



UML

unified modeling  
language



# Inhalt

- Definition
- Entwicklung
- Anwendungsfalldiagramm
- Klassendiagramm
- Aktivitätsdiagramm
- Kollaborationsdiagramm
- Sequenzdiagramm
- Zustandsdiagramm
- Komponentendiagramm
- Einsatzdiagramm



# Definition

- Sprache zur Spezifikation, Visualisierung, Konstruktion und Dokumentation von Modellen
- für Softwaresysteme, Geschäftsmodelle und andere Nicht-Softwaresysteme
- Satz von Notationen zur Formung einer allgemeinen Sprache zur Softwareentwicklung
- Entwurf und die Entwicklung von Softwaremodellen auf einheitlicher Basis diskutierbar



# Entwicklung

- Entwickelt von Grady Boch, Ivar Jacobsen und Jim Rumbaugh
- Seit 1998 durch OMG standardisiert
- Heute ISO standardisiert (ISO/IEC 19501)



# Modellelemente

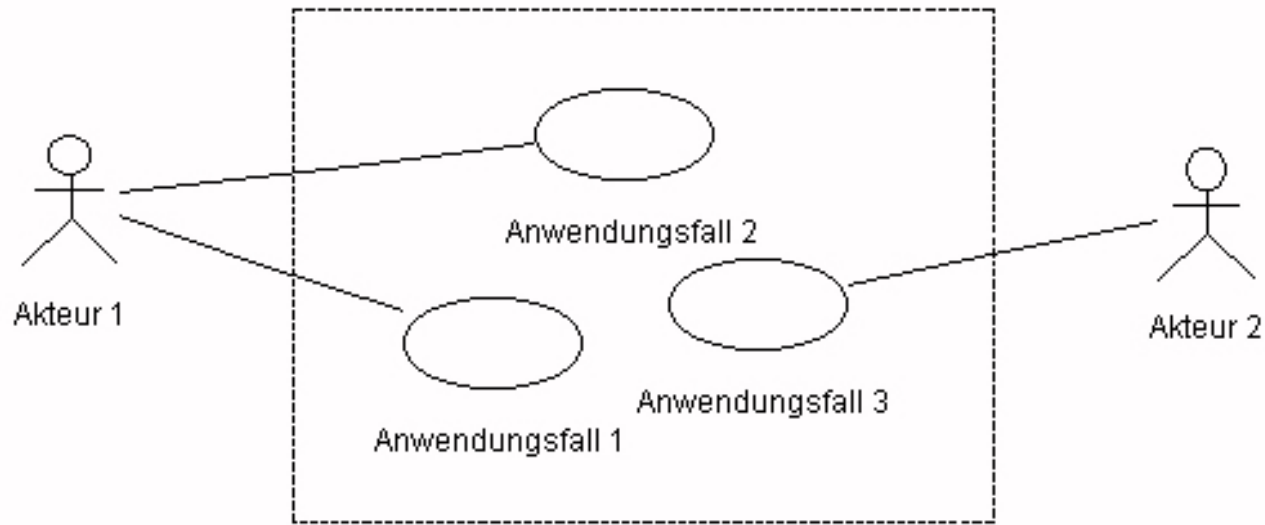
- Modellelemente der UML werden nach Diagrammtypen gegliedert:
- Anwendungsfalldiagramm
- Klassendiagramm
- Aktivitätsdiagramm
- Kollaborationsdiagramm
- Sequenzdiagramm
- Zustandsdiagramm
- Komponentendiagramm
- Einsatzdiagramm



# Anwendungsfalldiagramm (Use case diagram)

- beschreibt Zusammenhänge zwischen verschiedenen Anwendungsfällen untereinander und beteiligten Akteuren
- zeigt Struktur und Zusammenhänge von verschiedenen Geschäftsvorfällen

# Anwendungsfalldiagramm (Use case diagram)

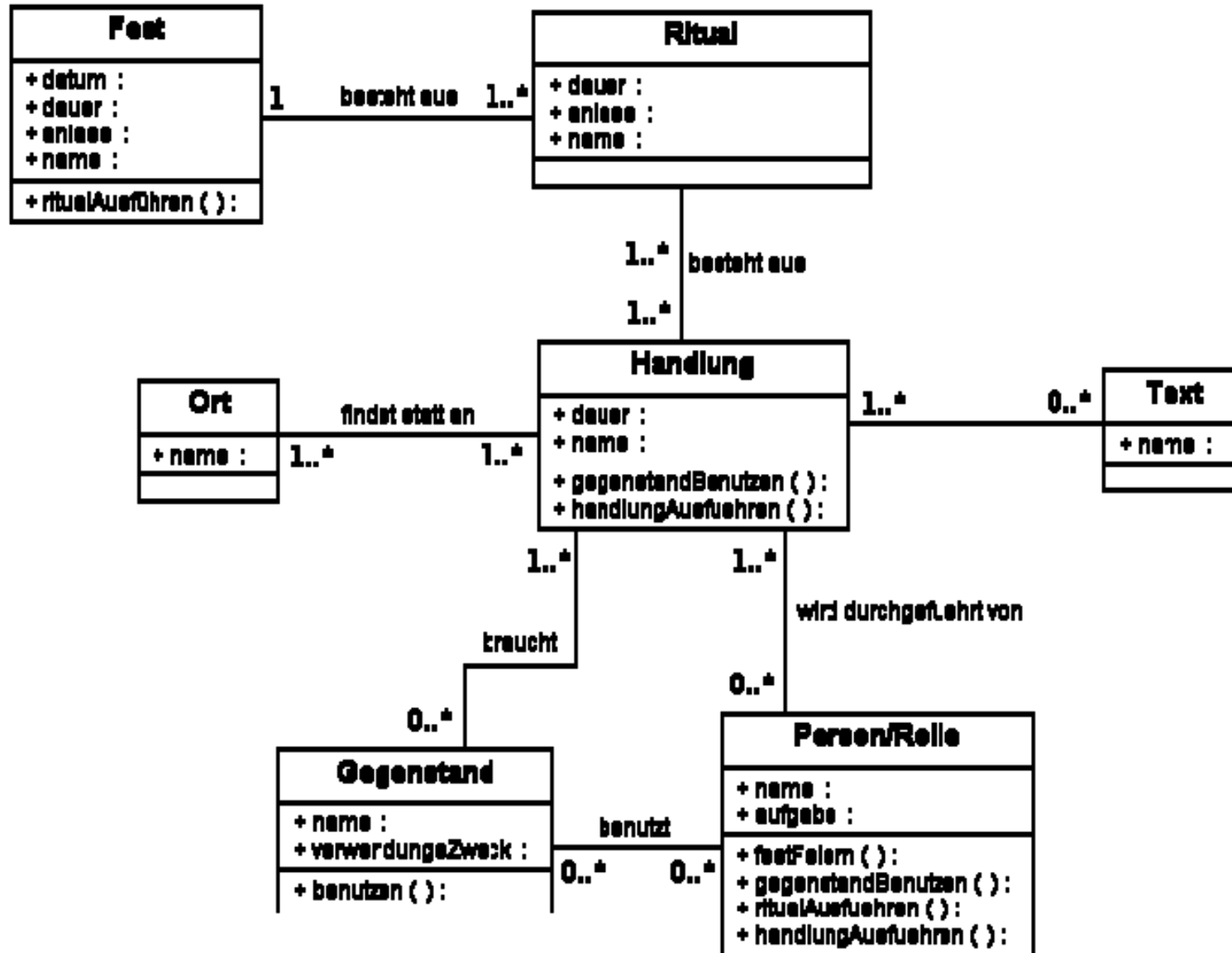




# Klassendiagramm

- Klasse =Zusammenfassung gleichartiger Objekte, enthalten Konstruktionsbeschreibung für Objekte die mit ihnen erzeugt werden
- Objekte sind agierenden Grundelemente einer Anwendung
- Gleichartigkeit bezieht sich auf Eigenschaften (Attribute) und/oder auf Fähigkeiten (Operationen/Methoden) der Objekte einer Klasse
- Verhalten der Objekte wird durch die Möglichkeit eines Objektes Nachrichten zu empfangen und zu verstehen beschrieben dazu benötigt Objekt bestimmte Operationen.

# Klassendiagramm

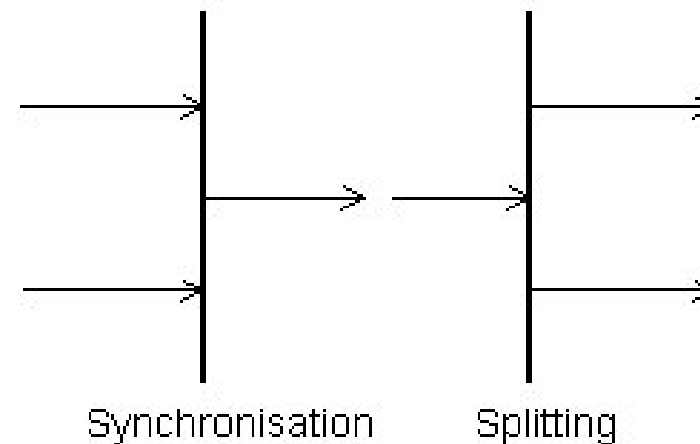
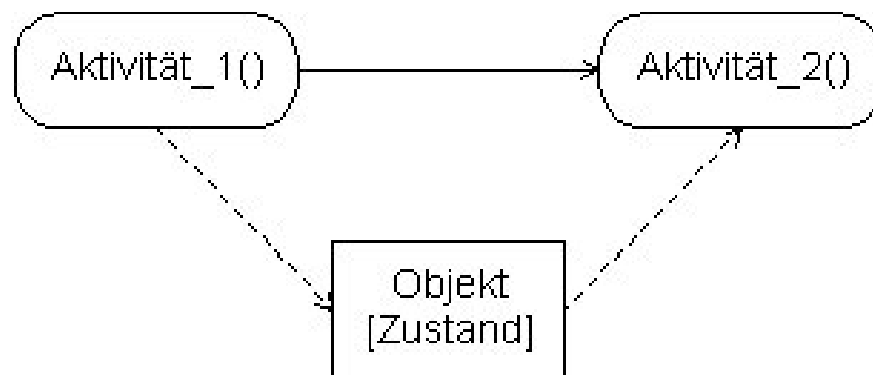
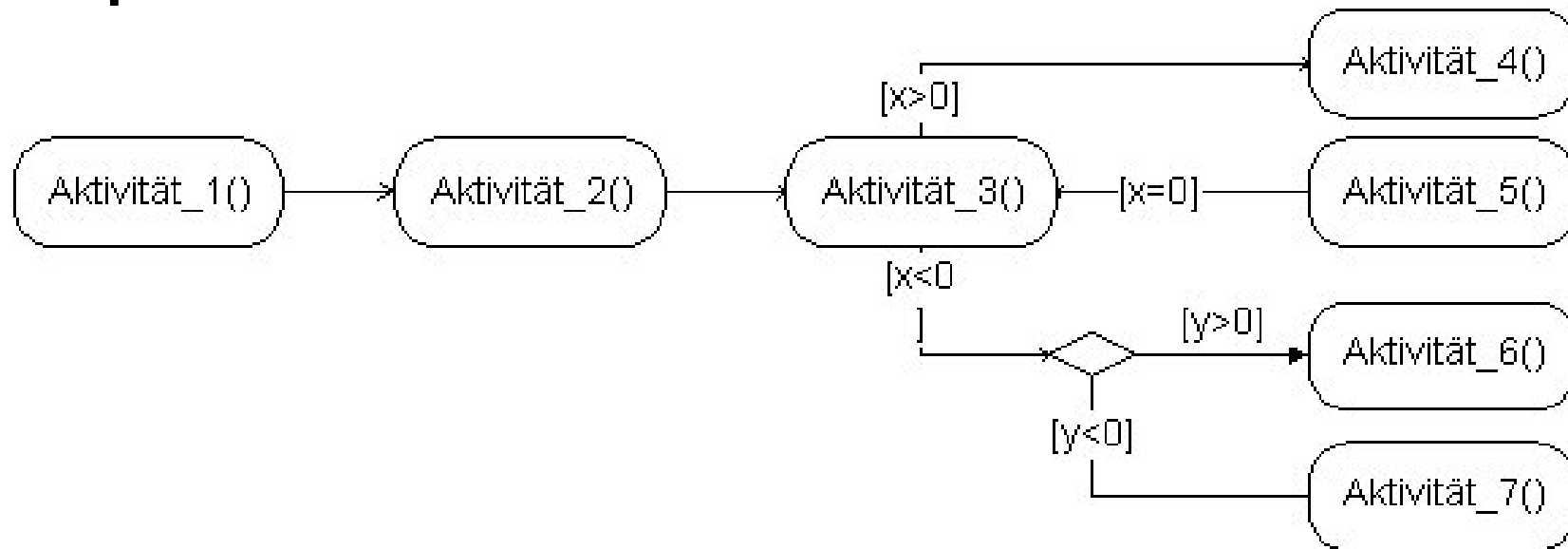




# Aktivitätsdiagramm

- Beschreibung von Objekte mittels der Aktivitäten, die sie während des Programmablaufes vollführen,
- Aktivität = einzelner Schritt innerhalb eines Programmablaufes, ein spezieller Zustand eines Modellelementes, eine interne Aktion sowie eine oder mehrere von ihm ausgehende Transitionen
- Aktivitätsdiagramm als Sonderform eines Zustandsdiagrammes, dessen Zustände der Modellelemente in der Mehrzahl als Aktivitäten definiert sind

# Aktivitätsdiagramm



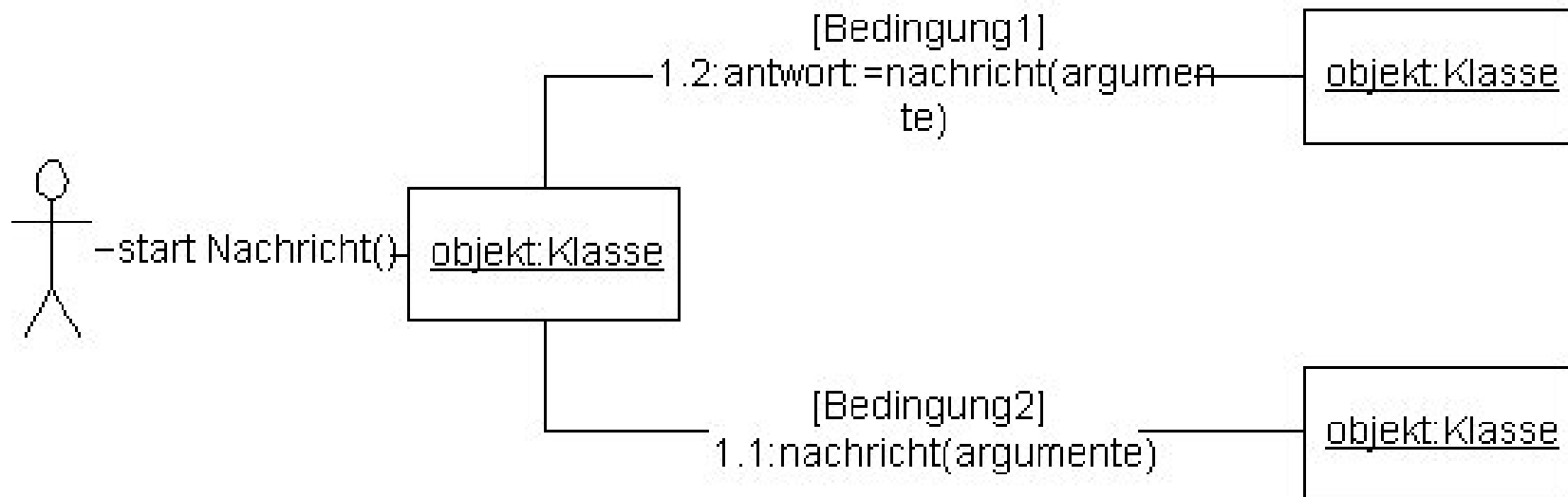


# Kollaborationsdiagramm

- visualisiert einzelnen Objekte und Zusammenarbeit untereinander
- für Programmablauf und Verständnis des selbigen wichtigen kommunikativen Aspekte zwischen den einzelnen Objekten ereignisbezogen dargestellt
- zeitlicher Verlauf der Interaktionen durch eine Nummerierung der Nachrichten symbolisiert
- einzelne Objekte können Nachrichten austauschen, ein Objekt kann sich auch stets selbst Nachrichten zusenden, ohne da eine Assoziation vorhanden sein müsste
- für die Darstellung von Entwurfs-Sachverhalten, beinhaltet stets kontextbezogene begrenzte Projektionen des Gesamtmodells



# Kollaborationsdiagramm



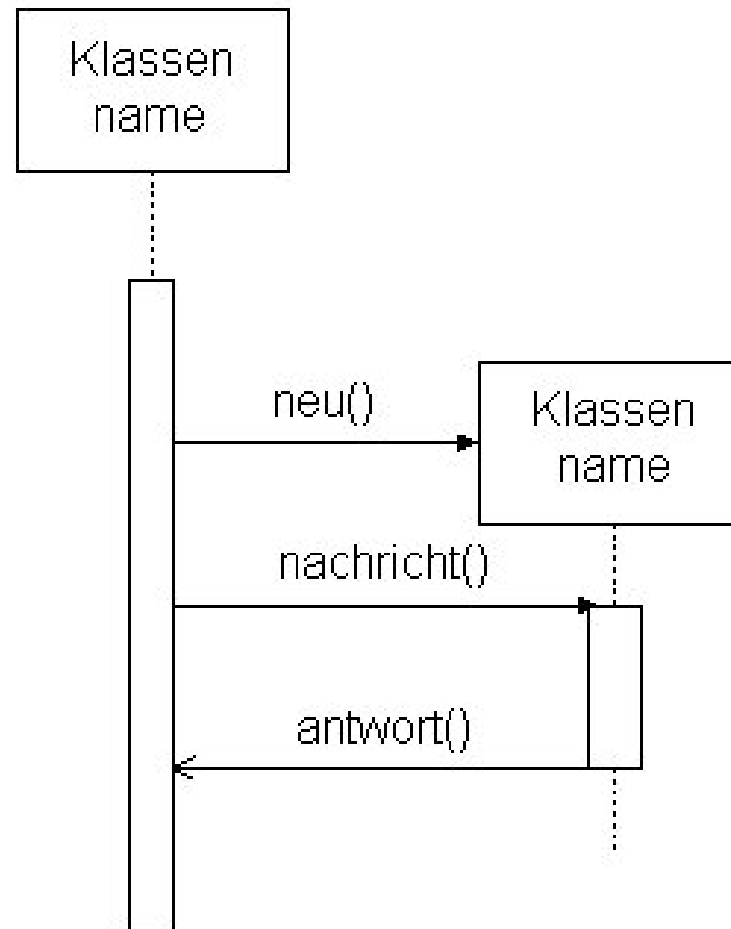


# Sequenzdiagramm

- beschreibt die Interaktionen zwischen Modellelementen ähnlich, wie bei einem Kollaborationsdiagramm, jedoch steht zeitliche Verlauf des Nachrichtenaustausches im Vordergrund
- Die Zeitlinie verläuft senkrecht Objekte werden durch senkrechte Lebenslinien beschrieben und die gesendeten Nachrichten waagerecht entsprechend ihres zeitlichen Auftretens eingetragen



# Sequenzdiagramm



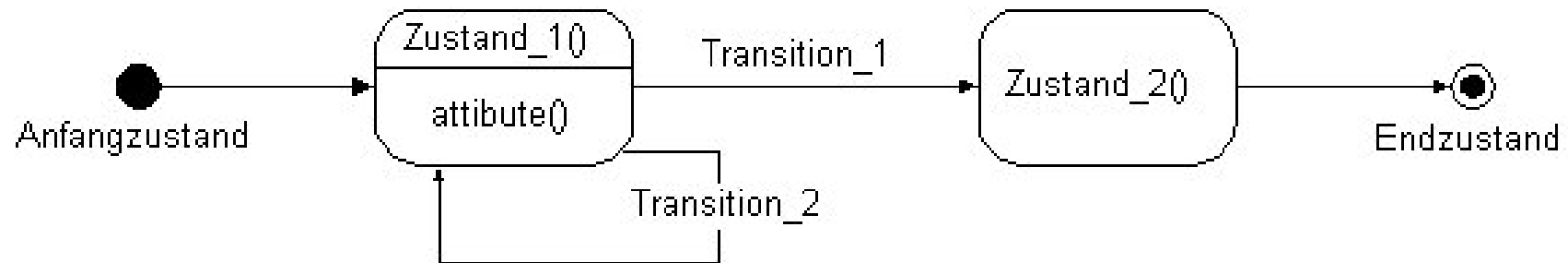


# Zustandsdiagramm

- beschreibt hypothetische Maschine, die sich zu jedem Zeitpunkt in einer Menge endlicher Zustände befindet. Sie besteht aus:
  - einem Anfangszustand
  - einer endlichen Menge von Zuständen
  - einer endlichen Menge von Ereignissen
  - einer endlichen Anzahl von Transitionen, die den Übergang des Objektes von einem zum nächsten Zustand beschreiben
  - einem oder mehrerer Endzustände



# Zustandsdiagramm



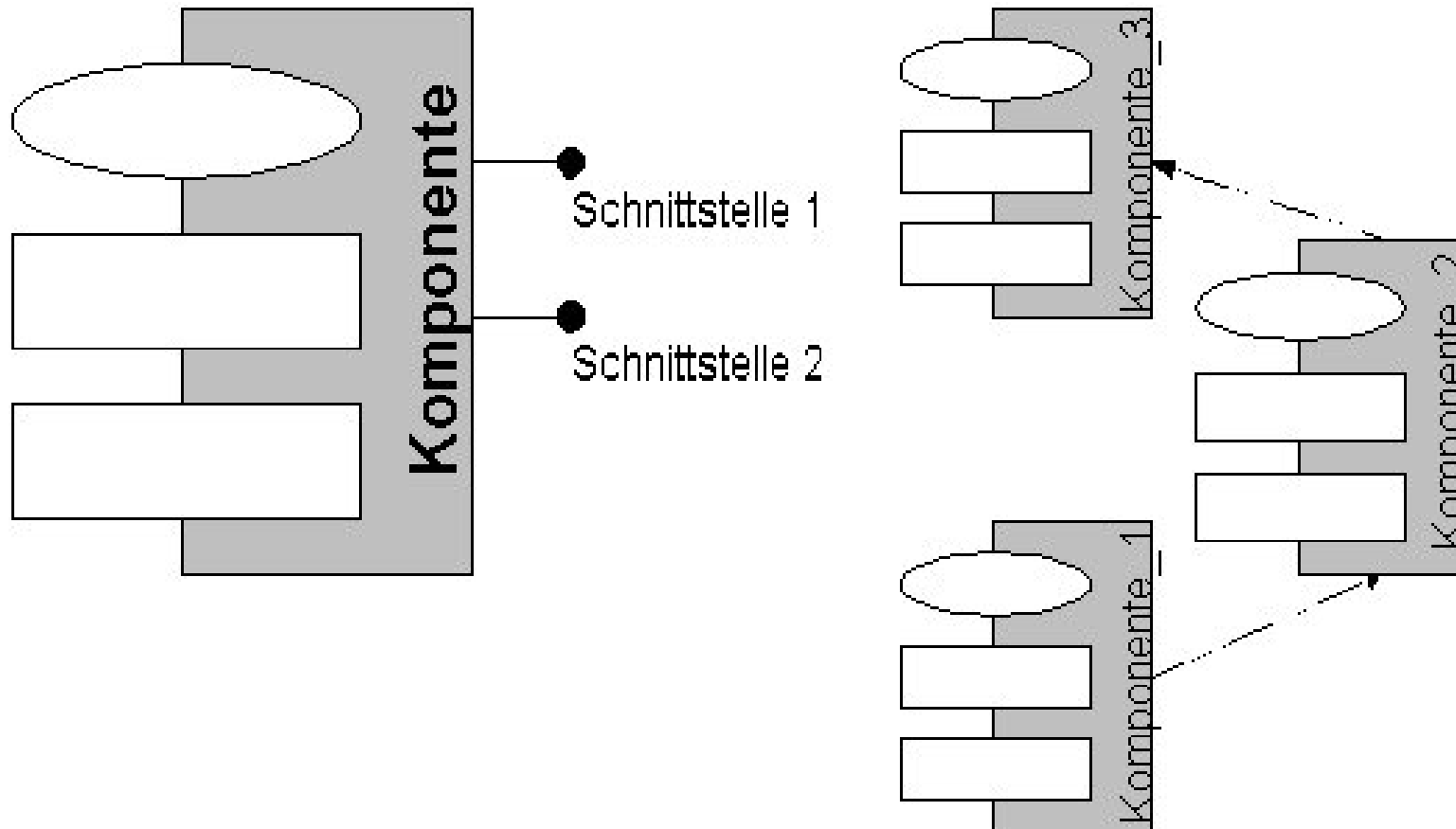


# Komponentendiagramm

- bei späterer Implementierung der Softwarelösung
- Wenn Compiler- und Laufzeitabhängigkeiten klar sind
- Darstellung von Zusammenhängen der einzelnen Komponenten der späteren Softwarelösung



# Komponentendiagramm



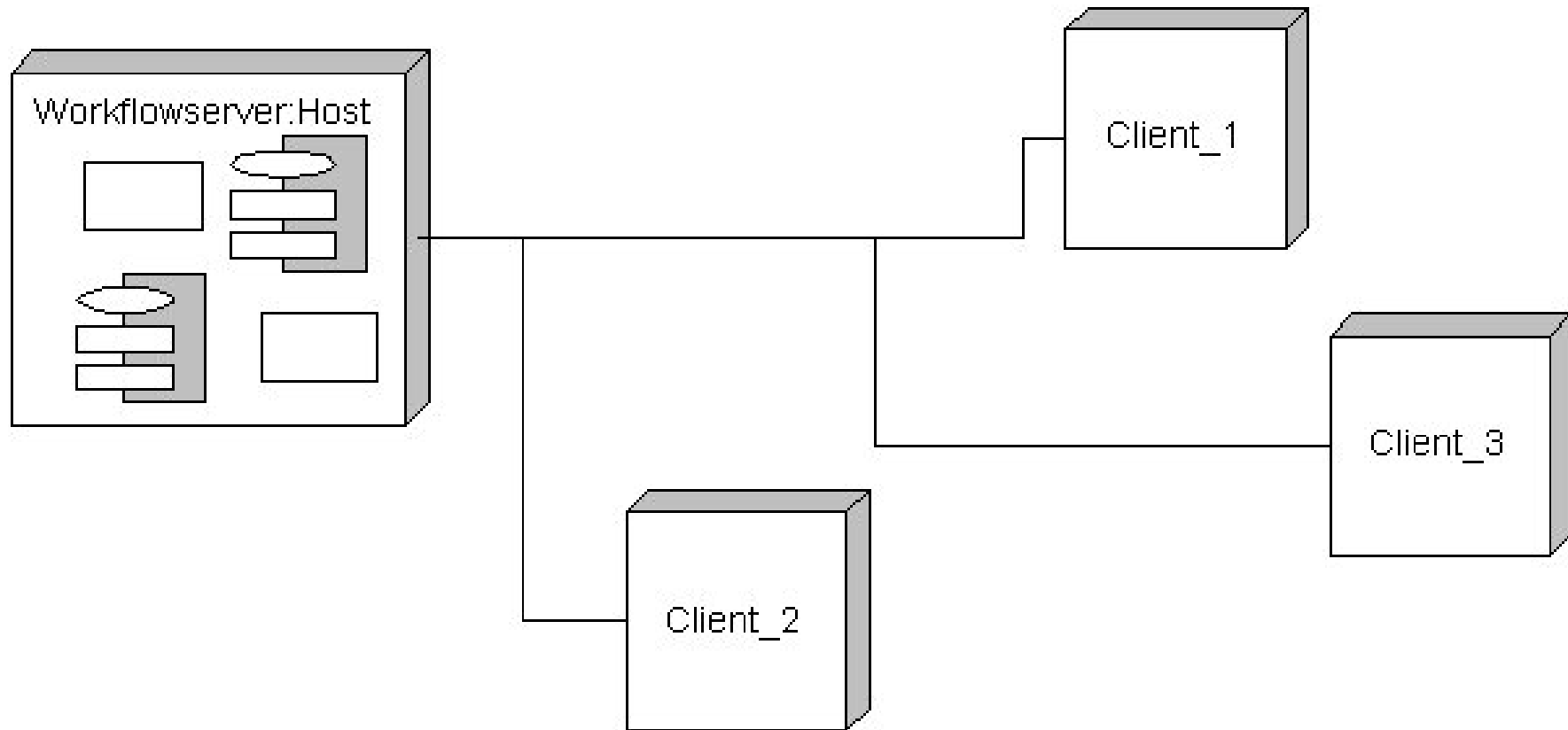


# Einsatzdiagramm

- Beschreibt welche Komponenten (Objekte) auf welchen Knoten ablaufen, d.h. wie diese konfiguriert sind und welche Abhängigkeiten bestehen



# Einsatzdiagramm





# Literatur

- [www.collide.info](http://www.collide.info)
- <http://ivs.cs.uni-magdeburg.de/>
- <http://de.wikipedia.org>