

Theorie: Modellierung/Diagrammtypen

Was ist Modellierung? Wozu benutzt man Modellierung?

Modellierung ist das Abbilden eines Ausschnitts der Realität in einem Modell. Je nach Problemstellung werden unterschiedliche Teile der Realität in Modellen abgebildet. In der Informatik gibt es dazu verschiedene Modelle, die benutzt werden, um Informationen kompakt und übersichtlich darzustellen. Jedes Modell erfüllt einen bestimmten Zweck: Beschreibung von Zuständen und Eigenschaften von Objekten, Darstellung von Beziehungen zwischen Objekten, Erläutern von Abläufen und Prozessen usw. Die informatischen Modelle können Informationen aus vielfältigen Kontexten darstellen. Die Gegenstände dieser Modelle sind reale oder fiktive Objekte innerhalb oder außerhalb der Informatik. Daher können die Modelle auch in Unterrichtsfächern abseits des Fachs Informatik verwendet werden. Sie können z.B. für folgende Aufgaben in den einzelnen Unterrichtsfächern benutzt werden:

- Beschreibung von Berechnungsalgorithmen im Fach Mathematik oder Abläufen in naturwissenschaftlichen Fächern
- Ordnen und Beschreiben von geometrischen Objekten in den Fächern Mathematik und Geometrisches Zeichnen
- Zusammenfassung der wesentlichen Aspekte eines Textes im Fach Deutsch oder in einer Fremdsprache
- Üben von Vokabeln und Erweitern des Wortschatzes in einer Fremdsprache
- Planung von Abläufen oder Beschreibung von Prozessen in naturwissenschaftlichen Fächern.

Ein entscheidender Aspekt für den Unterricht sind die vielfältigen Visualisierungsmöglichkeiten von Informationen und die Komprimierung der Information auf die relevanten Inhalte.

Dieser Beitrag gibt einen Überblick über verschiedene Diagrammtypen/Modelle, die in der Informatik verwendet werden, und die Informationen, die durch diese Modelle darstellbar sind. Außerdem werden exemplarische Beispiele präsentiert.

Ein weit verbreiteter Standard für die grafische Darstellung von Modellen in der Informatik ist UML (Unified Modeling Language). In UML gibt es verschiedene Modelle mit festgelegter Notation, welche für eine Vielzahl von Anwendungsfällen geeignet sind. Zu den Diagrammarten von UML gehören u.a. die Objekt-, Klassen- und Aktivitätsdiagramme. Neben diesen UML-Diagrammen werden aber auch andere grafische Darstellungen von Modellen benutzt, etwa das Gegenstand-Beziehung-Modell (auch Entity-Relationship-Diagramm genannt).

Modellarten

Je nachdem, welche Informationen der Realität man abbilden möchte, ist ein anderes Diagramm geeignet.

Grundlegend lässt sich zwischen Diagrammen unterscheiden, die einen Ablauf festlegen (wie z.B. das Aktivitätsdiagramm) und solchen, die z.B. statische Eigenschaften und Beziehungen von Gegenständen oder Personen beschreiben. Bei der zweiten Gruppe lässt sich zusätzlich unterscheiden, ob man ein konkretes Objekt (z.B. *mein Pferd*) oder eine Objektklasse (z.B. *das Pferd*) beschreiben möchte.

Je nachdem, welche Informationen man darstellen möchte, kann eines (oder auch mehrere) der unten angeführten Diagramme geeignet sein.

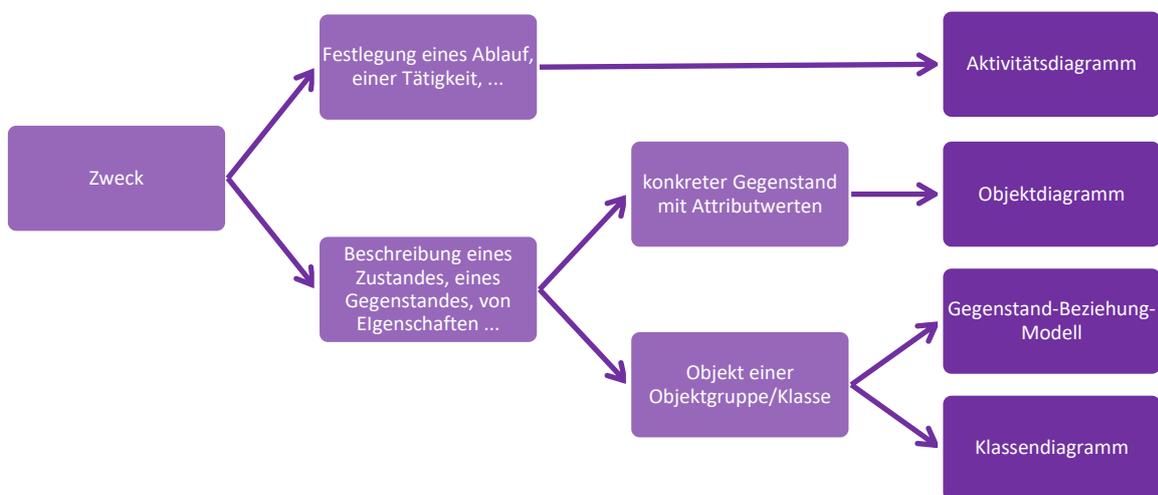


Abbildung 1: Diagrammtypen

- **Objektdiagramm:** Das Objektdiagramm stellt konkrete Objekte (z.B. mein Pferd) mit ihren Eigenschaften (z.B. Alter) und den Ausprägungen der Eigenschaften (z.B. 2 Jahre) dar. Weiters können Beziehungen zwischen Objekten dargestellt werden.
- **Gegenstand-Beziehung-Modell:** Die Beziehungen zwischen Gegenständen (bzw. Personen) sowie deren Eigenschaften werden dargestellt. Dabei geht es nicht um konkrete Gegenstände/Personen (z.B. die Schülerin Susi) oder Ausprägungen von Eigenschaften (z.B. Alter = 10 Jahre). Stattdessen werden die Gemeinsamkeiten von ähnlichen Gegenständen oder Personen dargestellt (z.B. *Jede Schülerin und jeder Schüler hat ein bestimmtes Alter.*).
- **Klassendiagramm:** Ähnlich dem Gegenstand-Beziehung-Modell stellt das Klassendiagramm Objektgruppen und ihre Beziehungen dar. Allerdings bietet das Klassendiagramm komplexere Beziehungstypen an, z.B. Generalisierungen und Teil-Ganzes-Beziehungen. Außerdem lässt sich das Verhalten von Objekten dieser Objektgruppen ebenfalls beschreiben. Im Klassendiagramm werden die Ausprägungen der Attribute einzelner Gegenstände nicht erfasst (anders als im Objektdiagramm).

- **Aktivitätsdiagramm:** In diesem Diagramm können Abläufe, Handlungsabfolgen bzw. Handlungsanweisungen beschrieben werden. Die Darstellung ist übersichtlicher als eine textuelle Beschreibung, vor allem, wenn die Handlung kein sequentieller Ablauf ist, sondern Verzweigungen und Wiederholungen zulässt.

Für dieses Arbeitspaket wird nur das Aktivitätsdiagramm behandelt, wodurch im Folgenden Abschnitt nur jener Diagrammtyp näher beleuchtet wird.

Aktivitätsdiagramm

Das Aktivitätsdiagramm beschreibt Abläufe. Diese Abläufe werden in einzelne Aktivitäten zerlegt (jede Aktivität erhält ein eigenes Feld), deren Reihenfolge durch Pfeile angegeben werden. Eine Aktivität ist z.B.: *Poppy verschwindet*.

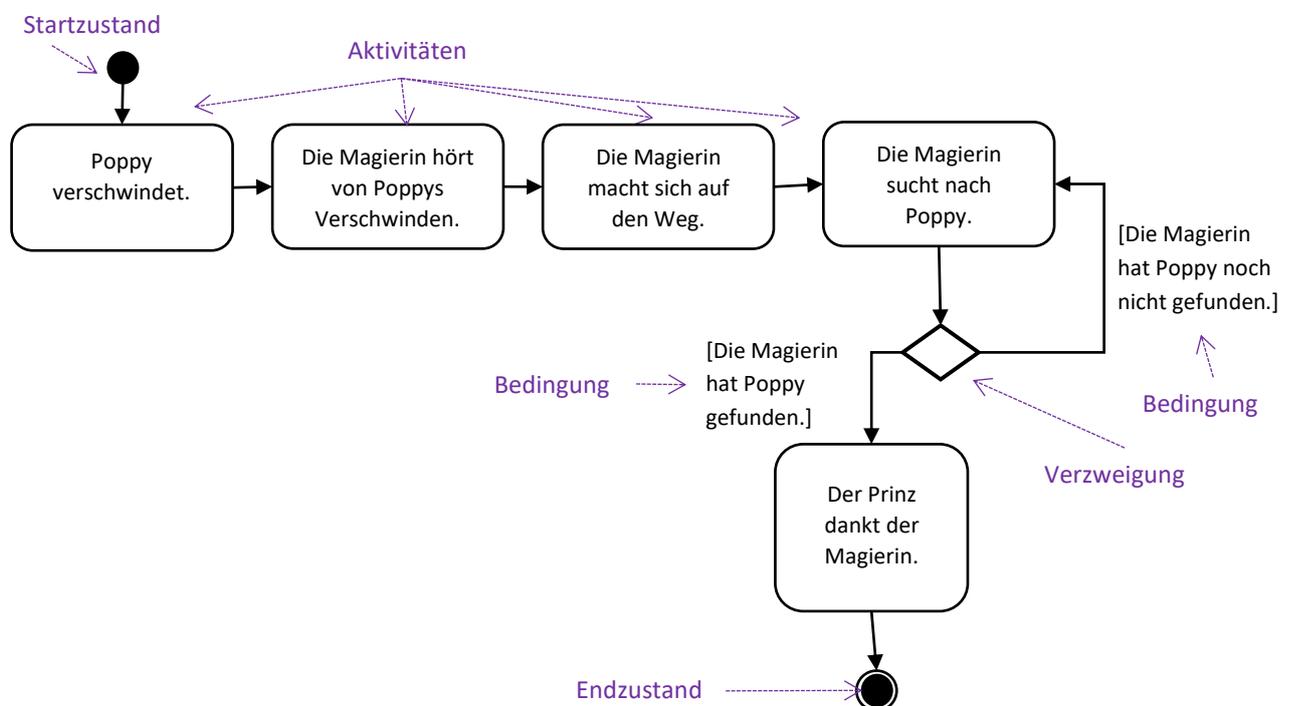


Abbildung 2: Beispiel eines Aktivitätsdiagramms

Jedes Aktivitätsdiagramm hat einen Startzustand und einen Endzustand, die Anfang und Ende markieren. Die Aktivitäten werden zwischen diesen Zuständen angeordnet.

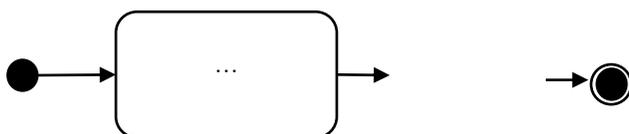


Abbildung 3: Start- und Endzustände im Aktivitätsdiagramm

Sequenz

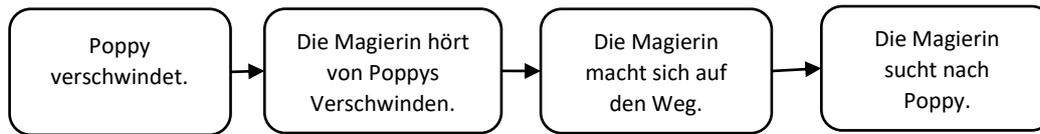


Abbildung 4: Sequenz in einem Aktivitätsdiagramm

Die Aktivitäten folgen ohne Verzweigung aufeinander. Auf jede Aktivität kann nur eine weitere Aktivität folgen. Diese werden durch Pfeile verbunden, die die Leserichtung angeben.

Verzweigung

Wenn auf eine Aktivität mehr als eine Aktivität folgen kann, verzweigt sich der Ablauf. Man muss eine Entscheidung treffen, welchem Weg man folgt. Die Entscheidung wird dadurch getroffen, dass man prüft, welche der Bedingungen erfüllt ist, z.B. geht die Geschichte anders weiter, je nachdem, ob die Magierin Poppy bereits gefunden hat oder nicht. Um eine Verzweigung zu zeichnen, endet eine Verbindung in einer Raute (\diamond), von der mehrere Verbindungen ausgehen können, von denen jede mit einer Bedingung versehen werden muss.

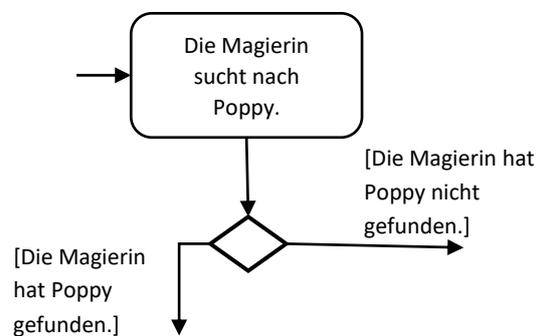


Abbildung 5: Verzweigung in einem Aktivitätsdiagramm

Schleife

Es kann passieren, dass eine der Verbindungen in einer bereits geschehenen Aktivität endet und diese Aktivität wiederholt werden muss. Wenn Poppy noch nicht gefunden wurde, muss die Magierin z.B. weitersuchen. Die Aktivitäten in der Schleife müssen solange wiederholt werden, bis eine andere Bedingung erfüllt ist, die aus der Schleife führt. Die Magierin sucht etwa solange nach Poppy, bis die Bedingung *Die Magierin hat Poppy gefunden* eintritt.

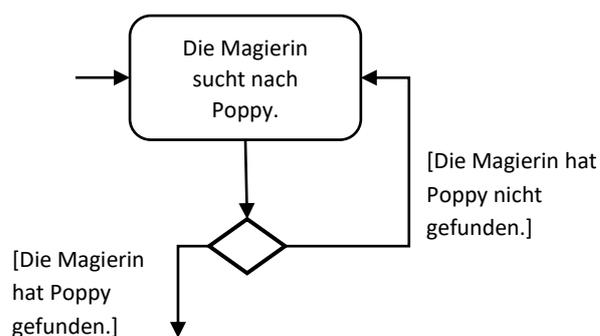


Abbildung 6: Schleife in einem Aktivitätsdiagramm