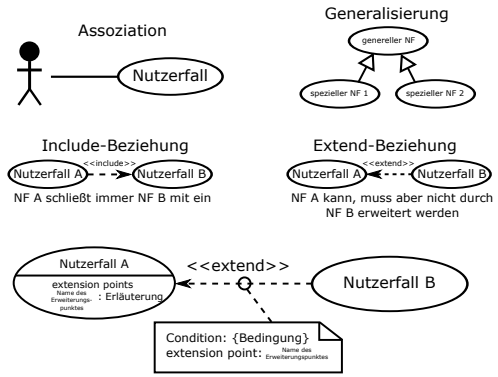
Anwendungsfalldiagramm



Zustandsdiagramm

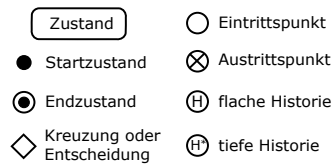
Verhaltenszustandsmaschine (BSM)

Transition ::= [Auslöser] [[Bedingung]] / [Aktion]
Auslöser ::= Ereignisse [(Zuweisungen)]
Ereignisse ::= Ereignisse (,Ereignisse)*
Zuweisungen ::= Zuweisung (,Zuweisung)*
Zuweisung ::= Attribut | Attribut : Typ

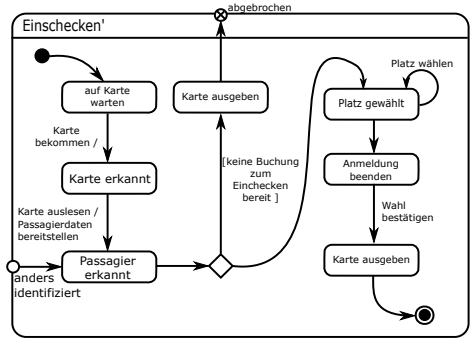
Protokollzustandsmaschine (PSM)

Transition ::= [[Vorbedingung]] Methodenaufruf / [[Nachbedingung]]

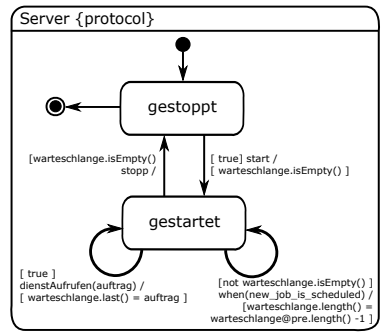
Notationen:



Beispiel: Verhaltenszustandmaschine



Beispiel: Protokollzustandmaschine



Klassendiagramm

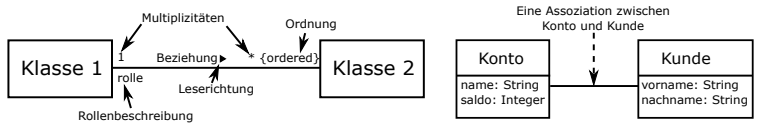
Analysemodell (fachliche Sicht)

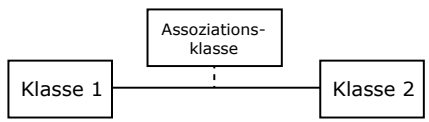
- Klassen sind Fachbegriffe
- Attribute
  - i. Allg. ohne Datentypen
  - ggfs. mit Multiplizitäten
- Methoden ohne Parameter und Rückgabewert
- Bidirektionale Assoziationen/Aggregationen
- Beschreibung von Assoziationen (Multiplizitäten, Rollennamen, Assoziationsnamen (mit Leserichtung)
  - ggfs. mit Qualifer
- Assoziationsklassen und n-äre Beziehungen
- Generalisierung / Spezialisierung (Vererbung)
- Abgeleitete Attribute und Methoden
- Aufzählungen (Enumerationen)

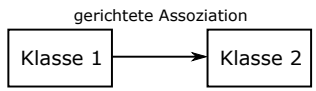
Entwurfsmodell (fachliche+tech. Sicht)

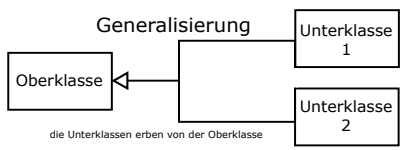
- Klassen -> abstrakt, Interface, Sterotyp, ggf. Klassen streichen, hinzufügen, umbennen
- Attribute -> Sichtbarkeiten, Ableitung, Klassenattribute, Initialisierung, weitere spezielle Eigenschaften
- Operationen -> Parameter, Sichtbarkeiten, Rückgabewert, Klassenoperation
- Assoziationen -> gerichtet, geordnet / sortiert
- Auflösen von Assoziationsklassen / n-äre Beziehungen
- Abhängigkeiten
- Packages
- Hilfsmethoden (Konstruktoren, getter/setter, toString() u.a)

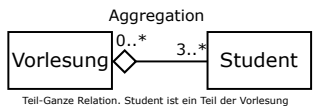
Eine Assoziation beschreibt eine Beziehung zwischen zwei oder mehr Klassen. An den Enden von Assoziationen sind häufig Multiplizitäten vermerkt. Diese drücken aus, wie viele dieser Objekte in Relation zu den anderen Objekten dieser Assoziation stehen.

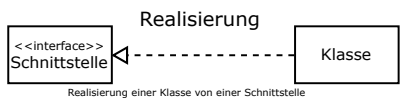


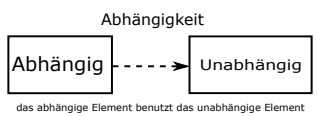












Syntax für Attribute

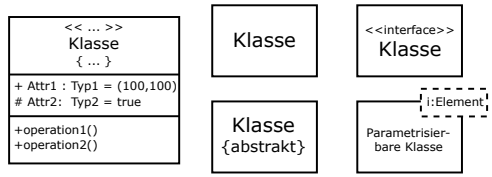
Sichtbarkeit Attributname : Paket :: Typ [Multiplizität Ordnung] = Initialwert {Eigenschaftswerte}
Eigenschaftswerte : {readOnly}, {ordered}, {composite}

Syntax für Operationen

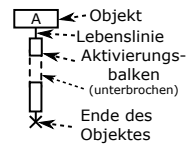
Sichtbarkeit Operationsname (Parameterliste) : Rückgabetyp

Sichtbarkeit:
+ public element
# protected element
– private element
~ package element

Parameterliste: Richtung Name : Typ = Standartwert
Richtung: in, out, inout



Sequenzdiagramm

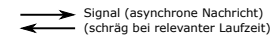


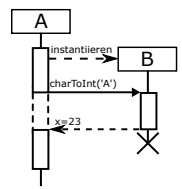
Beschreibung:

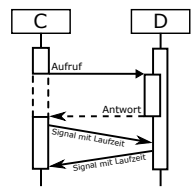
Ein synchroner Aufruf unterbricht den Aktivierungsbalken solange, bis eine synchrone Antwort eintrifft. Eine asynchrone Nachricht verändert nichts am Aktivierungsbalken.

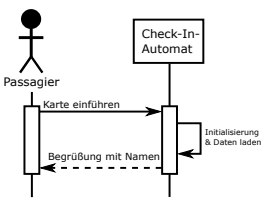
Notationen:

Nachricht ::= Aufruf | Antwort| Signal



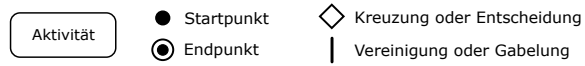






Aktivitätsdiagramm

Notationen:



Beispiel: Aktivitätsdiagramm