## Zustände von Programmen

Ab einer gewissen Komplexität können Programme oder deren Komponenten nicht mehr ohne vorgängige Planung erstellt werden. Für die Planung dienen grafische Hilfsmittel: Diagramme, die im sogenannten UML-Standard festgelegt sind, bieten für alle Aufgaben der Softwareentwicklung die richtigen Darstellungsarten an.

UML: Unified Modelling Language: Darstellungsvorschriften für Diagramme in der Softwareentwickling

UML Diagramme werden eingeteilt in sogenannte „Strukturdiagramme“, die also die statische Struktur eines Programmes mit allen enthaltenen Elementen und den Beziehungen untereinander abbilden, und Verhaltensdiagramme, die das Augenmerk auf das Verhalten eines Programms legen. Von den Verhaltensdiagrammen stehen verschiedene zur Verfügung:

* Aktivitätsdiagramm: beschreibt Abläufe mit Hilfe von Aktionen, Transitionen, Verzweigungen…
* Zustandsdiagramm: beschreibt Objektzustände und mögliche Zustandsübergänge

Das Zustandsdiagramm eignet sich sehr gut, um ereignisorientierte Programme darzustellen.

## Zustandsdiagramm (State-Event-Diagramm oder State Machine)

Ein Zustand eines Objektes ist bestimmt durch die gerade aktuellen Werte aller Eigenschaften dieses Objektes. Ein Zustandsdiagramm zeigt eine Folge von Zuständen (States), die ein Objekt einnehmen kann, und aufgrund welcher Ereignisse (Events) Zustandsänderungen stattfinden können.

Versteht man z.B. unter einem Objekt ein Formular eines Programmes, das angezeigt wird, ist sein Zustand sein momentanes Aussehen (welche Elemente sichtbar sind, welche Farben oder Texte angezeigt werden, usw). Eine Zustandsänderung wäre jetzt z. B. das Öffnen eines neuen Formulares, und das auslösende Ereignis könnte ein Klick auf einen Button durch den Benutzer gewesen sein.

Das Zustandsdiagramm eignet sich also sehr gut, um die Abläufe eines Windows-Form-Programmes darzustellen.

Regeln für Zustände oder Zustandsübergänge lauten:

* Ein System ist zu jedem Zeitpunkt immer in genau einem Zustand
* Ein Zustandsdiagramm enthält ein Minimum von zwei Zuständen und zwei Zustandsübergängen.
* Jeder Zustand hat einen Namen
* Ein Zustandsübergang ist mit einer Kondition und optional mit einer Aktion bezeichnet.

## Darstellungsregeln für ein Zustandsdiagramm

Ein Zustandsdiagramm setzt sich aus folgenden Elementen zusammen:

|  |  |
| --- | --- |
|  | Der Zustand wird als Rechteck mit abgerundeten Ecken dargestellt. |
|  | Der Zustandsübergang (Transition) wird als Pfeil mit einer durchgezogenen Linie und einer offenen Pfeilspitze dargestellt. |
|  | Der Startzustand ist ein ausgefüllter Kreis, von dem ein Zustandsübergang ausgeht, an dem aber kein Zustandsübergang endet. |
|  | Der Endzustand ist ein ausgefüllter Kreis, der noch mit einer weiteren Kreislinie umgeben ist. Von ihm geht kein Zustandsübergang aus, aber einer oder mehrere Zustandsübergänge enden an ihm. |
|  | Die Verzweigung wird eingesetzt, wenn ein Zustandsübergang von einem Quellzustand mehrere Zielzustände hat. |

Weitere Notationsregeln:

|  |  |
| --- | --- |
|  | Zustandsübergänge werden gewöhnlich durch Ereignisse ausgelöst, die auf den Pfeilen zwischen den Zuständen notiert werden. Auf dem Pfeil kann das Ereignis und die darauf folgende Aktion notiert werden. Getrennt werden Ereignis und Aktion durch einen horizontalen Strich oder bei einigen Software-Tools auch durch einen Querstrich. |
|  | Ein Pfeil kann auch auf den gleichen Zustand zurückführen. |
|  | Nach einer Verzweigung werden die aufteilenden Bedingungen an dem Pfeil in geschweiften Klammern notiert. |

Wendet man diese Diagrammform nun dazu an, um die Benutzeroberfläche eines Computerprogramms mit den zugehörigen Ereignisprozeduren abzubilden, dann werden die Fenster mit Zuständen beschrieben. Das jeweils aktive Fenster ist der aktuelle Zustand. Ändert sich durch ein Ereignis nichts am Fenster ist es auch möglich, dass der Quell- und der Zielzustand identisch sind, der Zustandsübergangspfeil also wieder auf sich selber zeigt.