte Modellierung
 - ◆ Standardisiert durch die OMG (Object Management Group) → www.omg.org
- Bietet zahlreiche Diagrammtypen für verschiedene Sichten von Software
 - ◆ statische Sichten auf die Struktur
 - ◆ dynamische Sichten auf das Verhalten

Bezug von UML zu gängigen OO Sprachen



- Entstanden aus der Zusammenführung dreier führender objekt-orientierten Methodologien
 - ◆ OMT (James Rumbaugh) – Klassendiagramme, Sequenzdiagramme, ...
 - ◆ OOSE (Ivar Jacobson) – Anwendungsfalldiagramme, ...
 - ◆ Booch (Grady Booch) – Software-Prozess → „Unified Process“

UML Diagrammtypen



Modellieren Sie mit verschiedenen Diagrammen die Simulation einer Uhr !!

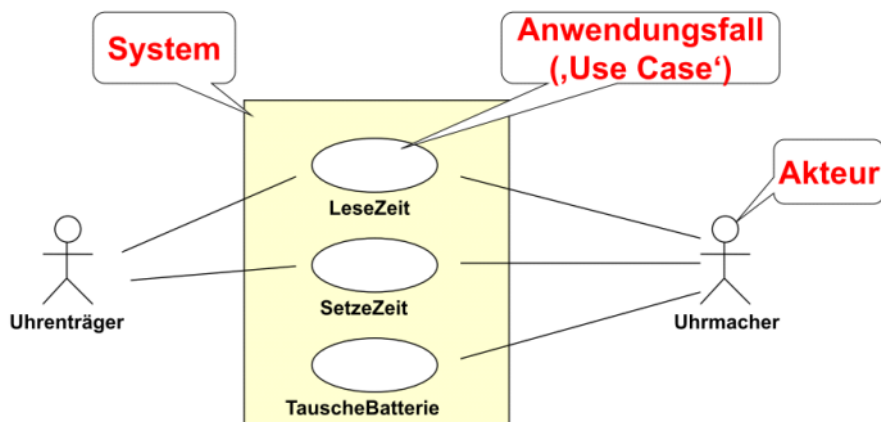
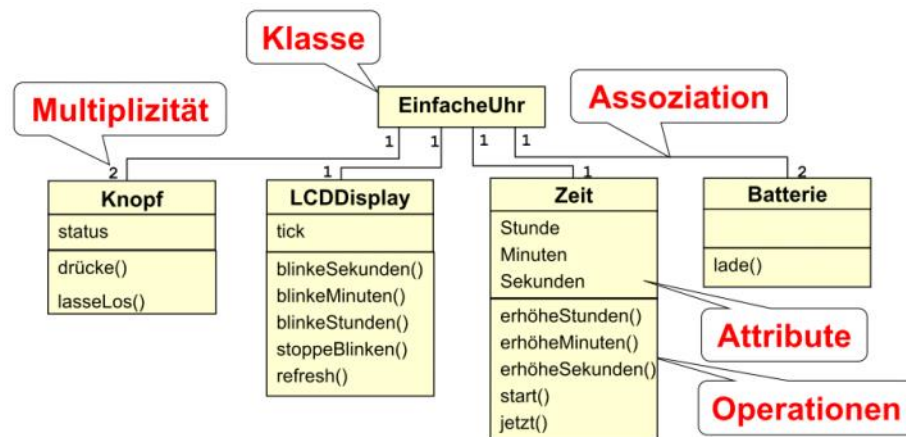
- Funktionale Aspekte
- Statische Aspekte
- Dynamische Aspekte



Modellieren Sie mit verschiedenen Diagrammen die Simulation einer Uhr !!

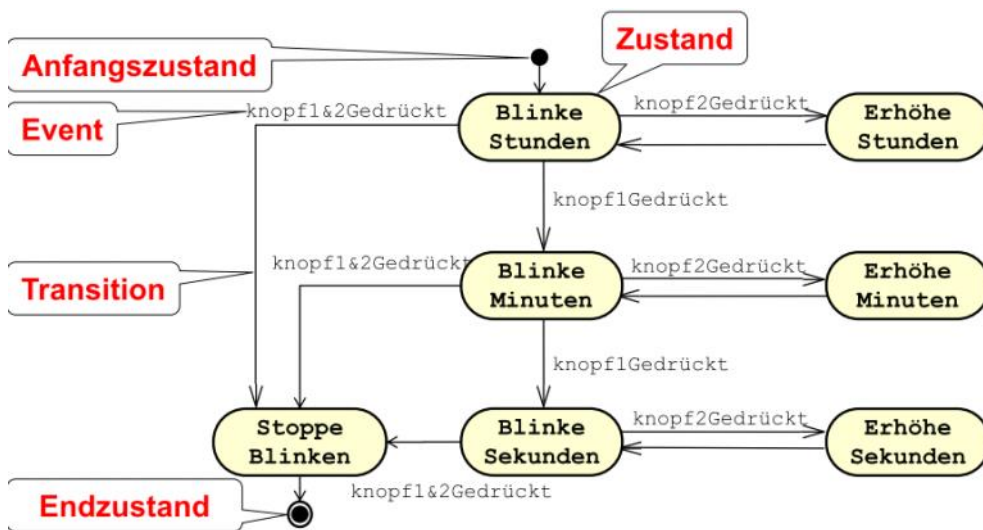
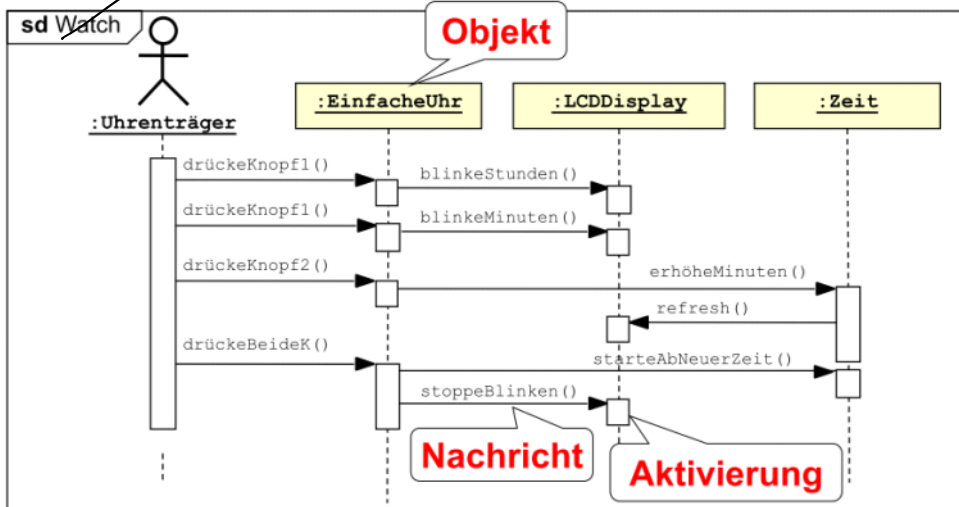


- Funktionale Aspekte
- Statische Aspekte
- Dynamische Aspekte

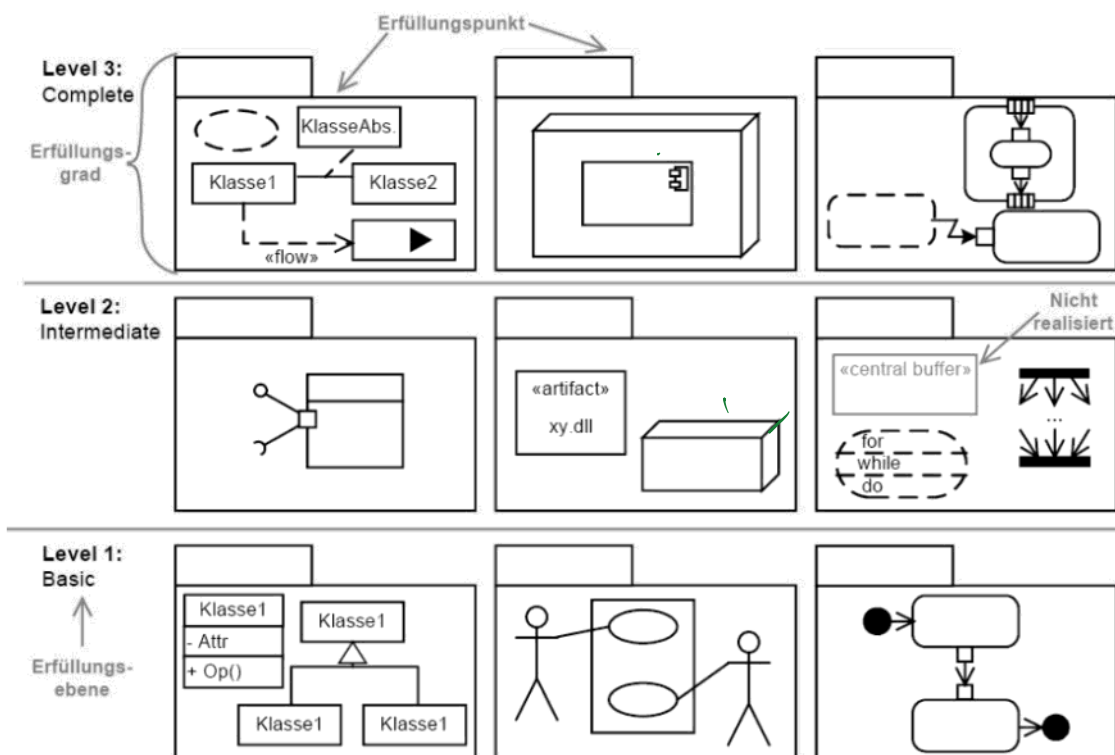


UML-Kurzübersicht: Sequenz-Diagramme

Beispiel



Erfüllungsebenen

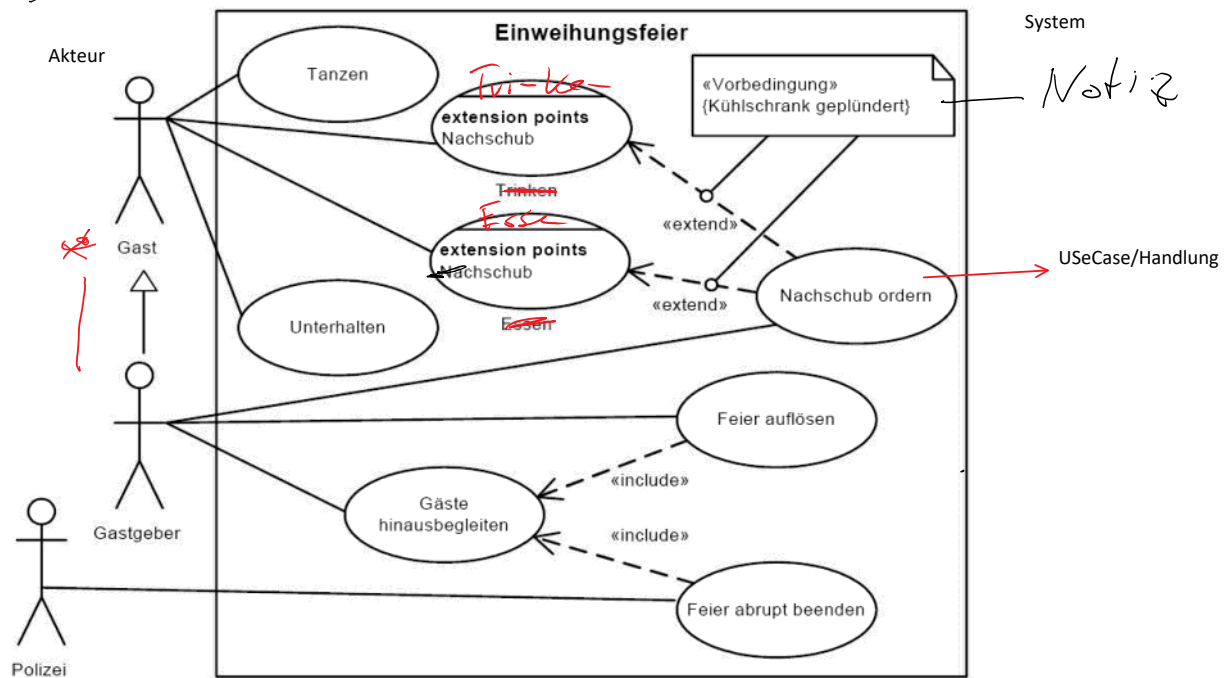


Je höher der Level, desto höher der Detailgrad und die Anzahl der verwendeten Diagramme

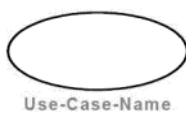
UseCase-Notation

Zeigt die funktionalen Anforderungen des Nutzers an das System

Durch die einfache Sprache entstehen keine Missverständnisse zwischen Kunde und Analyst
Der UseCase wird durch eine Dokumentation ergänzt (GeschäftsprozessSchablone)



Modellelemente



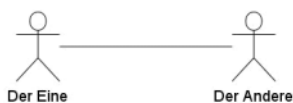
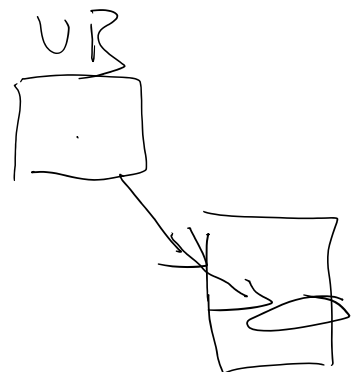
Anwendungsfall

Beschreibt die Funktionalität aus Nutzersicht
Es kann keinen UseCase geben, der nicht in irgendeiner Weise mit einem Nutzer verbunden ist
Hinter dem UseCase steht die Beschreibung des Ablaufs



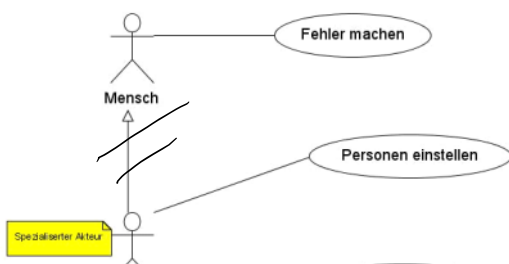
Stellt die externe Rolle dar, welche die Funktionalität des Systems nutzen will.

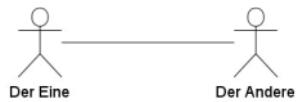
Dies kann auch eine Fremdsystem sein



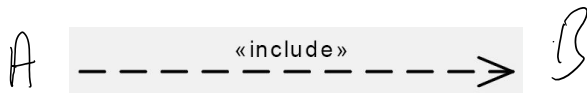
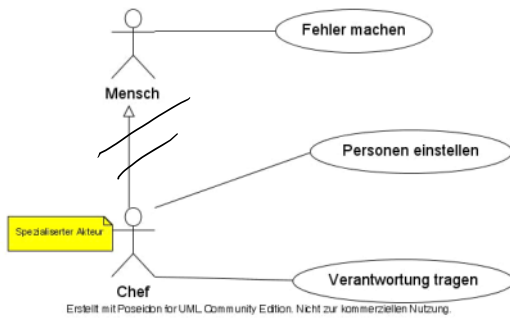
Beziehungen

Bilden die Interaktionen zwischen den verschiedenen Elementen des UC-Diagramms ab
Vererbung zwischen Akteuren ist möglich

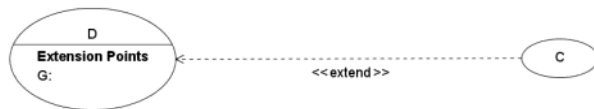




Beziehungen
Bilden die Interaktionen zwischen den verschiedenen Elementen des UC-Diagramms ab
Vererbung zwischen Akteuren ist möglich



UC A bindet **immer** den UC B mit ein.
UC B ist sozusagen nur ausgelagert. UC B kann damit mehrfach verwendet werden, er kann aber auch alleine stehen



UC C **erweitert** UC D beim Eintreten von best. Bedingungen
d.h. UC C **kann, muss aber nicht** eintreffen

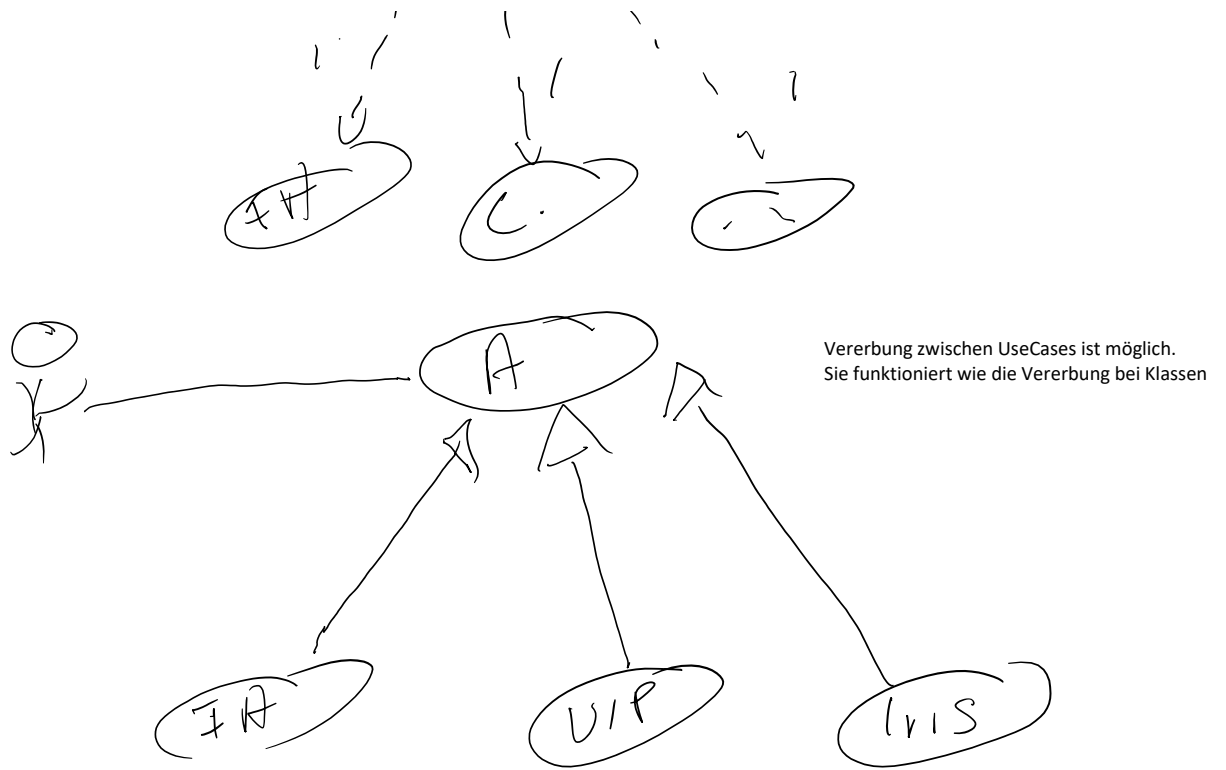
Durch Nachschub kann Essen unter best. Bedingungen erweitert

ExtensionPoints definieren die jeweilige Bedingung



Vererbung:
- Siehe Vererbung im Klassendiagramm
- Die Unter-UseCases **ersetzen** den Ober-UseCase





UseCase - Übung

Akteure:

Die Auftraggeber einer Online-Videothek haben eine erste Vorstellung, wie eine Benutzer-verwaltung aussehen und welche Funktionen es geben soll. Ein Entwickler macht sich in einem Gespräch diesbezüglich folgende Notizen:

UseCase;

Um die Funktionen der Online-Videothek zu nutzen, müssen Kunden sich zuerst einloggen. Dabei wird der Benutzerstatus überprüft. Die Kunden können anschließend Filme ausleihen, oder ihr Guthaben auffüllen.

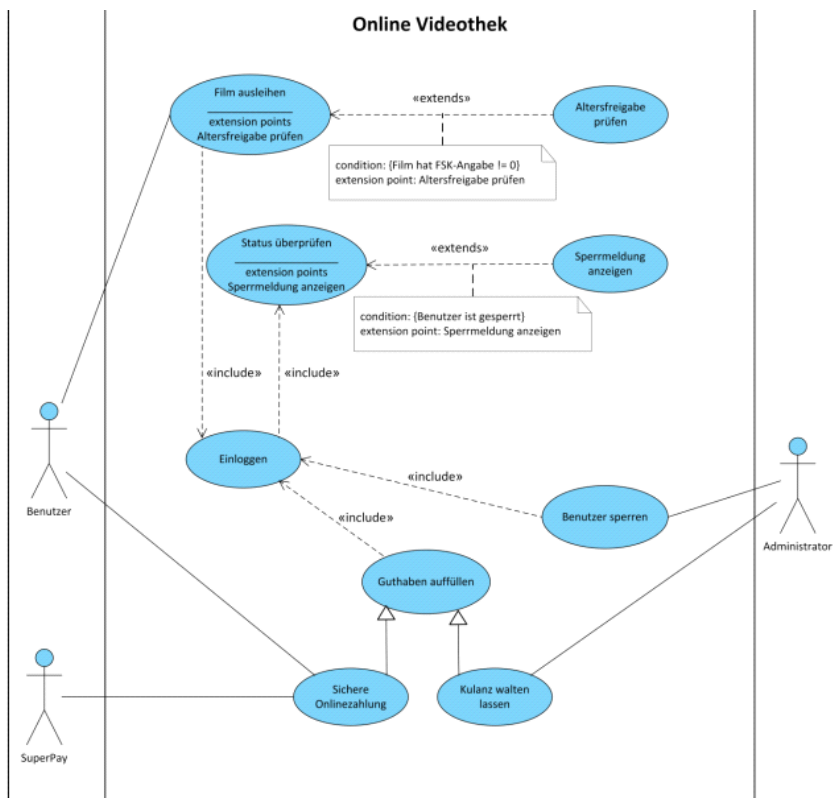
Bei der Ausleihe wird das Alter des Kunden mit der FSK-Angabe des Films abgeglichen und gegebenenfalls die Ausleihe des Films verweigert. Für die sichere Online-Zahlung bei der Guthabensauffüllung verwendet das System den externen Dienstleister „SuperPay“.

Ein Administrator soll (ebenfalls nach einem Login) in der Lage sein, aus Kulanz direkt das Guthaben der Kunden zu erhöhen.

Der Administrator kann ferner Benutzer sperren, wenn Sie gegen die AGB verstoßen haben. In diesem Fall kann ein Benutzer sich nicht mehr am System anmelden. Ein Anmeldeversuch wird mit einer entsprechenden Meldung abgebrochen.

a) Erstellen Sie für obige Notizen ein Use Case-Diagramm, das die relevanten Anwendungsfälle darstellt und untereinander in Beziehung setzt. Identifizieren Sie die relevanten Akteure.

Diskussionen



Aufgabe zu UseCase

- a) Modellieren Sie die folgenden Zusammenhänge in einem *Anwendungsfalldiagramm*.

Erstellt werden soll ein System zur Vermittlung von Mitfahrgelegenheiten. Die Nutzer des Systems können in Fahrer und Mitfahrer unterteilt werden. Fahrer können ihre Fahrten im System eingeben. Dabei kann zwischen dem Eingeben von so genannten **Etappenfahrten** und **einfachen Fahrten** unterschieden werden. Beide Eingabevorgänge weisen viele Übereinstimmungen, jedoch auch einige Besonderheiten auf.

Mitfahrer können nach Mitfahrgelegenheiten suchen. Sowohl das Suchen als auch das Eingeben von Fahrten beinhaltet das Erfassen von Streckenpunkten. Außerdem können sowohl Fahrer als auch Mitfahrer Mitfahrgelegenheiten absagen. (6T)

Sommer_2010

2. Handlungsschritt (25 Punkte)

Ein Ziel des Projektes ist die Verbesserung des Kundenservice. Im Rahmen dieses Vorhabens soll eine neue Internetpräsenz erstellt werden.

- a) Die neue Internetpräsenz soll unter anderem folgende Funktionalität bieten:

Alle Besucher der Webseite können allgemeine Informationen der Krake AG abrufen (Leistungs- und Produktpotfolio und AGB).

Ein Kunde kann Reservierungsanfragen stellen sowie Buchungen und Stornierungen durchführen. Von einem Neukunden werden zunächst die Kundendaten erfasst. Für Buchungen und Stornierungen muss sich ein Kunde einloggen; seine eingegebenen Daten werden überprüft.

Erstellen Sie ein entsprechendes Anwendungsfalldiagramm.

(15 Punkte)

So_2012_ga1_awp

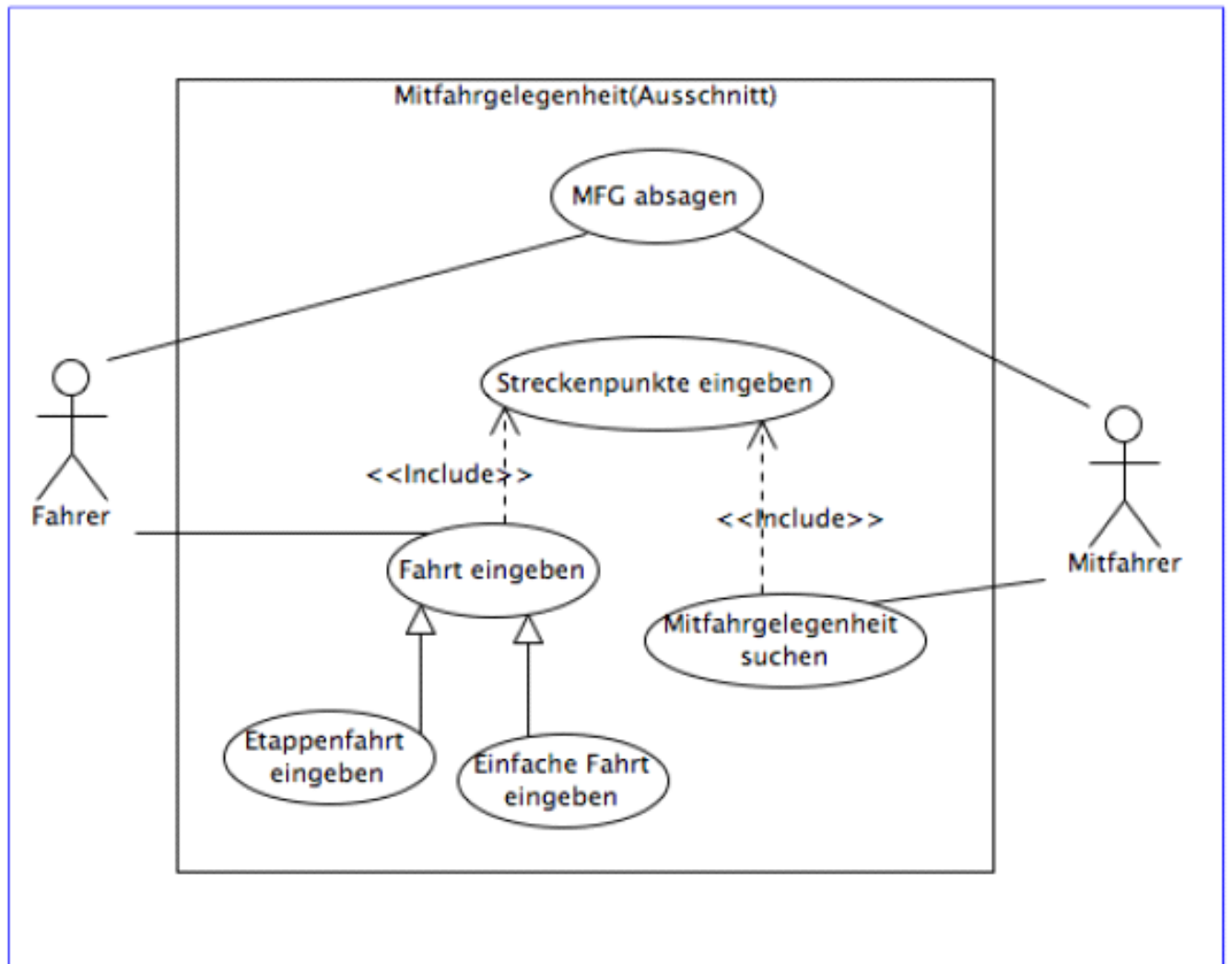
1. Handlungsschritt (25 Punkte)

Die B&G GmbH vermietet auch exklusive Ferienhäuser. Die Immo-IT GmbH soll dazu eine Anwendung entwickeln, über die im Internet Ferienhäuser angeboten und gebucht werden können.

- a) Das System soll Folgendes ermöglichen:
- Ein Vermieter stellt ein Ferienhaus ein.
 - Ein Kunde sucht ein Ferienhaus.
 - Ein Kunde prüft die Verfügbarkeit eines ausgewählten Ferienhauses.
 - Ein Kunde stellt einen Reservierungsauftrag und gibt alle erforderlichen Daten ein.
 - Ein Vermieter reserviert ein Ferienhaus und verschickt eine Reservierungsbestätigung an den Kunden.

Erstellen Sie ein UML-Anwendungsfalldiagramm.

(12 Punkte)



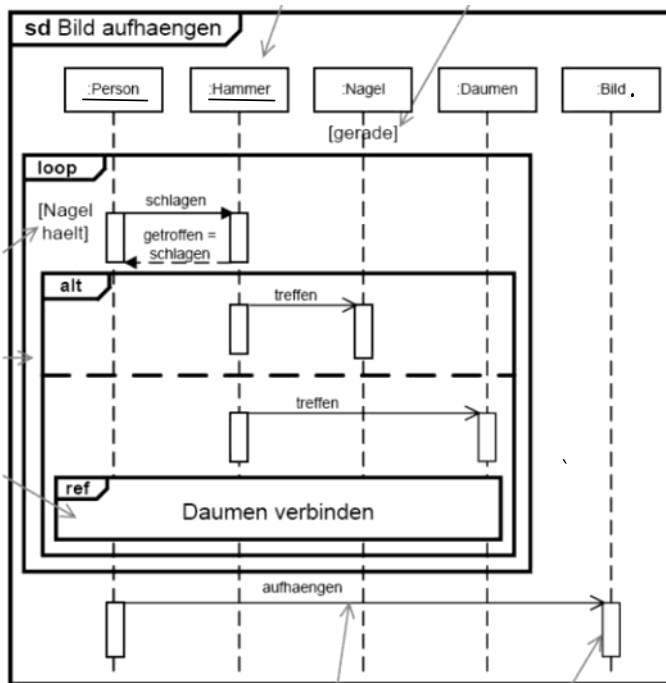
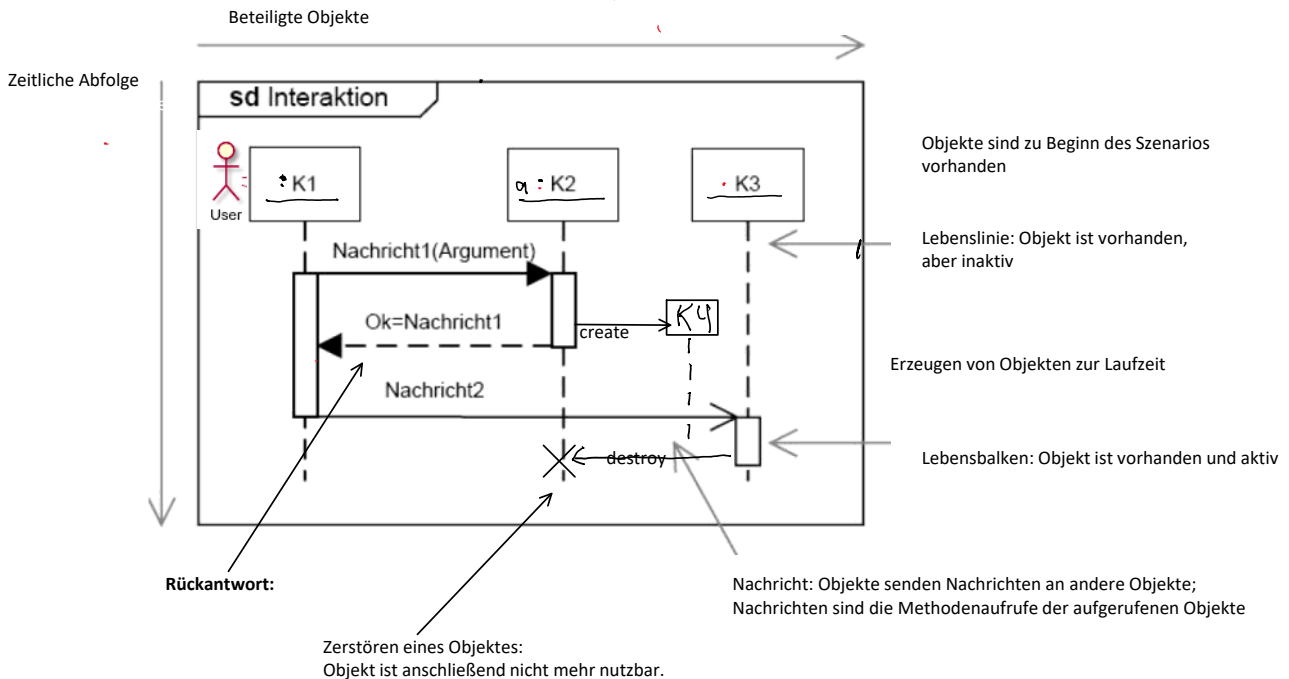
Jeweils einen halben Punkt auf: System, Fahrer, Mitfahrer, alle Anwendungsfälle, include bei Profileingabe, Vererbung bei Profileingabe, include bei Streckenpunkten.

Sequenzdiagramm



uml_seque
nz

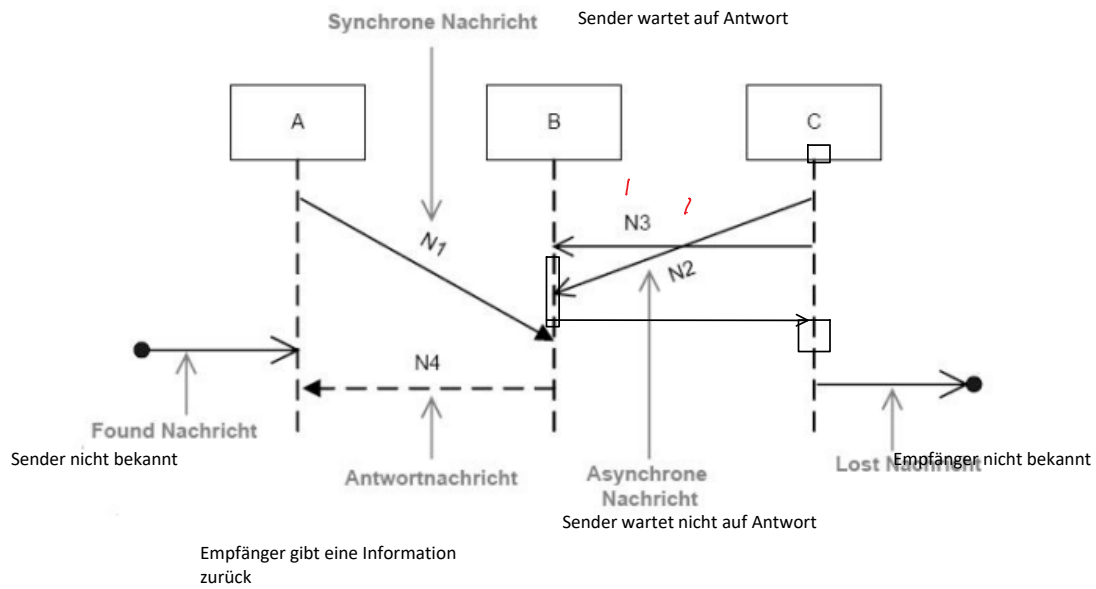
Sequenzdiagramme beschreiben häufig einen UseCase und stellen die Kommunikation zwischen den beteiligten Objekten innerhalb eines UseCases dar, d.h. die Abfolge der verschiedenen Methodenaufrufe zwischen den Objekten. Sie können auch Teilausschnitte eines UseCases abbilden und in ihrem Detailgrad(Methode + Parameter + Datentypen + Rückgabewerte) variieren.



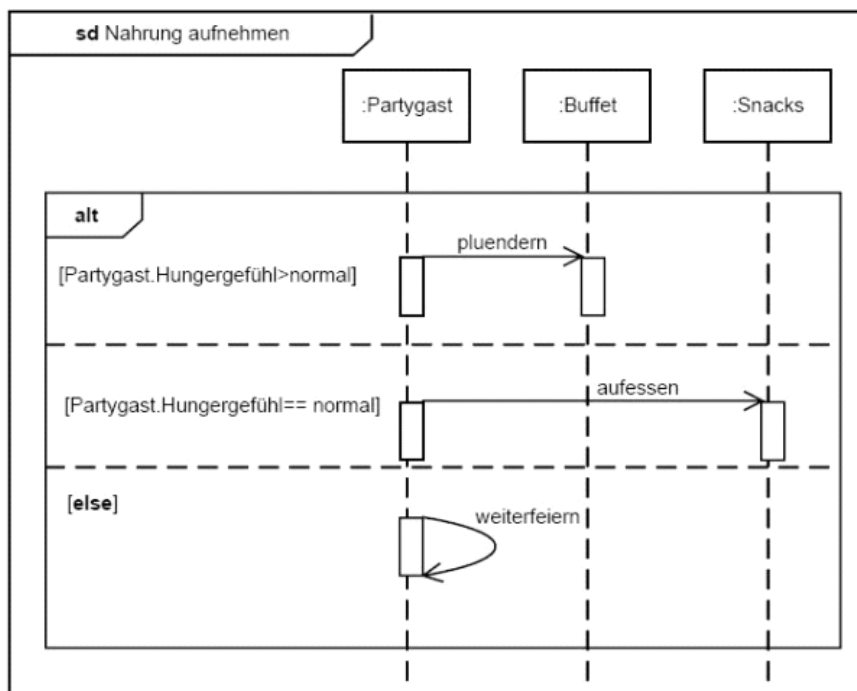
Komplexere Abläufe sind durch die sog. **Kombinierten Fragmente** darstellbar

Logische Programmstrukturen wie Schleifen und Bedingungen können damit abgebildet werden, erschweren aber gleichzeitig die Lesbarkeit des Diagramms

Nachrichten:
Sie sind die wichtigsten Komponenten des Sequenzdiagramms, weil dadurch die statische Struktur des Klassendiagramms eine dynamische Umsetzung erhält



Kombinierte Fragmente – alt



Alt = Alternativen (if else)
 Opt Optional
 Par = Parallel
 Loop Schleife

