

# *Modellierung und ausgewählte Modellierungssprachen*

*Ing. Peter Völkl, MA MSc mit Beiträgen von*

*Thomas Grabetz, BSc MA*

## Kurzbiographie der Autoren



Peter Völkl

Peter Völkl, Jahrgang 1984, begann seine berufliche Laufbahn, nach dem Abschluss seiner Ausbildung an der HTBLuVA für EDV und Organisation, 2005 als Softwareentwickler beim Österreichischen Roten Kreuz. Dort übernahm er 2008 die Leitung einer, für die IT Services des Blutbereiches verantwortlichen, Abteilung der Blutspendezentrale für Wien, Niederösterreich und das Burgenland. Seine Schwerpunkte lagen hierbei, neben der Führung der Abteilung, in der laufenden Koordination und Überwachung von IT Projekten und Prozessen, sowie der Entwicklung, Testung und Validierung von IT Systemen. Neben seiner beruflichen Tätigkeit begann er 2010 mit dem

Bachelorstudium der Wirtschaftsinformatik an der FernFH, welches er 2012 abschloss. Anschließend absolvierte er 2014 die Masterstudien Wirtschaftsinformatik an der FernFH und Information Management and IT Security an der FH Technikum Wien. Ab November 2014 war Peter Völkl als wissenschaftlicher Mitarbeiter an der FernFH tätig.



Thomas Grabetz

Thomas Grabetz, Jahrgang 1967, hat seinen akademischen Werdegang über den zweiten Bildungsweg erfolgreich absolviert. Nach dem Abschluss am Abendkolleg der HTL Spengergasse erwarb er einen Bachelor in Computer Studies von der University of Derby sowie 2014 einen Master-Abschluss an der Ferdinand Porsche FERNFH.

Seit 1998 ist Thomas Grabetz in der IT-Branche tätig und konnte in verschiedenen Rollen umfangreiche Erfahrung sammeln, zunächst als Software Engineer und später als Key Account Manager für renommierte internationale Unternehmen. Seit 2008 hat er sich auf die Bereiche Geschäftsprozessmanagement, Industrie 4.0 und Digitale Transformation spezialisiert.

Im September 2023 übernahm Thomas Grabetz die Position des Wissenschaftlichen Mitarbeiters an der Ferdinand Porsche FERNFH.

Dieses urheberrechtlich geschützte Werk wird Ihnen unter einer Attribution-NonCommercial 4.0 International-Lizenz zur Verfügung gestellt (CC BY-NC 4.0). Eine Kopie dieses Lizenzvertrages finden Sie unter: [creativecommons.org/licenses/by-nc/4.0/deed.de](https://creativecommons.org/licenses/by-nc/4.0/deed.de)

Die Rechte an zitierten Abbildungen liegen beiden in der jeweiligen Quellenangabe genannten Urheber\*innen. Auflage 202608b

# Inhaltsverzeichnis

## Inhalt

|           |                                                                   |    |
|-----------|-------------------------------------------------------------------|----|
| Lektion 1 | Einleitung .....                                                  | 6  |
| 1.1       | Der Modellbegriff .....                                           | 6  |
| 1.2       | Modellbildung und Anforderungen an ein Modell.....                | 8  |
| 1.3       | Die Grundsätze ordnungsmäßiger Modellierung (GoM) .....           | 10 |
| 1.3.1     | Grundsatz der Richtigkeit.....                                    | 10 |
| 1.3.2     | Grundsatz der Relevanz.....                                       | 11 |
| 1.3.3     | Grundsatz der Wirtschaftlichkeit.....                             | 11 |
| 1.3.4     | Grundsatz der Klarheit.....                                       | 11 |
| 1.3.5     | Grundsatz der Vergleichbarkeit .....                              | 11 |
| 1.3.6     | Grundsatz des systematischen Aufbaus.....                         | 12 |
| 1.3.7     | Erweiterung der GoM.....                                          | 12 |
| 1.4       | Einführung in Prozess-, Ablauf- und Geschäftsprozessmodelle ..... | 13 |
| 1.4.1     | Prozess- und Ablaufmodelle .....                                  | 13 |
| 1.4.2     | Geschäftsprozessmodelle .....                                     | 14 |
| 1.5       | Hinweis zur grafischen Darstellung und Werkzeugunterstützung..... | 14 |
| Lektion 2 | Flussdiagramme .....                                              | 16 |
| 2.1       | Aufbau eines Flussdiagramms.....                                  | 16 |
| 2.2       | Symbole in Flussdiagrammen.....                                   | 17 |
| 2.3       | Vorgehen bei der Erstellung .....                                 | 18 |
| 2.4       | Modellierungsregeln.....                                          | 19 |
| 2.5       | Beispiel: Online-Prüfungsanmeldung.....                           | 20 |
| 2.6       | Typische Fehler .....                                             | 21 |
| 2.7       | Arten von Flussdiagrammen .....                                   | 21 |
| Lektion 3 | Geschäftsprozessmodellierung .....                                | 23 |
| 3.1       | Ereignisgesteuerte Prozessketten (EPK).....                       | 23 |
| 3.1.1     | Aufbau einer EPK.....                                             | 25 |
| 3.2       | Business Process Model and Notation (BPMN).....                   | 30 |
| 3.2.1     | Aufbau der BPMN .....                                             | 30 |
| 3.2.2     | Beispiel eines BPMN.....                                          | 39 |

|           |                                                                                              |    |
|-----------|----------------------------------------------------------------------------------------------|----|
| 3.3       | DMN – Decision Model and Notation .....                                                      | 40 |
| 3.3.1     | Schritt-für-Schritt-Anleitung: .....                                                         | 41 |
| 3.3.2     | Erweiterung des Beispiels: Zerlegung in mehrere Entscheidungen .....                         | 48 |
| 3.3.3     | Interpretation des DMN-Modells.....                                                          | 49 |
| 3.3.4     | Typische Fehler bei DMN-Modellen.....                                                        | 50 |
| 3.3.5     | Abgrenzung zu BPMN .....                                                                     | 51 |
| 3.3.6     | Übung: .....                                                                                 | 52 |
| Lektion 4 | ArchiMate als Modellierungssprache für Unternehmensarchitekturen .....                       | 53 |
| 4.1       | Archimate Elemente: .....                                                                    | 55 |
| 4.1.1     | Kernelemente .....                                                                           | 55 |
| 4.1.2     | Motivation-, Strategie-, Implementierungs- und Migrationselemente .....                      | 56 |
| 4.1.3     | Beziehungen und Verknüpfungspunkte .....                                                     | 58 |
| 4.2       | Erstellen eines ArchiMate Modells – Step by Step.....                                        | 60 |
| 4.2.1     | Architektur & Scope .....                                                                    | 60 |
| 4.2.2     | Modellierung vorbereiten .....                                                               | 60 |
| 4.2.3     | Modellierung .....                                                                           | 65 |
| 4.3       | Ebenen verbinden und analysieren.....                                                        | 72 |
| 4.3.1     | Mögliche Verbindungen.....                                                                   | 72 |
| 4.3.2     | Mögliche Analysefälle aus der Praxis .....                                                   | 76 |
| Lektion 5 | Zusammenfassung: Stärken und Einsatzbereiche der vorgestellten<br>Modellierungssprachen..... | 77 |
| 5.1       | Flussdiagramme.....                                                                          | 77 |
| 5.2       | Ereignisgesteuerte Prozesskette (EPK) .....                                                  | 77 |
| 5.3       | Business Process Model and Notation (BPMN) .....                                             | 78 |
| 5.4       | Decision Model and Notation (DMN) .....                                                      | 78 |
| 5.5       | ArchiMate.....                                                                               | 78 |
| 5.6       | Vorteile professioneller Modellierungswerkzeuge.....                                         | 79 |
| 5.7       | Übersicht der vorgestellten Modellierungssprachen.....                                       | 80 |

### **Hinweis zur Verwendung von Quellen und KI-gestützten Werkzeugen**

Für die Erstellung dieses Skripts wurden zahlreiche Fachquellen herangezogen, inhaltlich zusammengefasst und entsprechend den geltenden wissenschaftlichen Standards zitiert. Zur Verbesserung der sprachlichen Verständlichkeit und Lesbarkeit wurden einzelne Textpassagen mithilfe von ChatGPT (Version 5.6) überarbeitet. Die verwendeten Quellen, die übernommenen fachlichen Inhalte sowie die KI-gestützten Formulierungsvorschläge wurden vom Autor sorgfältig geprüft und bei Bedarf überarbeitet. Die Verantwortung für sämtliche Inhalte und Formulierungen des Skripts liegt beim Autor.

# Lektion 1    Einleitung

Modellierung und Modellierungssprachen sind ein essentieller Bestandteil der meisten Themenfelder der Wirtschaftsinformatik. Die Modellierung bildet dabei einen existierenden, erwarteten oder geplanten Sachverhalt in Form eines Modells ab. Dabei handelt es sich um eine Darstellung, welche der betroffenen Zielgruppe und Zielsetzung angepasst wurde. In den nachfolgenden Lektionen werden der Modellbegriff und die ordnungsgemäße Modellierung, sowie die Grundlagen einzelner, in der Wirtschaftsinformatik besonders relevanter, Modellierungsformen erläutert.

## 1.1 Der Modellbegriff

Bei einem Modell handelt es sich um eine vereinfachte Darstellung der Realität oder ihres Soll-Zustandes. Durch diese Abstraktion ist eine genauere Betrachtung oder Planung bestimmter Einzelaspekte möglich. Es betrachtet nur so viele Details wie nötig sind, um das zugrunde liegende Problem vollständig zu beschreiben. Der Begriff „Modell“ leitet sich aus dem lateinischen Wort „modulus“ ab und bezeichnete einen Maßstab im Bereich der Architektur. Laut dem deutschen Duden Fremdwörterbuch (2010), bedeutet Modell: Objekt, Gebilde, das die inneren Beziehungen und Funktionen von etwas abbildet bzw. [schematisch] veranschaulicht [und vereinfacht, idealisiert].

Für Wirtschaftsinformatiker und Wirtschaftsinformatikerinnen gehören Modelle und Modellierungstechniken zum unumgänglichen Handwerkszeug, insbesondere die Modelle, die in Informationssystemen abgebildet werden und jene, die wir in der Softwareentwicklung verwenden.

Modelle gibt es nicht nur in der Wirtschaftsinformatik, sondern auch in allen anderen Themenfeldern der Wissenschaft, Forschung und Entwicklung. So werden beispielsweise in der Wirtschaftsinformatik Datenmodelle verwendet und anatomische Modelle in der Medizin.

Die gängigen Modelle in der Wirtschaftsinformatik sind:

- **Datenmodelle,**
- **Prozess- und Ablaufmodelle,**
- **Geschäftsprozessmodelle** und
- **objektorientierte Modelle.**

Während Sie in anderen Lehrveranstaltungen über Daten- und objektorientierten Modellen unterrichtet werden, fokussieren wir uns in dieser Lehrveranstaltung auf Prozess- und Ablaufmodelle sowie Geschäftsprozessmodelle.

Auch die Erstellung eines Test- oder Entwicklungssystems in der Software- und Systementwicklung ist ein (Soll-)Modell des Echtsystems. Meist hat es jedoch einen sehr geringen Abstrahierungsgrad, da es dem Echtssystem so nahe wie möglich kommen soll.

Die richtige Wahl eines Modells ist essenziell für den Erfolg der gewünschten Betrachtung, da es durch eine zu starke oder falsche Abstraktion der Darstellung auch zu fehlerhaften Annahmen und Resultaten führen kann.

Jedes Modell lässt sich selbst wiederum in einem Metamodell beschreiben, welches die Modellierung selbst definiert. Ein Metamodell definiert somit die einzelnen Bausteine eines Modells und wie sie anzuwenden sind. Die drei Hauptmerkmale des allgemeinen Modellbegriffs nach Herbert Stachowiak, dem Begründer der allgemeinen Modelltheorie, sind (Stachowiak, 1973, S. 131ff):

- **Abbildungsmerkmal:** Modelle sind Abbildungen oder Repräsentationen natürlicher oder künstlicher Objekte, die selbst wiederum Modelle sein können. Es handelt sich dabei auch um die Zuordnung von Original-Attributen zu Modell-Attributen.
- **Verkürzungsmerkmal:** Modelle bestehen, in der Regel, nicht aus allen, sondern nur aus den relevanten, Attributen des betrachteten Objektes. Um die korrekten Attribute für ein Modell auszuwählen ist sowohl das Wissen über alle möglichen Attribute des Originalobjektes als auch über die Attribute des Modells notwendig.
- **Pragmatisches Merkmal:** Modelle sind nicht alleine einem Zweck, sondern auch einem bestimmten Anwenderkreis und Zeitraum zugeordnet. *Eine pragmatisch vollständige Bestimmung des Modellbegriffs hat nicht nur die Frage zu berücksichtigen, wovon etwas Modell ist, sondern auch, für wen, wann und wozu bezüglich seiner je spezifischen Funktionen es Modell ist.*

Ein Beispiel für ein einfaches Modell und seine Attribute ist das eines Smartphones:

Je nach Abstrahierungsgrad ändern sich die mögliche Darstellung und die relevanten Attribute. Eine mögliche Aufstellung der Attribute und Abbildungen unterschiedlicher Abstrahierungsgrade ist in Tabelle 1 dargestellt.

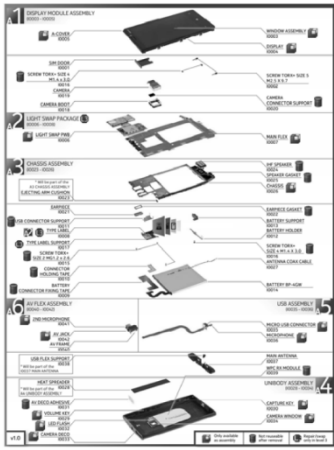
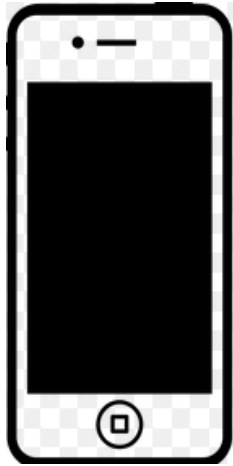
| Anwendungsfall: | Technischer Aufbau eines konkreten Smartphones                                                                                                                                                                                                      | Abbildung eines konkreten Smartphones                                                                         | Piktogramm für Smartphone                                                            |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| Attribute:      | <ul style="list-style-type: none"> <li>• Gerätetyp</li> <li>• Betriebssystem</li> <li>• Hersteller</li> <li>• Gehäuse</li> <li>• Schrauben</li> <li>• Kabel</li> <li>• Platinen</li> <li>• Speicher</li> <li>• Anschlüsse</li> <li>• ...</li> </ul> | <ul style="list-style-type: none"> <li>• Gerätetyp</li> <li>• Betriebssystem</li> <li>• Hersteller</li> </ul> | <ul style="list-style-type: none"> <li>• Gerätetyp</li> </ul>                        |
| Abbildung       |                                                                                                                                                                   |                            |  |

Tabelle 1: Einfaches Modell eines Smartphones

## 1.2 Modellbildung und Anforderungen an ein Modell

Bei der Modellbildung geht es um die ausreichende Abbildung eines realen Systems zur Beschreibung einer bestimmten Fragenstellung oder eines Zustandes. Dazu können entweder bestehende Modelle verwendet werden, oder ein eigenes, geeignetes, Metamodell geschaffen werden. Meist handelt es sich bei den gängigen Modellierungsaufgaben in der Wirtschaftsinformatik jedoch um Situationen, in denen bestehende Modellarten zur Anwendung kommen können und sollten. Die Verwendung eines weit verbreiteten, und im eigenen Fachbereich allgemein bekannten, Modells führt zu einem schnelleren Verständnis für alle beteiligten Personen.

Die Modellbildung lässt sich generell in vier Schritten beschreiben und durchführen (Oeser, 2012, S. 7):

<sup>1</sup> Produktabbildung von <http://www.nokia.com/at-de/smartphones-handys/mobiltelefone/lumia930/> 7

- Schritt 1: Vereinfachung und Formulierung des Problems
- Schritt 2: Formalisierung und Quantifizierung des Problems
- Schritt 3: Verarbeitung des Datenmaterials und Erstellung des Modells
- Schritt 4: Interpretation und Rückschluss auf den ursprünglichen Sachverhalt

In Schritt 1 und 2 wird deutlich, dass die Anforderungen an ein Modell durch die Formulierung des zugrunde liegenden Problems definiert werden. Dieser Planungsschritt ist sehr wichtig, da die Modellauswahl für den weiteren Erfolg der Modellierung, und der damit verbundenen Behandlung der Problemstellung, essenziell ist.

Im 4. Schritt wird die Erfüllung der Zielsetzung des Modells nochmals überprüft und damit sichergestellt, dass es den Sachverhalt problemadäquat widerspiegelt.

Der Modellierer hat, durch seine Tätigkeit, eine hohe Verantwortung gegenüber der Zielsetzung des Modells. Durch eine zu hohe Abstrahierung kann es zu Fehlern in allen, anhand des Modells abgeleiteten, Folgerungen kommen. Diese Fehler wirken sich dann auf alle Bereiche aus, welche das Modell einsetzen. Wesentliche Fehler können in der Praxis auch sehr schlimme Folgen haben, wie das nachfolgende Beispiel beschreibt<sup>2</sup>.

Auszug aus der Frankfurter Allgemeinen Zeitung vom 22.09.2006:

### **23 Menschen sterben in den Trümmern des Transrapid**

*Bei dem schweren Transrapid-Unglück im emsländischen Lathen sind 23 Menschen getötet worden. Das teilte die Polizei am Freitag abend mit. Zehn Menschen sollen die Katastrophe verletzt überlebt haben. Der Transrapid war am Vormittag auf einen Reinigungswagen geprallt.*

Der Transrapid nutzt eine neue Antriebsform und ein spezielles Leitsystem mit einigen Sicherheitsmaßnahmen. Der Zug wird durch Magnete in der Schiene bewegt. Weiter umgreift das Fahrwerk des Zuges die Schiene und haltet ihn in Position, was wiederum ein Entgleisen verhindert. Zusätzlich ist jeder Zug mit einem Ortungssystem ausgestattet, welches seine Daten kontinuierlich mit dem Leitsystem der Streckenanlage abgleicht. Ein Zusammenstoß zweier Züge ist daher, zumindest dem Modell nach, durch die zentrale Steuerung des Streckenantriebs und die Positionsbestimmung nicht möglich. Das System galt daher als enorm sicher.

Leider gab es in diesem Modell eine Komponente, welche nicht beachtet wurde. Die Streckenfahrzeuge und Reinigungswagen sind mit herkömmlichen Dieselantrieben ausgestattet und verfügten über keine Ortungssysteme. Daher war es auch möglich, dass ein Reinigungsfahrzeug übersehen wurde und in dessen Folge 23 Menschen starben.

<sup>2</sup><http://www.faz.net/aktuell/gesellschaft/magnetbahn-unglueck-23-menschen-sterben-in-den-truemmern-des-transrapid-1357701.html>, Aufruf der Seite am 03.11.2014

## 1.3 Die Grundsätze ordnungsmäßiger Modellierung (GoM)

Bei den Grundsätzen ordnungsgemäßer Modellierung (GoM) handelt es sich um semantische Empfehlungen zur bedarfsgerechten Informationsmodellierung. Damit wurde eine bewusste Einschränkung der Modellierungsfreiheit, zugunsten der Zielgerichtetheit der Modellierung selbst, eingeführt. Dies soll wiederum die Modellqualität erhöhen. Die GoM wurden von Becker, Rosemann und Schütte, in Anlehnung an den Grundsätzen ordnungsgemäßer Buchführung (GoB), entwickelt. (Becker, Rosemann, & Schütte, 1995, S. 437)

Die GoM definieren einen Ordnungsrahmen, bestehend aus sechs Grundsätzen, welchen wiederum ein Metamodell zugrunde liegt, in dem die einzelnen GoM Objekte, Attribute und ihre Beziehungen untereinander definiert werden (Becker, Rosemann, & Schütte, 1995, S. 437ff):

- Grundsatz der Richtigkeit,
- Grundsatz der Relevanz,
- Grundsatz der Wirtschaftlichkeit,
- Grundsatz der Klarheit,
- Grundsatz der Vergleichbarkeit und dem
- Grundsatz des systematischen Aufbaus.

Zusammengefasst sollen sie die Qualität von Informationsmodellen sicherstellen.

### 1.3.1 Grundsatz der Richtigkeit

Ein Modell kann sowohl semantisch als auch syntaktisch richtig sein. Die syntaktische Richtigkeit bezieht sich dabei auf die korrekte Einhaltung der Vorgaben des Metamodells. Bei der semantischen Richtigkeit handelt es sich um die korrekte Abbildung des Modellierungsobjektes. Dies bedeutet nicht nur, dass das Modellierungsobjekt, dem Zweck entsprechend, vollständig abgebildet sein muss, sondern auch, dass es in sich keine Widersprüche aufweisen darf. Die Richtigkeit beinhaltet auch die eindeutige und korrekte Benennung von modellierten Objekten. Hierbei sollen Synonyme<sup>3</sup> und Homonyme<sup>4</sup> vermeiden werden.

---

<sup>3</sup> Synonyme sind unterschiedliche Benennungen für den gleichen Begriff. (z.B. kann für den Begriff Auto auch Kraftfahrzeug, Wagen oder Automobil verwendet werden).

<sup>4</sup> Ein Homonym ist die gleiche Benennung für zwei oder mehrere Begriffe (z.B. Golf ist ein Sport, eine Meeresbucht und ein Fahrzeug).

### **1.3.2 Grundsatz der Relevanz**

Bei der Modellierung müssen, die mit ihr bezweckten Ziele eingehalten werden. Die in dem Modell enthaltenen Elemente und Beziehungen sind relevant, wenn das Modellierungsziel bei ihrer Reduzierung nicht mehr erfüllt werden würde. Das Modell muss sich wieder in eine Beschreibung des beschriebenen Objektes zurückführen lassen, ohne für die Betrachtung relevante Aspekte zu verlieren.

### **1.3.3 Grundsatz der Wirtschaftlichkeit**

Der Grundsatz der Wirtschaftlichkeit limitiert die Tiefe und den Umfang eines Modells hinlänglich der, durch die Modellierung selbst, entstehenden Kosten. Es handelt sich dabei um eine Abwägung zwischen dem, durch die Modellierung entstehenden, Nutzen durch eine bessere Abbildbarkeit des zugrunde liegenden Problems und der Verbesserung der Interaktion zwischen den Modellnutzern und den Kosten, die durch die Modellierung entstehen. Je umfangreicher und tiefergehend ein Modell gebildet wird, desto mehr Aufwand muss in seine Erstellung investiert werden.

### **1.3.4 Grundsatz der Klarheit**

Die Anwendbarkeit des Grundsatzes der Klarheit ist, ähnlich der Relevanz, stark von den Anwendern des Modells abhängig. So ist es nötig, die Darstellung eines Modells der Zielgruppe anzupassen. Analog zur syntaktischen Richtigkeit ist auch eine klare Darstellung des Modells essenziell. Dabei geht es insbesondere um die verwendeten Symbole auf Ebene des Metamodells, als auch um die Anordnung der Symbole innerhalb des Modells selbst. Dieser Grundsatz kann beispielsweise bedeuten, dass ein sehr komplexes Modell in weitere Abstrahierungsebenen geteilt werden muss, um für den Anwender verständlich zu bleiben. Hierfür werden meist die Möglichkeiten von Substitutionen und Detailmodellen geschaffen.

### **1.3.5 Grundsatz der Vergleichbarkeit**

hnlich dem Grundsatz der Richtigkeit wird beim Grundsatz der Vergleichbarkeit in eine syntaktische und eine semantische Vergleichbarkeit unterschiedenen. Die beiden Grundsätze der Richtigkeit und Vergleichbarkeit lehnen sich dabei auch aneinander an. Bei der syntaktischen Vergleichbarkeit handelt es sich um eine Vergleichbarkeit unterschiedlicher Modelle, mit derselben zugrunde liegenden Modellierungsanforderung. Dadurch soll auch eine Überführung von Modellen mit unterschiedlichen Metamodellen ermöglicht werden. Die semantische Vergleichbarkeit bezieht sich auf die inhaltliche Vergleichbarkeit. Damit soll der Vergleich unterschiedlicher Modelle zueinander ermöglicht werden.

### 1.3.6 Grundsatz des systematischen Aufbaus

Bei in unterschiedlichen Sichten aufgebauten Modellen müssen die Sichten untereinander konsistent sein. Das bedeutet, dass die Bezüge der unterschiedlichen Sichten zueinander vorhanden sein müssen. So muss beispielsweise ein Ablaufmodell, welches auf ein Datenmodell referenziert, auch sicherstellen, dass das entsprechende Datenmodell entsprechend modelliert ist und umgekehrt. Um hierbei alle relevanten Sichten zu integrieren, liegen solchen Modellen übergreifende Architekturkonzepte zu Grunde.

### 1.3.7 Erweiterung der GoM

Die eingeführten Grundsätze ordnungsgemäßer Modellierung wurden von ihrem Mitbegründer Reinhard Schütte in Form der „neuen Grundsätze ordnungsgemäßer Modellierung“ überarbeitet. Diese erweiterten Grundsätze weichen von den herkömmlichen in einigen Teilen ab. Das Ziel der Überarbeitung war, die theoretische Begründung und die praktische Prüfbarkeit der Grundsätze zu verbessern.

Diese „neuen Grundsätze ordnungsgemäßer Modellierung“ sind (Schütte, 1997):

- Grundsatz der Konstruktionsadäquanz,
- Grundsatz der Sprachadäquanz,
- Grundsatz des systematischen Aufbaus,
- Grundsatz der Klarheit und der
- Grundsatz der Vergleichbarkeit.

Die erweiterten Grundsätze selbst werden aus, als allgemeingültig erachteten Zwecken abgeleitet, welche in Abbildung 1 dargestellt sind.

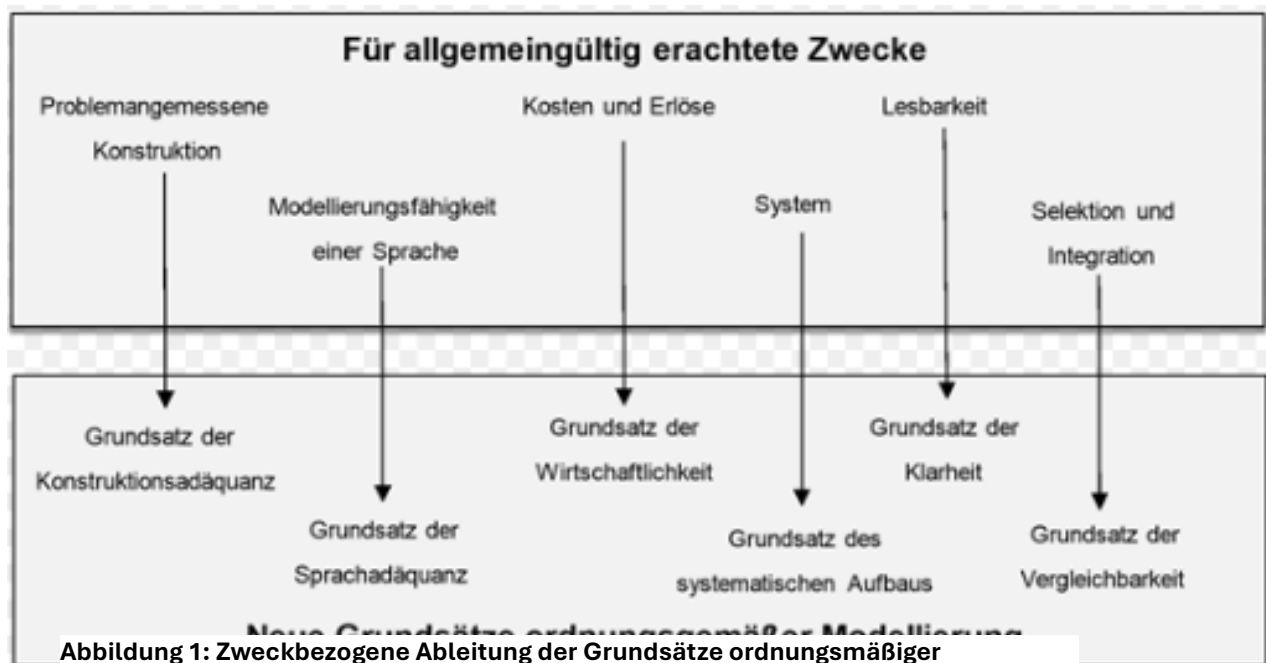


Abbildung 1: Zweckbezogene Ableitung der Grundsätze ordnungsmäßiger Modellierung (Schütte, 1997)

GoM:

- Grundsatz der Konstruktionsadäquanz und der
- Grundsatz der Sprachadäquanz.

Diese beiden sollten den Grundsatz der Richtigkeit und den Grundsatz der Relevanz ersetzen. Inhaltlich wurden jedoch auch bei den anderen Grundsätzen einige Ergänzungen durchgeführt.

Der **Grundsatz der Konstruktionsadäquanz** ist dabei als wichtigster Grundsatz definiert. Hierbei wird der Nutzen des Modells, für seinen jeweiligen Modellierungszweck, in den Vordergrund gestellt und der Konsens der Modellersteller herangezogen, da sie bewerten können, ob der Abstrahierungsgrad des Modells die Realität für die Aufgabenstellung angemessen abbildet. Es muss somit zwischen den Anwendern eine Einigkeit über die, ausreichend durch das Modell abgebildete, Problemstellung und die Art der gewählten Darstellung bestehen. Auch muss ein Konsens bezüglich einheitlicher Namenskonventionen herrschen. Ein inhaltlich identes Attribut oder Objekt muss auch in allen Modellen dieselbe Bezeichnung haben.

Der **Grundsatz der Sprachadäquanz** beschreibt dabei die Spracheignung und Sprachrichtigkeit in Relation zum Modellsystem. Die Spracheignung bezieht sich auf die Auswahl der geeigneten Modellierungstechnik für den jeweiligen Modellierungszweck. Die Sprachrichtigkeit bezieht sich dagegen auf die korrekte Anwendung der, im Metamodell definierten, Sprachsyntax.

Diese neuen GoM werden jedoch, im Gegensatz zu ihrer ursprünglichen Version, kaum angewendet, obwohl sie sich durch eine hohe theoretische Begründung und praktische Prüfbarkeit auszeichnen.

## 1.4 Einführung in Prozess-, Ablauf- und Geschäftsprozessmodelle

In den folgenden Kapiteln werden wir auf die wichtigsten Notationsformen der Prozess- und Ablaufmodelle sowie der Geschäftsmodelle eingehen. Doch zunächst gilt es, die wesentlichen Unterscheidungsmerkmale darzustellen.

### 1.4.1 Prozess- und Ablaufmodelle

sind der Oberbegriff für Modelle, die zeigen, **wie eine Arbeit oder ein Vorgang abläuft**. Sie können sehr unterschiedliche Bereiche betreffen, etwa einen technischen Ablauf, einen Produktionsschritt, einen Softwareprozess oder eine interne Arbeitsanweisung.

Beispiel:

„Eine Rechnung wird geprüft, freigegeben und bezahlt.“

### 1.4.2 Geschäftsprozessmodelle

sind eine spezielle Form davon. Sie beschreiben Abläufe **im betrieblichen bzw. organisatorischen Kontext**. Dabei stehen meist nicht nur die einzelnen Schritte im Mittelpunkt, sondern auch:

- beteiligte Personen oder Abteilungen,
- eingesetzte Systeme,
- Entscheidungen und Ausnahmen,
- Informationen und Dokumente,
- Nutzen bzw. Ergebnis für Kund:innen oder das Unternehmen.

#### Beispiel eines Geschäftsprozesses:

„Eine Kundin bestellt ein Produkt → Auftrag wird erfasst → Lager prüft Verfügbarkeit → Ware wird versendet → Rechnung wird erstellt.“

#### Gegenüberstellung:

| Begriff                           | Bedeutung                                                                              |
|-----------------------------------|----------------------------------------------------------------------------------------|
| <b>Prozess- und Ablaufmodelle</b> | Allgemeine Darstellung von Abläufen, auch außerhalb eines Unternehmens                 |
| <b>Geschäftsprozessmodelle</b>    | Darstellung betrieblicher Abläufe mit Rollen, Organisation, Informationen und Systemen |

Ein Geschäftsprozessmodell ist also **immer ein Prozess- bzw. Ablaufmodell**, aber nicht jedes Ablaufmodell ist automatisch ein Geschäftsprozessmodell.

## 1.5 Hinweis zur grafischen Darstellung und Werkzeugunterstützung

In den Kapiteln 2 bis 4 werden unterschiedliche Modellierungsansätze vorgestellt. Dabei können sich die verwendeten Symbole je nach Modellierungswerkzeug hinsichtlich Farbe, Form oder einzelner grafischer Details unterscheiden. Dies liegt daran, dass viele Hersteller die standardisierten Notationen an die Gestaltung ihrer Softwareprodukte anpassen.

Die grundlegende Bedeutung der Elemente sowie die Regeln und Konventionen der jeweiligen Modellierungssprache bleiben davon unberührt. Unterschiede können auch zwischen verschiedenen Produkten oder Produktversionen desselben Herstellers auftreten, beispielsweise zwischen ARIS Express und ARIS Architect. Entscheidend ist daher nicht die konkrete grafische Ausgestaltung eines Symbols, sondern dessen fachliche Bedeutung innerhalb des Modells.

Wichtig ist außerdem, professionelle Modellierungswerkzeuge von reinen Zeichenprogrammen zu unterscheiden. Bei der Modellierung werden nicht lediglich grafische Formen miteinander verbunden. Die verwendeten Modellelemente werden vielmehr als strukturierte Objekte in einem zentralen Repository gespeichert und können dadurch in unterschiedlichen Modellen wiederverwendet, miteinander verknüpft und systematisch ausgewertet werden. Diese strukturierte

Datenbasis ermöglicht weiterführende Anwendungen wie Prozessanalysen, automatisierte Berichte, Konsistenzprüfungen oder – abhängig vom eingesetzten Werkzeug – die technische Ausführung und Automatisierung von Prozessen.

## Lektion 2 Flussdiagramme

*Die Lektion 2 fasst im Wesentlichen den Begriff und die Modellierung von Flussdiagrammen zusammen, welche auf der Webseite der Firma Miro beschrieben wird. Sofern nicht anders angegeben, sind die entsprechenden Abbildungen ebenfalls von der Webseite entnommen (Miro DACH GmbH., 2026).*

Flussdiagramme helfen dabei, Prozesse, Abläufe und Entscheidungen übersichtlich darzustellen. Mithilfe standardisierter Symbole und verbindender Pfeile werden einzelne Schritte in einer klaren, meist sequenziellen Reihenfolge visualisiert.

Mit einem geeigneten Online-Tool lassen sich Flussdiagramme einfach und rasch erstellen. Dadurch können auch komplexe Zusammenhänge verständlich aufbereitet, dokumentiert und kommuniziert werden.

Flussdiagramme, auch Flowcharts genannt, sind ein bewährtes Instrument zur Dokumentation und Visualisierung von Prozessen. Aufgrund ihrer Vielseitigkeit kommen sie in vielen Bereichen zum Einsatz – etwa bei der Entscheidungsfindung, der Problemlösung oder der Analyse und Verbesserung von Abläufen und Systemen.

Ein Flussdiagramm ist eine grafische Darstellung eines Prozesses oder Arbeitsablaufs. Mithilfe standardisierter Symbole und Pfeile werden einzelne Arbeitsschritte, Entscheidungen und deren Reihenfolge übersichtlich dargestellt. Dadurch wird sichtbar, wie ein Prozess abläuft und welche Wege innerhalb des Prozesses möglich sind.

Flussdiagramme werden daher auch als Fließdiagramme oder Ablaufdiagramme bezeichnet. Sie sind in vielen Bereichen verbreitet – etwa in der Technik, im Bildungswesen, in der Programmierung, im Projektmanagement oder in der Prozessanalyse.

Sie eignen sich sowohl zur Darstellung einfacher als auch komplexer Abläufe. Ziel ist es, dass Betrachter:innen einen schnellen und verständlichen Überblick über den gesamten Prozess erhalten.

In einem Flussdiagramm steht jedes Symbol für einen bestimmten Schritt oder eine bestimmte Entscheidung. Häufig werden dabei Aktionen, Informationen, Materialien, Dienstleistungen, Eingaben und Ergebnisse dargestellt. Besonders bei unklaren oder komplexen Abläufen kann ein Flussdiagramm helfen, Zusammenhänge besser zu verstehen und Entscheidungen nachvollziehbar vorzubereiten.

Damit Flussdiagramme sinnvoll eingesetzt werden können, werden im Folgenden ihre grundlegende Bedeutung sowie die wichtigsten Symbole und deren Verwendung erklärt. (Miro DACH GmbH., 2026)

### 2.1 Aufbau eines Flussdiagramms

Ein Flussdiagramm stellt einen Ablauf grafisch dar. Es besteht aus standardisierten Symbolen und Verbindungslinien beziehungsweise Pfeilen. Die verwendeten Symbole stehen für unterschiedliche Elemente eines Prozesses, zum Beispiel für den Beginn oder das Ende eines Ablaufs, für Tätigkeiten, Dateneingaben, Ausgaben oder Entscheidungen. Die Pfeile verbinden diese Elemente miteinander und zeigen, in welcher Reihenfolge die einzelnen Schritte durchgeführt werden.

Üblicherweise wird ein Flussdiagramm von oben nach unten oder von links nach rechts gelesen. Die Darstellung sollte dabei so klar und übersichtlich sein, dass auch außenstehende Personen den

Ablauf ohne zusätzliche mündliche Erklärung verstehen können. Lange, sich kreuzende oder unübersichtliche Verbindungslinien sollten daher möglichst vermieden werden.

Jedes Flussdiagramm beginnt mit einem Startsymbol und endet mit einem oder mehreren Endsymbolen. Dazwischen werden die einzelnen Prozessschritte dargestellt. Dazu gehören Tätigkeiten, Ein- und Ausgaben von Daten sowie Entscheidungen.

Prozessschritte werden möglichst kurz und eindeutig als Tätigkeiten formuliert, zum Beispiel „Antrag prüfen“, „Daten erfassen“ oder „Bestätigung senden“. Entscheidungen werden dagegen als Fragen formuliert, etwa „Antrag vollständig?“ oder „Zahlung eingegangen?“. Von einem Entscheidungssymbol müssen mindestens zwei klar beschriftete Wege ausgehen, beispielsweise „Ja“ und „Nein“. Dadurch wird ersichtlich, wie der Prozess abhängig vom Ergebnis der Entscheidung weitergeführt wird.

## **2.2 Symbole in Flussdiagrammen**

Für die Erstellung von Flussdiagrammen werden meist standardisierte Formen und Symbole verwendet. Jedes Symbol steht für einen bestimmten Schritt, eine Entscheidung oder ein anderes Element innerhalb eines Prozesses beziehungsweise einer Ablauffolge.

Je nach Anwendungsbereich kann sich die genaue Bedeutung einzelner Symbole leicht unterscheiden. Die folgenden Symbole helfen dir dabei, Prozesse übersichtlich zu strukturieren und verständlich darzustellen. (Miro DACH GmbH., 2026)

| Symbol                                                                              | Bezeichnung                                   | Erklärung                                                                                                                                                   |
|-------------------------------------------------------------------------------------|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | <b>Oval / Terminal</b>                        | Kennzeichnet den Beginn oder das Ende eines Prozesses.                                                                                                      |
|    | <b>Parallelogramm</b>                         | Steht für die Eingabe oder Ausgabe von Daten, etwa das Erfassen einer Information oder die Anzeige eines Ergebnisses.                                       |
|    | <b>Rechteck / Prozess</b>                     | Beschreibt eine Tätigkeit oder einen Arbeitsschritt innerhalb des Ablaufs.                                                                                  |
|    | <b>Raute / Entscheidung</b>                   | Stellt eine Entscheidung oder Prüfung dar. Von ihr führen meist unterschiedliche Wege weiter, beispielsweise „Ja“ und „Nein“.                               |
|    | <b>Flusslinie / Pfeil</b>                     | Zeigt die Reihenfolge und Richtung der einzelnen Prozessschritte an.                                                                                        |
|    | <b>Kreis / Konnektor</b>                      | Verbindet getrennte Stellen innerhalb eines komplexen Flussdiagramms, ohne lange oder unübersichtliche Linien zeichnen zu müssen.                           |
|  | <b>Klammer / Kommentar</b>                    | Ergänzt Erläuterungen zu einem bestimmten Bereich oder Schritt. Eine gestrichelte Linie kann auf das betreffende Element verweisen.                         |
|  | <b>Dokumentensymbol</b>                       | Kennzeichnet ein Dokument, das innerhalb des Prozesses erstellt, verwendet oder verarbeitet wird. Mehrere überlagerte Symbole stehen für mehrere Dokumente. |
|  | <b>Hexagon / Vorbereitung</b>                 | Stellt einen vorbereitenden Schritt dar, der vor der eigentlichen Prozessausführung notwendig ist.                                                          |
|  | <b>Vertikaler Zylinder / Datenbank</b>        | Kennzeichnet eine Datenbank oder einen anderen strukturierten Datenspeicher.                                                                                |
|  | <b>Horizontaler Zylinder / Direktspeicher</b> | Steht für einen Datenspeicher mit direktem Zugriff auf die gespeicherten Informationen.                                                                     |
|  | <b>Umgedrehtes Dreieck / Speicherung</b>      | Kennzeichnet die Ablage oder Speicherung von Daten, Unterlagen oder Zwischenergebnissen.                                                                    |

**Tabelle 2: Symbole in Flussdiagrammen**

## 2.3 Vorgehen bei der Erstellung

Die Erstellung eines Flussdiagramms beginnt nicht mit dem Zeichnen, sondern mit dem Verstehen des Ablaufs. Zunächst wird festgelegt, welcher Prozess oder welche Entscheidungssituation modelliert werden soll. Danach werden Start- und Endpunkt bestimmt. Anschließend werden die wesentlichen Tätigkeiten in eine sinnvolle Reihenfolge gebracht. Erst danach werden Entscheidungen ergänzt, die den Ablauf verzweigen oder zu Schleifen führen können.

Für die Modellierung ist es hilfreich, den Ablauf zunächst in kurzen Sätzen zu beschreiben. Aus diesen Sätzen können anschließend die Elemente des Diagramms abgeleitet werden. Tätigkeiten werden zu Rechtecken, Fragen zu Entscheidungsrauten und eindeutige Ergebnisse zu Endpunkten. Wird während der Erstellung deutlich, dass ein Schritt zu groß oder zu unklar ist, sollte er weiter unterteilt werden. Umgekehrt sollte ein Flussdiagramm nicht jeden technischen Klick darstellen, wenn dadurch die Übersichtlichkeit verloren geht.

Ein gutes Flussdiagramm ist also weder eine vollständige Kopie der Realität noch eine zu stark vereinfachte Skizze. Es reduziert die Realität auf jene Schritte und Entscheidungen, die für das Verständnis oder die Analyse des Ablaufs wesentlich sind.

## 2.4 Modellierungsregeln

**Eindeutiger Start und klares Ende:** Der betrachtete Ablauf muss erkennbar beginnen und abgeschlossen werden. Mehrere Endpunkte sind zulässig, wenn sie unterschiedliche fachliche Ergebnisse darstellen.

**Tätigkeiten mit Verben formulieren:** Prozessschritte sollten aktiv beschrieben werden, zum Beispiel „Kundendaten erfassen“ statt nur „Kundendaten“. Dadurch wird sichtbar, was tatsächlich geschieht.

**Entscheidungen als Fragen formulieren:** Eine Raute enthält eine prüfbare Frage, zum Beispiel „Voraussetzungen erfüllt?“. Die ausgehenden Pfeile werden eindeutig beschriftet, etwa mit „Ja“ und „Nein“.

**Leserichtung einhalten:** Der Ablauf sollte möglichst von oben nach unten **oder** von links nach rechts verlaufen. Achten Sie darauf, dass Sie die Leserichtung einhalten und nicht mischen. Rücksprünge sind erlaubt, sollten aber als Schleifen klar erkennbar sein.

**Kreuzende Pfeile vermeiden:** Viele kreuzende Linien erschweren die Lesbarkeit. Bei komplexen Abläufen sollte das Diagramm vereinfacht oder in Teilprozesse zerlegt werden.

**Angemessener Detaillierungsgrad:** Das Diagramm soll den Ablauf erklären und analysierbar machen. Zu viele Details machen es unübersichtlich, zu wenige Details machen es fachlich unbrauchbar.

## 2.5 Beispiel: Online-Prüfungsanmeldung

Das folgende Beispiel zeigt einen einfachen Ausschnitt einer Online-Prüfungsanmeldung. Ein Student gibt seine Zugangsdaten ein. Sind diese falsch, wird eine Fehlermeldung angezeigt und der Student kann die Eingabe wiederholen. Sind die Zugangsdaten korrekt, wird die Prüfung ausgewählt und die Anmeldefrist geprüft. Je nach Ergebnis endet der Prozess mit einer Fehlermeldung oder mit einer gespeicherten Anmeldung.

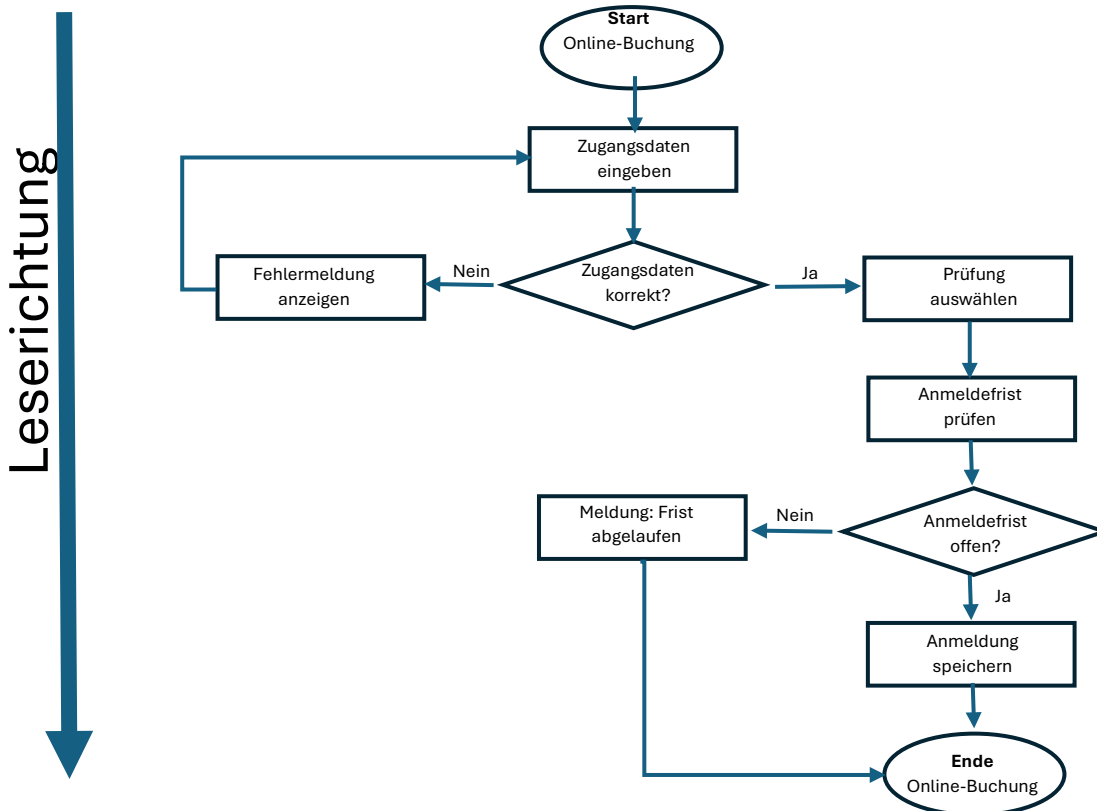
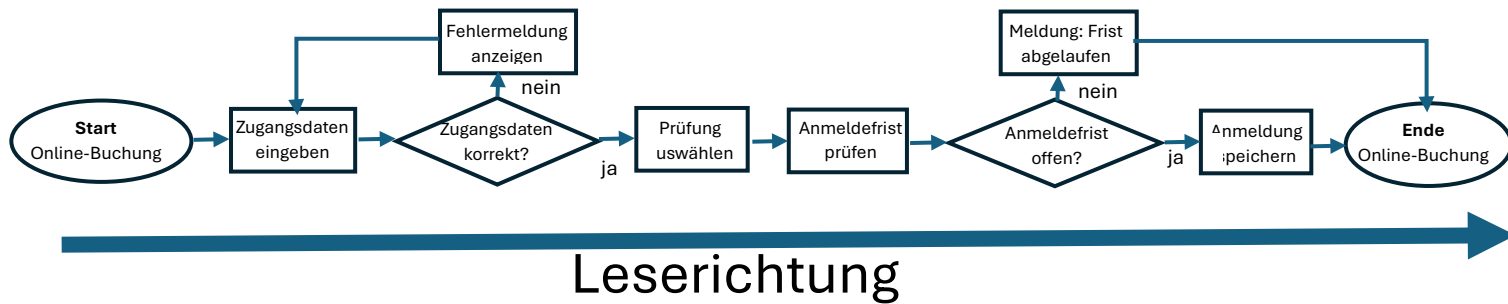


Abbildung 2: Flussdiagramm – Prüfung Online-Buchung



**Abbildung 3: Flussdiagramm - Prüfung Online-Buchung - horizontale Leserichtung**

An den Abbildungen ist erkennbar, dass Entscheidungen den Ablauf verzweigen. Die Frage „Zugangsdaten korrekt?“ führt entweder zur Fortsetzung des Prozesses oder zu einer Schleife zurück zur erneuten Eingabe. Die Frage „Frist offen?“ führt entweder zum Abbruch mit Meldung oder zur Speicherung der Anmeldung. Damit zeigt das Flussdiagramm nicht nur die Reihenfolge der Schritte, sondern auch die fachliche Logik des Prozesses.

## 2.6 Typische Fehler

Häufige Fehler entstehen, wenn Tätigkeiten und Entscheidungen verwechselt werden. Eine Formulierung wie „Zahlung eingegangen?“ ist keine Tätigkeit, sondern eine Entscheidung und gehört daher in eine Raute. Ebenso problematisch sind unbeschriftete Entscheidungsausgänge, weil dann nicht klar ist, welcher Pfeil für welchen Fall gilt. Weitere typische Fehler sind fehlende Endpunkte, zu große Prozessschritte wie „Anmeldung bearbeiten“ oder ein unübersichtliches Diagramm mit vielen kreuzenden Pfeilen.

## 2.7 Arten von Flussdiagrammen

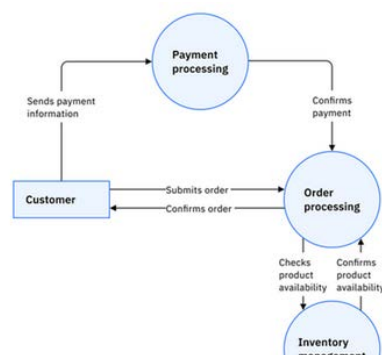
Neben der Darstellung von Prozessen können Flussdiagramme auch in vielen anderen Bereichen eingesetzt werden, zum Beispiel zur Visualisierung von Entscheidungen, zur Fehlersuche oder zur Beschreibung technischer Abläufe. Deshalb sollte vor der Erstellung eines Diagramms überlegt werden, welches Format und welche Darstellung am besten zur jeweiligen Aufgabenstellung passen.

### Datenflussdiagramm DFD

Ein Datenflussdiagramm (DFD) stellt grafisch dar, wie Daten innerhalb eines Informationssystems oder Geschäftsprozesses verarbeitet und weitergeleitet werden. Es zeigt, aus welchen Quellen Daten stammen, welche Prozesse sie verarbeiten, wo sie gespeichert werden und an welche Stellen sie weitergegeben werden.

Durch diese vereinfachte Darstellung lassen sich auch komplexe Systeme leichter verstehen und analysieren. Datenflussdiagramme werden daher häufig in der Softwareentwicklung, Systemanalyse, Prozessoptimierung und bei der Planung neuer Informationssysteme eingesetzt.

Das Modell unterstützt Fachleute dabei, Datenwege transparent zu machen, Schwachstellen zu erkennen und bestehende Prozesse gezielt effizienter und wirksamer zu gestalten. (IBM, 2026)

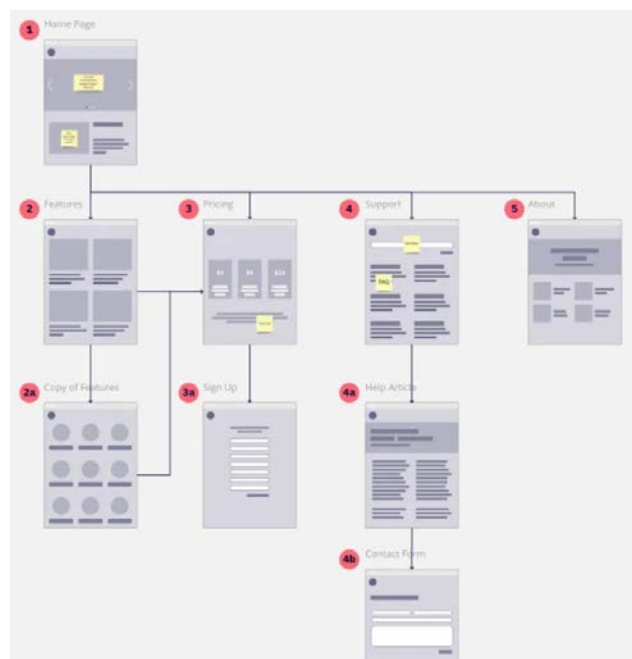


**Abbildung 4: Ein vereinfachtes logisches Datenflussdiagramm, das den Bestellvorgang auf einer E-Commerce-Website veranschaulicht (IBM, 2026)**

## Website Flussdiagramm

Die Erstellung einer Website ist deutlich komplexer, als es auf den ersten Blick erscheint. Neben der Gestaltung des Erscheinungsbildes müssen Inhalte sinnvoll strukturiert, Navigationsmenüs übersichtlich aufgebaut und einzelne Seiten logisch miteinander verknüpft werden. Auch die Auswahl geeigneter Bilder, Texte, Farben und Schriftarten spielt eine wichtige Rolle.

Darüber hinaus sollte eine Website auf unterschiedlichen Geräten – etwa Smartphones, Tablets und Computern – gut lesbar und einfach bedienbar sein. Eine klare Navigation, verständliche Inhalte und ein ansprechendes Design tragen dazu bei, dass Besucher:innen schnell die gewünschten Informationen finden und gerne auf der Website bleiben. (Miro DACH GmbH., 2026)



**Abbildung 5: Website Flussdiagramm (Miro DACH GmbH., 2026)**

## Entscheidungsbaum Diagramm

Ein Entscheidungsbaum ist eine besondere Form des Flussdiagramms. Er zeigt mögliche Entscheidungen, Handlungsalternativen und deren erwartete Ergebnisse. Dadurch unterstützt er dabei, die für ein Ziel passende Entscheidung zu treffen.

Der Entscheidungsbaum beginnt mit einer Ausgangsfrage oder Aufgabe, dem sogenannten Wurzelknoten. Von diesem gehen verschiedene Entscheidungswege ab. Jeder Zweig führt zu weiteren Fragen, Maßnahmen oder Ergebnissen, bis ein möglicher Endpunkt erreicht ist.

Entscheidungsbäume werden unter anderem in Medizin, Finanzwesen, Informatik und Unternehmensplanung eingesetzt. Sie können aber überall dort hilfreich sein, wo Entscheidungen nachvollziehbar strukturiert und verschiedene Folgen verglichen werden sollen. (Canva Pty Ltd, 2026)



**Abbildung 6: Entscheidungsbaum (Canva Pty Ltd, 2026)**

Die vorgestellten Beispiele zeigen nur einen kleinen Ausschnitt der vielfältigen Einsatzmöglichkeiten von Flussdiagrammen. Je nach Anwendungsbereich, Zielsetzung und gewünschtem Detaillierungsgrad gibt es zahlreiche weitere Diagrammarten, die Prozesse, Entscheidungen, Informationen oder Abläufe grafisch darstellen können, auf die wir im Rahmen dieser Lehrveranstaltung aber nicht weiter eingehen werden.

Weitere häufig verwendete Formen sind beispielsweise:

- Swimlane-Diagramme zur Darstellung von Verantwortlichkeiten
- Prozessflussdiagramme für betriebliche Abläufe
- Programmablaufpläne in der Softwareentwicklung
- Datenflussdiagramme zur Darstellung von Datenbewegungen
- Workflow-Diagramme für Arbeits- und Freigabeprozesse
- Systemflussdiagramme zur Darstellung technischer Systeme
- BPMN-Diagramme für Geschäftsprozesse
- Ursache-Wirkungs-Diagramme zur Analyse von Problemen

## **Lektion 3      Geschäftsprozessmodellierung**

Ergänzend zu den bereits betrachteten Modellen der Prozess- und Ablaufmodellierung gibt es noch einige spezielle Geschäftsprozessmodelle. In dieser Lektion werden nun die Grundlagen von zwei der diesbezüglich wichtigsten Modelle erläutert. Diese Modelle dienen der organisatorischen Betrachtung von Geschäftsprozessen und werden auch von Nicht-Technikern eingesetzt. Sie dienen oft auch als gemeinsame Basis bei der Spezifikation der Anforderungen von IT-Systemen und ihrer Unterstützung der Geschäftsprozesse.

### **3.1 Ereignisgesteuerte Prozessketten (EPK)**

Die Ereignisgesteuerten Prozessketten (EPK) wurden 1992 durch Keller, Nüttgens und Scheer entwickelt. Sie wurden von den Petri-Netzen abgeleitet und dienen der prozessorientierten Modellierung von Geschäftsprozessen. Sie sind Teil des Rahmenkonzeptes der Architektur

integrierter Informationssysteme (ARIS) und dort im Bereich der Steuerungssicht angesiedelt (Keller, 1992).

Geschäftsprozesse werden als Prozessketten dargestellt, in denen Funktionen durch Start- und Endereignisse miteinander verbunden sind. Ereignisse beschreiben betriebswirtschaftlich relevante Zustände von Informationsobjekten und beeinflussen oder steuern den weiteren Prozessablauf. Sie können Funktionen auslösen oder deren Ergebnis darstellen (Software AG, 2019).

Im Unterschied zu Funktionen, die einen zeitlichen Ablauf beschreiben, beziehen sich Ereignisse auf einen bestimmten Zeitpunkt. Sie können entweder das erstmalige Auftreten eines Informationsobjekts kennzeichnen, etwa „Kundenanfrage ist eingetroffen“, oder eine Statusänderung ausdrücken, beispielsweise „Angebot ist abgelehnt“ (Software AG, 2019).

Da Informationsobjekte und ihre Attribute in der Datensicht von ARIS beschrieben werden, verbinden ereignisgesteuerte Prozessketten die Daten- und Funktionssicht. Sie werden daher der Steuerungssicht von ARIS zugeordnet. Grafisch werden Ereignisse als Sechsecke dargestellt. Ihre Bezeichnung sollte stets das Informationsobjekt und dessen Zustand enthalten, etwa „Auftrag ist eingetroffen“ (Software AG, 2019).



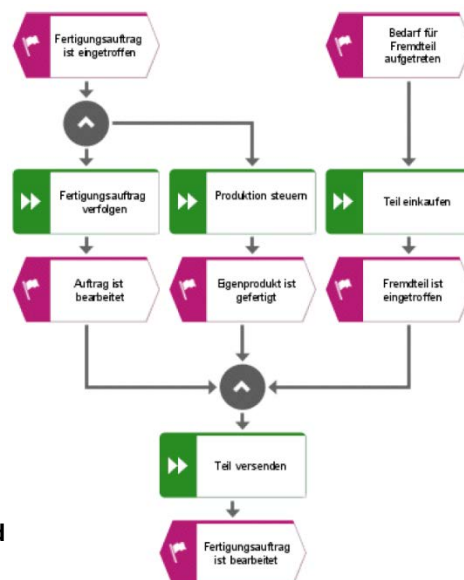
**Abbildung 7: Ereignis** **stellung (Software AG, 2019)**

### 3.1.1 Aufbau einer EPK

Ereignisse können Funktionen auslösen und zugleich das Ergebnis einer ausgeführten Funktion darstellen. Durch die abwechselnde Abfolge von Ereignissen und Funktionen entstehen ereignisgesteuerte Prozessketten, kurz EPK. Sie bilden den logisch und zeitlich geordneten Ablauf eines Geschäftsprozesses ab (Software AG, 2019).

Eine EPK beginnt und endet immer mit einem Ereignis. Ereignisse beschreiben, unter welcher Bedingung eine Funktion gestartet wird, oder welchen Zustand diese Funktion nach ihrer Durchführung erreicht. Von einem Ereignis können mehrere Funktionen gleichzeitig ausgehen. Ebenso kann eine Funktion mehrere unterschiedliche Ereignisse als Ergebnis haben (Software AG, 2019).

Damit Verzweigungen, Zusammenführungen oder Bearbeitungsschleifen dargestellt werden können, verwendet man in einer EPK Regeln in Form eines Kreises. Die Verbindungslinien haben dabei nicht nur eine grafische Funktion: Sie zeigen auch die logischen Beziehungen zwischen den einzelnen Elementen des Prozesses (Software AG, 2019).



Abbildung

2019)

Der Prozessablauf wird über einen Token-Ansatz abgearbeitet. Das bedeutet, dass immer ganz bestimmte Teilpfade des Kontrollflusses aktiv sind, wobei es möglich ist, eine parallele Ausführung mehrerer Pfade durchzuführen, indem der aktuelle Token auf mehrere Teilpfade aufgeteilt wird. Ein Zusammenführen oder Synchronisieren mehrerer Pfade zu einem einzigen Pfad ist ebenfalls möglich. Diese Teilung und Zusammenführung erfolgt über logische Operatoren. Folgende Operatoren sind dabei möglich:

- exklusiver Oder-Operator (XOR),
- inklusiver Oder-Operator (OR) und
- Und-Operator (AND).

Die Darstellung der logischen Operatoren erfolgt mittels Kreissymbol mit dem jeweiligen Typ darin. Die Darstellungsformen der Operatoren-Typen sind in den folgenden Abbildungen angeführt (Keller, 1992).

### 3.1.1.1 DIE VERKNÜPFUNG VON AUSLÖSENDEN EREIGNISSEN

#### UND-Regel

Die Funktion kann erst gestartet werden, wenn alle Ereignisse eingetreten sind.

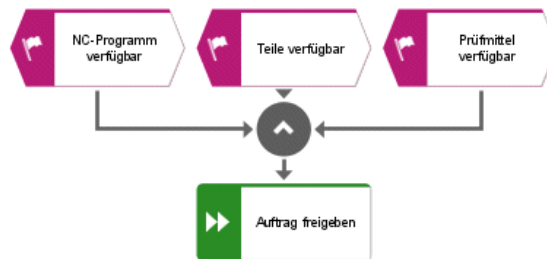


Abbildung 9: UND – Regel (Software AG, 2019)

#### ODER – Regel

Die Funktion wird ausgeführt, wenn mindestens eines der Ereignisse eingetreten ist.



Abbildung 10: ODER – Regel (Software AG, 2019)

#### EXKLUSIV - ODER - Regel

Die Funktion wird gestartet, wenn genau ein und nur ein Ereignis eingetreten ist.



Abbildung 11: Exklusiv - ODER - Verknüpfung (XOR) (Software AG, 2019)

### 3.1.1.2 DIE VERKNÜPFUNG VON ERZEUGTEN EREIGNISSEN

#### UND-Regel

Die Funktion führt dazu, dass alle Ereignisse eintreten.



Abbildung 12: UND - Verknüpfung erzeugter Ereignisse (Software AG, 2019)

#### ODER – Regel

Die Ausführung der Funktion führt dazu, dass mindestens eines der Ereignisse eintritt.



Abbildung 13: ODER - Verknüpfung erzeugter Ereignisse (Software AG, 2019)

#### EXKLUSIV - ODER – Regel (Entweder – oder – Regel XOR)

Die Ausführung der Funktion führt dazu, dass höchstens eines der Ereignisse eintritt.



Abbildung 14: Exklusiv - oder - Verknüpfung XOR erzeugter Ereignisse (Software AG, 2019)

Werden in einer EPK durch einen Operator mehrere Prozesspfade erzeugt, müssen diese im weiteren Verlauf grundsätzlich wieder mit demselben Operator zusammengeführt werden. Das heißt: Wurde ein Ablauf beispielsweise durch einen UND-, ODER- oder XOR-Operator aufgeteilt,

ist für die Zusammenführung **derselbe Operatortyp** zu verwenden. Dadurch bleibt eindeutig nachvollziehbar, unter welchen Bedingungen der Prozess weitergeführt wird.

Endet ein einzelner Pfad oder mehrere Pfade, ohne wieder zusammengeführt zu werden, müssen diese jeweils mit einem entsprechenden Endereignis abgeschlossen werden.

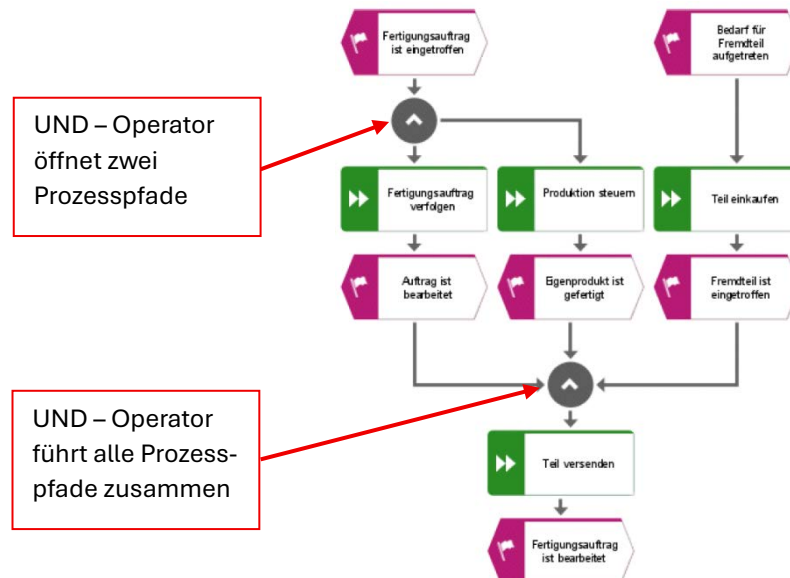
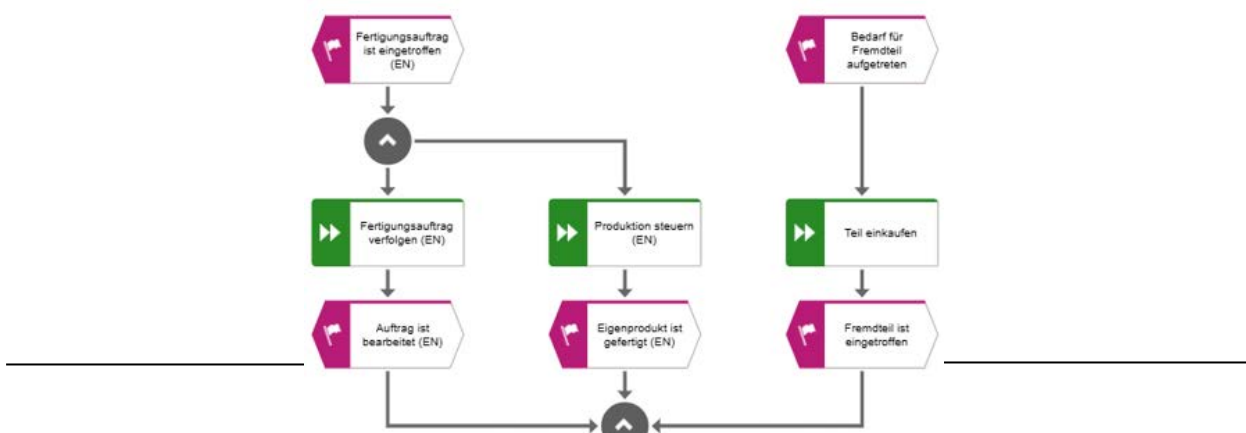


Abbildung 15: EPK - Zusammenführen mehrerer paralleler Kontrollflüsse

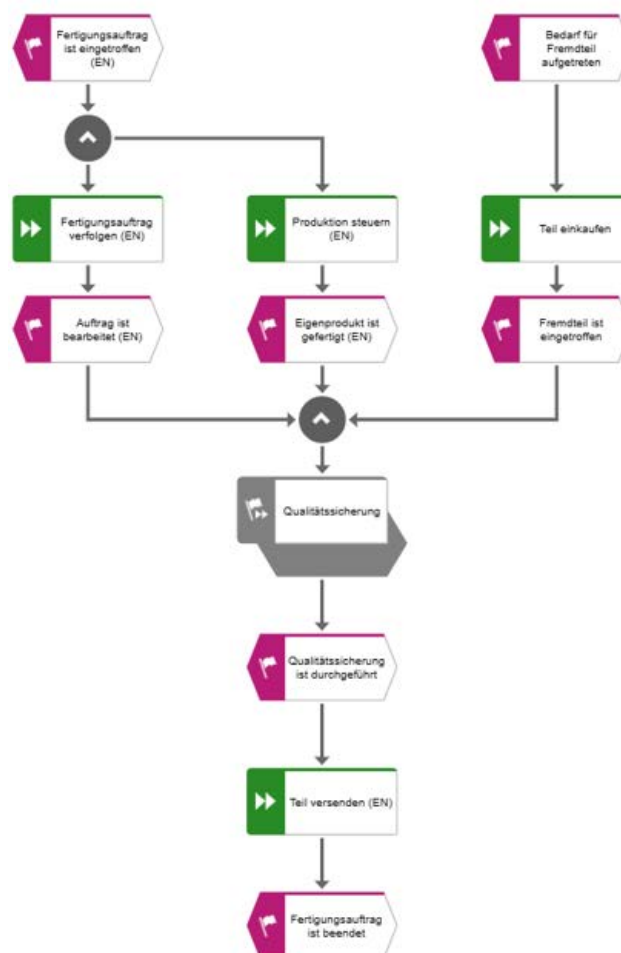
### 3.1.1.3 Subprozess Aufruf

Ein Subprozess ist ein ausgelagerter Teilprozess, der in einer EPK nicht vollständig im Hauptprozess dargestellt wird. Er beschreibt einen Ablauf, der aus mehreren eigenen Prozessschritten bestehen kann und an anderer Stelle genauer modelliert wird. Subprozesse werden vor allem dann eingesetzt, wenn ein Prozess sehr umfangreich oder unübersichtlich werden würde. Durch die Auslagerung bleibt die Haupt-EPK übersichtlich, während wiederkehrende oder komplexe Abläufe dennoch vollständig beschrieben werden können.

In der EPK wird ein Subprozess über einen sogenannten Subprozess-Aufruf eingefügt. Dieser steht an jener Stelle im Hauptprozess, an der der ausgelagerte Ablauf durchgeführt werden soll. Der Subprozess-Aufruf wird wie eine Funktion in den Prozess eingebunden, das heißt, er wird durch ein Ereignis ausgelöst und führt nach seiner Bearbeitung wieder zu einem Ereignis. Subprozesse können sowohl innerhalb eines Prozessflusses oder am Ende des Prozesses aufgerufen werden. Der detaillierte Ablauf des Subprozesses wird in einer eigenen EPK dargestellt, die ebenfalls mit einem Starterereignis beginnt und mit einem Endereignis abschließt.



**Abbildung 16: Subprozess Aufruf nach Prozessdurchlauf**



**Abbildung 17: Subprozess Aufruf innerhalb des Prozessflusses**

## 3.2 Business Process Model and Notation (BPMN)

Die Business Process Model and Notation (BPMN) wurde ursprünglich von Stephen A. White entwickelt und 2004 von der Business Process Management Initiative (BPMI) veröffentlicht. Seit Februar 2011 gibt es die BPMN nun in ihrer aktuellen Version 2.0, welche offiziell von der Object Management Group (OMG) verabschiedet und publiziert wurde. BPMN dient der Modellierung von Prozessen und kann daher andere Strukturen, wie eine übergeordnete Prozesslandschaft und Ablauforganisationen, nicht abbilden (Freund & Rücker).

Alle Elemente und der Aufbau von BPMN 2.0 sind in der zugehörigen, 538 Seiten langen, Spezifikation (Object Management Group (OMG), 2011) genau definiert. Nachfolgend werden nun die davon wichtigsten Grundaspekte erläutert.

### 3.2.1 Aufbau der BPMN

**Hinweis:** Bitte beachten Sie, dass die Darstellung der Objekte je nach verwendetem Modellierungswerkzeug unterschiedlich sein kann. Farben, Formen und einzelne Details können daher voneinander abweichen. Da in diesem Skript verschiedene Quellen verwendet wurden, können auch gleiche Objekte teilweise leicht unterschiedlich dargestellt sein.

Die BPMN kann in fünf Hauptelementtypen gegliedert werden:

- Flussobjekte,
- verbindende Objekte,
- Artefakte,
- Teilnehmer und
- Daten.

Die Flussobjekte beschreiben alle Entscheidungen, Aktivitäten und Ereignisse. Sie werden selbst wiederum über die verbindenden Objekte vernetzt und mit Hilfe der Artefakte beschrieben. Ergänzend können dann noch die Teilnehmer und relevanten Daten des Prozesses in das Modell eingefügt werden.

Generell wird, ähnlich wie bei den EPK, ein Token-Ansatz verwendet. Der Prozessfluss findet also entlang des definierten Pfades statt, welcher von einem Token durchlaufen wird. Dieser kann wiederum in mehrere parallele Token geteilt und umgekehrt wiederum zu einem einzelnen zusammengeführt werden.

Die möglichen **Flussobjekte** sind

- die Aktivität,
- das Event und
- das Gateway.

Die grafische Darstellung der Flussobjekte ist in der folgenden Abbildung ersichtlich.



**Abbildung 18: BPMN Flussobjekte**

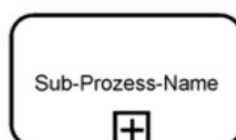
Die Aktivität beschreibt dabei eine konkrete Tätigkeit, die bei ihrem Aufruf durchgeführt wird. Diese kann in komplexeren Modellen wiederum selbst aus Unteraktivitäten bestehen. Auf diese Form der weiteren Untergliederung wird hier jedoch nicht weiter eingegangen. Eine Aktivität hat immer eine Bezeichnung und kann zusätzlich eine bestimmte Ausprägung haben. Diese wird in der linken oberen Ecke in Form eines Symbols angezeigt. Aktivitäten werden oft auch als Aufgaben bezeichnet, wenn sie selbst nicht mehr in Teilaufgaben teilbar sind. Ein Beispiel hierfür wäre die Zubereitung eines Kaffees mit einer Kaffeemaschine. Das „Zubereiten eines Espressos“ wäre eine Aktivität, da sie auch weiter in „Tasse unterstellen“ und „Espressotaste drücken“ geteilt werden könnte. Der Druck auf die Espressotaste hingegen wäre eine Aufgabe, da sie nicht weiter teilbar ist.

Eine Aktivität ohne spezielle Ausprägung wird als abstrakt bezeichnet. Die unterschiedlichen Ausprägungen und ihre Symbole sind in Abbildung 56 aufgelistet und beschrieben.

| Aktivitätsausprägung     | Symbol                                                                              | Beschreibung                                                                                       |
|--------------------------|-------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| Service-Aktivität        |    | wird durch ein automatisches System durchgeführt                                                   |
| Sende-Aktivität          |    | führt das Senden einer Nachricht durch (Erstellung und Versand)                                    |
| Empfangs-Aktivität       |    | führt das Empfangen einer Nachricht durch (Empfang und Verarbeitung)                               |
| Benutzer-Aktivität       |   | wird durch einen oder mehrere Menschen durchgeführt                                                |
| Manuelle-Aktivität       |  | wird durch einen oder mehrere Menschen manuell und ohne IT Unterstützung durchgeführt              |
| Geschäftsregel-Aktivität |  | führt eine BPMN Geschäftsregel aus (wird in dieser Lektion nicht weiter behandelt)                 |
| Script-Aktivität         |  | führt ein Script innerhalb der Prozessumgebung aus (wird in dieser Lektion nicht weiter behandelt) |

**Tabelle 3: BPMN Aktivitäten mit Ausprägung (Object Management Group)**

Weiter ist es möglich, über eine Aktivität einen Sub-Prozess aufzurufen. Damit kann das Modell vereinfacht und mehrfach ausgeführte Prozessabläufe zusammengefasst werden. Ein Sub-Prozess wird durch ein kleines Quadrat mit einem Pluszeichen in der Mitte des unteren Bereiches der Aktivität dargestellt. Der Inhalt des Sub-Prozesses wird in Form eines eigenen Pools modelliert, welcher in diesem Abschnitt weiter unten genauer beschrieben wird. Ein Beispiel für einen Sub-Prozess Aufruf ist in Abbildung 39 dargestellt



**Abbildung 19: BPMN Sub-Prozess Aufruf (Object Management Group)**

Ein Event ist ein bestimmtes Ereignis, welches eintreten und ausgelöst werden kann. Es kann dabei sowohl von außerhalb ausgelöst werden, als auch durch den modellierten Prozess selbst. Dabei wird zwischen Start-, Zwischen- und End- Events unterschieden. Ein Start-Event ist ein Ereignis, welches außerhalb des modellierten Prozesses ausgelöst wird und damit einen Einfluss auf einen Teil des Prozesses von außen hat. Start-Events werden daher auch zum Start des Prozesses eingesetzt. End-Events beenden einzelne Abläufe des Prozesses. Ist kein Teil des Prozesses mehr aktiv, da jeder Pfad in einem End-Event geendet hat, ist damit der gesamte Prozess beendet. Eine Ausnahme ist hierbei das End-Event vom Typ Terminierung, welches bei seinem Aufruf den gesamten Prozess beendet. Weiter gibt es noch Events, welche innerhalb des Prozesses ausgelöst werden und wiederum einen anderen Teil des eigenen Prozesses beeinflussen. Diese Events werden als Zwischen-Events bezeichnet.

Je nachdem um welchen Event Typ es sich handelt wird die Umrandung des Kreises anders dargestellt. Die unterschiedlichen Event-Typen sind in Abbildung 58 dargestellt.



**Abbildung 20: BPMN - Event Typen**

Der Start eines Prozessflusses wird immer durch ein oder mehrere Start-Events begonnen und durch ein oder mehrere End-Events beendet.

Zusätzlich kann der Typ eines Events durch einen zugehörigen Auslöser oder ein Resultat beschrieben werden. Die wichtigsten Auslöser oder Resultate sind:

- Nachricht,
- Signal,
- Timer,
- Bedingung,
- Verknüpfung und
- Terminierung.

Bei einer Nachricht handelt es sich um eine Benachrichtigung von außen an den Prozess oder umgekehrt, sowie möglicher Benachrichtigungen zwischen Aktivitäten. Signale sind, im Gegensatz zu Nachrichten, Benachrichtigungen welche ohne bestimmte Empfänger an alle gerichtet sind, die sie auffangen können. Timer erlauben, zwischen den Aktivitäten ein bestimmtes Zeitintervall oder einen Zeitpunkt abzuwarten und Start-Events in einem bestimmten Intervall oder Zeitpunkt zu starten. Bei Bedingungen handelt es sich um Events, welche unter einer bestimmten Bedingung ausgeführt werden oder auf eine bestimmte

Bedingung warten. Bei den Verknüpfungen handelt es sich um einen Sonderfall von Events, welche eine grafische Möglichkeit darstellen den Prozessfluss innerhalb des Diagramms zu unterbrechen und an einer anderen Stelle fortzusetzen. Dabei ist es wichtig, die Verknüpfung an beiden Stellen im Diagramm gleich zu benennen. Die Terminierung beendet den gesamten Ablauf unmittelbar nach ihrem Eintreten, egal ob noch parallele Aktivitäten offen sind, oder nicht.

In Tabelle 4: BPMN Eventauslöser findet sich eine Tabelle der wichtigsten Symbole der Auslöser und Resultate und in welchen Event-Typen ihre Verwendung zulässig ist.

|              | Symbol                                                                              | Zulässig als Start-Event | Zulässig als Zwischen-Event | Zulässig als Ende-Event |
|--------------|-------------------------------------------------------------------------------------|--------------------------|-----------------------------|-------------------------|
| Nachricht    |    | Ja                       | Ja                          | Ja                      |
| Signal       |    | Ja                       | Ja                          | Ja                      |
| Timer        |   | Ja                       | Ja                          | Nein                    |
| Bedingung    |  | Ja                       | Ja                          | Nein                    |
| Verknüpfung  |  | Nein                     | Ja                          | Nein                    |
| Terminierung |  | Nein                     | Nein                        | Ja                      |

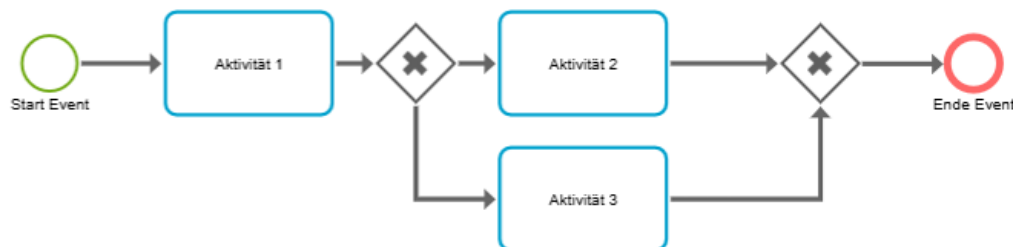
**Tabelle 4: BPMN Eventauslöser**

Gateways dienen dem Ablauffluss zwischen den Aktivitäten untereinander und zwischen Aktivitäten und Events. Sie ermöglichen den Ablauf zu teilen und wieder zusammenzuführen. Der Ablauf findet dabei immer nach dem Token-Ansatz statt. Das bedeutet, dass es immer einen bestimmten Teil des Pfades gibt, der gerade aktiv ist. Durch die Teilung des Prozess-Flusses mittels Gateways können mehrere parallele Abarbeitungen erfolgen und auch bestimmte Pfade unter bestimmten Bedingungen ausgewählt werden.

Die wichtigsten Gateway-Typen sind:

- Exklusives Oder (Entweder-Oder Verbindung),
- Inklusives Oder (Und-Oder Verbindung),
- Parallel (Und Verbindung) und
- Komplex (Kombination mehrerer Verbindungsarten).

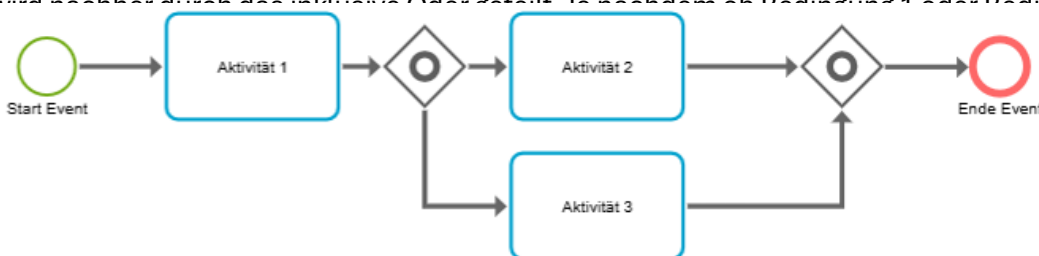
Mit einem exklusiven Oder-Gateway werden mögliche Pfade vorgegeben, von denen genau einer eingeschlagen werden kann. In Abbildung 59 ist die mögliche Verwendung von Gateways mit exklusivem Oder dargestellt. Hier löst das Event Start die Aktivität 1 aus und wird nachher durch das exklusive Oder geteilt. Je nachdem ob Bedingung 1 oder Bedingung 2 erfüllt ist, wird der Ablauf bei Aktivität 2 oder Aktivität 3 fortgesetzt. Damit beide Aktivitäten am Event Ende abschließen können, werden die Pfade nach den beiden möglichen Aktivitäten wieder über ein exklusives Oder zusammengeführt. Grundsätzlich werden Pfade immer mit demselben Gateway-Typ zusammengeführt, mit dem sie auch geteilt wurden.



**Abbildung 21: BPMN Exclusives Gateway**

Bei einem inklusiven Oder-Gateway werden mögliche Pfade vorgegeben, von denen einer oder mehrere eingeschlagen werden können. In Abbildung 60 ist die mögliche Verwendung von Gateways mit inklusivem Oder dargestellt. Hier löst das Event Start ebenfalls die Aktivität 1 aus und wird nachher durch das inklusive Oder geteilt. Je nachdem ob Bedingung 1 oder Bedingung 2,

oder Aktivität 2, oder Aktivität 3, oder beide Aktivitäten, oder alle drei Aktivitäten, oder keine der Aktivitäten.

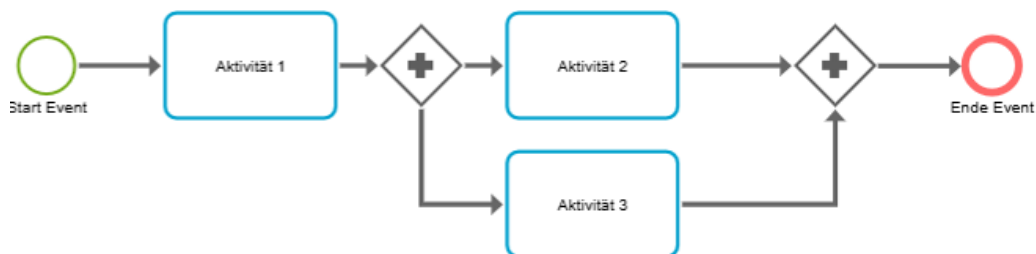


**Abbildung 22: BPMN - Inklusives Gateway**

Damit es bei der Aufteilung der Pfade nicht zum Stillstand des Gesamtablaufes kommen kann wird empfohlen, bei Oder-Bedingungen, einen zusätzlichen Standardfluss ohne Bedingung zu

modellieren welcher ausgeführt wird, wenn keine der übrigen Bedingungen zutrifft (ähnlich dem Default-Zweig eines Select-Case Statements).

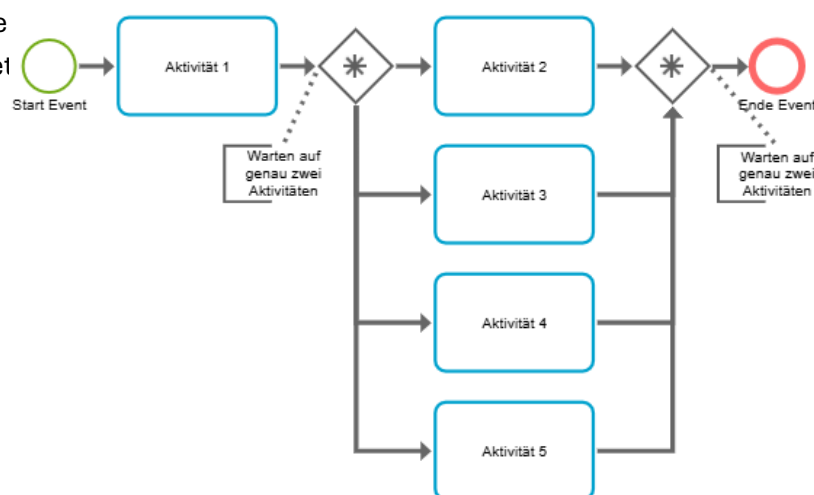
Bei einem Parallel-Gateway werden mögliche Pfade vorgegeben, welche alle gleichzeitig, ohne zusätzliche Bedingungen, eingeschlagen werden. In Abbildung 23 ist die mögliche Verwendung von Parallel Gateways dargestellt. Hier löst das Event Start wieder die Aktivität 1 aus und wird nachher durch das Parallel Gateway geteilt. Der Ablauf wird daher parallel bei Aktivität 2 und Aktivität 3 fortgesetzt. Damit der Prozess-Fluss, nach der Ausführung beider Aktivitäten, am Event Ende abschließen kann, werden die Pfade wieder über ein Parallel Gateway zusammengeführt. Das bedeutet, dass das Gateway abwartet, bis beide seiner Eingangspfade, nach dem Abschluss aller parallel ausgeführten Aktivitäten, aktiv sind und erst dann die Ausführung mit der Auslösung des Events Ende fortsetzt.



**Abbildung 23: BPMN - Paralleles Gateway**

Ein komplexes Gateway kann zur Kombination mehrerer verketteter Gateway-Arten in einem einzelnen Gateway genutzt werden. Die Aktivierung der einzelnen Pfade ist dabei von einer rein verbalen Bedingung abhängig, welche beliebig komplex sein kann (z.B.: zwei Aktivitäten aus fünf möglichen müssen ausgeführt werden). Genauso verhält es sich auch beim Zusammenführen paralleler Pfade. In der folgenden Abbildung ist die mögliche Verwendung von komplexen Gateways dargestellt. Hier löst das Event Start wieder die Aktivität 1 aus und wird nachher durch das Gateway geteilt, wobei **zwei aus fünf** möglichen Aktivitäten ausgeführt werden müssen. Nach dem Abschluss der

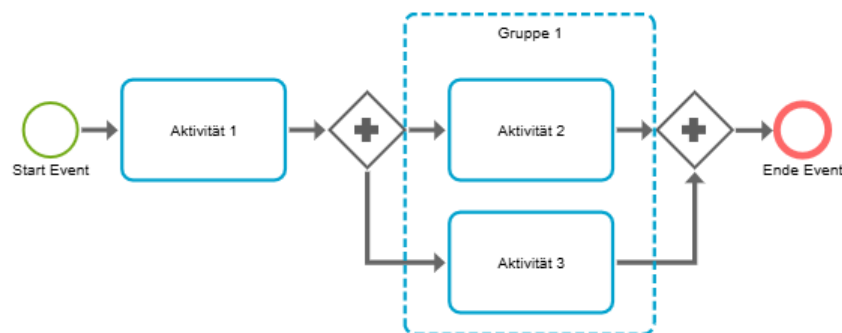
Events Ende fortsetzt die Ausführung des Events Ende fortsetzt.



**Abbildung 24: BPMN - komplexes Gateway mit zwei aus fünf möglichen Aktivitäten**

Wie im vorherigen Beispiel der komplexen Gateways bereits eingesetzt, gibt es die Möglichkeit bestimmte Freitext- Anmerkungen zu einzelnen Elementen zu machen, indem eine eckige Klammer mit einer gestrichelten Linie mit dem jeweiligen Element verbunden wird. Hinter die Klammer wird dann die Freitext-Anmerkung geschrieben. Diese Anmerkungen gehören zur Gruppe der Artefakte.

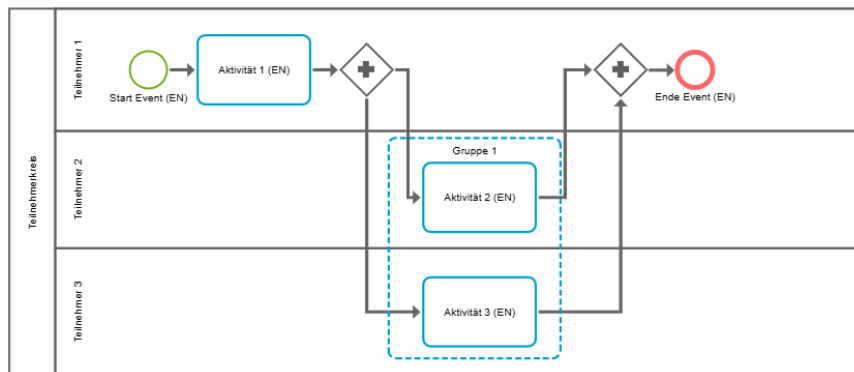
Ein weiteres Artefakt ist die Gruppierung. Dabei können mehrere Teile des Diagramms durch ein gepunktet-gestricheltes Rechteck eingefasst werden, um ihre logische Zusammengehörigkeit zu verdeutlichen. Dies hat jedoch keinerlei Auswirkung auf die Funktionalitäten des Prozesses. In das System zusätzlich eingebrachte eigene Symbole gelten auch als Artefakte und haben ebenfalls keine Auswirkung auf den Ablauf. Ein Beispiel für eine Gruppierung ist in Abbildung 45: BPMN - Gruppierung dargestellt. Hier sind die Elemente Aktivität 2 und Aktivität 3 zu einer Gruppe mit der Bezeichnung Gruppe 1 zusammengefasst. Auf den Ablauf des Modells hat dies jedoch keinerlei Auswirkung.



**Abbildung 25: BPMN - Gruppierung**

Die Teilnehmer eines Prozesses werden über Pools mit darin optional enthaltenen Lanes dargestellt. Dabei ist zu beachten, dass es sich bei einem Teilnehmer nicht immer um eine Person, sondern auch um eine Personengruppe, Rolle, Organisationseinheit oder auch um ein bestimmtes System handeln kann. Innerhalb eines Pools werden all die Prozesssteile dargestellt, die von seinen Teilnehmern ausgeführt werden. Innerhalb eines Pools können weitere Unterteilnehmer definiert werden, welche den definierten Teilnehmerkreis weiter unterteilen. Diese Unterteilungen werden als Lanes bezeichnet und können wiederum weiter ineinander verschachtelt werden. Wichtig ist dabei, dass diese Unterteilnehmer innerhalb des übergeordneten Teilnehmerkreises an demselben Prozess beteiligt sind. Eine Aktivität wird dabei immer in der Lane des Teilnehmers eingezeichnet, der sie durchführt. Soll eine Aktivität von mehreren Teilnehmern durchgeführt werden, muss sie in jeder Lane separat eingezeichnet werden. Eine Aktivität darf nicht über mehrere Lanes hinweg gezeichnet werden. Der Grund dieser Regel liegt in der Eindeutigkeit der Aktivitätsdurchführung und Zuständigkeit. Wird beispielsweise eine Aktivität entweder von Teilnehmer 1 oder Teilnehmer 2 ausgeführt, wird sie vorab über ein exklusives Oder-Gateway in zwei idente Aufgaben in beiden Lanes aufgeteilt und anschließend wieder zusammengeführt. Damit ist sichergestellt, dass für ihre Ausführung genau einer der beiden Teilnehmer erforderlich ist. In Abbildung 64 ist ein einfaches Prozessdiagramm dargestellt, welches das Zusammenspiel mehrerer Lanes in einem Pool zeigt. Hier führt nach dem Start des Prozesses Teilnehmer 1 die Aktivität 1 aus. Danach werden Aktivität 2 von Teilnehmer 2 und

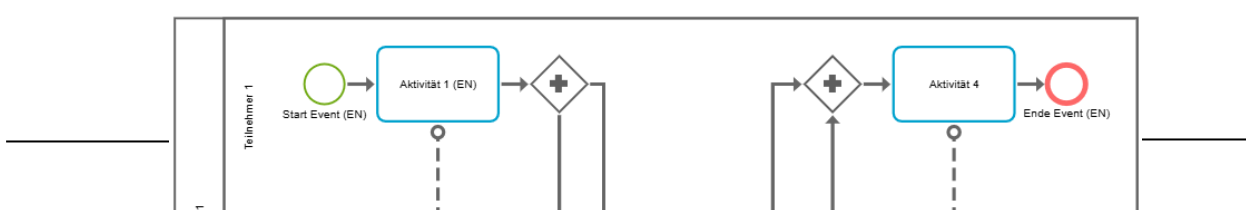
Aktivität 3 von Teilnehmer 3 gestartet. Sobald beide Aktivitäten abgeschlossen sind, wird noch Aktivität 4 von Teilnehmer 1 ausgeführt und der Prozessfluss danach abgeschlossen. Es wurde in diesem Beispiel weiter davon ausgegangen, dass Aktivität 2 und Aktivität 3 thematisch zusammengehören und daher eine Gruppierung eingezeichnet. Dies hat jedoch keinen Einfluss auf den Ablauf und dient einzig der Übersichtlichkeit.



**Abbildung 26: BPMN - Pool mit Lanes**

Gibt es in einem Prozessdiagramm **mehrere Pools**, stellen diese getrennte Teilnehmerkreise mit getrennten Prozessabläufen dar. Es handelt sich dabei um kollaborative Prozesse. Hierbei ist zu beachten, dass die Prozesse nicht direkt über Gateways oder Aktivitäten miteinander verknüpft werden dürfen, da sie in getrennten Systemen stattfinden. Zur Interaktion mehrerer Pools werden **Nachrichtenflüsse** verwendet. Diese werden als gerichtete gestrichelte Linie dargestellt. Dabei muss beachtet werden, dass ein Gateway keine direkten Aufrufe von Aktivitäten in einem anderen Pool durchführen darf und auch nicht mit Gateways des anderen Pools verknüpft werden kann. Befindet sich ein Nachrichtenfluss direkt zwischen zwei Aktivitäten, handelt es sich um eine Abstimmung der Aktivitäten mittels Nachrichten. Zwischen einer Aktivität und einem Ereignis wird ein Ereignis ausgelöst oder empfangen. Zusätzlich ist es möglich, nicht mit einem Prozessteil eines Pools, sondern mit seinen Teilnehmern zu interagieren. Dazu wird ein Nachrichtenfluss direkt zur Außenkante des gewünschten Pools gezeichnet. Damit ist es auch möglich, Nachrichten an und von Teilnehmern darzustellen, deren Prozess inhaltlich nicht modelliert wird.

Die unterschiedlichen Formen von Nachrichtenflüssen sind in Abbildung 47: Mehrere Pools und Nachrichtenflüsse dargestellt. Hier gibt es zwei verschiedene Prozessabläufe zweier unterschiedlicher Teilnehmerkreise und damit unterschiedlicher Pools. Der *Teilnehmerkreis 1* teilt sich selbst wiederum in zwei unterschiedliche Teilnehmer, welche mittels Lanes in den Pool integriert sind. Der erste Nachrichtenfluss findet zwischen *Aktivität 1* und dem Startevent im Pool von *Teilnehmerkreis 2* statt. Dabei wird durch die Ausführung von *Aktivität 1* eine Nachricht ausgelöst, welche gleichzeitig auch der Start-Event von Teilnehmerkreis 2 ist. Der nächste Nachrichtenfluss geschieht zwischen *Aktivität B* und *Aktivität 3*. Hier benötigt *Aktivität 3* Informationen aus *Aktivität B*. Der letzte Nachrichtenfluss geschieht durch Ausführung von *Aktivität 4*. Diese benachrichtigt jedoch kein Element in *Teilnehmerkreis 2*, sondern den *Teilnehmerkreis*

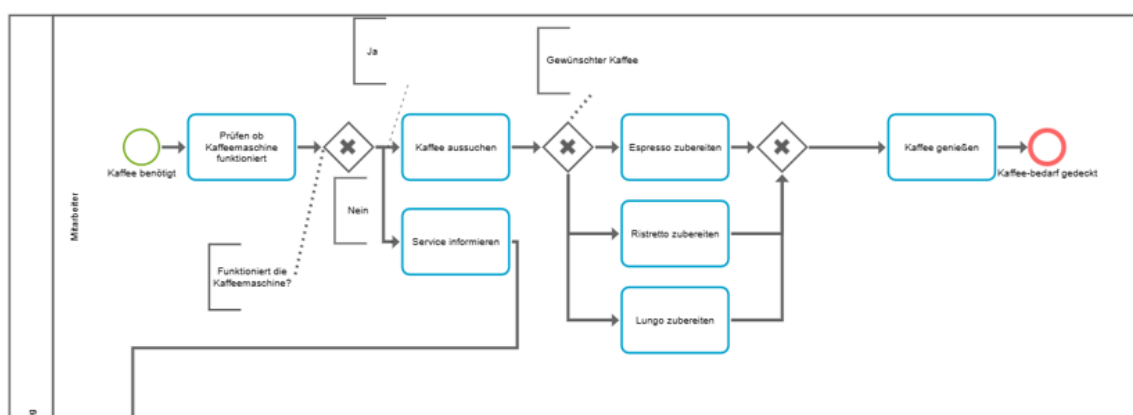


**Abbildung 27: Mehrere Pools und Nachrichtenflüsse**

### 3.2.2 Beispiel eines BPMN

In diesem Beispiel soll der Prozess einer Kaffeezubereitung in einer Büroküche modelliert werden. Die Teilnehmer des Prozesses sind alle Mitarbeiter der Abteilung. Diese untergliedern sich in die Mitarbeiter, die Kaffee trinken wollen, und den Mitarbeiter, der ebenfalls für die Wartung der Kaffeemaschinezuständig ist.

Sobald ein Mitarbeiter einen Kaffee möchte, prüft er zuerst ob die Kaffeemaschine korrekt funktioniert. Ist dies der Fall, sucht er sich einen Kaffee (Espresso, Ristretto oder Lungo) aus und bereitet diesen zu. Nach der Zubereitung trinkt er den Kaffee und hat seinen Kaffeebedarf damit gestillt. Sollte die Kaffeemaschine nicht funktionieren informiert er den Service-Verantwortlichen der Abteilung. Dieser überprüft wiederum die Kaffeemaschine hinsichtlich ihrer möglichen Probleme. Ist sie verkalkt, entkalkt er sie, fehlen Bohnen, füllt er Bohnen nach und wenn sie defekt ist, beauftragt er den Hersteller-Service mit einer Reparatur. Alle Probleme können auch gleichzeitig vorhanden sein, daher ist es auch möglich, dass er mehrere der Servicetätigkeiten parallel vornimmt. Im Falle eines Gerätedefektes wird der Service-Verantwortliche wieder vom Hersteller-Service informiert, sobald die Kaffeemaschine repariert ist. Sind alle Servicearbeiten beendet, und die Maschine damit wieder einsatzbereit, informiert der Service-Verantwortliche Kollege die Mitarbeiter der Abteilung. Stellt er hingegen fest, dass überhaupt kein Problem besteht, informiert er direkt die Mitarbeiter der Abteilung, ohne Servicetätigkeiten durchzuführen



### 3.3 DMN – Decision Model and Notation

DMN steht für **Decision Model and Notation™** und dient dazu, geschäftliche Entscheidungen und die dazugehörigen Regeln präzise und nachvollziehbar zu modellieren (OMG - Object Management Group, 2026).

#### Entscheidungen mit DMN modellieren

Der Einsatz von DMN unterstützt Organisationen dabei, komplexe Entscheidungsbereiche verständlich darzustellen. Mithilfe gut lesbarer Diagramme können alle Beteiligten besser nachvollziehen, welche Entscheidungen getroffen werden müssen, welche Informationen dafür notwendig sind und welche Regeln dabei gelten. Dadurch entsteht eine gemeinsame Grundlage für Diskussionen und Abstimmungen über Umfang und Art betrieblicher Entscheidungen (OMG - Object Management Group, 2026).

DMN kann außerdem dazu beitragen, Aufwand und Risiko bei der Automatisierung von Entscheidungen zu verringern. Anforderungen werden grafisch zerlegt und strukturiert dargestellt. Geschäftsregeln können klar und eindeutig in Entscheidungstabellen beschrieben werden. Solche Tabellen sind leicht verständlich und ermöglichen eine verlässliche Definition der Entscheidungslogik. Gleichzeitig können DMN-Modelle als Grundlage für Systeme dienen, die Entscheidungen automatisch prüfen, ausführen oder weiterverarbeiten (OMG - Object Management Group, 2026).

Ein weiterer Vorteil von DMN besteht darin, dass Entscheidungsmodelle in einen geschäftlichen Zusammenhang eingebettet werden können. Entscheidungen lassen sich mit Organisationseinheiten, Kennzahlen und Unternehmenszielen verbinden. Dadurch wird sichtbar, welche Rolle eine Entscheidung im Gesamtzusammenhang eines Geschäftsprozesses oder eines Falls spielt (OMG - Object Management Group, 2026).

DMN ist so gestaltet, dass es gemeinsam mit BPMN und CMMN verwendet werden kann. Während BPMN Geschäftsprozesse beschreibt und CMMN fallbezogene Abläufe unterstützt, ergänzt DMN diese Modelle um die Entscheidungslogik. So kann zum Beispiel eine Aufgabe in einem BPMN-Prozess mit einem DMN-Entscheidungsmodell verknüpft werden. Das Entscheidungsmodell legt dann fest, welche Entscheidung im Rahmen dieser Aufgabe zu treffen ist und welche Regeln dafür angewendet werden (OMG - Object Management Group, 2026).

Ein zentrales Element von DMN ist das **Decision Requirements Diagram**, kurz **DRD**. Ein DRD zeigt, welche Entscheidungen in einem bestimmten Geschäftsbereich getroffen werden müssen und wie diese voneinander abhängen. Außerdem stellt es dar, welche Eingangsdaten benötigt werden und welches fachliche Wissen in die Entscheidung einfließt. Ein DRD kann als übersichtliche Analyse eines Entscheidungsbereichs verwendet werden oder als Grundlage für einen ausführbaren Entscheidungsservice dienen (OMG - Object Management Group, 2026).

DMN unterscheidet grundsätzlich drei Ebenen der Modellierung. Die erste Ebene beschreibt die **Entscheidungsanforderungen**. Dabei wird grafisch dargestellt, welche Entscheidungen erforderlich sind, welche Abhängigkeiten bestehen und welche Informationen benötigt werden. Die zweite Ebene umfasst sogenannte **Boxed Expressions**. Damit können Bestandteile der

Entscheidungslogik grafisch und strukturiert dargestellt werden. Besonders wichtig sind dabei Entscheidungstabellen, mit denen Geschäftsregeln übersichtlich und eindeutig formuliert werden können. Die dritte Ebene ist die eigentliche **Entscheidungslogik**. Dafür verwendet DMN die Ausdruckssprache **FEEL**. Mit FEEL können Bedingungen, Berechnungen, einfache Datenstrukturen und externe Logik formal beschrieben werden (OMG - Object Management Group, 2026).

Wenn ein DMN-Modell vollständig spezifiziert ist, kann es nicht nur zur Dokumentation und Analyse dienen, sondern auch automatisiert ausgeführt werden. Damit verbindet DMN fachliche Verständlichkeit mit technischer Umsetzbarkeit (OMG - Object Management Group, 2026).

### 3.3.1 Schritt-für-Schritt-Anleitung:

#### Von der Entscheidungstabelle zum DMN-Modell

DMN, kurz für Decision Model and Notation, wird verwendet, um fachliche Entscheidungen und die dazugehörigen Regeln übersichtlich zu modellieren. Dabei geht es nicht darum, einen gesamten Geschäftsprozess darzustellen, sondern konkrete Entscheidungen nachvollziehbar zu beschreiben.

Ein DMN-Modell beantwortet zum Beispiel Fragen wie:

- Erhält ein Kunde einen Rabatt?
- Wie hoch ist der Rabatt?
- Wird ein Antrag genehmigt?
- Welche Risikoklasse wird einer Person zugeordnet?

Im folgenden Beispiel soll entschieden werden, **wie hoch der Rabatt für eine Kundenbestellung ist**.

#### 1. Ausgangssituation

Ein Unternehmen möchte abhängig vom Kundentyp und vom Bestellwert automatisch entscheiden, welchen Rabatt ein Kunde erhält.

Die fachlichen Regeln lauten:

Bestandskunden erhalten bei einem Bestellwert ab 500 Euro einen Rabatt von 10 %. Bei einem Bestellwert unter 500 Euro erhalten sie 5 %. Neukunden erhalten ab einem Bestellwert von 500 Euro einen Rabatt von 5 %. Liegt der Bestellwert darunter, erhalten sie keinen Rabatt.

Die Entscheidung lautet daher: Wie hoch ist der Kundenrabatt?

#### 2. Eingabedaten bestimmen

Zuerst wird festgelegt, welche Informationen für die Entscheidung benötigt werden. Diese Informationen nennt man in **DMN Input Data**, also Eingabedaten.

Für unser Beispiel werden zwei Eingabedaten benötigt:

| Eingabedatum       | Beschreibung                                                  | Mögliche Werte             |
|--------------------|---------------------------------------------------------------|----------------------------|
| <b>Kundentyp</b>   | Gibt an, ob es sich um einen Neu- oder Bestandskunden handelt | Neukunde, Bestandskunde    |
| <b>Bestellwert</b> | Gibt den Wert der Bestellung in Euro an                       | Zahl, z. B. 250, 500, 1200 |

**Tabelle 5: Beispiel - Eingabedaten**

### 3. Ausgabe der Entscheidung festlegen

Anschließend wird definiert, welches Ergebnis die Entscheidung liefern soll. In unserem Beispiel ist das Ergebnis der Entscheidung die Höhe des Rabatts.

| Entscheidung                  | Ergebnis          |
|-------------------------------|-------------------|
| <b>Kundenrabatt bestimmen</b> | Rabatt in Prozent |

Mögliche Ergebnisse sind:

Rabatt → 0%,5%,10%

### 4. Regeln in eine Entscheidungstabelle überführen

Die fachlichen Regeln werden nun in einer Entscheidungstabelle dargestellt. Jede Zeile beschreibt eine Regel. Die Spalten auf der linken Seite enthalten die Bedingungen, die Spalte auf der rechten Seite enthält das Ergebnis.

#### Entscheidungstabelle: Kundenrabatt bestimmen

|                                                    |                               |                  |                    |                          |
|----------------------------------------------------|-------------------------------|------------------|--------------------|--------------------------|
| Bezeichnung                                        | <b>Kundenrabatt bestimmen</b> |                  |                    |                          |
|                                                    | <b>Regel</b>                  | <b>Kundentyp</b> | <b>Bestellwert</b> | <b>Rabatt</b>            |
| Bedingungen<br>Input Entry                         | 1                             | Bestandskunde    | >= 500             | 10 %                     |
|                                                    | 2                             | Bestandskunde    | < 500              | 5 %                      |
|                                                    | 3                             | Neukunde         | >= 500             | 5 %                      |
|                                                    | 4                             | Neukunde         | < 500              | 0 %                      |
| <b>Tabelle 6:DMN-Entscheidungstabelle – Aufbau</b> |                               |                  |                    |                          |
|                                                    |                               |                  |                    | Ergebnis<br>Output Entry |

Diese Tabelle ist bereits eine einfache Form der Entscheidungslogik. Sie zeigt uns, unter welchen Bedingungen welches Ergebnis entsteht.

Wichtig ist, dass die Regeln vollständig und eindeutig sind. Für jede mögliche Kombination aus Kundentyp und Bestellwert muss klar sein, welcher Rabatt gilt. Außerdem darf es keine widersprüchlichen Regeln geben.

### 5. Hit Policy festlegen

In DMN wird bei Entscheidungstabellen eine sogenannte **Hit Policy** verwendet. Sie gibt an, wie viele Regeln zutreffen dürfen und wie das Ergebnis bestimmt wird.

In unserem Beispiel trifft für jede Bestellung genau eine Regel zu. Ein Kunde ist entweder Neukunde oder Bestandskunde, und der Bestellwert ist entweder kleiner als 500 Euro oder mindestens 500 Euro.

Daher verwenden wir die Hit Policy:  $U \rightarrow \text{Unique}$  = Genau eine Regel darf zutreffen.

Die Entscheidungstabelle kann daher in DMN so dargestellt werden:

| Kundenrabatt bestimmen |               |             |        |
|------------------------|---------------|-------------|--------|
| <b>U</b>               | Kundentyp     | Bestellwert | Rabatt |
| 1                      | Bestandskunde | $\geq 500$  | 10%    |
| 2                      | Bestandskunde | $< 500$     | 5%     |
| 3                      | Neukunde      | $\geq 500$  | 5%     |
| 4                      | Neukunde      | $< 500$     | 0%     |

**Tabelle 7: DMN-Entscheidungstabelle**

Hit Policy

Mehrere Bedingungen werden in DMN standardmäßig mit „UND“ verknüpft!  
Wenn „Neukunde UND Bestellwert  $\geq 500$ “ dann 5% Rabatt

Klare Trennung zwischen Input- und Output-Entry mittels Doppelstrich (DMN Standard)

Das „U“ links oben steht für **Unique**. Es zeigt an, dass pro Entscheidung genau eine Regel angewendet werden soll.

### Hinweis:

Es ist auch möglich, mehrere Bedingungen mittels **ODER** zu verknüpfen. In der folgenden Tabelle erhalten sowohl Neukunden als auch Bestandskunden bei einem Bestellwert ab 500 einen Rabatt von 5%.

Mehrere Bedingungen können auch mittels „**ODER**“ verknüpft werden  
Wenn „Neukunde ODER Bestandskunde UND Bestellwert  $\geq 500$ “ dann 5% Rabatt

| Kundenrabatt bestimmen |                         |             |        |
|------------------------|-------------------------|-------------|--------|
| <b>U</b>               | Kundentyp               | Bestellwert | Rabatt |
| 1                      | Bestandskunde           | $\geq 500$  | 10%    |
| 2                      | Bestandskunde           | $< 500$     | 5%     |
| 3                      | Neukunde, Bestandskunde | $\geq 500$  | 5%     |

**Tabelle 8: DMN Entscheidungstabelle mit ODER Verknüpfung**

Der Vorteil an der Entscheidungstabelle ist, dass man sehr gut erkennen kann, auf Basis welcher Daten welche Entscheidungen getroffen werden und die entsprechenden Regeln dazu.

### Mögliche Hit Policies in DMN:

Im vorangegangenen Beispiel haben wir mit der Hit Policy „U“ (Unique) gearbeitet. Neben dieser Policy gibt es eine Reihe weiterer nützlicher Policies. Grundsätzlich legt eine **Hit Policy** in einer DMN-Entscheidungstabelle fest, wie mit zutreffenden Regeln umgegangen wird. Sie bestimmt also, ob genau eine Regel, mehrere Regeln oder eine bestimmte Kombination von Regeln als Ergebnis verwendet werden darf.

Die folgende Tabelle gibt einen Überblick über die wichtigsten Hit Policies:

| Hit Policy   | Bezeichnung   | Bedeutung                                                                                              | Typischer Einsatz                                                                |
|--------------|---------------|--------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|
| <b>U</b>     | Unique        | Genau eine Regel darf zutreffen.                                                                       | Wenn sich Regeln gegenseitig eindeutig ausschließen.                             |
| <b>A</b>     | Any           | Mehrere Regeln dürfen zutreffen, müssen aber zum gleichen Ergebnis führen.                             | Wenn Überschneidungen erlaubt sind, aber kein Widerspruch entstehen darf.        |
| <b>F</b>     | First         | Mehrere Regeln dürfen zutreffen, aber die erste passende Regel wird verwendet.                         | Wenn Regeln bewusst in einer Prioritätsreihenfolge angeordnet sind.              |
| <b>R</b>     | Rule Order    | Alle zutreffenden Regeln werden in der Reihenfolge der Tabelle ausgegeben.                             | Wenn mehrere Ergebnisse relevant sind und die Reihenfolge wichtig ist.           |
| <b>C</b>     | Collect       | Alle zutreffenden Ergebnisse werden gesammelt.                                                         | Wenn mehrere Ergebnisse gleichzeitig zurückgegeben werden sollen.                |
| <b>C+</b>    | Collect Sum   | Alle zutreffenden numerischen Ergebnisse werden addiert.                                               | Wenn Zuschläge, Punkte oder Beträge summiert werden.                             |
| <b>C&lt;</b> | Collect Min   | Aus allen zutreffenden Ergebnissen wird der kleinste Wert genommen.                                    | Wenn der niedrigste Wert relevant ist.                                           |
| <b>C&gt;</b> | Collect Max   | Aus allen zutreffenden Ergebnissen wird der größte Wert genommen.                                      | Wenn der höchste Wert relevant ist.                                              |
| <b>C#</b>    | Collect Count | Die Anzahl der zutreffenden Regeln wird gezählt.                                                       | Wenn gezählt werden soll, wie viele Bedingungen erfüllt sind.                    |
| <b>P</b>     | Priority      | Mehrere Regeln dürfen zutreffen, aber das Ergebnis mit der höchsten fachlichen Priorität wird gewählt. | Wenn Ergebnisse eine festgelegte Rangordnung haben.                              |
| <b>O</b>     | Output Order  | Alle zutreffenden Ergebnisse werden nach ihrer fachlichen Priorität sortiert ausgegeben.               | Wenn mehrere Ergebnisse relevant sind und nach Priorität geordnet werden sollen. |

Die Wahl der Hit Policy hängt davon ab, wie die Entscheidung fachlich funktionieren soll. Für einfache Entscheidungstabellen wird häufig **Unique** verwendet, weil genau eine Regel zutreffen soll. Wenn mehrere Regeln gleichzeitig zutreffen dürfen, kommen Hit Policies wie **Collect**, **Rule Order**, **Priority** oder **Output Order** zum Einsatz.

Die Hit Policy beantwortet die Frage, was passiert, wenn keine, eine oder mehrere Regeln einer Entscheidungstabelle zutreffen?

Deshalb sollte die Hit Policy immer bewusst gewählt und fachlich begründet werden.

## 6. Von der Entscheidungstabelle zum DMN-Modell

Die Entscheidungstabelle beschreibt die Logik der Entscheidung. In einem DMN-Modell wird zusätzlich grafisch dargestellt, welche Eingabedaten für welche Entscheidung benötigt werden. Insbesondere sehr komplexe Entscheidungen lassen sich nicht allzuleicht in einer Entscheidungstabelle darstellen. Dazu wird das **Decision Requirements Diagram**, kurz **DRD**, erstellt.

Ein DRD besteht im Wesentlichen aus folgenden Objekten:

| Darstellung                                                                         | Objekt                          | Bedeutung                                                           |
|-------------------------------------------------------------------------------------|---------------------------------|---------------------------------------------------------------------|
|    | <b>Decision</b>                 | Eine Entscheidung, die ein Ergebnis liefert                         |
|   | <b>Input Data</b>               | Eingabedaten, die für eine Entscheidung benötigt werden             |
|  | <b>Business Knowledge Model</b> | Wiederverwendbares Fachwissen oder eine Berechnungslogik            |
|  | <b>Knowledge Source</b>         | Quelle des fachlichen Wissens, z. B. Richtlinie, Gesetz, Verordnung |

Tabelle 9: DRD - Objekte

In unserem Beispiel gibt es:

| DMN-Element       | Inhalt im Beispiel     |
|-------------------|------------------------|
| <b>Input Data</b> | Kundentyp              |
| <b>Input Data</b> | Bestellwert            |
| <b>Decision</b>   | Kundenrabatt bestimmen |

Tabelle 10: DMN Elemente für Beispiel

Die Eingabedaten werden mit der Entscheidung verbunden. Dadurch wird sichtbar, dass die Entscheidung „Kundenrabatt bestimmen“ auf den Informationen „Kundentyp“ und „Bestellwert“ basiert.

## 7. Grafische Darstellung als DMN-Modell

Das einfache DMN-Modell sieht folgendermaßen aus:



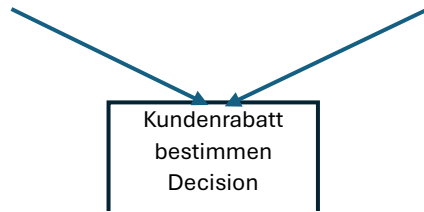


Abbildung 29: Grafische Darstellung als DMN Modell

Die beiden oberen Elemente sind Eingabedaten. Sie liefern die notwendigen Informationen für die Entscheidung. Die Entscheidung selbst wird darunter dargestellt.

Die Pfeile zeigen, dass die Eingabedaten in die Entscheidung einfließen.

### 8. Entscheidung im DMN-Modell mit Logik verbinden

Das grafische DMN-Modell zeigt, **welche Informationen benötigt werden** und **welche Entscheidung getroffen wird**. Die konkrete Logik der Entscheidung wird in der Entscheidungstabelle hinterlegt.

Man kann sich das Zusammenspiel daher so vorstellen:

#### DMN-Diagramm:

Welche Eingabedaten werden für welche Entscheidung benötigt?

#### Entscheidungstabelle:

Welche Regeln führen zu welchem Ergebnis?

Für unser Beispiel bedeutet das:



Die Entscheidung „Kundenrabatt bestimmen“ enthält intern die, bereits bekannte, Entscheidungstabelle:

| Kundenrabatt bestimmen |               |             |        |
|------------------------|---------------|-------------|--------|
| U                      | Kundentyp     | Bestellwert | Rabatt |
| 1                      | Bestandskunde | >= 500      | 10%    |

|   |               |        |    |
|---|---------------|--------|----|
| 2 | Bestandskunde | < 500  | 5% |
| 3 | Neukunde      | >= 500 | 5% |
| 4 | Neukunde      | < 500  | 0% |

**Tabelle 11: Kundenrabatt bestimmen - interne Entscheidungstabelle**

### 3.3.2 Erweiterung des Beispiels: Zerlegung in mehrere Entscheidungen

Bei einfachen Beispielen reicht oft eine einzige Entscheidung aus. In der Praxis werden Entscheidungen jedoch häufig in mehrere Teilschritte zerlegt.

Angenommen, das Unternehmen möchte zuerst den Kunden bewerten und danach den Rabatt bestimmen. Dann könnten zwei Entscheidungen modelliert werden:

1. **Kundenkategorie bestimmen**
2. **Kundenrabatt bestimmen**

Die Kundenkategorie könnte zum Beispiel vom Kundentyp und von der bisherigen Anzahl an Bestellungen abhängen.

#### Neue Eingabedaten

| Eingabedatum                   | Beschreibung                               |
|--------------------------------|--------------------------------------------|
| Kundentyp                      | Neukunde oder Bestandskunde                |
| Anzahl bisheriger Bestellungen | Anzahl der bereits getätigten Bestellungen |
| Bestellwert                    | Wert der aktuellen Bestellung              |

**Tabelle 12: Eingabedaten zur Erweiterung des Beispiels**

#### Entscheidungstabelle: Kundenkategorie bestimmen

| Kundenkategorie bestimmen |               |                                |                 |
|---------------------------|---------------|--------------------------------|-----------------|
| U                         | Kundentyp     | Anzahl bisheriger Bestellungen | Kundenkategorie |
| 1                         | Neukunde      | 0                              | Neu             |
| 2                         | Bestandskunde | < 5                            | Standard        |
| 3                         | Bestandskunde | >= 5                           | Stammkunde      |

**Tabelle 13: Kundenkategorie bestimmen**

#### Entscheidungstabelle: Kundenrabatt bestimmen

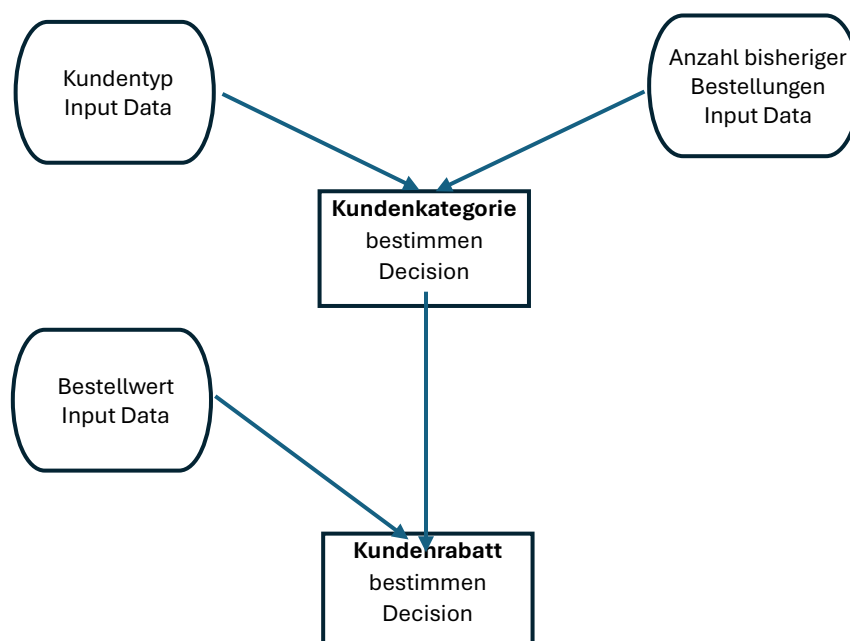
| Kundenrabatt bestimmen |                 |             |        |
|------------------------|-----------------|-------------|--------|
| U                      | Kundenkategorie | Bestellwert | Rabatt |
| 1                      | Neu             | < 500       | 0      |

|   |            |            |    |
|---|------------|------------|----|
| 2 | Neu        | $\geq 500$ | 5  |
| 3 | Standard   | $< 500$    | 5  |
| 4 | Standard   | $\geq 500$ | 10 |
| 5 | Stammkunde | $< 500$    | 10 |
| 6 | Stammkunde | $\geq 500$ | 15 |

**Tabelle 14: Kundenrabatt bestimmen**

### Grafisches DMN-Modell mit mehreren Entscheidungen

Das erweiterte DMN-Modell zeigt nun, dass eine Entscheidung das Ergebnis einer anderen Entscheidung verwendet.



**Abbildung 30: Grafische Darstellung mit mehreren Entscheidungen – DMN Modell**

Wie Sie sehen können, benötigt die Entscheidung **Kundenkategorie bestimmen** die Eingabedaten „Kundentyp“ und „Anzahl bisheriger Bestellungen“. Das Ergebnis dieser Entscheidung wird anschließend für die Entscheidung **Kundenrabatt bestimmen** verwendet. Zusätzlich benötigt diese zweite Entscheidung den „Bestellwert“.

### 3.3.3 Interpretation des DMN-Modells

Ein DMN-Modell beantwortet mehrere Fragen gleichzeitig:

| Frage                                | Antwort im Beispiel                                    |
|--------------------------------------|--------------------------------------------------------|
| Welche Entscheidung wird getroffen?  | Kundenrabatt bestimmen                                 |
| Welche Eingabedaten werden benötigt? | Kundentyp, Anzahl bisheriger Bestellungen, Bestellwert |
| Gibt es Zwischenentscheidungen?      | Ja, Kundenkategorie bestimmen                          |
| Welche Regeln gelten?                | Regeln in den Entscheidungstabellen                    |
| Welches Ergebnis wird erzeugt?       | Rabatt in Prozent                                      |

**Tabelle 15: Welche Fragen werden durch das DMN beantwortet?**

### 3.3.4 Typische Fehler bei DMN-Modellen

Beim Erstellen von DMN-Modellen treten häufig folgende Fehler auf:

| Fehler                                   | Erklärung                                                                                         |
|------------------------------------------|---------------------------------------------------------------------------------------------------|
| Unklare Entscheidungsnamen               | Die Entscheidung ist zu allgemein formuliert, z. B. „Kunde prüfen“ statt „Kundenrabatt bestimmen“ |
| Fehlende Eingabedaten                    | Eine Entscheidung benötigt Informationen, die im Modell nicht dargestellt sind                    |
| Überschneidende Regeln                   | Zwei Regeln können gleichzeitig zutreffen, obwohl nur eine Regel erlaubt ist                      |
| Unvollständige Regeln                    | Für bestimmte Fälle gibt es kein Ergebnis                                                         |
| Vermischung von Prozess und Entscheidung | Prozessschritte gehören in BPMN, Entscheidungslogik gehört in DMN                                 |

**Tabelle 16: Typische Fehler bei der Erstellung von DMN Modellen**

### 3.3.5 Abgrenzung zu BPMN

BPMN beschreibt den Ablauf eines Geschäftsprozesses. DMN beschreibt die Logik einer Entscheidung.

Ein BPMN-Prozess könnte zum Beispiel den Schritt enthalten:

Rabatt berechnen

Die genaue Entscheidung, welcher Rabatt berechnet wird, wird jedoch nicht im BPMN-Prozess modelliert, sondern in DMN.

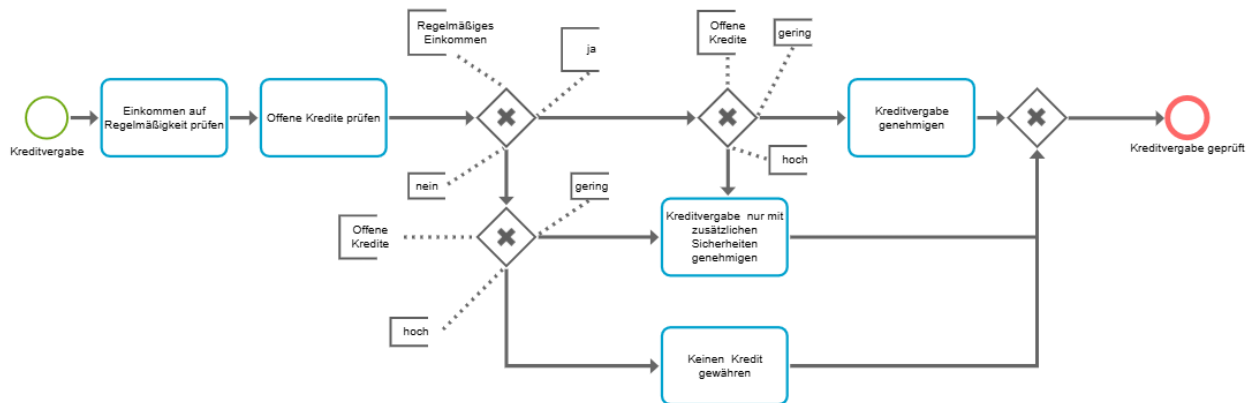
**Die Verbindung zwischen BPMN und DMN kann daher so verstanden werden:**

**BPMN** zeigt: **Wann** wird entschieden?

**DMN** zeigt: **Wie** wird entschieden?

### 3.3.6 Übung:

Erstellen Sie aus folgendem BPMN eine DMN Entscheidungstabelle:



| Kreditvergabe |                             |                |                          |
|---------------|-----------------------------|----------------|--------------------------|
| U             | Einkommen                   | Offene Kredite | Kreditvergabe            |
| 1             | Regelmäßiges Einkommen      | niedrig        | JA                       |
| 2             | Kein Regelmäßiges Einkommen | niedrig        | JA-zusätzl. Sicherheiten |
| 3             | Regelmäßiges Einkommen      | hoch           | JA-zusätzl. Sicherheiten |
| 4             | Kein Regelmäßiges Einkommen | Hoch           | NEIN                     |

## Lektion 4 ArchiMate als Modellierungssprache für Unternehmensarchitekturen

ArchiMate ist eine von The Open Group standardisierte Modellierungssprache zur Beschreibung von Unternehmensarchitekturen. Sie ermöglicht es, unterschiedliche Aspekte eines Unternehmens auf einer übergeordneten Ebene darzustellen und miteinander in Beziehung zu setzen. Ursprünglich standen dabei drei zentrale Ebenen im Mittelpunkt: die Geschäfts-Ebene, die Anwendungs-Ebene und die Technologie-Ebene. In der aktuellen Version wurde ArchiMate um weitere Ebenen ergänzt, sodass heute ein umfassender Rahmen zur Modellierung von Unternehmensarchitekturen zur Verfügung steht (SPARX Systems Europe, 2026).

1. Strategie-Ebene
2. Geschäfts-Ebene
3. Anwendungs-Ebene
4. Technologie-Ebene – inkl. physische Technologien
5. Implementierungs & Migrations Ebene

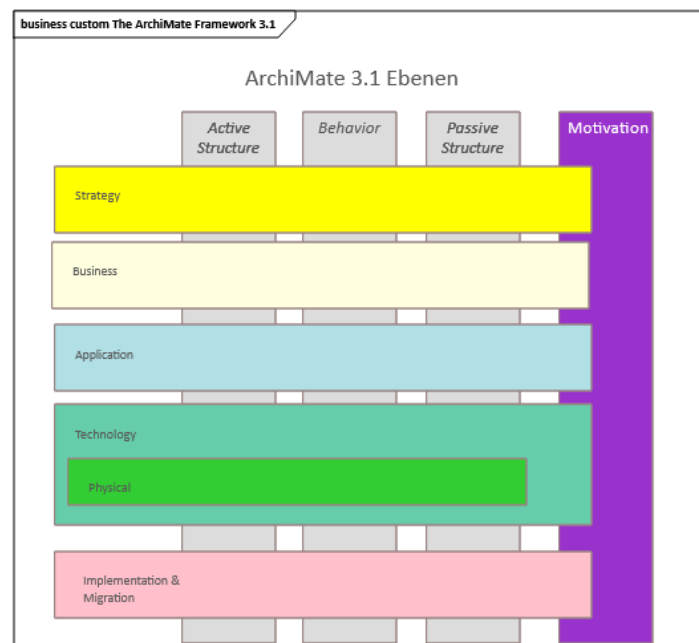


Tabelle 17: ArchiMate Ebenen (SPARX Systems Europe, 2026)

Die Sprachelemente von ArchiMate lassen sich grundsätzlich in drei Kategorien einteilen: **aktive Strukturen, Verhalten** und **passive Strukturen**. Aktive Strukturen beschreiben jene Elemente, die etwas ausführen, zum Beispiel Personen, Rollen, Organisationseinheiten oder Anwendungen. Verhalten beschreibt Tätigkeiten, Prozesse oder Funktionen. Passive Strukturen stehen für Objekte, Informationen oder Daten, auf die im Rahmen eines Verhaltens zugegriffen wird (SPARX Systems Europe, 2026).

Ein einfaches Beispiel dafür ist die Aussage: „John liest ein Buch.“ John ist dabei die aktive Struktur, das Lesen beschreibt das Verhalten, und das Buch ist die passive Struktur. Dieses

Grundmuster findet sich auf mehreren ArchiMate-Ebenen wieder. Je nach Ebene stehen dabei unterschiedliche fachliche Aspekte im Vordergrund (SPARX Systems Europe, 2026).

Ein wesentliches Merkmal moderner Modellierungswerkzeuge wie Enterprise Architect ist die Trennung zwischen Modelldaten und deren grafischer Darstellung. Die Modelldaten bilden einen zentralen Wissensbestand und können in unterschiedlichen Diagrammen wiederverwendet werden. Wird ein Modellelement geändert, kann diese Änderung automatisch in allen Diagrammen sichtbar werden, in denen das Element verwendet wird. Dadurch entsteht ein konsistenter „Single Point of Truth“ (SPARX Systems Europe, 2026).

Durch die standardisierte Sprache von ArchiMate können Modellelemente und deren Beziehungen klar interpretiert werden. In Enterprise Architect können diese Informationen auch mit Elementen anderer Modellierungssprachen verknüpft werden. So kann beispielsweise ein ArchiMate Business Process durch ein BPMN-Prozessmodell detaillierter beschrieben werden. Ebenso können ArchiMate-Datenobjekte oder Anwendungskomponenten durch UML-Klassen oder technische Komponenten weiter verfeinert werden. Damit lassen sich unterschiedliche Sichten für verschiedene Stakeholder in einem gemeinsamen Modell verbinden (SPARX Systems Europe, 2026).

ArchiMate umfasst mehrere Ebenen, die jeweils unterschiedliche Fragestellungen der Unternehmensarchitektur unterstützen (SPARX Systems Europe, 2026).

#### **Die Motivations-Ebene**

beschreibt Treiber, Ziele, Anforderungen und Interessen von Stakeholdern. Sie hilft zu verstehen, warum bestimmte Veränderungen oder Entscheidungen notwendig sind.

#### **Die Strategie-Ebene**

konzentriert sich auf strategische Fähigkeiten, Ressourcen, Prinzipien und langfristige Ausrichtungen eines Unternehmens.

#### **Die Geschäfts-Ebene**

beschreibt organisatorische Strukturen, Rollen, Geschäftsprozesse, Geschäftsobjekte und Services. Sie zeigt, wie ein Unternehmen Werte schafft, Leistungen bereitstellt und mit Kundinnen und Kunden interagiert.

#### **Die Anwendungs-Ebene**

beschreibt Anwendungen, Anwendungskomponenten, Schnittstellen und Datenobjekte. Sie zeigt, welche IT-Systeme Geschäftsprozesse unterstützen und welche Daten dabei verwendet werden.

Dadurch lassen sich zum Beispiel Fragen beantworten wie: Welche Anwendung unterstützt welchen Geschäftsprozess? Welche Services fallen aus, wenn eine Anwendung nicht verfügbar ist? Welche personenbezogenen Daten werden in welchen Systemen verarbeitet?

#### **Die Technologie-Ebene**

beschreibt die technologische Infrastruktur, auf der Anwendungen betrieben werden. Dazu zählen etwa Systemsoftware, Netzwerke, Datenbanken und technische Dienste. Sie

hilft zu verstehen, welche technischen Komponenten für den Betrieb der Anwendungen erforderlich sind und welche Auswirkungen technische Ausfälle auf Anwendungen und Geschäftsprozesse haben können.

#### Die Physische Ebene

beschreibt konkrete physische Ressourcen wie Server, Geräte, Speicher oder Netzwerkkomponenten. Während die Technologie-Ebene stärker technische Dienste und Systemsoftware abbildet, stehen hier konkrete Hardware- und Infrastrukturkomponenten im Mittelpunkt.

#### Die Umsetzungs- und Migrations-Ebene

beschreibt Veränderungen in der Unternehmensarchitektur. Dazu gehören Projekte, Arbeitspakete, Zielarchitekturen und Migrationsschritte. Sie unterstützt die Planung und Umsetzung von Veränderungen.

Diese Ebenen bilden gemeinsam einen Rahmen, mit dem Unternehmen ihre Architektur analysieren, planen und kommunizieren können. Dabei ist es nicht notwendig, immer alle Ebenen vollständig zu modellieren. Entscheidend ist, welche Fragestellungen beantwortet werden sollen und welche Stakeholder welche Informationen benötigen (SPARX Systems Europe, 2026).

ArchiMate bietet dafür eine standardisierte Sprache, ist aber kein vollständiges Vorgehensmodell. Für eine erfolgreiche Modellierung braucht es neben der Sprache auch ein geeignetes Werkzeug, eine passende Methode und praktische Erfahrung. In der Praxis wird ArchiMate häufig mit TOGAF kombiniert, da TOGAF ein Vorgehensmodell für die Entwicklung von Unternehmensarchitekturen bereitstellt (SPARX Systems Europe, 2026).

ArchiMate ist besonders in Bereichen wie Informationstechnologie, Telekommunikation, Finanzwesen und öffentlicher Verwaltung verbreitet. Die Sprache eignet sich vor allem dann, wenn Geschäftsprozesse, Anwendungen und Technologien auf einer höheren Abstraktionsebene gemeinsam betrachtet und miteinander verknüpft werden sollen (SPARX Systems Europe, 2026).

Gleichzeitig bringt ArchiMate auch Herausforderungen mit sich. Die Sprache enthält viele Konzepte, Beziehungen und Notationen. Für ungeübte Anwenderinnen und Anwender kann dies zunächst komplex wirken. Deshalb ist es wichtig, die Modellierung schrittweise aufzubauen und sich an klaren Fragestellungen zu orientieren (SPARX Systems Europe, 2026).

## **4.1 Archimate Elemente:**

### **4.1.1 Kernelemente**

Die Kernelemente bilden die Grundlage der ArchiMate-Modellierung. Mit ihnen beschreiben werden, **wer oder was handelt** (*aktive Struktur*), **welches Verhalten ausgeführt wird** (*Verhalten*), **welche Informationen oder Objekte dabei verwendet oder erzeugt werden** (*passive Struktur*) und **welche Leistungen nach außen angeboten werden** (*Services*).

Diese Grundstruktur findet sich in der

- Business-,
- Application- und
- Technology-Schicht

wieder. So kann beispielsweise ein Geschäftsakteur einen Geschäftsprozess ausführen, der einen Geschäftsservice bereitstellt und dabei auf ein Geschäftsobjekt zugreift. Neben konkreten Elementen enthält ArchiMate auch allgemeine Elemente wie *Location*, *Grouping* und *Plateau*, mit denen Orte, zusammengehörige Modellelemente oder bestimmte Zustände einer Architektur dargestellt werden können.

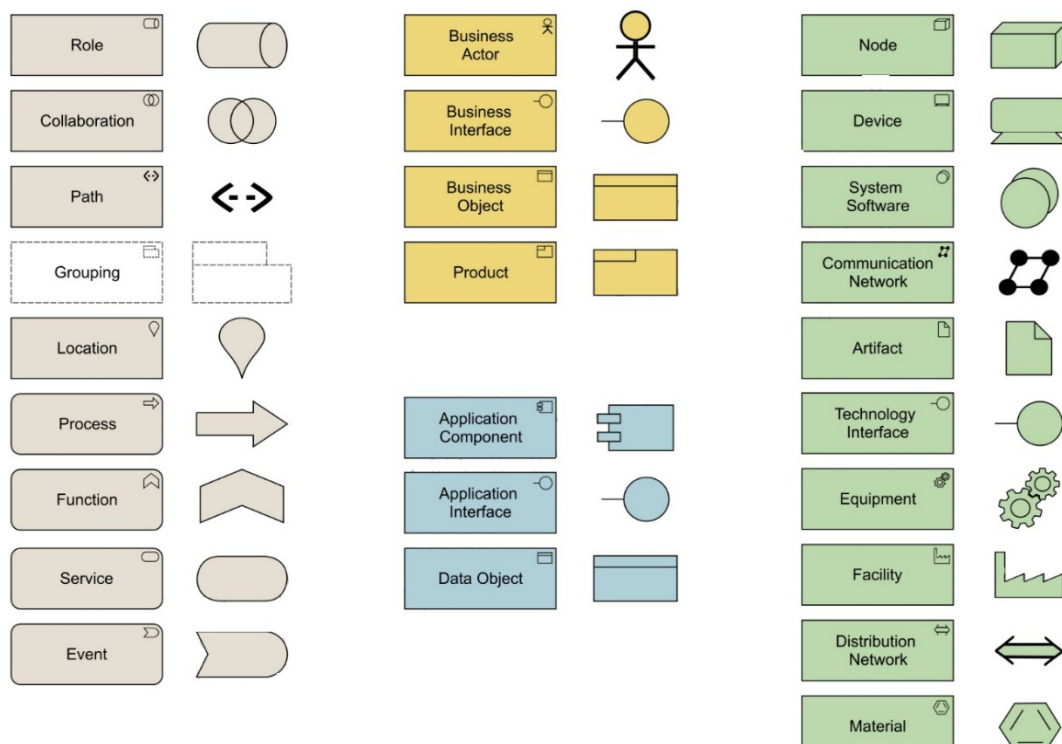


Abbildung 31: ArchiMate Kernelemente (The Open Group, 2026, S 123)

## Motivationselemente

Motivationselemente erklären, **warum eine Architektur gestaltet oder verändert wird**. Sie dienen dazu, die Beweggründe, Ziele und Anforderungen hinter einer Lösung nachvollziehbar darzustellen. Mit *Stakeholdern* werden betroffene oder interessierte Personen und Gruppen beschrieben. *Driver* stellen interne oder externe Einflussfaktoren dar, während *Assessments* deren Bewertung wiedergeben. Daraus können *Goals* und konkretere *Outcomes* abgeleitet werden. *Principles*, *Requirements* und *Constraints* legen fest, nach welchen Grundsätzen und unter welchen Bedingungen die Architektur entwickelt werden soll. Ein *Meaning*-Element beschreibt die Bedeutung eines Sachverhalts, während ein *Value*-Element den erwarteten Nutzen ausdrückt. Motivationselemente werden üblicherweise mit Elementen anderer Ebenen verknüpft, um zu zeigen, welche Architekturbausteine bestimmte Ziele oder Anforderungen verwirklichen.

## Strategielemente

Strategieelemente dienen dazu, die **strategische Ausrichtung eines Unternehmens und deren Umsetzung** zu beschreiben. Eine *Resource* stellt eine verfügbare oder benötigte Ressource dar, beispielsweise Mitarbeitende, Wissen, Kapital oder technische Infrastruktur. Eine *Capability* beschreibt eine Fähigkeit, über die eine Organisation verfügt oder die sie entwickeln möchte. Mit einem *Value Stream* wird dargestellt, durch welche aufeinanderfolgenden wertschöpfenden Aktivitäten ein Nutzen für Kundinnen, Kunden oder andere Stakeholder entsteht. Ein *Course of Action* beschreibt schließlich eine strategische Vorgehensweise oder Maßnahme. Strategieelemente schaffen damit die Verbindung zwischen den Unternehmenszielen und der späteren Ausgestaltung von Geschäftsprozessen, Anwendungen und Technologien.

### Implementierungs- und Migrationselemente

Mit den Implementierungs- und Migrationselementen wird dargestellt, **wie eine bestehende Architektur in eine angestrebte Zielarchitektur überführt werden soll**. Ein *Work Package* beschreibt ein zeitlich und inhaltlich abgegrenztes Arbeitspaket, beispielsweise die Einführung einer neuen Anwendung. Ein *Deliverable* bezeichnet ein überprüfbares Ergebnis eines solchen Arbeitspakets. Ein *Implementation Event* stellt ein für die Umsetzung relevantes Ereignis oder einen Meilenstein dar. Mit einem *Plateau* wird ein stabiler Zustand der Architektur zu einem bestimmten Zeitpunkt beschrieben, etwa die Ist-, Übergangs- oder Zielarchitektur. Ein *Gap* kennzeichnet den Unterschied zwischen zwei solchen Zuständen. Diese Elemente unterstützen daher insbesondere die Planung von Transformationsvorhaben und die Darstellung einer Architektur-Roadmap.

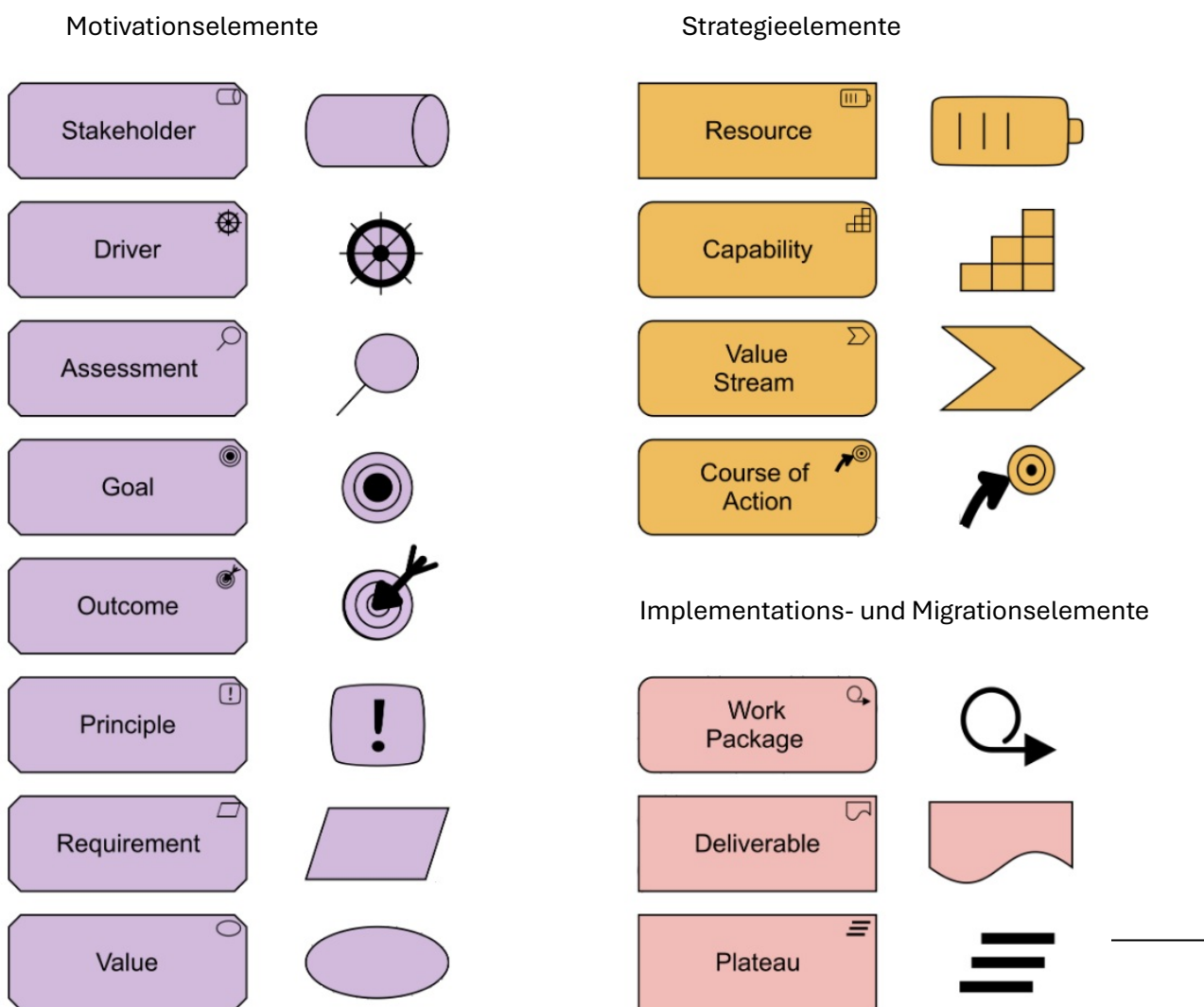


Abbildung 32: Motivation-, Strategie-, Implementierungs- und Migrationselemente (The Open Group, 2026, S124)

### 4.1.3 Beziehungen und Verknüpfungspunkte

Beziehungen legen fest, **wie die einzelnen ArchiMate-Elemente miteinander verbunden sind und welche Bedeutung diese Verbindung besitzt**. Strukturbeziehungen wie Composition, Aggregation, Assignment und Realization beschreiben beispielsweise Teil-Ganzes-Beziehungen, Verantwortlichkeiten oder die konkrete Umsetzung eines abstrakten Elements. Abhängigkeitsbeziehungen wie Serving, Access, Influence und Association zeigen unter anderem, dass ein Element eine Leistung für ein anderes bereitstellt, auf Informationen zugreift oder ein anderes Element beeinflusst. Dynamische Beziehungen wie Triggering und Flow stellen zeitliche Auslöser sowie den Austausch von Informationen, Gütern oder Werten dar. Die Specialization-Beziehung kennzeichnet eine speziellere Ausprägung eines allgemeinen Elements.

Junctions beziehungsweise Verknüpfungspunkte ermöglichen es, mehrere Beziehungen desselben Typs logisch zusammenzuführen oder aufzuteilen. **Dabei wird zwischen UND-Verknüpfungen und ODER-Verknüpfungen unterschieden**. Sie können beispielsweise ausdrücken, dass mehrere Voraussetzungen gemeinsam erfüllt sein müssen oder dass eine von mehreren Alternativen ausreicht. Verknüpfungspunkte sollten gezielt eingesetzt werden, da sie ausschließlich Beziehungen miteinander verbinden und keine eigenständigen fachlichen oder technischen Objekte darstellen.

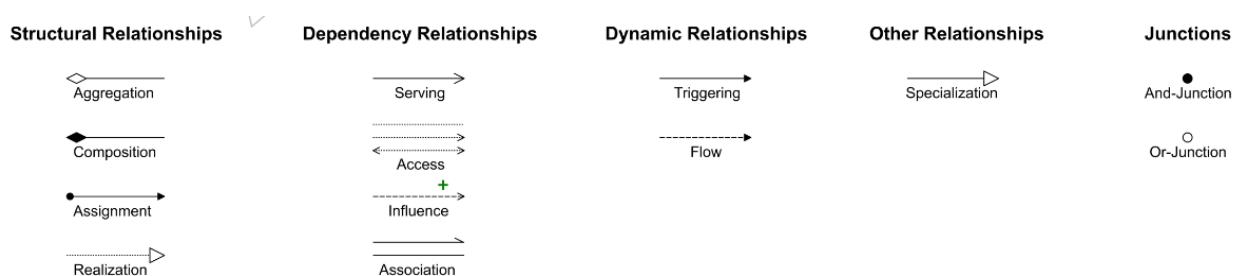


Abbildung 33: ArchiMate Beziehungen und Verknüpfungspunkte (The Open Group, 2026)S 124

Beziehungen sind ein wesentlicher Bestandteil jedes Modells. Durch sie entsteht die eigentliche Aussagekraft des Modells. Daher ist es notwendig, die spezifischen Beziehungen so zu wählen, dass diese fachlich sicher begründet werden können.

| Beziehung | Lesart | Beispiel |
|-----------|--------|----------|
|-----------|--------|----------|

|             |                                      |                                            |
|-------------|--------------------------------------|--------------------------------------------|
| Assignment  | führt aus / ist verantwortlich für   | Studienservice führt den Prozess aus.      |
| Realization | realisiert ein abstrakteres Element  | Der Prozess realisiert den Anmeldeservice. |
| Serving     | stellt Funktionalität bereit         | Die API dient dem Geschäftsprozess.        |
| Access      | liest oder schreibt passive Struktur | Die Funktion greift auf Anmeldedaten zu.   |
| Triggering  | löst nachfolgendes Verhalten aus     | MVP löst Pilotbetrieb aus.                 |
| Flow        | überträgt Information oder Wert      | Anmeldedaten fließen zur Bestätigung.      |
| Composition | besteht exklusiv aus Teilen          | Der Node umfasst System Software.          |
| Influence   | wirkt auf ein Motivationselement     | Bearbeitungszeit beeinflusst das Ziel.     |
| Association | unspezifischer Zusammenhang          | Nur wenn keine präzisere Beziehung passt.  |

**Tabelle 18: Beziehungen richtig auswählen**

Beispiele:

- Serving zeigt vom Anbieter zum Nutzer
- Realization zeigt vom konkreten zum abstrakteren Element.
- Access verbindet Verhalten mit passiver Struktur.

## 4.2 Erstellen eines ArchiMate Modells – Step by Step

Im Folgenden beschäftigen wir uns mit der Vorgehensweise zur Erstellung eines ArchiMate Modells anhand eines kleinen Übungsbeispiels.

### Ausgangssituation:

„Digitale Kursanmeldung“

Eine Fachhochschule wickelt Kursanmeldungen teilweise per E-Mail und Tabellenkalkulation ab. Studierende erhalten spät Rückmeldung, das Studienservice überträgt Daten mehrfach, und der Bearbeitungsstand ist nicht transparent. Künftig sollen sich Studierende im Campus-Portal selbst anmelden und sofort eine Bestätigung erhalten. Die Lösung muss datenschutzkonform sein und schrittweise eingeführt werden

### 4.2.1 Architektur & Scope

Bevor mit der Modellierung begonnen wird, müssen einige grundlegende Fragen beantwortet werden:

Z.B.:

- Wie kann ein Campusportal eine durchgängige Kursanmeldung ermöglichen?
- Wann startet der Anmeldeprozess und wie wird dieser Beendet?
- Wer sind die Stakeholder?
- Wie sieht der zeitliche Ablauf des Projektes aus?

In der Praxis ist es Hilfreich, diese Fragestellungen kompakt in einer Tabelle festzuhalten.

| Festlegung       | Ausprägung                                                                 |
|------------------|----------------------------------------------------------------------------|
| Architekturfrage | Wie ermöglicht ein Campus-Portal eine durchgängige digitale Kursanmeldung? |
| Umfang           | Vom Aufruf des Portals bis zur Bestätigung                                 |
| Nicht betrachtet | Lehrplanung, Gebühren und Prüfungsverwaltung                               |
| Stakeholder      | Studierende, Studienservice, IT-Betrieb und Datenschutz                    |
| Zeithorizont     | Ist-Zustand, MVP/Pilot und Zielzustand                                     |

**Tabelle 19: Fragen zu Architektur & Scope**

### 4.2.2 Modellierung

### vorbereiten

Bitte beachten Sie, dass es sich bei der Modellierung nicht um das reine Zeichnen von Objekten handelt. Die entsprechenden Tools arbeiten mit einem Repository, das es Ihnen ermöglicht, einzelne Elemente nur einmal anzulegen und in mehreren Sichten wieder zu verwenden. Das ist insbesondere dann wichtig, wenn es zu Änderungen kommt oder über die Modelle entsprechende Reports angelegt werden.

Die Modellierung selbst beginnen Sie am besten mit einer Fragestellung. Die Frage sollte so formuliert sein, dass diese bereits zu den benötigten Ebenen und Beziehungen hinweist. In unserem Fall wäre das z.B:

„Welche Geschäftsleistung wird durch welche Anwendung und durch welche technische Plattform ermöglicht?

Basierend auf dieser Fragestellung erhalten Sie bereits einen ersten Überblick über den benötigten Elemente-Katalog:

| <b>Ebene</b> | <b>Standardobjekte</b>                                                        | <b>Leitfrage</b>                 |
|--------------|-------------------------------------------------------------------------------|----------------------------------|
| Motivation   | Stakeholder, Driver, Assessment, Goal, Outcome, Requirement, Constraint       | Warum ist die Veränderung nötig? |
| Geschäft     | Business Actor, Role, Process, Service, Object, Event                         | Wer erbringt welche Leistung?    |
| Anwendung    | Application Component, Function, Service, Data Object                         | Welche IT unterstützt?           |
| Technologie  | Node, Device, System Software, Technology Function/Service, Artifact, Network | Worauf wird sie betrieben?       |
| Umsetzung    | Work Package, Deliverable, Plateau, Gap                                       | Wie erfolgt der Wandel?          |

Wie wir bereits gesehen **Tabelle 20: Benötigter Elemente-Katalog** haben, ordnet ArchiMate die Elemente gleichzeitig nach zwei Dimensionen. Die Ebenen beantworten, in welchem fachlichen Bereich ein Element liegt. Die Aspekte erklären, welche grundsätzliche Rolle das Element dort einnimmt. Erst die Kombination beider Dimensionen bestimmt die Bedeutung eines Elements.

### Hinweis:

Die folgende Tabelle zeigt die ArchiMate-Matrix in einer angepassten Farbgebung. Da die nachfolgenden Beispielmodelle mit **ARIS Architect** der Software AG erstellt wurden, wurden auch die Farben der Matrix an die Darstellung in diesem Modellierungswerkzeug angepasst. Die inhaltliche Bedeutung und Zuordnung der ArchiMate-Elemente bleiben dadurch unverändert.

|                                                 | Active Structure                                                                                  | Behavior                                                          | Passive Structure                                |  |
|-------------------------------------------------|---------------------------------------------------------------------------------------------------|-------------------------------------------------------------------|--------------------------------------------------|--|
| <b>Strategy</b>                                 | Resource                                                                                          | Capability / Value Stream / Cours of Action                       |                                                  |  |
| <b>Business</b>                                 | Actor / Role<br>Collaboration / Interface                                                         | Process / Function /<br>Interaction / Service                     | Business Oject /<br>Contract /<br>Representation |  |
| <b>Application</b>                              | Component /<br>Collaboration / Interface                                                          | Function / Process /<br>Interaction / Event /<br>Service          | Data Object                                      |  |
| <b>Technology /<br/>Physical</b>                | Node / Device / System /<br>Software / Facility /<br>Equipment / Network                          | Technology Function /<br>Process Interaction /<br>Evant / Service | Artivact / Material                              |  |
| <b>Implementation<br/>/ Migration</b>           |                                                                                                   | Work Package                                                      | Deliverable                                      |  |
| Begründet und beeinflusst Elemente aller Ebenen |                                                                                                   |                                                                   |                                                  |  |
| <b>Motivation</b>                               | Stakeholder / Driver / Assessment / Goal / Outcome / Principle / Requirement / Constraint / Value |                                                                   |                                                  |  |

**Tabelle 21: Zusammenhang zwischen Ebenen, Aspekten und Motivation**

### Hinweis:

Wie bereits zu Beginn dieses Skripts erläutert, kann sich die grafische Darstellung von Modellelementen je nach verwendetem Modellierungswerkzeug unterscheiden. Da die nachfolgenden Modelle mit ARIS Architect in der Version 10 erstellt wurden, zeigt die folgende Abbildung die darin verwendeten Objekte in der ARIS-spezifischen Darstellungsform. Die Symbole orientieren sich an den Vorgaben der Open Group, weisen jedoch das für ARIS typische Erscheinungsbild auf.

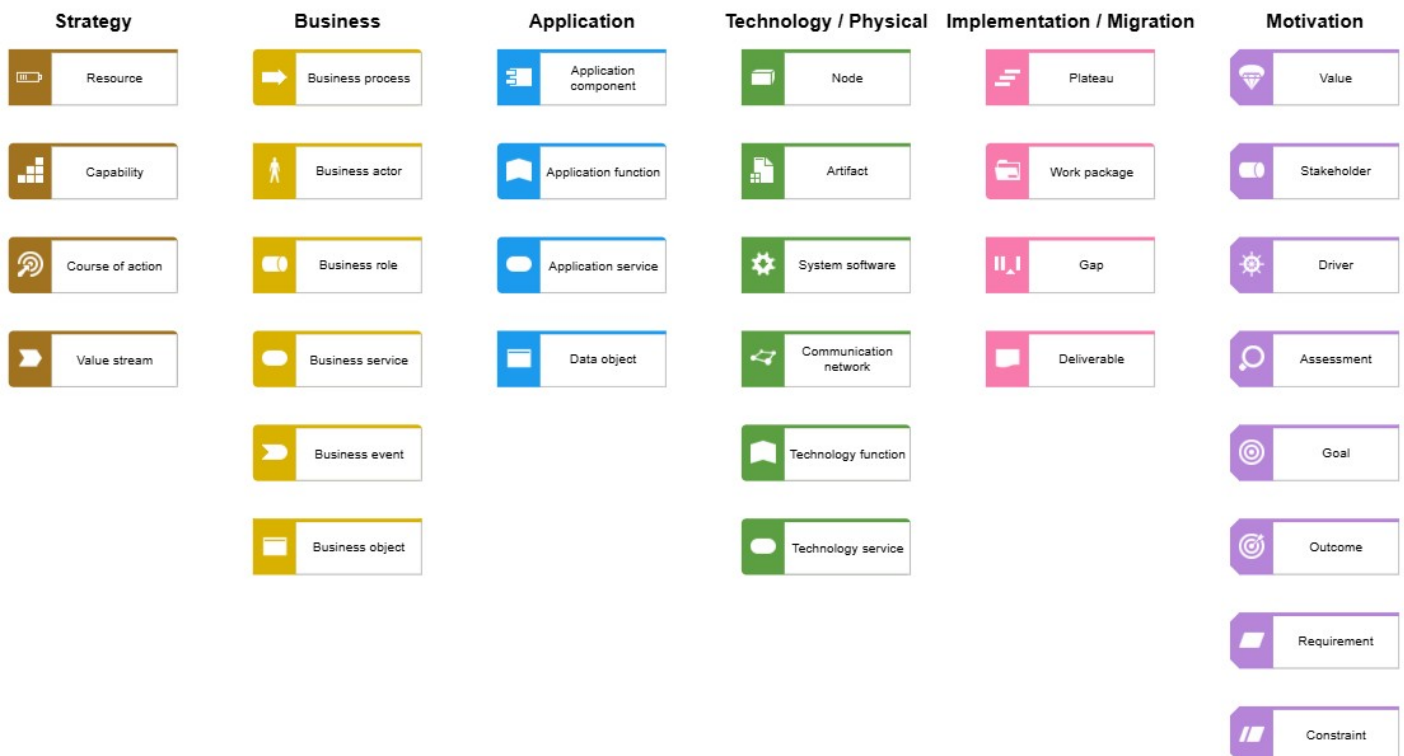


Abbildung 34: Verwendete Objekte in der ARIS-spezifischen Darstellungsform

Basierend auf dem ArchiMate Ebenen- und Aspekte Konzept kann man als nächstes die Überlegung anstellen, nämlich auf welchen Ebenen das jeweilige Element liegt.

| Ebene                      | Schwerpunkt                                                               | Beispiel                                                                    |
|----------------------------|---------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| Strategy                   | Ressourcen, Fähigkeiten und strategische Handlungsoptionen                | Resource „IT-Kompetenz“, Capability „Digitale Studierendenservices“         |
| Business                   | Organisation, fachliches Verhalten, Leistungen und Informationen          | Business Role „Studienservice“, Business Process „Kursanmeldung bearbeiten“ |
| Application                | Softwarebausteine, Anwendungsverhalten, Services und Daten                | Application Component „Campus-Portal“, Data Object „Anmeldedatensatz“       |
| Technology / Physical      | Technische und physische Infrastruktur, Plattformleistungen und Artefakte | Node „Applikationsserver“, Equipment, Artifact „Container-Artefakt“         |
| Implementation & Migration | Umsetzung, Ergebnisse und Architekturzustände                             | Work Package „Portal-MVP“, Deliverable „MVP bereitgestellt“, Plateau        |

**Tabelle 22: Wo liegt das Element**

Mit den Aspekten werden die Rollen der Elemente ausgedrückt.

Wichtig: Ebene und Aspekt dürfen nicht verwechselt werden. Die Ebene – beispielsweise **Application** – zeigt, welchem Bereich der Unternehmensarchitektur ein Element zugeordnet ist. Der Aspekt – beispielsweise **Behavior** – beschreibt, welche Rolle das Element innerhalb dieser Ebene erfüllt.

Ein Application Process gehört daher:

- zur **Application-Ebene** und
- zum Aspekt **Behavior**.

| Aspekt            | Leitfrage                                                               | Typische Beziehung                                          |
|-------------------|-------------------------------------------------------------------------|-------------------------------------------------------------|
| Active Structure  | Wer oder was kann Verhalten ausführen?                                  | Assignment verbindet aktive Struktur mit Verhalten.         |
| Behavior          | Was geschieht oder welche Leistung wird angeboten?                      | Realization verbindet internes Verhalten mit einem Service. |
| Passive Structure | Woran wird Verhalten ausgeführt oder welche Information wird verwendet? | Access verbindet Verhalten mit passiver Struktur.           |

**Tabelle 23: Die Aspekte geben an, welche Rolle ein Element hat**

### Was bei der Motivation zu beachten ist

Motivation ist keine zusätzliche operative Schicht zwischen Strategy und Business. Motivation beschreibt vielmehr, warum Architekturentscheidungen getroffen werden. Stakeholder, Driver, Assessments, Goals, Outcomes, Principles, Requirements, Constraints und Values können Elemente jeder Ebene begründen, beeinflussen oder einschränken.

### Nicht jede Zelle in der Matrix muss befüllt sein

Das Schema ist kein starres Raster, in dem jede Ebene zwingend aktive Struktur, Verhalten und passive Struktur besitzen muss.

In **Strategy** gibt es mit Resource ein aktives Strukturelement und mit Capability, Value Stream sowie Course of Action Verhaltenselemente, aber kein eigenes passives Strukturelement.

**Implementation & Migration** enthält mit Work Package vor allem Verhalten und mit Deliverable passive Struktur. Plateau und Gap beschreiben Architekturstände beziehungsweise Unterschiede und lassen sich nicht sinnvoll auf die drei Kernaspekte reduzieren.

Auch **Physical** ergänzt die Technology-Ebene insbesondere um aktive Strukturen wie Equipment und Facility sowie passive Struktur wie Material; technisches Verhalten wird mit den Technology-Behavior-Elementen modelliert.

### Über die Ebenen hinweg zeigt sich damit dasselbe Denkprinzip:

Eine aktive Struktur führt Verhalten aus; das Verhalten greift auf passive Struktur zu oder erzeugt sie; ein Service macht ausgewähltes Verhalten für andere Elemente nutzbar. Motivation erklärt, weshalb dieses Zusammenspiel benötigt wird.

| Ebene                                 | Active Structure                  | Behavior                                   | Passive Structure  |
|---------------------------------------|-----------------------------------|--------------------------------------------|--------------------|
| <b>Strategy</b>                       | Resource                          | Capability, Value Stream, Course of Action | kein eigener Typ   |
| <b>Business</b>                       | Business Actor / Role             | Business Process / Function / Service      | Business Object    |
| <b>Application</b>                    | Application Component             | Application Function / Process / Service   | Data Object        |
| <b>Technology / Physical</b>          | Node, Device, Equipment, Facility | Technology Function / Process / Service    | Artifact, Material |
| <b>Implementation &amp; Migration</b> | kein eigener Typ                  | Work Package                               | Deliverable        |

**Tabelle 24: : Grundmuster der ArchiMate Matrix**

## 4.2.3 Modellierung

In den nächsten Schritten werden die jeweiligen Modelle erstellt:

### 4.2.3.1 Strategie Modell

Die Strategie-Ebene verbindet Motivation mit der operativen Architektur. Sie zeigt, welche Ressourcen und Fähigkeiten benötigt werden und mit welcher strategischen Handlungsoption diese aufgebaut oder genutzt werden.

| ArchiMate-Elementtyp | Elementname                   |
|----------------------|-------------------------------|
| Resource             | IT-Kompetenz                  |
| Capability           | Digitale Studierendenservices |
| Course of Action     | Portal zuerst                 |

|                  |                          |
|------------------|--------------------------|
| Value Stream     | Digitale Student Journey |
| Business Process | Kursanmeldung bearbeiten |

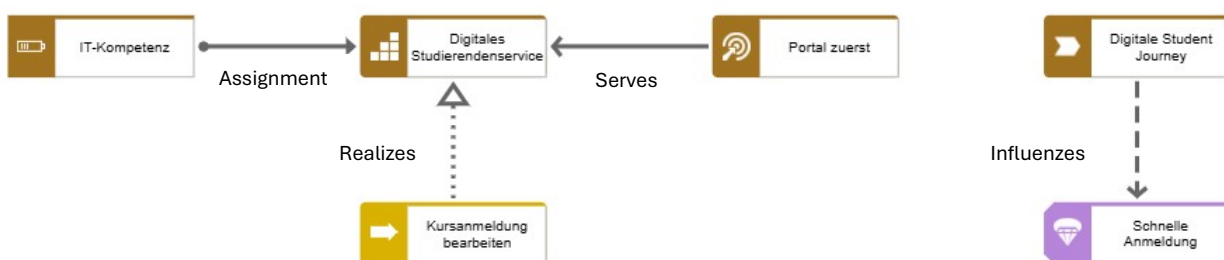
**Tabelle 25: Strategie Modell - Elemente aus unserem Beispiel**

**Es soll die Frage beantwortet werden:**

Welche Fähigkeiten und Ressourcen benötigt werden?

**Vorgehen:**

1. Benötigte Ressourcen identifizieren.
2. Fähigkeit als Capability formulieren.
3. Strategische Handlungsoption als Course of Action anlegen.
4. Value Stream aus Sicht des Nutzens beschreiben.
5. Geschäftsverhalten mit der Capability verbinden.



**Abbildung 35: Strategische Sicht der digitalen Kursanmeldung (Erstellt mit Aris Architect 10)**

**Hinweis:**

Lesen Sie jede Beziehung vom Ursprung zum Ziel als fachlichen Satz. Ist der Satz nicht verständlich, prüfen Sie Elementtyp, Richtung und Beziehung.

Z.B: IT-Kompetenz wird dem Digitalen Studierendenservice zugewiesen.

#### 4.2.3.2 Motivations-Modell

Mit dem Motivations Modell soll zum Ausdruck gebracht werden, weshalb die Architektur verändert werden soll. Das Modell liegt grundsätzlich quer zu den Ebenen und verhindert, dass sofort über Systeme gesprochen wird, bevor Ziel und Nutzen geklärt sind.

| ArchiMate-Elementtyp | Elementname       |
|----------------------|-------------------|
| Stakeholder          | Studierende       |
| Driver               | Manueller Aufwand |

|             |                                       |
|-------------|---------------------------------------|
| Assessment  | Lange Bearbeitungszeit                |
| Goal        | Digitale Anmeldung                    |
| Outcome     | Anmeldung in weniger als fünf Minuten |
| Requirement | Self-Service ermöglichen              |
| Constraint  | Datenschutz einhalten                 |

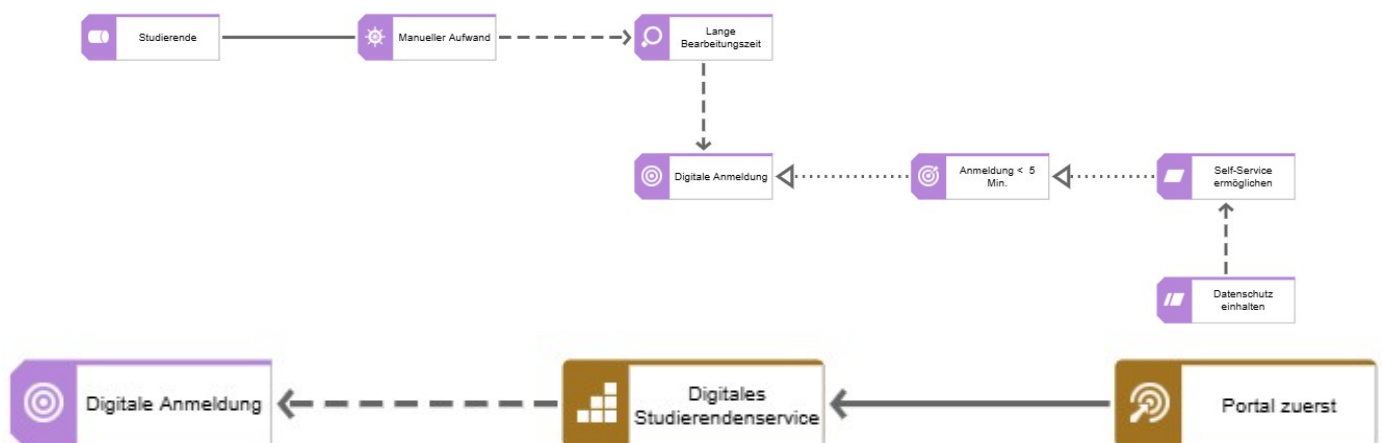
**Tabelle 26: Motivations Modell: Elemente aus unserem Beispiel**

**Es soll die Frage beantwortet werden:**

Warum soll die Architektur verändert werden?

### Vorgehen

1. Treiber und Bewertung identifizieren.
2. Ziel und messbares Ergebnis unterscheiden.
3. Requirement und Constraint aus dem Ziel ableiten.
4. Influence und Realization in Leserichtung prüfen.



**Abbildung 36: Motivations-Modell**

#### 4.2.3.3 Geschäftsarchitektur Modell

Die Geschäftsarchitektur beschreibt die fachliche Leistung. Business Role ist aktive Struktur, Business Process und Business Service sind Verhalten, Business Object ist passive Struktur.

**Tabelle 27: Geschäftsarchitektur Modell: Elemente aus unserem Beispiel**

| ArchiMate-Elementtyp       | Elementname                                          |
|----------------------------|------------------------------------------------------|
| Business Actor / Role      | Studierende / Studienservice – Active Structure      |
| Business Process / Service | Kursanmeldung bearbeiten / Anmeldeservice – Behavior |
| Business Object            | Anmeldedaten – Passive Structure                     |
| Business Event             | Anmeldung abgeschlossen – Behavior                   |

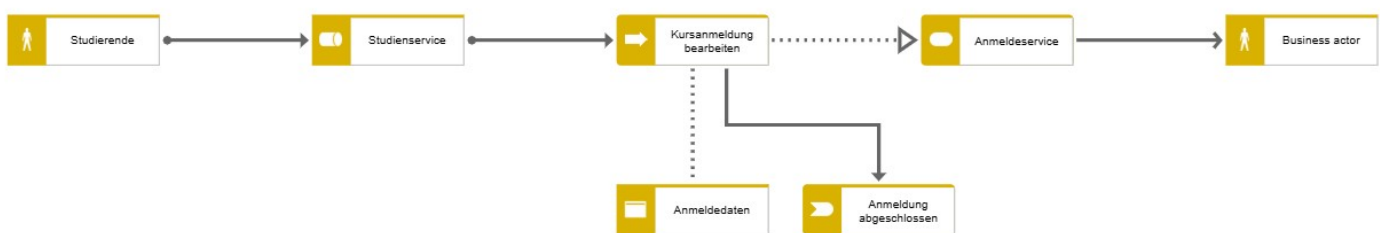
**Tabelle 28: Geschäftsarchitektur Modell: Elemente aus unserem Beispiel**

**Es soll die Frage beantwortet werden:**

Wer erbringt welche fachliche Leistun?

#### Vorgehen

1. Actor und Role anlegen.
2. Role dem Process mit Assignment zuweisen.
3. Process realisiert Service.
4. Service dient dem Actor über Serving.
5. Process greift über Access auf Business Object zu.



**Abbildung 37: Geschäftsarchitektur-Modell**

#### 4.2.3.4 Application Modell

Auf der Anwendungsebene bildet Application Component die aktive Struktur, Function und Service das Verhalten und Data Object die passive Struktur. Das Grundmuster bleibt somit gleich, nur die fachliche Ebene ändert sich.

| ArchiMate-Elementtyp           | Elementname                                             |
|--------------------------------|---------------------------------------------------------|
| Application Component          | Campus-Portal – Active Structure                        |
| Application Function / Service | Anmeldung verwalten / Kursanmeldungs-API – Behavior     |
| Data Object                    | Anmeldedatensatz – Passive Structure                    |
| Business Process               | Kursanmeldung bearbeiten – bedientes Geschäftsverhalten |

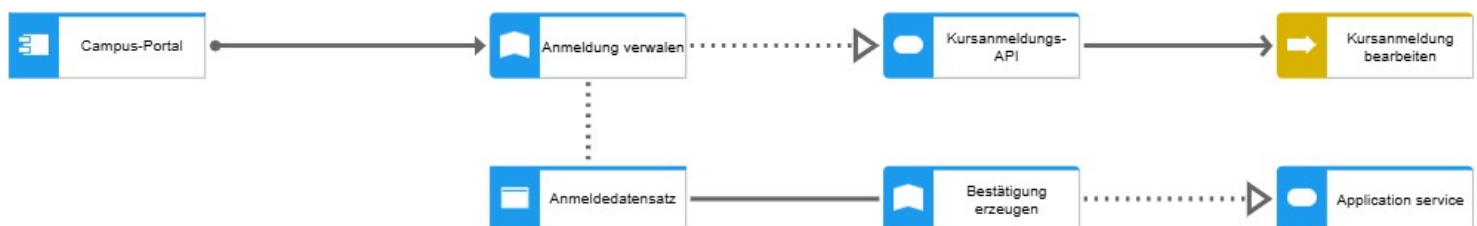
**Tabelle 29: Application Modell: Elemente aus unserem Beispiel**

**Es soll die Frage beantwortet werden:**

Welche Anwendung unterstützt den Geschäftsprozess?

#### Vorgehen

1. Component ist der Function zugewiesen.
2. Function realisiert Application Service.
3. Application Service dient dem Business Process.
4. Function greift auf Data Object zu.
5. Flow benennt übertragene Information.



**Abbildung 38: Application Modell**

#### 4.2.3.5 Technologie und physisches Modell

Node, Device, System Software, Equipment und Facility gehören zur aktiven Struktur. Technology Function, Process und Service beschreiben Verhalten. Artifact und Material bilden passive Struktur.

| ArchiMate-Elementtyp          | Elementname                                               |
|-------------------------------|-----------------------------------------------------------|
| Node / System Software        | Applikationsserver / Container Runtime – Active Structure |
| Technology Function / Service | Container ausführen / Hosting-Service – Behavior          |
| Artifact                      | Container-Artefakt – Passive Structure                    |
| Communication Network         | Hochschulnetz – Active Structure                          |

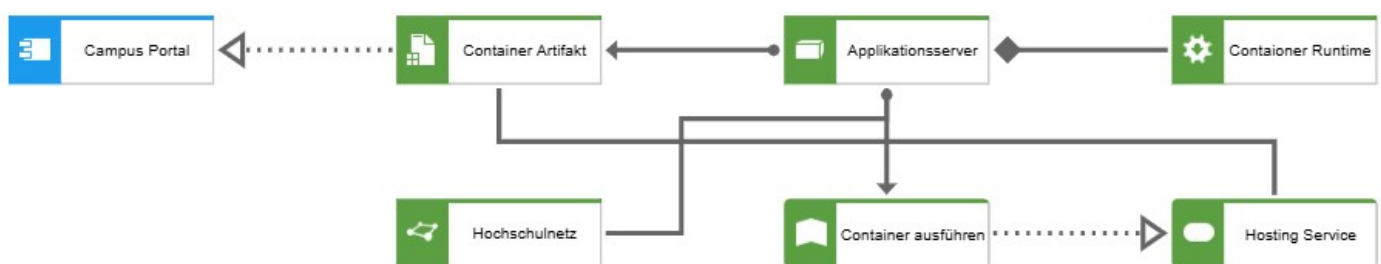
**Tabelle 30: Technologie und physisches Modell: Elemente aus unserem Beispiel**

**Es soll die Frage beantwortet werden:**

Worauf wird die Anwendung betrieben?

#### Vorgehen

1. Artifact realisiert Application Component.
2. Artifact wird einem Node zugewiesen.
3. Node umfasst System Software.
4. Technology Function realisiert Technology Service.
5. Technology Service dient Application Component.



**Abbildung 39: Technologie- und physisches Modell**

#### 4.2.3.6 Umsetzungs- und Migrations-Modell

In Implementation & Migration ist Work Package ein Verhaltenselement und Deliverable ein passives Strukturelement. Plateau und Gap beschreiben Zustände beziehungsweise Unterschiede und gehören nicht zum Dreierschema.

| ArchiMate-Elementtyp | Elementname                                                  |
|----------------------|--------------------------------------------------------------|
| Work Package         | Portal-MVP und Pilotbetrieb – Behavior                       |
| Deliverable          | MVP bereitgestellt und Pilot ausgewertet – Passive Structure |
| Plateau              | Ist- und Ziel-Architektur – Architekturzustände              |
| Gap                  | Medienbruch – Unterschied zwischen Zuständen                 |
| Requirement          | Self-Service ermöglichen – Motivation                        |

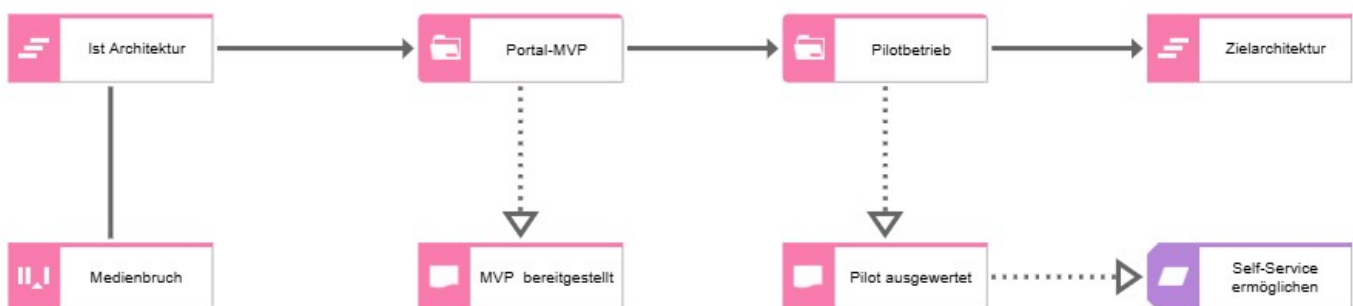
**Tabelle 31: Umsetzungs- und Migrations-Modell: Elemente aus unserem Beispiel**

**Es soll die Frage beantwortet werden:**

Wie wird die Zielarchitektur erreicht?

### Vorgehen

1. Ist- und Ziel-Plateau anlegen.
2. Wesentliche Lücke als Gap beschreiben.
3. Veränderung in Work Packages zerlegen.
4. Deliverables den Work Packages zuordnen.
5. Deliverables mit Requirements oder Architekturbausteinen verbinden.

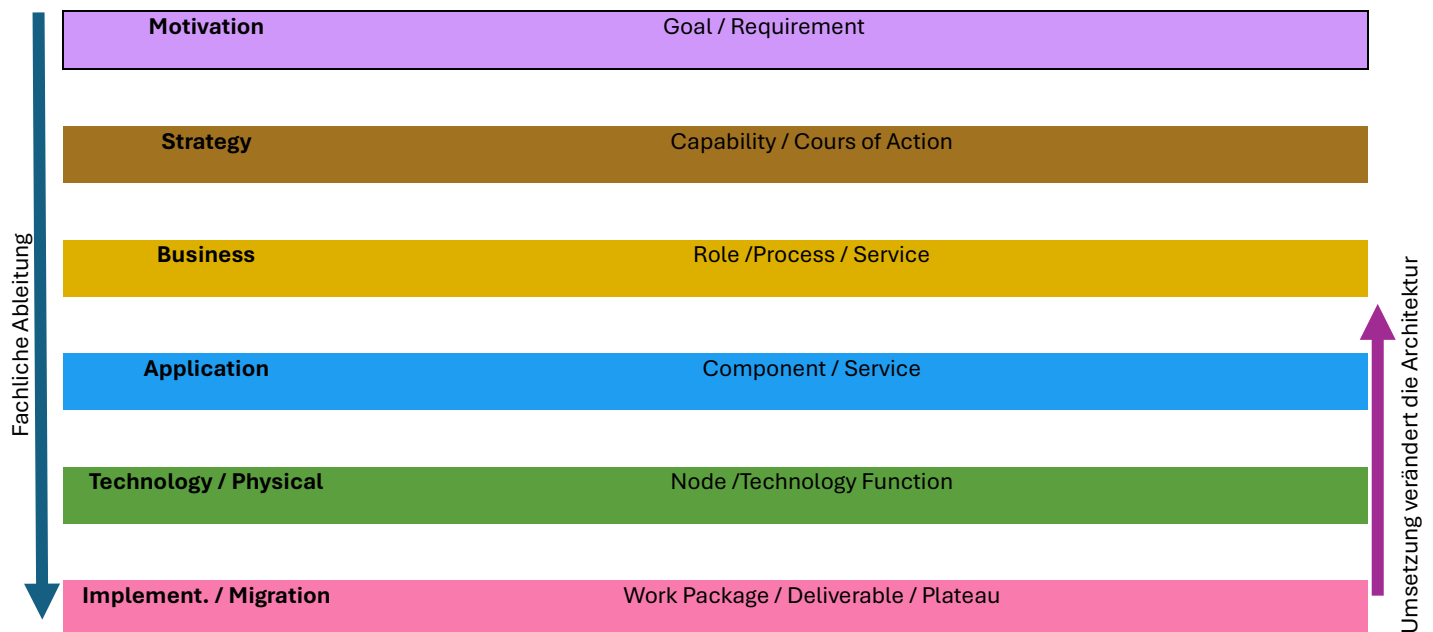


**Abbildung 40: Umsetzungs- und Migrations-Modell**

## 4.3 Ebenen verbinden und analysieren

Durch die Verknüpfung der ArchiMate-Ebenen entsteht ein zusammenhängendes Architekturmodell. Darin kann nachvollzogen werden, wie strategische Ziele und Anforderungen mit Geschäftsprozessen, Anwendungen, technischen Komponenten und konkreten Umsetzungsmaßnahmen verbunden sind. Dadurch lassen sich Abhängigkeiten und die Auswirkungen von Änderungen oder Ausfällen über mehrere Ebenen hinweg analysieren.

Die Ebenen werden nicht nur untereinander angeordnet. Standardisierte Beziehungen machen nachvollziehbar, warum eine Fähigkeit benötigt wird, wodurch sie umgesetzt wird und welche technischen sowie projektbezogenen Abhängigkeiten bestehen.



### Fachliche Ableitung → Schrittweise Konkretisierung der Architektur:

- Aus einem Ziel wird eine benötigte Fähigkeit abgeleitet.
- Die Fähigkeit wird durch Geschäftsprozesse umgesetzt.
- Die Geschäftsprozesse werden durch Anwendungen unterstützt.
- Die Anwendungen benötigen eine technische Infrastruktur.

### Umsetzung verändert Architektur → Umsetzungsmaßnahmen führen zur Zielarchitektur

- Das Work Package „Campus-Portal entwickeln“ erzeugt ein Deliverable.
- Dieses Deliverable führt zur Einführung oder Änderung des Campus-Portals.
- Dadurch verändern sich Application-, Business- und möglicherweise Technology-Elemente.
- Der neue Zustand wird als Ziel-Plateau dargestellt.

#### 4.3.1 Mögliche Verbindungen

Die Ausgangstabelle beschreibt sechs inhaltliche Verbindungen. In der praktischen Modellierung wird jede Verbindung durch konkrete ArchiMate-Elemente und zulässige Beziehungen ausgedrückt. Dadurch entsteht eine durchgängige Nachverfolgbarkeit von den Beweggründen bis zur technischen und organisatorischen Umsetzung.

| Inhaltliche Verbindung   | Konkretes Beispiel                                                     | ArchiMate-Beziehung       |
|--------------------------|------------------------------------------------------------------------|---------------------------|
| Motivation → Strategy    | Goal „Digitale Anmeldung“ → Capability „Digitale Studierendenservices“ | Influence                 |
| Strategy → Business      | Business Process „Kursanmeldung bearbeiten“ → Capability               | Realization               |
| Motivation → Geschäft    | Business Process → Requirement „Self-Service ermöglichen“              | Realization               |
| Application → Business   | Application Service „Kursanmeldungs-API“ → Business Process            | Serving                   |
| Technology → Application | Technology Service „Hosting-Service“ → Application Component           | Serving                   |
| Umsetzung → Architektur  | Work Package → Deliverable → Requirement → Ziel-Plateau                | Realization / Association |

**Tabelle 32: Mögliche Verbindungen zwischen den Ebenen**

#### 4.3.1.1 Motivation mit Strategie verbinden

**Es soll die Frage beantwortet werden:**

Warum?



**Abbildung 41: Motivation mit Strategie verbinden**

Das Ziel „Digitale Anmeldung“ wird durch „Digitales Studierendenservice“ beeinflusst. „Portal zuerst“ realisiert bzw. unterstützt deren Aufbau.

#### 4.3.1.2 Motivation mit Business verbinden

**Es soll die Frage beantwortet werden:**

Welche Anforderung / welches Geschäftselement erfüllt sie?



**Abbildung 42: Motivation mit Business verbinden**

Der Business Process „Kursanmeldung bearbeiten“ erfüllt das Requirement „Self-Service ermöglichen“ und realisiert zugleich den nach außen angebotenen Business Service „Anmeldeservice“.

#### 4.3.1.3 Strategie mit Business verbinden

**Es soll die Frage beantwortet werden:**

Welche Fähigkeit / welches Geschäftsverhalten setzt sie um?



**Abbildung 43: Strategie mit Business verbinden**

Der Geschäftsprozess „Kursanmeldung bearbeiten“ realisiert die strategische Capability „Digitales Studierendenservice“. Die Business Rolle „Studienservice“ wird dem Prozess über Assignment zugewiesen.

Die strategische Capability „Digitales Studierendenservice“ beschreibt, was die Fachhochschule beherrschen muss. Der Business Process „Kursanmeldung bearbeiten“ beschreibt das konkrete operative Verhalten, durch das diese Fähigkeit realisiert wird. Mittels Assignment der Business Rolle „Studienservice“ wird zusätzlich gezeigt, welche Business Role den Prozess ausführt.

**Eine mögliche Analysefrage wäre:**

„Welche Geschäftsprozesse, Rollen und Services müssen verändert werden, wenn die Capability „Digitale Studierendenservices“ ausgebaut wird.“

#### 4.3.1.4 Application mit Business verbinden

**Es soll die Frage beantwortet werden:**

Welche Anwendungsleistung unterstützt welches Geschäftsverhalten?



**Abbildung 44: Application mit Business verbinden**

Die Anwendungskomponente „Campus-Portal“ realisiert den Application Service „Kursanmeldungs-API“. Dieser wiederum dient dem Geschäftsprozess „Kursanmeldung bearbeiten“.

Die Serving-Beziehung verläuft vom bereitstellenden Application Service zum nutzenden Business Process. Damit lässt sich bei einer Änderung oder einem Ausfall des Campus-Portals unmittelbar erkennen, welche fachlichen Abläufe betroffen sind.

**Eine mögliche Analysefrage wäre:**

Welche Geschäftsprozesse können nicht oder nur eingeschränkt ausgeführt werden, wenn die Kursanmeldungs-API nicht verfügbar ist?

#### 4.3.1.5 Technology mit Application verbinden

**Es soll die Frage beantwortet werden:**

Welche technische Leistung trägt welche Anwendung?



**Abbildung 45: Technologie mit Application verbinden**

Der Node „Applikationsserver“ stellt über den Technologieservice „Hosting-Service“ die technische Funktionalität für den Service „Campus Portal“ bereit.

Auch hier wird Serving vom Anbieter zum Nutzer gelesen: Der Hosting-Service dient dem Campus-Portal. Die Anwendungskomponente wird dadurch mit ihrer technischen Betriebsgrundlage verbunden.

**Eine mögliche Analysefrage wäre:**

Welche Anwendungen und in weiterer Folge welche Geschäftsprozesse sind betroffen, wenn der Hosting-Service ausfällt?

#### 4.3.1.6 Umsetzung mit Architektur verbinden

**Es soll die Frage beantwortet werden:**

Welche Umsetzungsergebnisse verändern die Architektur?



**Abbildung 46: Umsetzung mit Architektur verbinden**

Das Workpackage „Portal-MVP“ erzeugt ein Deliverable „Portal bereitgestellt“ welches ein Requirement „Self-Service ermöglichen“ erfüllt. Die dadurch erstellte Architektur wird im Ziel-Plateau „Zielarchitektur“ beschrieben.

Work Packages beschreiben die auszuführende Projektarbeit, Deliverables deren Ergebnisse. Die Ergebnisse erfüllen Requirements und führen zu einer veränderten Architektur, die in einem Plateau als stabiler Zustand zusammengefasst werden kann.

**Eine mögliche Analysefrage wäre:**

Welches Arbeitspaket und welches Deliverable sind für die Erfüllung eines Requirements verantwortlich – und welcher Zielzustand hängt davon ab?

### 4.3.2 Mögliche Analysefälle aus der Praxis

Wie Sie bereits gesehen haben, ist das eigentliche Ziel der Verknüpfung, Auswirkungen über mehrere Ebenen hinweg verfolgen zu können. Eine typische Analyse beginnt bei einem Ereignis oder einer Änderung und folgt den Beziehungen durch das Modell.

Die nachfolgende Tabelle beschreibt einige Szenarien, welche wir bereits mit unserem kleinen Modell auswerten können:

| Inhaltliche Verbindung     | Konkretes Beispiel                                                              | ArchiMate-Beziehung                            |
|----------------------------|---------------------------------------------------------------------------------|------------------------------------------------|
| <b>Technischer Ausfall</b> | Hosting-Service → Campus-Portal → Kursanmeldungs-API → Kursanmeldung bearbeiten | betroffene Anwendungen und Prozesse            |
| <b>Neue Anforderung</b>    | Requirement → Business Process → Application Service → Technology Service       | notwendige fachliche und technische Änderungen |
| <b>Strategieänderung</b>   | Goal → Capability ← Business Process ← Application Service                      | betroffene Fähigkeiten und deren Realisierung  |
| <b>Projektverzug</b>       | Work Package → Deliverable → Requirement → Ziel-Plateau                         | gefährdete Anforderungen und Zielarchitektur   |

**Tabelle 33: Mögliche Analysefälle aus der Praxis**

## Lektion 5      Zusammenfassung: Stärken und Einsatzbereiche der vorgestellten Modellierungssprachen

Wie wir bereits gesehen haben (siehe Kapitel 1.1) reduzieren Modellierungssprachen die Komplexität betrieblicher Sachverhalte, indem sie ausgewählte Aspekte strukturiert und nachvollziehbar darstellen.

Welche Sprache für ein Projekt geeignet ist, hängt daher vor allem von der Fragestellung, der Zielgruppe und dem gewünschten Detaillierungsgrad ab.

D.h: Die Wahl der Modellierungssprache sollte stets vom **Modellierungszweck** ausgehen.

Keine der in diesem Skript vorgestellten Modellierungssprachen ist für alle Modellierungsaufgaben gleichermaßen geeignet. Flussdiagramme, EPK, BPMN, DMN und ArchiMate setzen jeweils einen anderen Schwerpunkt.

Für einfache Abläufe ist häufig ein Flussdiagramm ausreichend. EPK und BPMN eignen sich für Geschäftsprozesse, wobei EPK stärker fachlich und BPMN stärker standardisiert sowie detaillierbar ausgerichtet ist.

DMN wird eingesetzt, wenn die Entscheidungslogik im Mittelpunkt steht.

ArchiMate schafft den übergeordneten Zusammenhang zwischen Strategie, Organisation, Informationssystemen und Technologie. In der Praxis werden die Sprachen daher oft kombiniert: ArchiMate liefert den architektonischen Kontext, BPMN oder EPK beschreiben den Prozessablauf und DMN präzisiert die darin enthaltenen Entscheidungen.

Die folgenden Kapitel sollen nochmals die Eigenschaften der jeweiligen Modellierungssprachen zusammenfassen:

### 5.1 Flussdiagramme

Flussdiagramme eignen sich besonders für die einfache und anschauliche Darstellung von Abläufen, Algorithmen und Entscheidungswegen. Ihre Stärke liegt in der leichten Verständlichkeit: Auch Personen ohne vertiefte Modellierungskenntnisse können einen überschaubaren Ablauf meist rasch erfassen. Sie sind daher gut für erste Entwürfe, Schulungsunterlagen, Arbeitsanweisungen oder die Kommunikation mit fachfremden Zielgruppen geeignet. Bei umfangreichen Geschäftsprozessen stoßen sie jedoch an Grenzen. Da ihre Notation nur begrenzt standardisiert ist, können komplexe Ereignisse, Verantwortlichkeiten, Nachrichten oder parallele Abläufe uneinheitlich und missverständlich dargestellt werden.

### 5.2 Ereignisgesteuerte Prozesskette (EPK)

Die EPK wurde speziell für die fachliche Modellierung von Geschäftsprozessen entwickelt. Durch den Wechsel von Ereignissen und Funktionen wird deutlich, welcher Zustand eine Tätigkeit auslöst und welches Ergebnis daraus entsteht. UND-, ODER- und XOR-Konnektoren ermöglichen außerdem die

Darstellung von Verzweigungen und parallelen Abläufen. Die EPK ist besonders geeignet, wenn Prozesse aus organisatorischer und fachlicher Sicht analysiert oder dokumentiert werden sollen, etwa im Prozessmanagement oder im Umfeld von ARIS. Ihre Schwäche liegt in der begrenzten internationalen Verbreitung und darin, dass technische Ausführungsdetails, Nachrichtenflüsse und komplexe Zusammenarbeit weniger präzise abgebildet werden können als mit BPMN.

### **5.3 Business Process Model and Notation (BPMN)**

BPMN ist ein international etablierter Standard für die Modellierung von Geschäftsprozessen. Die Notation kann sowohl für verständliche fachliche Übersichtsmodelle als auch für detaillierte, technisch orientierte Prozessmodelle verwendet werden. Pools und Lanes verdeutlichen Verantwortlichkeiten, Nachrichtenflüsse zeigen die Kommunikation zwischen Beteiligten, und verschiedene Ereignis- und Aufgabentypen ermöglichen eine präzise Beschreibung komplexer Prozesse. BPMN eignet sich daher besonders für abteilungs- oder organisationsübergreifende Prozesse sowie als Grundlage für Workflow- und Prozessautomatisierung. Die große Ausdrucksstärke ist zugleich eine Schwäche: Detaillierte BPMN-Modelle können ohne ausreichende Notationskenntnisse schwer verständlich werden und einen höheren Modellierungs- und Pflegeaufwand verursachen.

### **5.4 Decision Model and Notation (DMN)**

DMN ergänzt die Prozessmodellierung um die strukturierte Darstellung betrieblicher Entscheidungen. Entscheidungsanforderungsdiagramme zeigen, welche Informationen und Teilentscheidungen für ein Ergebnis erforderlich sind; Entscheidungstabellen legen die zugehörigen Regeln eindeutig und prüfbar fest. DMN ist besonders geeignet für regelbasierte Entscheidungen, beispielsweise bei der Kreditprüfung, Tarifberechnung oder Anspruchsprüfung. Die Trennung von Prozessablauf und Entscheidungslogik erleichtert die Wartung und Wiederverwendung von Regeln. DMN ist jedoch keine Sprache zur vollständigen Darstellung von Prozessabläufen oder Unternehmensarchitekturen. Bei Ermessensentscheidungen, unvollständigen Informationen oder stark qualitativen Kriterien lässt sich die Realität außerdem nur eingeschränkt in eindeutige Regeln überführen.

### **5.5 ArchiMate**

ArchiMate dient der integrierten Modellierung von Unternehmensarchitekturen. Die Sprache verbindet strategische Ziele und Fähigkeiten mit Geschäftsprozessen, Anwendungen, Daten, Technologien sowie Vorhaben zur Umsetzung und Migration. Ihre besondere Stärke liegt in der ebenenübergreifenden Nachvollziehbarkeit: So kann etwa gezeigt werden, welche strategische Anforderung einen Geschäftsprozess verändert, welche Anwendung diesen unterstützt und welche technische Infrastruktur dafür benötigt wird. ArchiMate eignet sich deshalb besonders für Enterprise-Architecture-Management, Transformationsvorhaben, Wirkungsanalysen und Roadmaps. Für die detaillierte Modellierung einzelner Prozessabläufe, Entscheidungen oder technischer Implementierungen ist die Sprache hingegen weniger geeignet. Umfangreiche ArchiMate-Modelle können zudem abstrakt wirken und erfordern eine sorgfältige Auswahl der Elemente und Sichten.

## 5.6 Vorteile professioneller Modellierungswerkzeuge

Professionelle Modellierungswerkzeuge dienen nicht nur dem grafischen Zeichnen von Modellen, sondern verwalten die verwendeten Modellelemente in einem zentralen Repository. Dadurch können beispielsweise Funktionen, Ereignisse, Rollen, Organisationseinheiten oder Anwendungssysteme in unterschiedlichen EPK- oder BPMN-Modellen einheitlich verwendet und miteinander verknüpft werden. Änderungen an einem Objekt lassen sich zentral nachvollziehen und müssen nicht in jedem Modell einzeln vorgenommen werden. Auf Basis des Repositorys können außerdem Auswertungen, Berichte, Abhängigkeitsanalysen und Konsistenzprüfungen durchgeführt werden. BPMN wird darüber hinaus häufig in Digitalisierungs- und Prozessautomatisierungsplattformen eingesetzt. Dort kann ein entsprechend detailliertes und technisch ergänztes Prozessmodell als Grundlage für die automatisierte Ausführung und Steuerung von Workflows dienen. Das Modell beschreibt somit nicht nur den Prozess, sondern kann – ergänzt um Formulare, Geschäftsregeln, Schnittstellen und technische Konfigurationen – unmittelbar zur Digitalisierung des Prozessablaufs beitragen.

## 5.7 Übersicht der vorgestellten Modellierungssprachen

| Sprache       | Schwerpunkt und geeigneter Zweck                                                   | Stärken                                                                                          | Grenzen                                                                                                  |
|---------------|------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| Flussdiagramm | Einfache Abläufe, Algorithmen, Entscheidungswege und erste Entwürfe                | Leicht verständlich; schnell zu erstellen; geringe Einstiegshürde                                | Begrenzt standardisiert; für komplexe Prozesse und Verantwortlichkeiten wenig präzise                    |
| EPK           | Fachliche Analyse und Dokumentation von Geschäftsprozessen                         | Klarer Ereignis-Funktions-Bezug; gute Darstellung fachlicher Prozesslogik; ARIS-Nähe             | International weniger verbreitet; Nachrichten und technische Ausführungsdetails nur begrenzt darstellbar |
| BPMN          | Standardisierte, organisationsübergreifende und automatisierbare Geschäftsprozesse | Hohe Ausdrucksstärke; klare Verantwortlichkeiten und Nachrichtenflüsse; internationaler Standard | Höherer Lern-, Modellierungs- und Pflegeaufwand; detaillierte Modelle können unübersichtlich werden      |
| DMN           | Nachvollziehbare Darstellung und Automatisierung regelbasierter Entscheidungen     | Klare Entscheidungslogik; prüfbare Regeln; gute Ergänzung zu BPMN                                | Keine vollständige Prozesssprache; begrenzt geeignet für qualitative oder intuitive Entscheidungen       |
| ArchiMate     | Unternehmensarchitektur, Transformation, Wirkungsanalyse und Roadmaps              | Verbindet Strategie, Geschäft, Anwendungen und Technologie; unterstützt Traceability             | Hoher Abstraktionsgrad; nicht für detaillierte Prozesse oder Implementierungen gedacht                   |

**Tabelle 34: Übersicht über die vorgestellten Modellierungssprachen**

## Literaturverzeichnis

- Canva Pty Ltd. (30. 07 2026). *Entscheidungsbaum erstellen: Anleitung, Vorlagen & Tipps*. Von [https://www.canva.com/de\\_de/online-whiteboard/entscheidungsbaum/](https://www.canva.com/de_de/online-whiteboard/entscheidungsbaum/) abgerufen
- Freund, J., & Rücker, B. (kein Datum). *Praxishandbuch BPMN 2.0*. München: Carl Hanser Verlag GmbH.
- IBM. (30. 06 2026). *Was ist ein Datenflussdiagramm*. Von <https://www.ibm.com/de-de/think/topics/data-flow-diagram> abgerufen
- Keller, G. (1992). *Semantische Prozessmodellierung auf der Grundlage "Ereignisgesteuerter Prozessketten (EPK)*. Saarbrücken: Universität des Saarlandes - Institut für Wirtschaftsinformatik.
- Miro DACH GmbH. (30. 06 2026). *Miro*. Von <https://www.miro.com> abgerufen
- OMG - Object Management Group. (09. 07 2026). *OMG Standards Development Organization*. Von <https://www.omg.org/dmn/> abgerufen
- Osterwalder, A., & Pigneur, Y. (2011). *Business Model Generation*. Frankfurt am Main: Campus Verlag.
- Reisig. (2010). *Petrinetze - Modellierungstechnik, Analysemethoden, Fallstudien*. Wiesbaden: Vieweg + Teuber Verlag; Springer Fachmedien Wiesbaden GmbH.
- Schütte, R. (1997). Die neuen Grundsätze ordnungsmäßiger Modellierung. *Forschungsforum* . Leipzig.
- Software AG. (2019). *ARIS Methodenhandbuch*. Darmstadt: Eigenverlag.
- SPARX Systems Europe. (10. 07 2026). [www.sparxsystems.de](https://www.sparxsystems.de). Von [https://www.sparxsystems.de/sprachen/archimate/?gad\\_source=1&gad\\_campaignid=21794505552&gclid=EAIaIQobChMIzuHO0NHHLQMVOvF5BB0XYRokEAMYAiAAEgI\\_xfD\\_BwE](https://www.sparxsystems.de/sprachen/archimate/?gad_source=1&gad_campaignid=21794505552&gclid=EAIaIQobChMIzuHO0NHHLQMVOvF5BB0XYRokEAMYAiAAEgI_xfD_BwE) abgerufen
- The Open Group. (2026). *ArchiMate(R) 4 Specification*. The Open Group Standard.

# Abbildungsverzeichnis

|                                                                                                                                                   |    |
|---------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Abbildung 1: Zweckbezogene Ableitung der Grundsätze ordnungsmäßiger Modellierung (Schütte, 1997).....                                             |    |
| Abbildung 2: Flussdiagramm – Prüfung Online-Buchung.....                                                                                          |    |
| Abbildung 3: Flussdiagramm - Prüfung Online-Buchung - horizontale Leserichtung .....                                                              |    |
| Abbildung 4: Ein vereinfachtes logisches Datenflussdiagramm, das den Bestellvorgang auf einer E-Commerce-Website veranschaulicht (IBM, 2026)..... | 22 |
| Abbildung 5: Website Flussdiagramm (Miro DACH GmbH., 2026).....                                                                                   |    |
| Abbildung 6: Entscheidungsbaum (Canva Pty Ltd, 2026).....                                                                                         | 23 |
| Abbildung 7: Ereignisse -grafische Darstellung (Software AG, 2019) .....                                                                          |    |
| Abbildung 8: Beispiel einer EPK (Software AG, 2019).....                                                                                          |    |
| Abbildung 9: UND – Regel (Software AG, 2019) .....                                                                                                |    |
| Abbildung 10: ODER – Regel (Software AG, 2019) .....                                                                                              |    |
| Abbildung 11: Exklusiv - ODER - Verknüpfung (XOR) (Software AG, 2019).....                                                                        |    |
| Abbildung 12: UND - Verknüpfung erzeugter Ereignisse (Software AG, 2019).....                                                                     |    |
| Abbildung 13: ODER - Verknüpfung erzeugter Ereignisse (Software AG, 2019) .....                                                                   |    |
| Abbildung 14: Exklusiv - oder - Verknüpfung XOR erzeugter Ereignisse (Software AG, 2019) .....                                                    |    |
| Abbildung 15: EPK - Zusammenführen mehrerer paralleler Kontrollflüsse.....                                                                        |    |
| Abbildung 16: Subprozess Aufruf nach Prozessdurchlauf.....                                                                                        |    |
| Abbildung 17: Subprozess Aufruf innerhalb des Prozessflusses .....                                                                                |    |
| Abbildung 18: BPMN Flussobjekte .....                                                                                                             |    |
| Abbildung 19: BPMN Sub-Prozess Aufruf (Object Management Group).....                                                                              |    |
| Abbildung 20: BPMN - Event Typen.....                                                                                                             |    |
| Abbildung 21: BPMN Exclusives Gateway.....                                                                                                        |    |
| Abbildung 22: BPMN - Inklusives Gateway.....                                                                                                      |    |
| Abbildung 23: BPMN - Paralleles Gateway .....                                                                                                     |    |
| Abbildung 24: BPMN - komplexes Gateway mit zwei aus fünf möglichen Aktivitäten .....                                                              |    |
| Abbildung 25: BPMN - Gruppierung.....                                                                                                             |    |
| Abbildung 26: BPMN - Pool mit Lanes .....                                                                                                         |    |
| Abbildung 27: Mehrere Pools und Nachrichtenflüsse.....                                                                                            |    |
| Abbildung 28: BPMN - Beispiel Kaffeeküche .....                                                                                                   |    |

|                                                                                                                   |    |
|-------------------------------------------------------------------------------------------------------------------|----|
| Abbildung 29: Grafische Darstellung als DMN Modell.....                                                           |    |
| Abbildung 30: Grafische Darstellung mit mehreren Entscheidungen – DMN Modell.....                                 |    |
| Abbildung 31: ArchiMate Kernelemente (The Open Group, 2026, S 123).....                                           |    |
| Abbildung 32: Motivation-, Strategie-, Implementierungs- und Migrationselemente (The Open Group, 2026, S124)..... |    |
| Abbildung 33: ArchiMate Beziehungen und Verknüpfungspunkte (The Open Group, 2026)S 124.....                       |    |
| Abbildung 34: Verwendete Objekte in der ARIS-spezifischen Darstellungsform .....                                  |    |
| Abbildung 35: Strategische Sicht der digitalen Kursanmeldung (Erstellt mit Aris Architect 10) .....               | 66 |
| Abbildung 36: Motivations-Modell.....                                                                             |    |
| Abbildung 37: Geschäftsarchitektur-Modell.....                                                                    |    |
| Abbildung 38: Application Modell.....                                                                             |    |
| Abbildung 39: Technologie- und physisches Modell .....                                                            |    |
| Abbildung 40: Umsetzungs- und Migrations-Modell .....                                                             |    |
| Abbildung 41: Motivation mit Strategie verbinden.....                                                             |    |
| Abbildung 42: Motivation mit Business verbinden .....                                                             |    |
| Abbildung 43: Strategie mit Business verbinden.....                                                               |    |
| Abbildung 44: Application mit Business verbinden.....                                                             |    |
| Abbildung 45: Technologie mit Application verbinden .....                                                         |    |
| Abbildung 46: Umsetzung mit Architektur verbinden .....                                                           |    |

## Tabellenverzeichnis

|                                                                             |    |
|-----------------------------------------------------------------------------|----|
| Tabelle 1: Einfaches Modell eines Smartphones .....                         | 8  |
| Tabelle 2: Symbole in Flussdiagrammen.....                                  | 18 |
| Tabelle 3: BPMN Aktivitäten mit Ausprägung (Omject Management Group).....   | 32 |
| Tabelle 4:BPMN Eventauslöser .....                                          | 34 |
| Tabelle 5: Beispiel - Eingabedaten.....                                     | 42 |
| Tabelle 6:DMN-Entscheidungstabelle – Aufbau .....                           | 42 |
| Tabelle 7: DMN-Entscheidungstabelle .....                                   | 43 |
| Tabelle 8: DMN Entscheidungstabelle mit ODER Verknüpfung .....              | 44 |
| Tabelle 9: DRD - Objekte.....                                               | 46 |
| Tabelle 10: DMN Elemente für Beispiel.....                                  | 46 |
| Tabelle 11: Kundenrabatt bestimmen - interne Entscheidungstabelle.....      | 48 |
| Tabelle 12: Eingabedaten zur Erweiterung des Beispiels .....                | 48 |
| Tabelle 13: Kundenkategorie bestimmen.....                                  | 48 |
| Tabelle 14: Kundenrabatt bestimmen.....                                     | 49 |
| Tabelle 15: Welche Fragen werden durch das DMN beantwortet?.....            | 50 |
| Tabelle 16: Typische Fehler bei der Erstellung von DMN Modellen .....       | 50 |
| Tabelle 17: ArchiMate Ebenen (SPARX Systems Europe, 2026) .....             |    |
| Tabelle 18: Beziehungen richtig auswählen .....                             | 59 |
| Tabelle 19: Fragen zu Architektur & Scope .....                             | 60 |
| Tabelle 20: Benötigter Elemente-Katalog.....                                | 61 |
| Tabelle 21: Zusammenhang zwischen Ebenen, Aspekten und Motivation .....     | 62 |
| Tabelle 22: Wo liegt das Element .....                                      | 64 |
| Tabelle 23: Die Aspekte geben an, welche Rolle ein Element hat .....        | 64 |
| Tabelle 24: : Grundmuster der ArchiMate Matrix .....                        | 65 |
| Tabelle 25: Strategie Modell - Elemente aus unserem Beispiel .....          | 66 |
| Tabelle 26: Motivations Modell: Elemente aus unserem Beispiel.....          | 67 |
| Tabelle 27: Geschäftsarchitektur Modell: Elemente aus unserem Beispiel..... | 68 |
| Tabelle 28: Geschäftsarchitektur Modell: Elemente aus unserem Beispiel..... | 68 |

|                                                                                    |    |
|------------------------------------------------------------------------------------|----|
| Tabelle 29: Application Modell: Elemente aus unserem Beispiel.....                 | 69 |
| Tabelle 30: Technologie und physisches Modell: Elemente aus unserem Beispiel.....  | 70 |
| Tabelle 31: Umsetzungs- und Migrations-Modell: Elemente aus unserem Beispiel ..... | 71 |
| Tabelle 32: Mögliche Verbindungen zwischen den Ebenen .....                        | 73 |
| Tabelle 33: Mögliche Analysefälle aus der Praxis .....                             | 76 |
| Tabelle 34: Übersicht über die vorgestellten Modellierungssprachen .....           | 80 |