

Technik

# Lesen von Zustandsdiagrammen

## Stichworte

Zustand, Übergang, Mutationseingangs, Aktion

## Motivation

Ein gegebenes (einfaches) Zustandsdiagramm muss gelesen und verstanden werden.

## Ein Beispiel zum Lesen

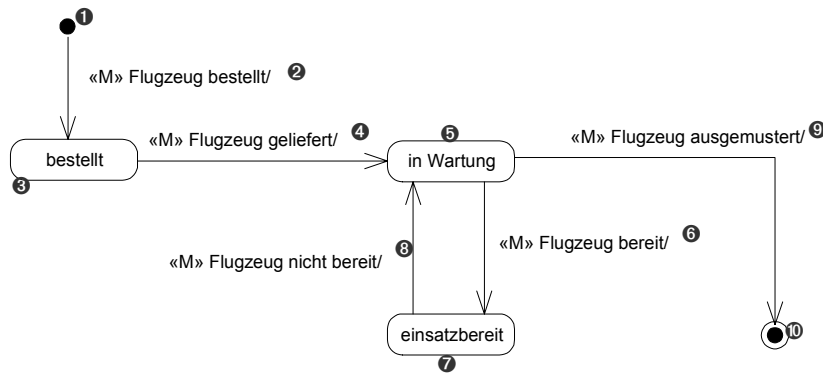


Abbildung 1 Ein Zustandsdiagramm mit Ereignissen

Das Diagramm in Abbildung 1 zeigt alle Zustände, in denen das Objekt *Flugzeug* im Laufe seines Lebens sein kann. Ferner zeigt es die möglichen Übergänge zwischen den Zuständen und die Ereignisse, die diese Übergänge auslösen.

Jedes Objekt der Klasse *Flugzeug* kommt aus dem Nichts<sup>1</sup> (Startzustand) und verschwindet (in der Regel) wieder im Nichts<sup>10</sup> (Endzustand). Dies gilt in der Regel für alle Klassen, d.h. auf den meisten Zustandsdiagrammen finden man einen Startzustand<sup>1</sup> und einen Endzustand<sup>10</sup>.

Im Verlaufe seines Lebens kann ein *Flugzeug* (Achtung: hier ist die Rede vom *Objekt* Flugzeug und nicht vom realen Flugzeug) in unserem IT-System 3 Zustände einnehmen: *bestellt*<sup>3</sup>, *in Wartung*<sup>5</sup> und *einsatzbereit*<sup>7</sup>.

Das Ereignis «M» *Flugzeug bestellt*<sup>2</sup> führt dazu, dass aus dem „Nichts“<sup>1</sup> ein neues Objekt *Flugzeug* im IT-System erzeugt wird (Geburt). Dieses Objekt ist, gleich nachdem es erzeugt worden ist, im Zustand *bestellt*<sup>3</sup>.

Tritt das Ereignis «M» *Flugzeug geliefert*<sup>4</sup> ein und das Flugzeug ist im Zustand *bestellt*<sup>3</sup>, dann wechselt es in den Zustand *in Wartung*<sup>5</sup>. Ist das Flugzeug in einem anderen Zustand als *bestellt*, passiert nichts!

Durch die Ereignisse «M» *Flugzeug bereit*<sup>6</sup> und «M» *Flugzeug nicht bereit*<sup>6</sup> wechselt das Flugzeug beliebig oft zwischen den Zuständen *in Wartung*<sup>5</sup> und *einsatzbereit*<sup>7</sup>.

Am Ende seines Lebens verschwindet das *Flugzeug*-Objekt durch das Ereignis «M» *Flugzeug ausgemustert*<sup>9</sup> im Nichts<sup>10</sup>, d.h. es wird gelöscht (Tod).

Abbildung 2 zeigt weitere Elemente, die in Zustandsdiagrammen vorkommen können.

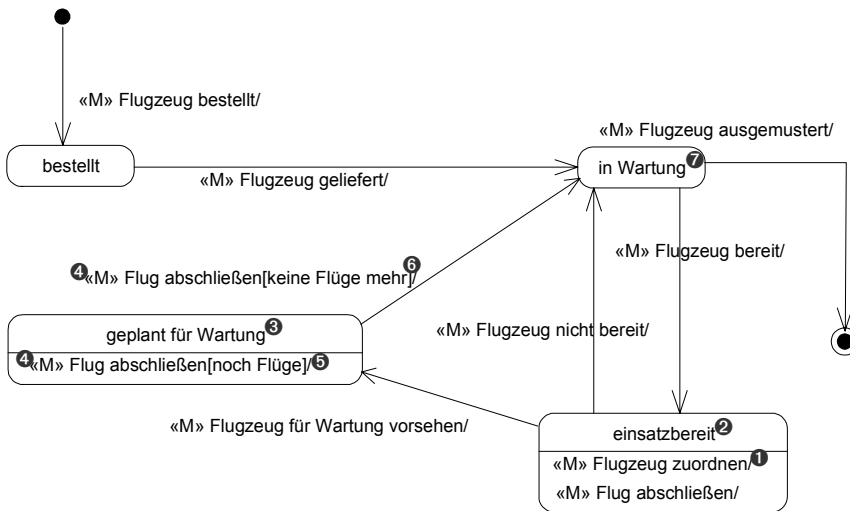


Abbildung 2 Ein Zustandsdiagramm mit internen Übergängen und Wächterbedingungen

Neben den bereits erklärten Übergängen, gibt es noch die internen Übergänge. Durch das Ereignis «M» Flugzeug zuordnen<sup>1</sup>, das auftritt, wenn das Flugzeug einem Flug zugeordnet wird, wird kein Übergang in einen andern Zustand ausgelöst. Vielmehr verbleibt das Flugzeug im Zustand *einsatzbereit*<sup>2</sup>. Es findet ein *interner Übergang* statt, das Flugzeug-Objekt ist vor und nach dem Ereignis in demselben Zustand *einsatzbereit*<sup>2</sup>.

Eine Wächterbedingung erlaubt es, ein Ereignis abhängig von einer Bedingung zu akzeptieren oder abzulehnen. Tritt im Zustand *geplant für Wartung*<sup>3</sup> das Ereignis «M» Flug abschließen<sup>4</sup> auf, hängt die Reaktion des Objektes von der in eckigen Klammern angegebenen Wächterbedingung ab. Ist die Bedingung *[noch Flüge]*<sup>5</sup> wahr (d.h. dem Flugzeug sind noch Flüge zugeteilt) dann findet ein interner Übergang statt. Das Flugzeug verbleibt im Zustand *geplant für Wartung*<sup>3</sup>. Ist hingegen die Bedingung *[keine Flüge mehr]*<sup>6</sup> wahr (d.h. dem Flugzeug sind keine Flüge mehr zugeteilt) dann findet ein Übergang auf den Zustand *in Wartung*<sup>7</sup> statt.

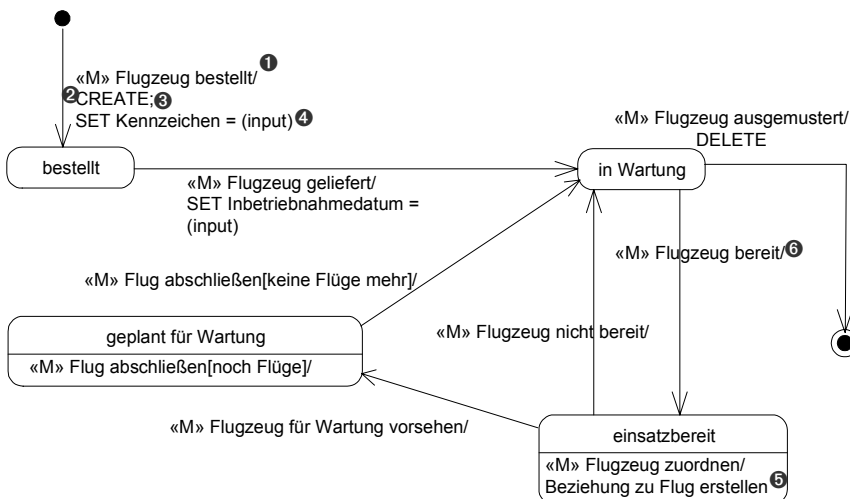


Abbildung 3 Ein Zustandsdiagramm mit Aktionen

Die Aktionen geben an, wie ein Objekt auf ein Mutationsereignis reagiert. Abbildung 3 zeigt verschiedene Arten von Aktionen. Eine Aktion folgt immer hinter dem Schrägstrich<sup>1</sup> nach dem Ereignis. Dem Mutationsereignis «M» Flugzeug bestellt folgen die Aktionen *CREATE*<sup>2</sup> die angibt, dass ein neues Objekt erzeugt wird, und *SET Kennzeichen = (input)*<sup>4</sup> die angibt, dass dem Attribut Kennzeichen ein vom Benutzer im Anwendungsfall eingegebener Wert zugeordnet wird. Einzelne Aktionen werden durch eine ;<sup>3</sup> getrennt. Neben diesen semi-

formalen Aktionen wie CREATE und SET können Aktionen auch als freier Text beschrieben werden. Dem Mutationsereignis «M» *Flugzeug zuordnen* folgt die Aktion *Beziehung zu Flug erstellen*<sup>⑤</sup> die angibt, dass ein Beziehung zu einem Objekt *Flug* erstellt wird. Steht keine Aktion bei einem Ereignis<sup>⑦</sup> dann bedeutet das entweder, dass die Aktion noch nicht spezifiziert ist, oder dass das Objekt lediglich in einen anderen Zustand übergeht. Vertieft man sich in das Fallbeispiel dann wird man feststellen, dass das Zustandsdiagramm in Abbildung 3 noch um weitere Zustände und Ereignisse erweitert werden muss.

Ein Zustandsdiagramm, das alle möglichen Lebenswege eines Objektes dokumentiert, kann nicht einfach sequentiell gelesen werden. Es dient dem Leser jedoch zur Beantwortung einer Reihe von typischen Fragen:

- Was passiert mit dem Objekt, wenn ein bestimmtes Ereignis eintrifft? Da die Antwort auf diese Frage jeweils von aktuellen Zustand des Objektes abhängt, sollte die Frage eigentlich lauten:  
Wie reagiert ein Objekt, das sich in einem bestimmten Zustand befindet, auf ein bestimmtes Ereignis?
- Welche Ereignisse sind für das Objekt relevant?
- Wie, d.h. durch welche Ereignisse, kann ein bestimmter Zustand verlassen werden?
- Wie, d.h. durch welche Ereignisse, kann ein bestimmter Zustand erreicht werden?

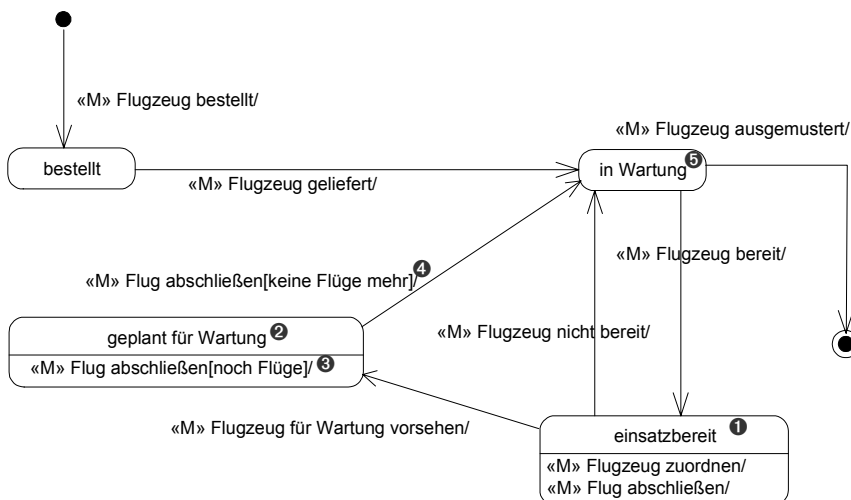


Abbildung 4 Selektives lesen im Zustandsdiagramm

Betrachten wir einige dieser Fragen anhand des Zustandsdiagramms der Klasse *Flugzeug* in Abbildung 4.

- Wie reagiert ein *Flugzeug*-Objekt, das sich im Zustand *einsatzbereit*<sup>①</sup> befindet, auf das Ereignis «M» *Flugzeug zuordnen*?  
Um die Frage zu beantworten muss als erstes geprüft werden, ob im Zustand *einsatzbereit*<sup>①</sup> das Ereignis «M» *Flugzeug zuordnen* überhaupt vorkommt. Das Ereignis ist vorgesehen, wenn ein Übergang (einen Pfeil) in einen anderen Zustand existiert, der mit dem Ereignisnamen bezeichnet ist, oder wenn ein interner Übergang (einen Eintrag im unteren Teil des Zustands-Symbols) vorhanden ist. In unserem Beispiel existiert kein Übergang in einen anderen Zustand jedoch ein interner Übergang, was folgendes bedeutet:  
Ein *Flugzeug*-Objekt im Zustand *einsatzbereit*<sup>①</sup> akzeptiert das Ereignis «M» *Flugzeug zuordnen* und befindet sich danach immer noch im Zustand *einsatzbereit*<sup>①</sup>.
- Wie reagiert ein *Flugzeug*-Objekt, das sich im Zustand *geplant für Wartung*<sup>②</sup> befindet, auf das Ereignis «M» *Flug abschließen*?  
Um die Frage zu beantworten prüft man als erstes, ob im Zustand *geplant für Wartung*<sup>②</sup> das Ereignis «M» *Flug abschließen* überhaupt vorkommt. In unserem Beispiel existiert sowohl ein Übergang in einen anderen Zustand, als auch ein interner Übergang. Da nur ein Übergang möglich ist (das *Flugzeug*-Objekt soll sich nach dem Übergang in genau einem definierten Zustand und nicht in zwei Zuständen befinden), braucht es Kriterien, um

zu entscheiden, welcher Übergang stattfinden soll. Dabei helfen die Wächterbedingungen [*noch Flüge*] <sup>3</sup> und [*keine Flüge mehr*] <sup>4</sup>. Wir müssen also prüfen, ob dem Flugzeug noch Flüge zugeordnet sind. In unserem Fall gehen wir davon aus, dass dem Flugzeug keine weiteren Flüge mehr zugeteilt sind. Dies bedeutet:

Ein *Flugzeug*-Objekt im Zustand *geplant für Wartung* <sup>2</sup> akzeptiert das Ereignis «M» *Flug abschließen* und wechselt, da ihm keine weiteren Flüge zugeteilt sind, in den Zustand *in Wartung* <sup>5</sup>.

- Wie reagiert ein *Flugzeug*-Objekt, das sich im Zustand *geplant für Wartung* <sup>2</sup> befindet, auf das Ereignis «M» *Flugzeug zuordnen*?

Um die Frage zu beantworten prüft man als erstes, ob im Zustand *geplant für Wartung* <sup>2</sup> das Ereignis «M» *Flugzeug zuordnen* überhaupt vorkommt. In unserem Beispiel existiert weder ein Übergang in einen anderen Zustand noch ein interner Übergang. Dies bedeutet:

Ein *Flugzeug*-Objekt im Zustand *geplant für Wartung* <sup>2</sup> akzeptiert das Ereignis «M» *Flugzeug zuordnen* nicht<sup>1</sup>.

- Welche Ereignisse sind für ein *Flugzeug*-Objekt relevant?  
Die Antwort lautet: Alle Ereignisse, die auf dem Zustandsdiagramm der Klasse *Flugzeug* enthalten sind, d.h. alle Ereignisse, die in mindestens einem Zustand akzeptiert werden. Alle anderen Ereignisse spielen für ein *Flugzeug*-Objekt keine Rolle. Das bedeutet:

Für ein *Flugzeug*-Objekt sind nur die Ereignisse «M» *Flugzeug bestellt*, «M» *Flugzeug geliefert*, «M» *Flugzeug bereit*, «M» *Flugzeug nicht bereit*, «M» *Flugzeug zuordnen*, «M» *Flug abschließen*, «M» *Flugzeug für Wartung vorsehen* und «M» *Flugzeug ausgemustert* relevant.

- Durch welche Ereignisse kann ein *Flugzeug*-Objekt den Zustand *in Wartung* <sup>5</sup> verlassen?  
Um die Frage zu beantworten sucht man alle Übergänge (Pfeile), die aus dem Zustand *in Wartung* <sup>5</sup> in einen andern Zustand führen. In unserem Beispiel gibt es zwei solche Übergänge. Dies bedeutet:  
Ein *Flugzeug*-Objekt im Zustand *in Wartung* <sup>5</sup> kann diesen Zustand nur durch das Ereignis «M» *Flugzeug bereit* oder das Ereignis «M» *Flugzeug ausgemustert* verlassen.
- Durch welche Ereignisse kommt ein *Flugzeug*-Objekt in den Zustand *einsatzbereit* <sup>1</sup>?  
Um die Frage zu beantworten sucht man alle Übergänge (Pfeile), die in den Zustand *einsatzbereit* <sup>1</sup> führen. In unserem Beispiel gibt es genau einen solchen Übergang. Dies bedeutet:  
Ein *Flugzeug*-Objekt kann nur durch das Ereignis «M» *Flugzeug bereit* in den Zustand *einsatzbereit* <sup>1</sup> kommen (und zwar aus dem Zustand *in Wartung* <sup>5</sup>).

Die besprochenen Fragen haben bereits gezeigt: In einem Zustandsdiagramm ist das, was nicht da steht, genauso wichtig wie das, was da steht. Ereignisse, die in einem bestimmten Zustand nicht vorkommen, werden *nicht akzeptiert*, wenn das Objekt in diesem Zustand ist. Dies bedeutet, dass das nicht akzeptierte Ereignis im IT-System *nicht* erfolgreich abgearbeitet werden kann. Es muss eine entsprechende Fehlermeldung erzeugt werden. Ereignisse, die in *keinem* Zustand vorkommen, werden immer *ignoriert*. Dadurch lassen sich für unser *Flugzeug*-Objekt folgende Aussagen aus dem Zustandsdiagramm lesen:

- Wird ein Flugzeug geliefert so ist es nie direkt im Zustand *einsatzbereit*, es ist immer zuerst im Zustand *in Wartung*.
- Ein einsatzbereites Flugzeug kann *nicht* ausgemustert werden. Wird dies trotzdem versucht dann scheitert das Mutationsereignis mit einer entsprechenden Fehlermeldung.

## Referenzen

1. KnowSolution "Dynamik von Geschäftsobjekten T3": Notation von Zustandsdiagrammen
2. Patrick Grässle, Henriette Baumann, Philippe Baumann: UML projektorientiert - Geschäftsprozeßmodellierung, IT-System-Spezifikation und Systemintegration mit der UML, Galileo Press, 2000, ISBN 3-934358-58-6

---

<sup>1</sup> Das IT-System sollte den Anwender über den Grund, warum die das Zuordnen des Flugzeugs nicht geklappt hat, informieren.