



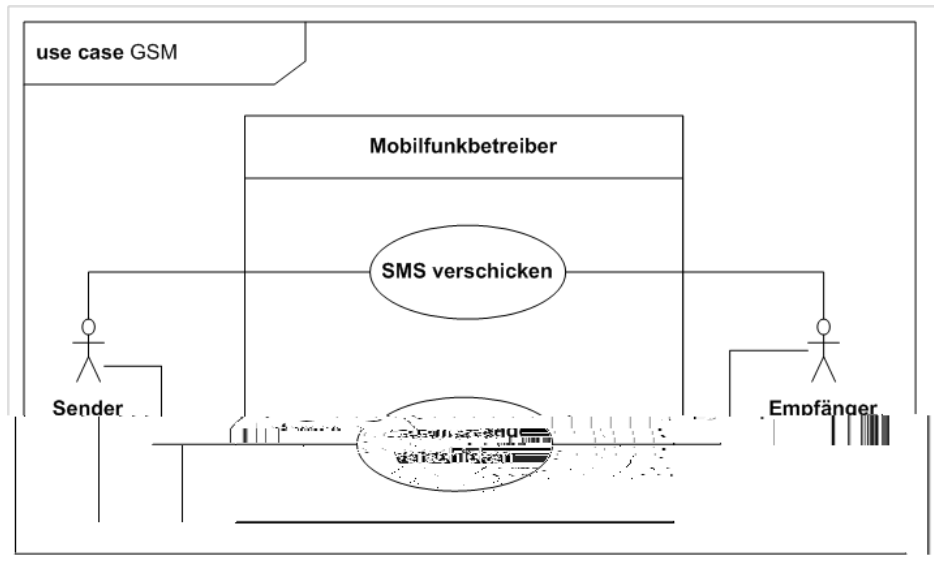
UML

Verhaltens- und Interaktionsdiagramme

1. Verhaltensdiagramme:

Das Anwendungsfalldiagramm ist ein *Verhaltensdiagramm*. Es zeigt eine bestimmte Sicht auf das erwartete Verhalten eines Systems und wird deshalb für die Spezifikation der Anforderungen an ein System eingesetzt. In einem Anwendungsfalldiagramm werden typischerweise Anwendungsfälle und Akteure mit ihren Abhängigkeiten und Beziehungen dargestellt.

Ein Anwendungsfalldiagramm stellt keine Ablaufbeschreibung dar. Es dient als Hilfsmittel zur Anforderungsermittlung für Systeme.

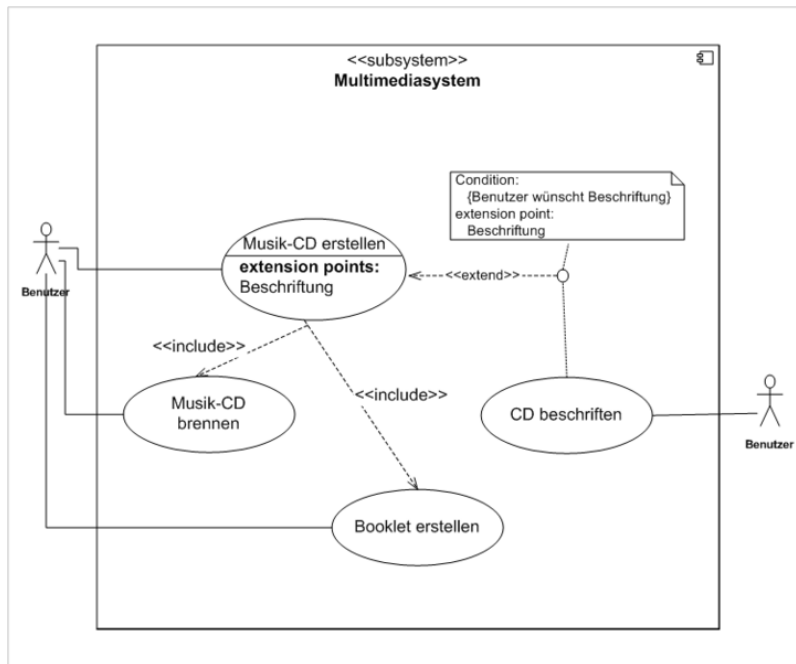


Die Grundelemente des Anwendungsfalldiagramms sind die Akteure („Sender“, „Empfänger“) und die Aktivitäten („SMS verschicken“, „Fotomessage verschicken“), die in einem System („Mobilfunkbetreiber“) ablaufen. Diese Elemente wiederum werden zu einem Use-Case zusammen gefasst („GSM“).

Mittels der Beziehung „include“ werden Anwendungen in ihre Teilanwendung zerlegt und dargestellt. Dies nennt man auch funktionale Dekomposition.

Die Beziehung „extends“ beschreibt eine optionale Anwendung in einem Anwendungsfall. Die Anwendung wird durch eine andere Anwendung erweitert.

Auch eine Beziehung zwischen Akteuren ist möglich. Hierbei handelt es sich um eine Vererbung. Je nach dem von wo man ausgeht, bezeichnet man diese als Generalisierung bzw. Spezialisierung. Wird der Akteur „Kunde“ in „Firmenkunde“ und „Privatkunde“ aufgeteilt, wäre dies eine Spezialisierung.



Dieses etwas komplexere Anwendungsfalldiagramm zeigt die Möglichkeiten der Beziehungen zwischen Anwendungsfällen. Der Anwendungsfall „Musik-CD erstellen“ inkludiert die Anwendungen „Musik-CD brennen“ und „Booklet erstellen“. Wird dieser ausgeführt werden die inkludierten Anwendungen zwangsläufig ebenfalls durchgeführt.

Optional gibt es noch den Anwendungsfall „CD beschriften“, welcher an eine Kondition gebunden ist.

2. Interaktionsdiagramme

Eine Interaktion ist die Spezifikation eines Verhaltens, das am besten über den Austausch von Meldungen zwischen eigenständigen Objekten beschrieben wird.

Wer Interaktionen modelliert, geht davon aus, dass das modellierte System aus einem Netzwerk von Objekten besteht, die untereinander Meldungen austauschen. Modelliert werden Interaktionen als Mengen von Lebenslinien, auf denen Aktionen und andere Verhaltensweisen ablaufen und zwischen denen Nachrichten ausgetauscht werden.

Interaktionen modelliert man graphisch in Kommunikationsdiagrammen, in Sequenzdiagrammen oder in Zeitverlaufdiagrammen.

2.1. Zustandsdiagramm

Das Zustandsdiagramm stellt einen endlichen Automaten in einer UML-Sonderform grafisch dar und wird benutzt, um entweder das Verhalten eines Systems oder die zulässige Nutzung der Schnittstelle eines Systems zu spezifizieren. Ein Zustandsdiagramm zeigt eine Übersicht der Zustände, die der dargestellte Zustandsautomat – beispielsweise ein einzelnes Objekt oder auch ein (Teil-)System – zur Laufzeit annehmen kann und gibt an, aufgrund welcher Ereignisse Zustandsänderungen bzw. -übergänge stattfinden.

Der in einem Diagramm dargestellte Zustandsautomat besteht aus Knoten und (Zustands-) Übergängen, die einen Quell- und einen Zielknoten verbinden. Jeder Knoten ist entweder ein Zustand oder aber ein so genannter Pseudo-Zustand (siehe Abbildung rechts).

Ein Zustand modelliert eine Situation, in der eine bestimmte unveränderliche Bedingung gilt. Meistens ist diese Invariante nur implizit gegeben, will man sie explizit formulieren, kann man sie als Einschränkung dem Zustand zuordnen.

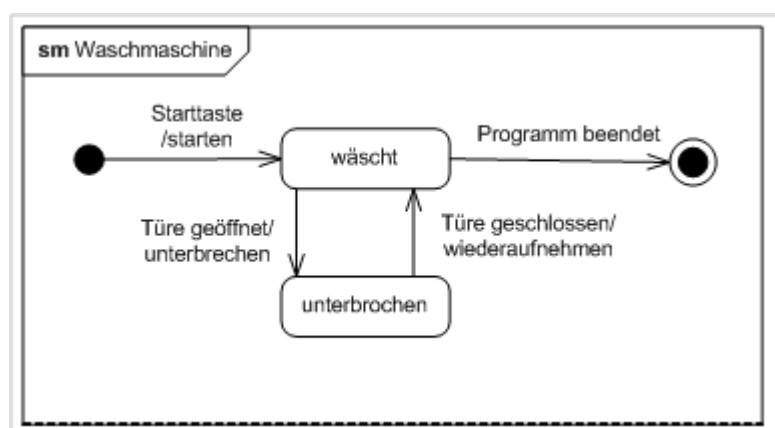
Dem Zustand können drei Verhaltensspezifikationen, zum Beispiel in Form einer Aktivität oder einer Interaktion, zugeordnet werden:

- ein Verhalten, das ausgeführt wird, wenn der Zustandsautomat in den Zustand eintritt
- ein Verhalten, das ausgeführt wird, wenn der Zustandsautomat den Zustand verlässt
- ein Verhalten, das ausgeführt wird, während sich der Zustandsautomat im Zustand befindet

Graphisch wird ein Zustand meistens als Rechteck mit abgerundeten Ecken dargestellt, siehe Beispiele in der Abbildung rechts.



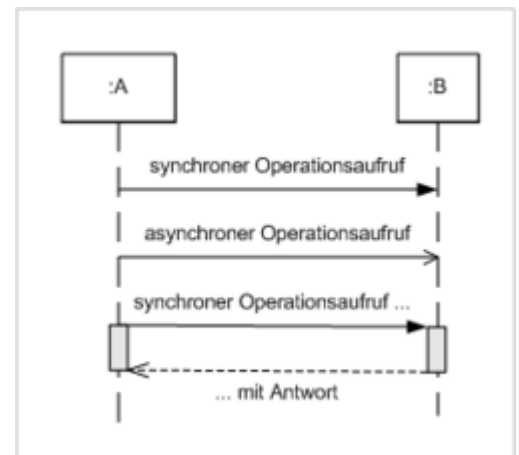
Ein Verhaltenszustandsautomat modelliert das Verhalten eines Modellelements. Der Zustandsautomat in der Abbildung spezifiziert zum Beispiel das Verhalten einer Waschmaschine.



2.2. Sequenzdiagramm

Das Sequenzdiagramm zeigt eine bestimmte Sicht auf die dynamischen Aspekte des modellierten Systems. Ein Sequenzdiagramm ist eine grafische Darstellung einer Interaktion und beschreibt den Austausch von Nachrichten zwischen Ausprägungen mittels Lebenslinien. Ein Sequenzdiagramm stellt in der Regel einen Weg durch einen Entscheidungsbaum innerhalb eines Systemablaufes dar.

Eine Nachricht wird in einem Sequenzdiagramm durch einen Pfeil dargestellt, wobei der Name der Nachricht über den Pfeil geschrieben wird. Synchroner Nachrichten werden mit einer gefüllten Pfeilspitze, asynchrone Nachrichten mit einer offenen Pfeilspitze gezeichnet. Nachrichten, die asynchronen Signalen entsprechen, werden gleich dargestellt wie asynchrone Operationsaufrufe. Die schmalen Rechtecke, die auf den Lebenslinien liegen, sind Aktivierungsbalken, die den „Focus of Control“ anzeigen, also jenen Bereich, in dem ein Objekt über den Kontrollfluss verfügt, und aktiv an Interaktionen beteiligt ist.



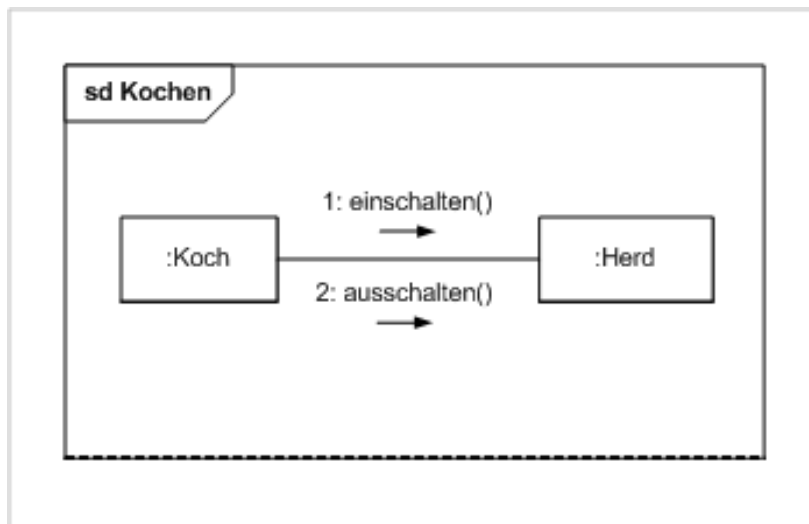
Die zeitliche Anordnung verläuft beim Sequenzdiagramm von Oben nach Unten.

Interaktionen können je nach modelliertem System sehr komplex werden. Wenn es keine Möglichkeit gäbe, Sequenzdiagramme zu modularisieren, wären die entsprechenden graphischen Darstellungen unübersichtlich und schwer verständlich.

Schlüsselwort	Bezeichnung	Beschreibung
alt	Alternatives Fragment	Durch einen alt-Operator können alternative Abläufe, die durch Bedingungen versehen sind, zusammengefasst werden.
assert	Zusicherung	Für eine Nachrichtenmenge kann mit Hilfe dieses Operators eine zwingend notwendige Ablaufreihenfolge angegeben werden.
break	Abbruchfragment	Der normale Ablauf wird unterbrochen, falls eine vorherige Bedingung erfüllt, bzw. verletzt wurde.
consider	Relevante Nachrichten	Mit Hilfe dieses Operators werden nur die angegebenen Aktionen ausgeführt, der Rest wird ignoriert.
critical	Kritischer Bereich	Falls diese Region betreten wird, so werden alle Aktionen ohne jegliche Unterbrechung ausgeführt.
ignore	Irrelevante Nachrichten	Bestimmte Aktionen können mit Hilfe dieses Operators an der Ausführung gehindert werden.
loop	Schleife	Mit Hilfe des loop-Operators können Schleifen definiert werden. Zur Vereinfachung findet man manchmal auch loop while oder loop until.
neg	Negation	Dieser Operator kapselt unzulässige Abläufe.
opt	Optionales Fragment	Die einfachste Form der Operatoren ist der opt-Operator, der optionale Teilabläufe umfasst.
par	Paralleles Fragment	Der par-Operator dient der Darstellung von parallelen Abläufen
seq	Lose Ordnung	Legt eine Reihenfolge für die Abfolge von Aktionen einer Lebenslinie vor.
strict	Strenge Ordnung	Ähnelt dem Aufbau des seq-Operators. Hier jedoch betrifft die Reihenfolge nicht nur eine Lebenslinie, sondern gleich alle Lebenslinien.

2.3. Kommunikationsdiagramm

Das Kommunikationsdiagramm zeigt eine bestimmte Sicht auf die dynamischen Aspekte des modellierten Systems und stellt Interaktionen grafisch dar, wobei der Austausch von Nachrichten zwischen Ausprägungen mittels Lebenslinien dargestellt wird.

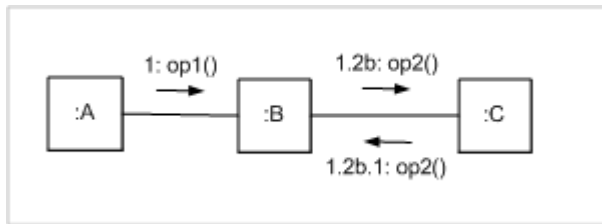


Ähnlich wie in einem Sequenzdiagramm werden in einem Kommunikationsdiagramm Lebenslinien als Rechtecke dargestellt. Dieses Symbol „Lebenslinie“ zu nennen mag etwas seltsam erscheinen, denn im Kommunikationsdiagramm wird im Unterschied zum Sequenzdiagramm die gestrichelte Linie, die für die Zeitachse beim Austausch von Nachrichten steht, nicht angezeigt. Eine Nachricht wird als kurzer Pfeil gezeichnet. Die Richtung des Pfeils zeigt vom Sender zum Empfänger der Nachricht. Der Pfeil ist beschriftet mit einer Sequenznummer und einer Signatur der Nachricht, zum Beispiel der Signatur der Operation, falls es sich bei der Nachricht um einen synchronen Aufruf einer Operation handelt.

Die zeitliche Abfolge von Nachrichten wird in Kommunikationsdiagrammen mit Sequenzausdrücken modelliert. Jeder modellierten Nachricht ist ein Sequenzausdruck zugeordnet, aus dem hervorgeht, was die Vorgängernachrichten sind und welche Nachrichten allenfalls parallel ablaufen.

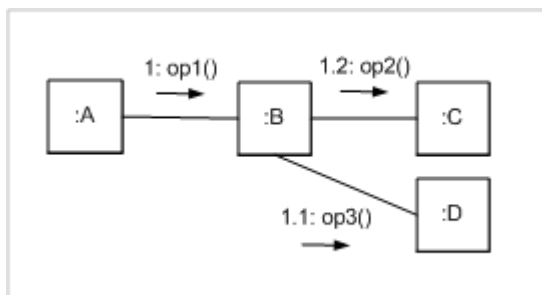
Ein Sequenzausdruck besteht aus einer Folge von mit einem Punkt (.) getrennten Sequenztermen, gefolgt von einem Doppelpunkt (:). Sequenzterme sind Zahlen, zum Beispiel 1, 10 oder 22, die optional von einem Kleinbuchstaben, zum Beispiel a oder b gefolgt werden. Folgende Zeichenfolgen sind also gültige Sequenzausdrücke: 1, 1.7, 1a.6, 1a.6c.2.

Die Bedeutung von Sequenzausdrücken ist durch drei Regeln gegeben:

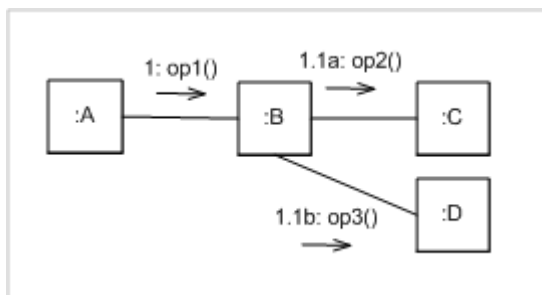


Erstens gibt die Folge der Terme im Sequenzausdruck die Verschachtelungstiefe in einer Meldung an. Eine Nachricht mit dem Sequenzausdruck 1.2b.6 wird also von einer Ausprägung gesendet, nachdem

diese eine Nachricht mit dem Sequenzausdruck 1.2b erhalten hat. In der Abbildung links folgt die Nachricht mit dem Sequenzausdruck 1.2b.1 auf die Nachricht mit dem Sequenzausdruck 1.2b, diese wiederum auf die Nachricht mit dem Sequenzausdruck 1.



Zweitens geben rein numerische Terme auf der gleichen Verschachtelungstiefe eine sequentielle Reihenfolge an. Die Abbildung links stellt also eine Interaktion dar, in der die Lebenslinie :B die Nachricht mit dem Sequenzausdruck 1.1 vor der Nachricht mit dem Sequenzausdruck 1.2 versendet.



Drittens bezeichnen Terme mit einem Buchstaben am Schluss Nachrichten, die parallel versendet werden. Im Diagramm links erkennt man also, dass die Nachricht mit dem Sequenzausdruck 1.1a und die Nachricht mit dem Sequenzausdruck 1.1b gleichzeitig versendet werden.