

# TUM

INSTITUT FÜR INFORMATIK

Das V-Modell XT 1.3 Metamodell

Thomas Ternité, Marco Kuhrmann



TUM-I0905  
Februar 09

TECHNISCHE UNIVERSITÄT MÜNCHEN

TUM-INFO-02-I0905-0/1.-FI  
Alle Rechte vorbehalten  
Nachdruck auch auszugsweise verboten

©2009

Druck: Institut für Informatik der  
Technischen Universität München

# Das V-Modell XT 1.3 Metamodell

## Kommentierte und erweiterte Fassung

Thomas Ternité<sup>1</sup>, Marco Kuhrmann<sup>2</sup>

<sup>1</sup>Technische Universität Clausthal  
Institut für Informatik, Software Systems Engineering  
Julius-Albert-Str. 4  
38678 Clausthal-Zellerfeld, Germany

<sup>2</sup>Technische Universität München  
Institut für Informatik, Software & Systems Engineering  
Boltzmannstr. 3  
85748 Garching, Germany

*thomas.ternite@tu-clausthal.de, kuhrmann@in.tum.de*

## Zusammenfassung

Dieser Bericht beschreibt das Metamodell des V-Modell XT. Unter einem Metamodell verstehen wir dabei die grundlegende Strukturierung der Beschreibungselemente des V-Modell XT, beispielsweise in Form der Beschreibungselemente *Vorgehensbaustein*, *Produkt* oder *Aktivität* mitsamt aller Abhängigkeiten zwischen diesen Elementen. Die grundlegende Strukturierung spiegelt sich unmittelbar in der V-Modell XT XML-Datei und im V-Modell XT Editor wider. Die PDF beziehungsweise HTML-Version des V-Modells gehen durch strukturelle Transformation aus der in diesem Bericht beschriebenen Struktur hervor. Dieser Bericht ist eine erweiterte und kommentierte Fassung der Metamodellokumentation, die dem V-Modell XT standardmäßig beiliegt.

## Keywords

Software Engineering, Process Engineering, Software Development Process, V-Modell XT, XML

**CR-Classification:** D.2

## **Danksagung**

Das V-Modell XT wurde im Projekt *Weit* (Weiterentwicklung des IT-Entwicklungsstandards des Bundes) entwickelt. Die grundlegende Überarbeitung des V-Modell XT Metamodells, das in diesem Bericht beschrieben wird, wurde in der Projektphase *Weit-IV* Stufe 2 in den Jahren 2007 und 2008 vorgenommen. An diesem Teilprojekt und auch an der Erstellung der Metamodelldokumentation wirkten weiterhin mit: Andreas Rausch, Ulrike Hammerschall, Jan Friedrich, Marc Sihling, Edward Fischer, Michael Deynet. Ihnen allen sei an dieser Stelle gedankt.

# Inhalt

1	Einleitung .....	1
1.1	Zielgruppen .....	1
1.2	Aufbau des Dokuments .....	1
2	Das V-Modell XT Metamodell .....	2
2.1	Überblick über die Pakete .....	2
2.2	Das Paket Basis .....	3
2.2.1	Querschnittliche Modellattribute .....	7
2.2.2	Nummerierung .....	8
2.3	Das Paket Statik .....	9
2.3.1	Submodelle .....	9
2.3.2	Vorgehensbausteine .....	9
2.3.3	Das Produktmodell .....	11
2.3.4	Produktabhängigkeiten .....	15
2.3.5	Das Aktivitätsmodell .....	16
2.3.6	Das Rollenmodell .....	18
2.3.7	Beziehungen .....	18
2.4	Das Paket Dynamik .....	25
2.4.1	PDS-Spezifikation .....	26
2.4.2	Ablaufbaustein .....	26
2.4.3	Ablaufbausteinspezifikation .....	32
2.4.4	Projekttypvariante .....	33
2.4.5	Entscheidungspunkt .....	34
2.5	Das Paket Anpassung .....	34
2.5.1	Projekttypen .....	35
2.5.2	Projekttypvarianten .....	36
2.5.3	Projektmerkmale und Projektmerkmalswerte .....	37
2.6	Das Paket Konventionsabbildung .....	38
3	Anpassungsmöglichkeiten des V-Modell XT .....	39
3.1	Aufteilung in Referenz- und Organisationsanteile .....	39
3.2	Erweiterungen des V-Modell XT .....	40
3.3	Anpassungen und Änderungen des V-Modell XT .....	41
3.3.1	Disziplinänderungen .....	42
3.3.2	Produktänderungen .....	43
3.3.3	Themenänderungen .....	45
3.3.4	Aktivitätenänderungen .....	46
3.3.5	Rollenänderungen .....	46
3.3.6	Tailoringänderungen .....	47
3.3.7	Entscheidungspunktänderungen .....	48
3.3.8	Beschreibungstextersetzungen .....	49
3.3.9	Beschreibungstextergänzungen .....	50
3.3.10	Beschreibungstextverschiebungen .....	51
3.3.11	Werkzeug-/Methodenänderungen .....	51
3.4	Vortailoring .....	52
4	Tailoring .....	52
4.1	Der Tailoring-Prozess .....	52
4.2	Zusammenhänge Tailoring – Metamodell .....	54
4.3	Konsequenzen für die Modellierung von Erweiterungen .....	55
5	Konsistenzbedingungen .....	56
6	Abkürzungsverzeichnis .....	58
7	Abbildungsverzeichnis .....	59
8	Referenzen .....	60



# 1 Einleitung

Das V-Modell XT (kurz V-Modell) ist das Standardvorgehensmodell der Bundesrepublik Deutschland im Bereich der Entwicklung von IT-Systemen. Neben inhaltlichen Besonderheiten, auf die z.B. [2] eingeht, basiert es *vollständig* auf einem formalen Metamodell. Dieses Metamodell stellt einen elementaren Baustein in der Flexibilität des V-Modells dar. Es definiert die grundlegenden Strukturen, die insbesondere für *Prozessingenieure* relevant sind, um entweder Anpassungen des V-Modells vorzunehmen oder neue Prozesse und Methoden auf Basis der V-Modell-Strukturen zu erstellen.

Dieser Bericht beschreibt das Metamodell des V-Modells. Unter einem Metamodell verstehen wir dabei die grundlegende Strukturierung der Beschreibungselemente des V-Modells, beispielsweise in Form der Beschreibungselemente *Vorgehensbaustein*, *Produkt* oder *Aktivität* mitsamt allen Abhängigkeiten zwischen diesen Elementen. Die grundlegende Strukturierung spiegelt sich unmittelbar in der V-Modell XT XML-Datei und im V-Modell XT Editor wider. Die PDF beziehungsweise HTML-Version des V-Modells gehen durch strukturelle Transformation aus der in diesem Bericht beschriebenen Struktur hervor.

**Hinweis:** Dieser Bericht ist eine erweiterte und kommentierte Fassung der Metamodellokumentation, die dem V-Modell standardmäßig beiliegt und über die V-Modell Webseite [1] verfügbar ist.

## 1.1 Zielgruppen

Zielgruppen dieses Berichts sind die Autoren des V-Modells selbst, sowie die Autoren organisationspezifischer Anpassungen des V-Modells bzw. von neuen Prozessen und Methoden auf Basis des V-Modells (im Folgenden immer als *Prozessingenieur* oder *PING* bezeichnet). Darüber hinaus ist die vorliegende Dokumentation für Werkzeughersteller von Interesse. Für sie bietet sich die Verwendung der V-Modell XT XML-Datei als Informationsbasis an. Der vorliegende Bericht beschreibt auch deren Aufbau.

Voraussetzung für das Lesen dieses Dokuments ist das Verständnis der Grundkonzepte des V-Modells, die im Teil 1 des V-Modells [1] oder in [2] beschrieben sind. Weiterhin sollten Kenntnisse in XML [4] und XML-Schema [5] vorhanden sein.

## 1.2 Aufbau

Dieser Bericht ist wie folgt aufgebaut: Kapitel 2 beschreibt die Strukturen des Metamodells, anhand einer Paket-artigen Gruppierung. In Kapitel 3 werden die Anpassungsoperationen des V-Modells beschrieben. Kapitel 4 beschreibt das Tailoring. Kapitel 5 enthält abschließend die Konsistenzbedingungen des Metamodells.

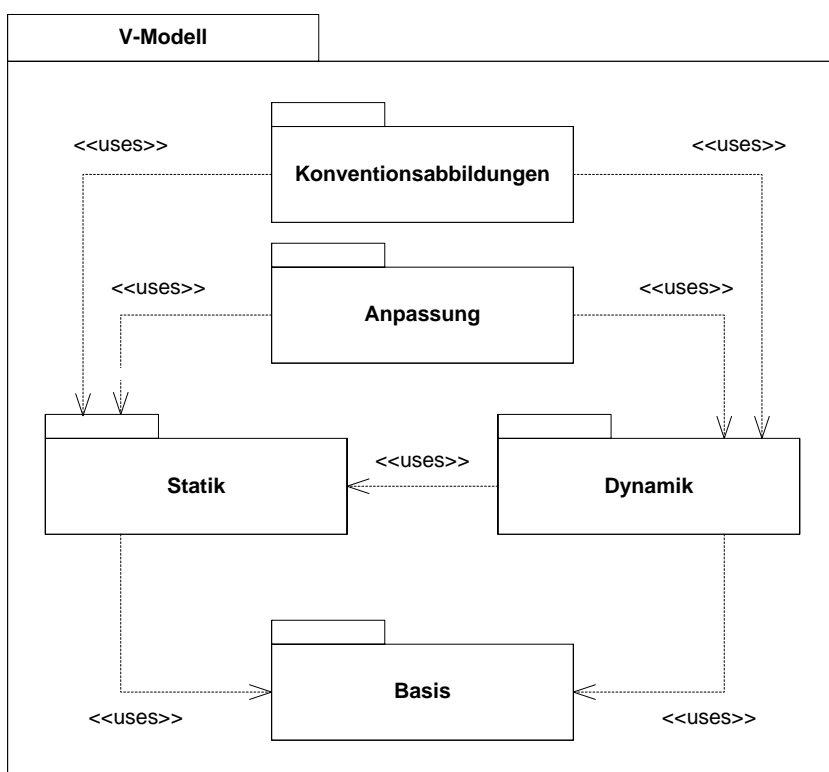
## 2 Das V-Modell XT Metamodell

Das Metamodell des V-Modells liegt in Form eines UML-Klassenmodells vor. Für die Beschreibung und das strukturelle Verständnis des Metamodells ist diese Beschreibung nicht immer optimal geeignet. Dieser Bericht bietet die Kernelemente als UML-Modell an, orientiert sich in der detaillierten Beschreibung aber an den Strukturen des XML-Schemas, welches das Metamodell technisch umsetzt.

Kapitel 2.1 gibt zunächst einen Überblick über eine Strukturierung des Metamodells in Form von UML-Paketen. Es beschreibt die in den Paketen enthaltenen Beschreibungselemente, ohne dabei auf die Zusammenhänge der Pakete einzugehen. Die einzelnen Pakete samt Inhalten werden in den folgenden Abschnitten detailliert beschrieben.

### 2.1 Überblick über die Pakete

Das Metamodell lässt sich in die in Abbildung 1 dargestellten Pakete *Basis*, *Statik*, *Dynamik*, *Anpassung* und *Konventionsabbildungen* strukturieren. Diese Pakete organisieren die Modellelemente des V-Modells und vereinfachen die Orientierung im Metamodell.



**Abbildung 1 Überblick über die Paketstruktur des Metamodells**

Das Paket *Basis* beschreibt den Aufbau der (Prozess-)Dokumentation und enthält Elemente, die außerhalb von fachlich begründeten Vorgehensbausteinen gekapselt werden, z.B. Elemente, die von den Paketen *Statik* und *Dynamik* referenziert werden.

Das Paket *Statik* beschreibt die Struktur von Vorgehensbausteinen und die darin enthaltenen Elementtypen. Damit enthält es die wesentlichen Grundelemente des V-Modells, mit denen die Vorgaben des Vorgehensmodells beschrieben werden.

Das Paket *Dynamik* beschreibt die Struktur aller Regeln und Elemente zur Konstruktion von Abläufen und Projektdurchführungsstrategien. Damit werden die zeitlichen Vorgaben und Spielräume des Vorgehensmodells erfasst, die die Reihenfolge der zu erstellenden Ergebnisse regeln.

Das Paket *Anpassung* dient dem Zweck, dem Anwender des V-Modells die Möglichkeit zu bieten, den Umfang des Vorgehensmodells auf seine Ansprüche hin anzupassen und insbesondere zu reduzieren. Hierzu soll der Anwender keine feingranularen Entscheidungen treffen müssen, sondern das durchzuführende Projekt z.B. anhand eines Projekttyps und konkreter Projektmerkmale zu charakterisieren.



Das Paket *Konventionsabbildungen* ist von den übrigen Paketen abhängig. Es beschreibt eine Abbildung der Begriffe aus anderen Konventionen auf die Konzepte des V-Modells. Das Paket referenziert daher potenziell die übrigen statischen und dynamischen Beschreibungselemente des Metamodells.

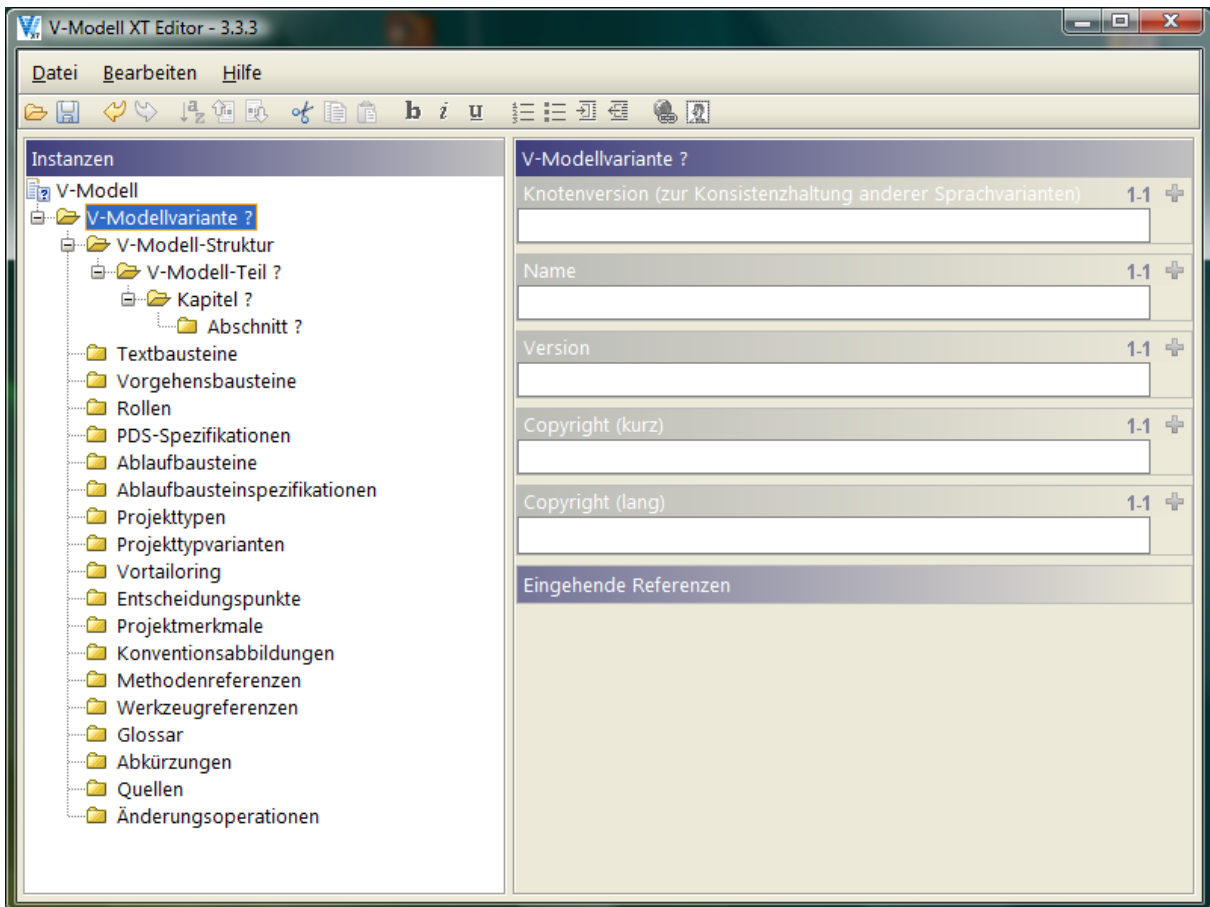
**Hinweis:** Die Paketstruktur ist ein logisches Konstrukt, das das Verständnis des Metamodells unterstützen soll. In der physischen Ausprägung finden sich diese Paketstrukturen nicht unmittelbar wieder. Entsprechende Zuordnungen lassen sich jedoch vornehmen.

## 2.2 Das Paket Basis

Das Paket Basis beschreibt die Basisstruktur und den prinzipiellen den Aufbau der Dokumentation und enthält Elemente, die außerhalb von fachlich begründeten Vorgehensbausteinen gekapselt werden, z.B. Elemente die von den Paketen *Statik* und *Dynamik* referenziert werden.

Element	Beschreibung
V-Modell-Struktur	Die V-Modell-Struktur modelliert den Aufbau und die Inhalte der V-Modell XT Prozessdokumentation. Diese Dokumentation ist untergliedert in verschiedene Teile, Kapitel und Abschnitte.
V-Modell-Teil	Ein V-Modell-Teil ist ein Teil der Dokumentation einer V-Modell-Variante. Ein V-Modell-Teil besteht aus mehreren Kapiteln.
Kapitel	Ein Kapitel ist ein Strukturierungselement eines V-Modell-Teils und kann weiter in Abschnitte untergliedert werden.
Abschnitt	Ein Abschnitt ist ein Teil eines Kapitels.
Anhang	Der Anhang ist ein spezialisierter V-Modell-Teil und besteht aus Methoden- und Werkzeugreferenzen, Glossar, Abkürzungen und Quellen.
Methodenreferenz	Methodenreferenzen fassen verschiedene Methoden des Software und Systems Engineering zusammen und beschreiben diese. Methoden können V-Modell-Modellelementen zugeordnet werden und eine detaillierte Arbeitsanweisung als Verfeinerung zum Aktivitätsmodell bereitstellen.
Werkzeugreferenz	Werkzeugreferenzen fassen Werkzeuge zusammen, die die Erarbeitung von V-Modell-Produkten und/oder die Durchführung von Aktivitäten unterstützen. Sie können V-Modell-Modellelementen zugewiesen werden und geben somit eine Hilfestellung als Verfeinerung zu den Inhalten z.B. der Produkt- oder Aktivitätsbeschreibungen.
Begriff	(Glossar-)Begriffe definieren ein Glossar. Diese können von Elementen aus den Paketen <i>Statik</i> und <i>Dynamik</i> referenziert werden.
Abkürzung	Abkürzungen dienen der Definition eines Abkürzungsverzeichnisses im V-Modell. Diese können von Elementen aus den Paketen <i>Statik</i> und <i>Dynamik</i> referenziert werden.
Quelle	Quellen legen ein Quellenverzeichnis fest. Diese können von Elementen aus den Paketen <i>Statik</i> und <i>Dynamik</i> referenziert werden.
V-Modell	V-Modell ist der sogenannte „root“-Knoten. Unterhalb dieses Elements sind <b>alle</b> V-Modellvarianten lokalisiert.
V-Modellvariante	Eine V-Modellvariante fasst alle Elemente eines V-Modells (Vorgehensbausteine, Projekttypen, Rollen etc.) zusammen. Sie ist auch Gegenstand der Anpassung. Mehrere V-Modellvarianten können im Rahmen einer organisationsspezifischen Anpassung zu einem organisationsspezifischen V-Modell zusammengefasst werden. Das Referenzmodell ist selbst als V-Modellvariante entworfen.

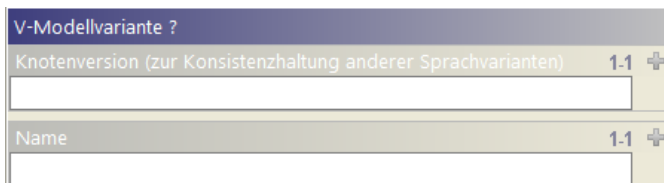
Diese Basisstruktur des V-Modells ist für Prozessingenieure sichtbar, wenn mit dem V-Modell Editor ein neues Dokument auf Basis des V-Modell XT Metamodells (hinterlegt in der Datei: *V-Modell-XT-Metamodell.xsd*) anlegt. Diese Sicht ist in Abbildung 2 zu sehen. Auch die oben angesprochene, nicht widerspiegelte Paketorientierung wird dort deutlich.



**Abbildung 2** Basisstruktur des V-Modell XT als XML-Baum im V-Modell XT Editor

In den folgenden Abschnitten werden die einzelnen Elemente dieser XML- bzw. Metamodellstruktur detailliert in Form der XML-Schemata vorgestellt. Die diskutierten Abbildungen sind jeweils Aspekte, die der *V-Modell-XT-Metamodell.xsd* direkt entnommen wurden.

Im Editor ist neben den einzelnen Feldern und Masken für Dateneingabe gleichzeitig auch die Semantik der jeweiligen Elemente sichtbar. Im rechten Bereich des Fensters sind die Eigenschaften des XML-Knotens für die V-Modellvariante gezeigt.



In dieser Maske sind Eingabefelder zu finden, die die Wertebelegung ermöglichen und die Anforderungen des Metamodells anzeigen. Die Eingabefelder haben dabei eine Bezeichnung, die dem Datenelement des Metamodells entspricht, für das ein Wert vergeben werden kann oder muss. Ob ein Wert festgelegt werden muss oder nicht, zeigt die Kardinalität am rechten Rand des Eingabefelds an. Folgende Semantik liegt dem zugrunde:

- 0-1: Dieses Feld ist optional und kann entweder genau einen oder keinen Wert erhalten.
- 1-1: Dies ist ein Pflichtfeld, das mit genau einem Wert belegt werden muss.
- 0-\*: Dieses Feld ist optional und kann keinen oder beliebig viele Werte erhalten.
- 1-\*: Dieses Feld ist ein Pflichtfeld, dem mindestens ein Wert zugewiesen werden muss, dem aber auch mehrere Werte zugewiesen werden können.

Eine detaillierte Erläuterung des Editors und seiner Benutzung findet sich in der Anwenderdokumentation, die ebenfalls über die V-Modell-Webseite [1] verfügbar ist.

**V-Modell-Struktur.** Die V-Modell-Struktur (Abbildung 3) dient dazu, die Prozessdokumentation aus den Inhalten des V-Modells zusammenzustellen.

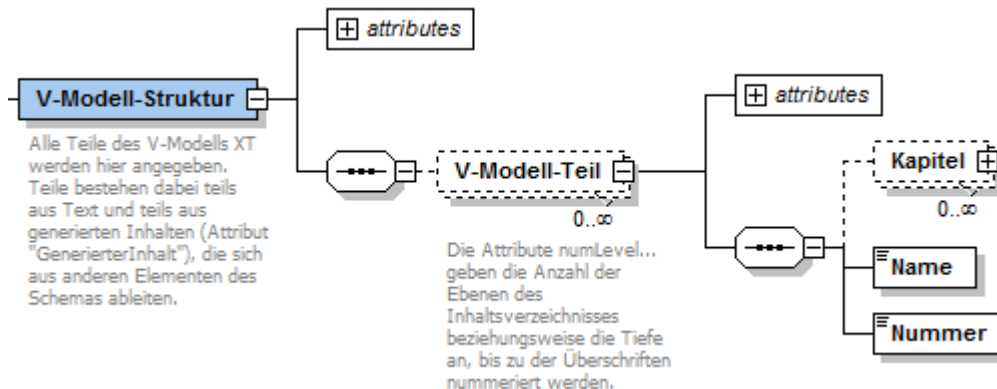


Abbildung 3 Das Basis-Paket, Bereich V-Modell-Struktur und Teile (Sicht: XML-Schema)

Dazu ist das V-Modell (genauer dessen Repräsentation in Form einer Prozessdokumentation) in verschiedene Teile gegliedert (standardmäßig sind dies die 9 Teile, z.B. „Teil 3 – Referenz Tailoring“). Über die Elemente *Name* und *Nummer* wird ein Name für den Dokumentationsteil vergeben und diesem Teil eine eindeutige Nummer zugeordnet (siehe vorangehendes Beispiel). Teile legen die groben Gliederungseinheiten des V-Modells, bzw. seiner Dokumentation, fest. Während des Tailorings können die V-Modell-Teile auch einzeln als Dokumentation in verschiedenen Formaten exportiert werden.

**Kapitel und Abschnitte.** Die „Nutzinformation“, also der Text, in den Teilen wird durch die Kapitel (Abbildung 4) erbracht. Kapitel und Abschnitte haben einen *Titel*, einen einleitenden *Text* und (optional) mehrere (*Unter-*)*Abschnitte*.

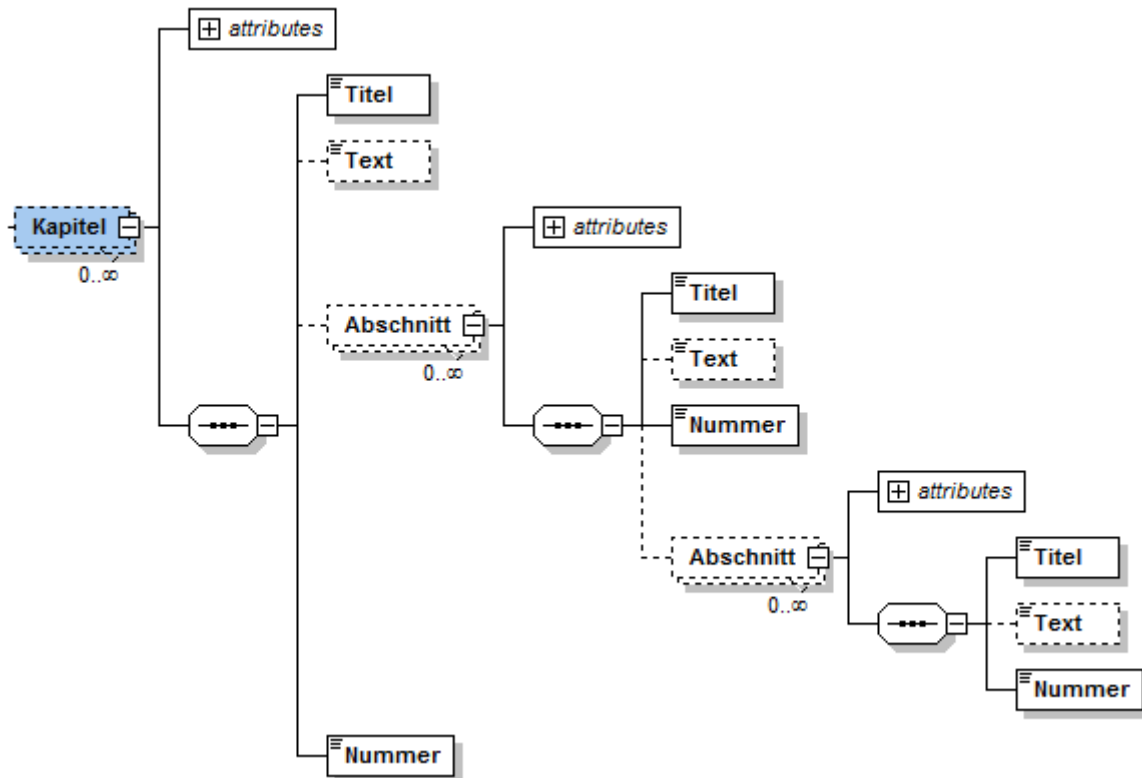
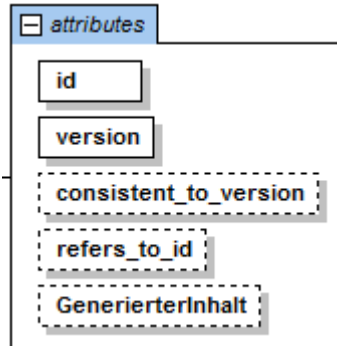


Abbildung 4 Das Basis-Paket, Bereich Kapitel und Abschnitt (Sicht: XML-Schema)

Die Prozessdokumentation, die in den Kapiteln und Abschnitten hinterlegt ist, kann entweder manuell mithilfe der Texteingabefunktionen des V-Modell XT Editors erstellt oder generiert werden. Im V-Modell sind alle Modellelemente (Produkte, Rollen, Aktivitäten etc.) *selbstdokumentierend*. Das heißt,

dass zu jedem Modellelement eine entsprechende Dokumentation bereits durch das betreffende Modellelement vorliegt. Soll eine Prozessdokumentation erstellt werden, müssen die betreffenden Dokumentationstexte nicht erneut erstellt werden. Sie können stattdessen aus den Modellelementen heraus generiert werden. Für Kapitel und Abschnitte gibt es daher zusätzlich zur Option der manuellen Texteingabe die Möglichkeit, mithilfe des Attributs *GenerierterInhalt* Modellelemente zu selektieren, und diese beim Export der Dokumentation als Quelle für die Beschreibungstexte heranzuziehen:



Betrachtet man z.B. den oben referenzierten Teil 3, so sind in der V-Modell-Struktur in diesem Teil die Beschreibungstexte für Vorgehensbausteine, Projekttypen etc. nicht redundant im Modell hinterlegt, sondern werden aus den betreffenden Modellelementen beim Export hierher generiert.

**Hinweis:** Dies zu verstehen ist wichtig, um zu verstehen, wie die konkrete Prozessdokumentation entsteht. Im Rahmen einer organisationsspezifischen Anpassung oder im Rahmen des Tailorings werden Modellinhalte ergänzt bzw. aus dem Modell entfernt. Diese Operationen sind durch die Selbstbeschreibung und die Generierung transparent. Wird z.B. im Rahmen einer organisationsspezifischen Anpassung ein Vorgehensbaustein ergänzt, sind keine weiteren Aktivitäten auszuführen, um diesen in die Prozessdokumentation mit aufzunehmen. Dies geschieht beim Export automatisch.

**Glossar, Abkürzungen und Quellen.** Für die Beschreibung des V-Modells oder die weitergehende Erläuterung von V-Modell Modellelementen und Inhalten stehen im Metamodell explizit Typen zur Modellierung von Glossaren, zur Auflistung von Abkürzungen und zur Sammlung von Literaturquellen (Abbildung 5) zur Verfügung.

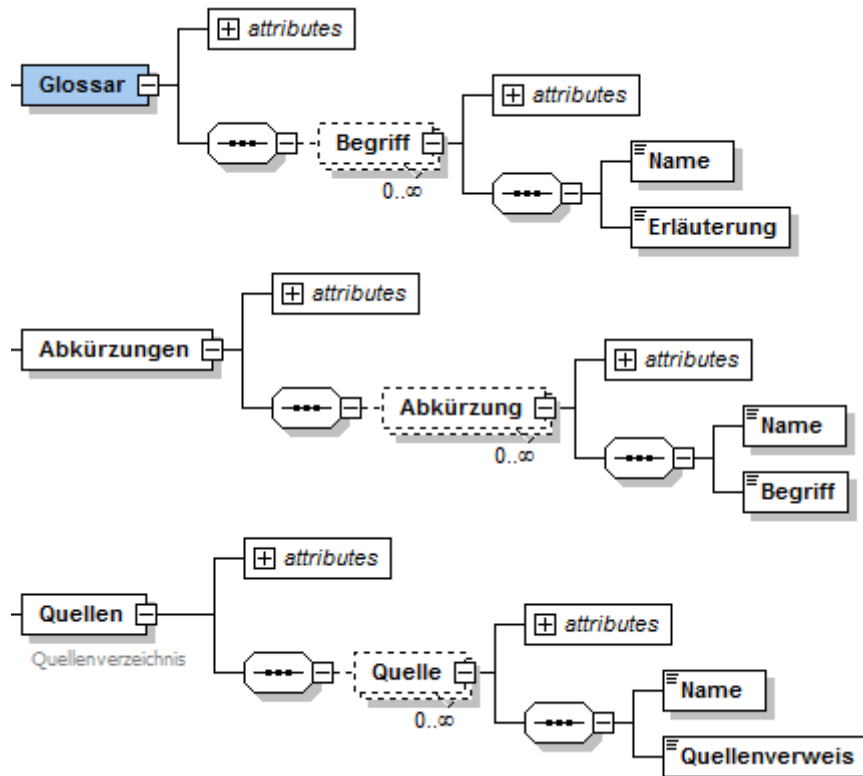


Abbildung 5 Das Basis-Paket, Bereiche: Glossar, Abkürzungen und Quellen (Sicht: XML-Schema)

Diese Modellelemente können befüllt und dann von verschiedenen anderen Modellelementen entweder explizit (per Modellreferenz) oder implizit (per Hyperlink) referenziert werden.

**Methodenreferenz.** Methodenreferenzen (Abbildung 6) dienen dazu, optionale oder detailliertere Verfahrensanweisungen im V-Modell zu hinterlegen. Methoden haben einen *Namen* und einen *Sinn\_und\_Zweck*, der die Methode selbst beschreibt. Zusätzlich können Methoden verschiedene Quellen per *QuelleRef* (siehe Abbildung 5) referenzieren. Methoden werden über explizite *Beziehungs*-Elemente (siehe Abschnitt 2.3.7) an V-Modell-Modellelemente gebunden.

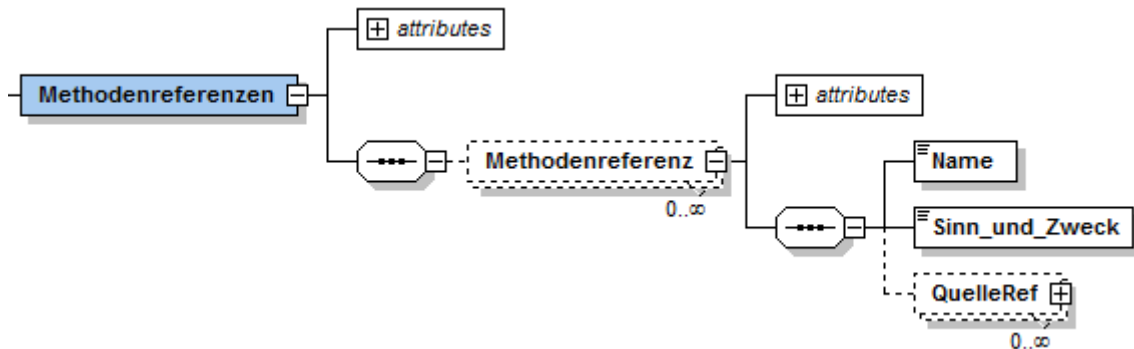


Abbildung 6 Das Basis-Paket, Bereich Methodenreferenz (Sicht: XML-Schema)

**Werkzeugreferenz.** Werkzeugreferenzen (Abbildung 7) dienen dazu, Optionen zur Unterstützung bei der Erarbeitung von V-Modell-Produkten oder bei der Durchführung von V-Modell-Aktivitäten, durch Werkzeuge aufzuzeigen und diese im Modell zu hinterlegen. Werkzeugreferenzen haben einen *Namen* und einen *Sinn\_und\_Zweck*, der das Werkzeug selbst bzw. seine Anwendung beschreibt. Zusätzlich können Werkzeuge verschiedene Quellen per *QuelleRef* (siehe Abbildung 5) referenzieren. Werkzeuge werden über explizite *Beziehungs*-Elemente (siehe Abschnitt 2.3.7) an V-Modell-Modellelemente gebunden.

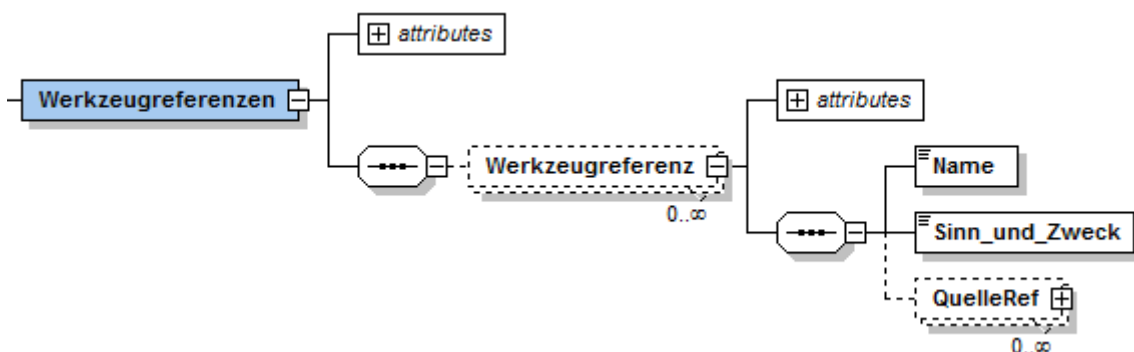


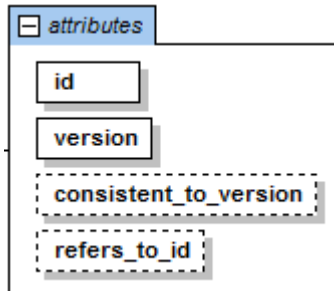
Abbildung 7 Das Basis-Paket, Bereich Werkzeugreferenz (Sicht: XML-Schema)

### 2.2.1 Querschnittliche Modellattribute

Weiter oben wurde bereits auf das Attribut *GenerierterInhalt* näher eingegangen. Neben diesem Attribut gibt es Attribute, die allen V-Modell Elementen, die referenzierbar sind, gemein sind. Im Detail sind das die Attribute *id*, *version*, *consistent\_to\_version* und *refers\_to\_id*.

- Das Attribut *id* wird mit einem eindeutigen Schlüsselwert belegt. Dies ist üblicherweise ein 16 Zeichen langer Hex-String, der vom Editor automatisch vergeben wird.
- Das Attribut *version* enthält als Wert eine Fließkommazahl, die die Versionsnummer des XML-Knotens angibt. Diese wird bei der Entwicklung des V-Modells benutzt, um die Internationalisierung des Modells zu vereinfachen.
- Die Attribute *consistent\_to\_version* und *refers\_to\_id* dienen der Unterstützung des V-Modells von verschiedenen Sprachen. Sie sind optional. Die deutsche Version des V-Modells ist dabei in der Regel der Master. Mithilfe von *consistent\_to\_version* wird eine Assoziation zwischen den verschiedenen Sprachversionen hergestellt. Das Attribut sagt dabei aus, zu welcher Knotenversion des Ursprungsmodells der vorliegende XML-Knoten konsistent ist. Mithilfe von *refers\_to\_id* wird eine Verknüpfung der Knoten selbst hergestellt, sodass ein Knoten z.B. im englischen V-Modell eindeutig seiner Entsprechung im deutschen V-Modell zugeordnet werden kann.

Diese Attribute sind in allen referenzierbaren Elementen des V-Modells zu finden:



Im V-Modell (in der XML-Datei) finden sich diese wie folgt wieder (Beispiel: Projekthandbuch):

```
<Produkt Initial="Ja" Extern="Nein" Produktvorlage="Nein" id="7d9bf684e9bf2c" version="2.70">
  <!-- Name Start 7d9bf684e9bf2c -->
  <Name>Projekthandbuch</Name><!-- 7d9bf684e9bf2c --> ...
```

## 2.2.2 Nummerierung

Nach Abbildung 4 sind für die Strukturierungselemente V-Modell-Teil, Kapitel und Abschnitt jeweils *Nummer*-Elemente zu vergeben. Die Vergabe der Nummern erfolgt nach einem bestimmten Schema. Die Nummern sollten 3-stellig aufgebaut werden. Fehlende Stellen sind durch führende Nullen aufzufüllen. Gültige Nummern sind z.B. „001“ oder „010“.

Die Vergabe der Nummern hat den Sinn, bei der Zusammenstellung der Dokumentation die Dokumentenhierarchien zu ermitteln und zu generieren. Werden beispielsweise zwei Abschnitten eines Kapitels die Nummern „001“ und „002“ gegeben, wird folgende Struktur daraus erzeugt:

	→ 1	Teil
	→ 1.1	Kapitel
Nummer: <b>001</b>	→ 1.1.1	Abschnitt
Nummer: <b>002</b>	→ 1.1.2	Abschnitt

Die Nummerierung erfolgt relativ innerhalb der jeweiligen Typen. Das heißt, dass das Feld *Nummer* die Reihenfolge von jenen Abschnitten festlegt, die zu demselben Kapitel gehören.

Das Nummerierungselement verfügt über zusätzliche Möglichkeiten zur Strukturierung. Wenn im Zuge einer organisationsspezifischen Anpassung Abschnitte ergänzt werden, die zwischen zwei Abschnitten eingeordnet werden sollen, so ist es nicht vorgesehen, die *Nummer* der bereits vorhandenen Abschnitte zu verändern, um Platz für den neuen Abschnitt zu schaffen. Stattdessen wird eine Unterstrukturierung vorgenommen:

	→ 1	Teil
	→ 1.1	Kapitel
Nummer: <b>001</b>	→ 1.1.1	Abschnitt
Nummer: <b>001.001</b>	→ 1.1.2	Abschnitt (org.-spezifisch)
Nummer: <b>002</b>	→ 1.1.3	Abschnitt

Da zwischen die Nummern 001 und 002 keine ganze Zahl hineinpasst, wird untergliedert, indem eine zusätzliche Hierarchieebene eingesetzt wird. Das oben dargestellte Beispiel (001 < 001.001 < 002) ist vergleichbar mit der Ordnungsbildung bei Fließkommazahlen (1 < 1,1 < 2). Um nun zwischen 001.001 und 002 noch weitere Themen einzufügen, stehen dem Anpasser zwei Möglichkeiten offen:

1. Eine Kennzeichnung in der Form x.y, wobei gilt  $x < 2^1$  und  $y > 1$ .<sup>2</sup>
2. Eine Kennzeichnung in der Form 001.001.z, wobei z eine beliebige Zahl darstellt.

Sobald zusätzliche Hierarchieebenen eingesetzt werden ist es dringend erforderlich, dass die angegebenen Zahlen 3-stellig sind. Das gilt auch für die erste Ebene. Während 01 und 001.001 als Nummern zulässig ist, wird 01.001 von der Werkzeugumgebung des V-Modells zurzeit nicht korrekt unterstützt.

Analog zu den Strukturierungselementen der Dokumentation, erfolgt die Nummerierung auch für Produkte, Aktivitäten und so weiter. Auch hier wird eine entsprechende Sortierung innerhalb der jeweiligen Elemente, z.B. die Reihenfolge der Themen innerhalb eines Produkts realisiert.

<sup>1</sup> Damit es vor dem Abschnitt mit der Nummer 002 erscheint.

<sup>2</sup> Damit es nach dem Abschnitt mit der Nummer 001.001 erscheint.

**Hinweis:** Eine Ausnahme stellt das Element *V-Modell-Teil* dar. Dies ist das einzige Element, bei dem die in diesem Feld eingetragene Nummer in der exportierten Dokumentation erscheint. Daher sind bei der Vergabe der Nummern für die Teile nur Nummern ohne vorangestellte Nullen verwendet worden, um zu verhindern, dass in der Dokumentation Teile mit den Bezeichnungen "V-Modell XT Teil 001" etc. erscheinen.

## 2.3 Das Paket Statik

Das Paket *Statik* beschreibt die Struktur von Vorgehensbausteinen und die darin enthaltenen Elementtypen. Damit enthält es die wesentlichen Grundelemente des V-Modells, mit denen die Vorgaben des Vorgehensmodells beschrieben werden.

### 2.3.1 Submodelle

Das Paket *Statik* ist sehr umfangreich, weshalb es weiter strukturiert und mit verschiedenen Sichten versehen ist. Im Wesentlichen fasst das Paket verschiedene Submodelle zusammen.

- Das *Aktivitätsmodell* fasst alle deskriptiven Elemente und Beziehungstypen zur Beschreibung von Aktivitäten (i.S. des V-Modells) zusammen.
- Das *Produktmodell* beschreibt alle Produkte, Themen und Beziehungstypen für die Produkte des V-Modells.
- Das *Rollenmodell* beschreibt die Rollen des V-Modells.

Über verschiedene Beziehungen werden die einzelnen Submodelle geeignet miteinander kombiniert. Im weiteren Verlauf werden die Inhalte der Submodelle sowie die Beziehungen detailliert dargestellt.

### 2.3.2 Vorgehensbausteine

Vorgehensbausteine stellen eines der **zentralen Konzepte** des V-Modells dar. Sie enthalten alle Inhalte des V-Modells, die für die Beschreibung der Ergebnisse sowie die Aufstellung von Ergebnisstrukturen erforderlich sind.

Abbildung 8 zeigt die Elemente des Vorgehensbausteins. Vorgehensbausteine sind einer V-Modellvariante zugeordnet und kapseln die wesentlichen Anteile des Produkt- und des Aktivitätsmodells.

**Hinweis:** Rollen sind, anders als in früheren V-Modell-Versionen, nicht mehr in Vorgehensbausteinen enthalten, sondern auf derselben Ebene wie Vorgehensbausteine unmittelbar einer V-Modellvariante zugeordnet. Vorgehensbausteine enthalten entsprechende Beziehungen, die die Verknüpfung von Produkten und Rollen herstellen.



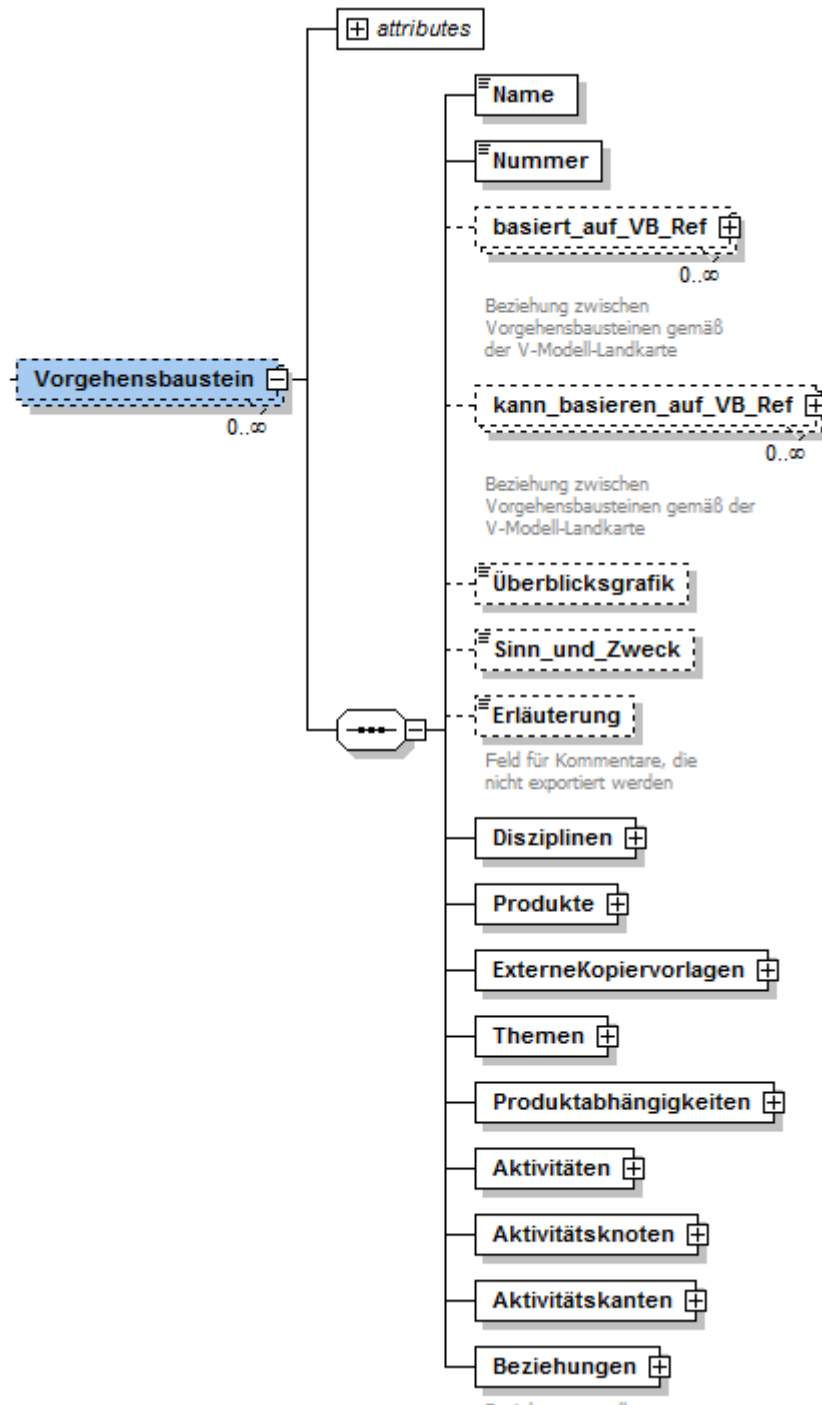


Abbildung 8 Struktur der Vorgehensbausteine im Metamodell (Sicht: XML-Schema)

Element	Beschreibung
Vorgehensbaustein	Vorgehensbausteine sind modulare Einheiten, die die grundlegenden Beschreibungselemente des V-Modells kapseln. Vorgehensbausteine kapseln Produkte, Aktivitäten (und die jeweiligen Verfeinerungen) sowie Disziplinen, Produktabhängigkeiten und Beziehungen.
Disziplinen	Eine Disziplin gehört zu einem Vorgehensbaustein. Mehrere Produkte und Aktivitäten können zu einer Disziplin zusammengefasst werden. Die zu einer Disziplin gehörenden Produkte und Aktivitäten müssen nicht aus dem Vorgehensbaustein stammen, aus dem die Disziplin stammt.



Element	Beschreibung
Produkte	Ein Produkt kann einer Disziplin zugeordnet werden. Einem Produkt sind immer eine verantwortliche Rolle und mehrere optionale, mitwirkenden Rollen zugewiesen. Zur Gliederung kann ein Produkt aus mehreren Themen bestehen. Weitere Informationen hierzu unter Abschnitt 2.3.3.
ExterneKopiervorlagen	Aus dem V-Modell heraus können für Produkt Vorlagen generiert werden. Alternativ können ergänzend zum Modell bereits eigene Vorlagen bereitgestellt werden. Als externe Kopiervorlage können sie Produkten zugeordnet werden und stellen eine Alternative bei der Erzeugung der Produktvorlagen während des Tailorings bereit: Anstelle der generierten Vorlagen kann einfach eine Kopiervorlage ausgewählt werden, die dann <i>kopiert</i> wird.
Produktabhängigkeiten	Produktabhängigkeiten beschreiben Abhängigkeiten zwischen Produkten, z.B. inhaltliche, erzeugende oder strukturelle. Weitere Informationen hierzu unter Abschnitt 2.3.4.
Themen	Ein Thema ist ein Element zur Strukturierung von Produkten. Themen können weiter in Unterthemen gegliedert sein. Weitere Informationen hierzu unter Abschnitt 2.3.3.
Aktivitäten	Aktivitäten beschreiben Vorgänge zur Bearbeitung/Fertigstellung von Produkten. Sie sind i.d.R. nur grob granular beschrieben, können jedoch durch sogenannte <i>Arbeitsschritte</i> verfeinernd beschrieben werden. Weitere Informationen hierzu unter Abschnitt 2.3.5.
Aktivitätsknoten	Aktivitätsknoten und Aktivitätskanten dienen der feineren Modellierung von Aktivitäten (vergleichbar mit dem Aktivitätsmodell der UML 2). Sie beschreiben z.B. Arbeitsschritte, Start- und Endeknoten oder auch Parallelisierungsknoten).
Aktivitätskanten	
Beziehungen	Beziehungen sind Modellelemente, die Modellelemente aus (verschiedenen) Submodellen miteinander verknüpfen, z.B. Rollen und Produkte. Mehr Informationen hierzu unter Abschnitt 2.3.7.

### 2.3.3 Das Produktmodell

Das Produktmodell fasst alle Beschreibungs- und Strukturelemente für die Modellierung von Produkten im V-Modell zusammen. Zum Produktmodell gehören verschiedene Bereiche:

- Produkte
- Themen und Unterthemen
- Produktabhängigkeiten

**Produkte.** Der Bereich Produkte ist in Abbildung 9 dargestellt. Produkte besitzen verschiedene Attribute, die für alle referenzierbaren V-Modell-Modellelemente gleichartig sind (siehe Abschnitt 2.2.1). Neben den Attributen besitzen Produkte noch eine Reihe weiterer Elemente. Jedes Produkt hat einen *Namen*. Dieser kennzeichnet den Produkttyp im V-Modell. Das Element *Nummer* muss für alle Produkte eindeutig mit einem ganzzahligen Wert belegt werden (siehe Abschnitt 2.2.2). Dieser Wert identifiziert ein Produkt und legt die Reihenfolge für den Export fest. *DisziplinRef* ordnet ein Produkt in genau eine Disziplin ein. Sofern ein Produkt im Rahmen der AG/AN-Schnittstelle relevant ist und vom Partnerprojekt übernommen wird, kann mithilfe von *SchnittstellenQuellproduktRef* das Ausgangsprodukt referenziert werden.

Jedes Produkt muss eine Beschreibung seines Sinns und Zwecks haben. Dafür sind die Elemente *Sinn\_und\_Zweck* oder alternativ *TextbausteinRef* zu belegen. Diese beiden Elemente tauchen in vielen Modellelementen des V-Modells auf und tragen immer die Informationen zur Beschreibung der jeweiligen Elemente. Aus diesen wird die Prozessdokumentation generiert. Sofern ein Textbaustein ausgewählt wurde, wird dieser vom Export bevorzugt. Die Elemente *BeispielProduktgestaltung* und

*Beispielprodukt* dienen der Ausgestaltung eines Produkts bzw. der Bereitstellung von Beispielen als Vorlagen.

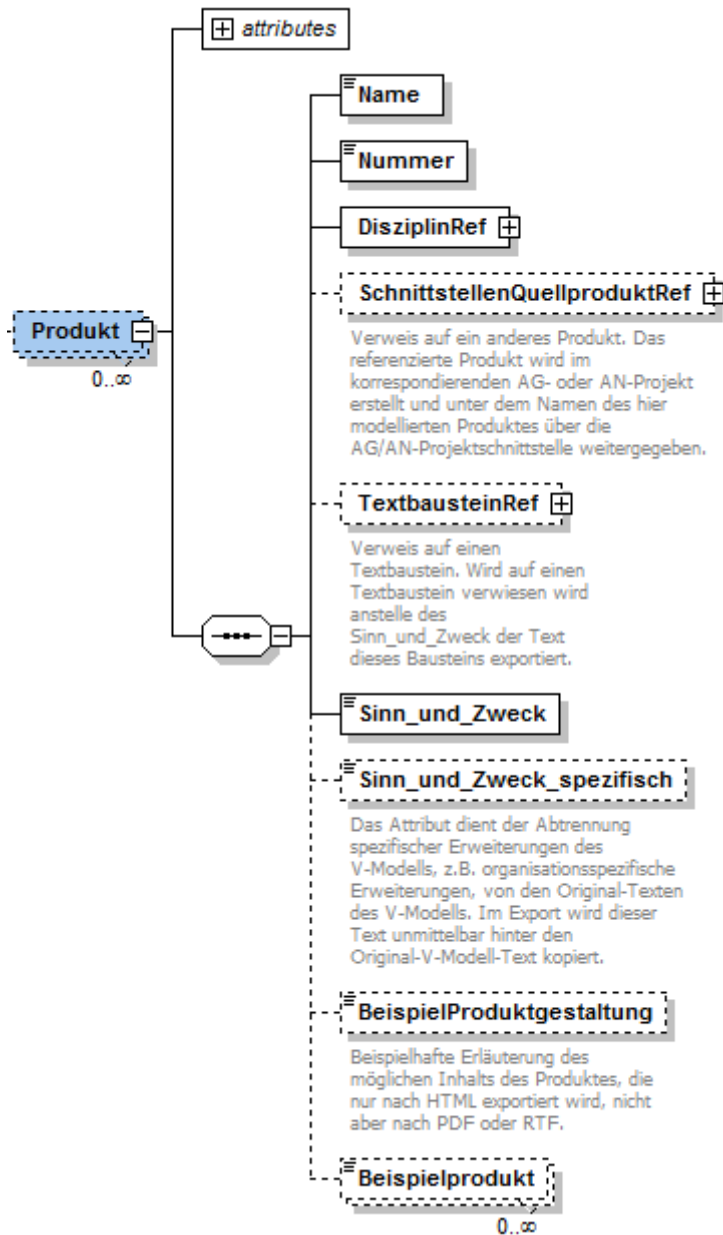
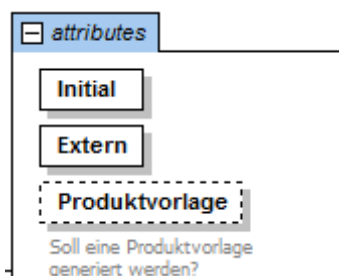


Abbildung 9 Das Produktmodell, Bereich Produkt (Sicht: XML-Schema)

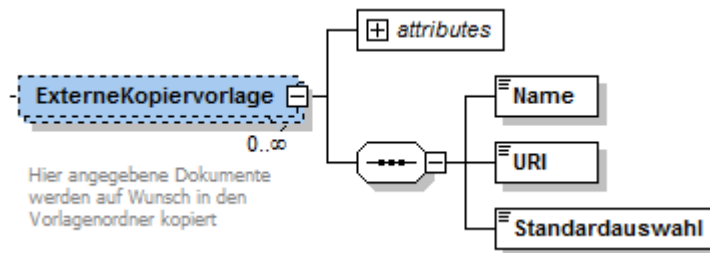
**Anmerkung:** Es existieren noch Felder, die den Suffix „spezifisch“ haben. Die betreffenden Felder existieren z.B. bei Produkten und Themen. Diese Felder wurden in früheren Versionen des V-Modells für organisationsspezifische Anpassungen verwendet. Sie sind aus Gründen der Kompatibilität noch enthalten. Von der Verwendung wird bei Neuanpassungen jedoch *dringend abgeraten*. Stattdessen sollte auf die *Änderungsoperationen* (siehe Abschnitt 3.3) zurückgegriffen werden.

Weiterhin erweitert der Elementtyp Produkt die oben gezeigte Attributliste um folgende Attribute:



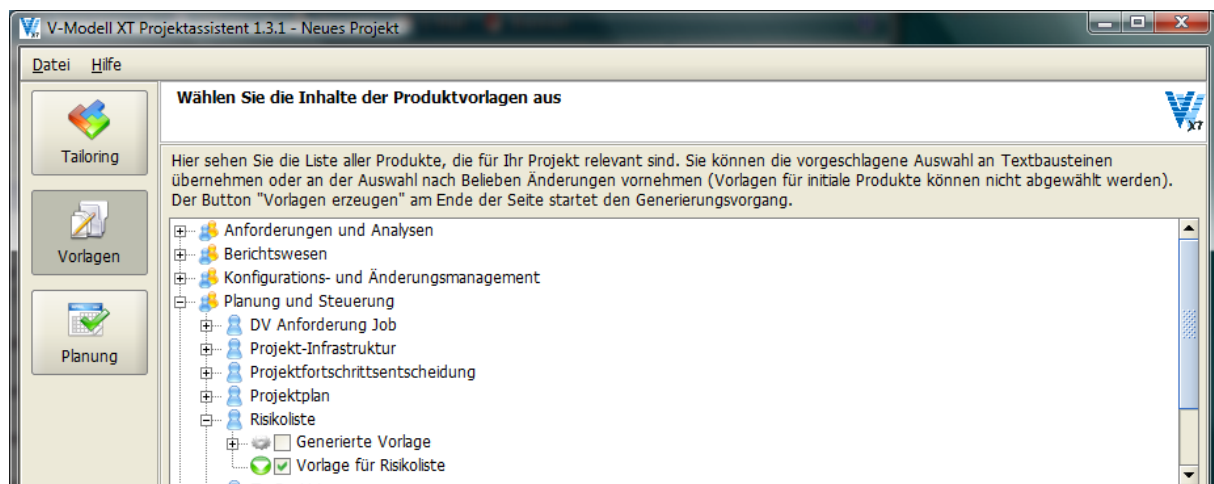
- Die Attribute *Initial* und *Extern* steuern das Erzeugungsverhalten von Produkten während der Instanziierung im Projekt. Mögliche Werte für die Attribute sind jeweils *Ja* und *Nein*.
- Das Attribut *Produktvorlage* muss mit dem Wert *Ja* belegt werden, um während des Tailorings mit dem Projektassistenten die Erstellung einer Produktvorlage zu gestatten. Ist eine externe Kopier-vorlage vorhanden, kann diese dann ergänzend angegeben werden.

**Externe Kopiervorlagen.** Externe Kopiervorlagen (Abbildung 10) sind vollständige Dokumente beliebigen Typs (z.B. Word-, Excel-, Java- oder sonstige Dateien), die als Vorlage beigelegt werden können. Während der Erzeugung der Produktvorlagen, kann der Anwender zwischen einer generierten Vorlage (aus den Inhalten des V-Modells) oder einer Kopiervorlage wählen. Die zur Verfügung stehenden Kopiervorlagen ergeben sich aus den im Modell hinterlegten Beziehungen (siehe Abschnitt 2.3.7).



**Abbildung 10 Das Produktmodell, Bereich externe Kopiervorlagen (Sicht: XML-Schema)**

Die Vorlage wird über das Element *URI* referenziert. Dieses ist ein relativer Pfad ausgehend von der V-Modell-XML-Datei. Die *Standardauswahl* zeigt an, ob anstelle der generierten Vorlage immer zuerst die Kopiervorlage bei der Produktvorlagenerzeugung angeboten werden soll (Abbildung 11).



**Abbildung 11 Externe Kopiervorlagen im V-Modell XT Projektassistenten**

**Themen.** Produkte können durch Themen strukturiert werden. Abbildung 12 zeigt die entsprechende Struktur des XML-Schemas für den Bereich Thema. Neben den weiter oben gezeigten Attributen verfügen Themen über weitere Elemente.

Die Attribute *Name* und *Nummer* sind analog zu den Elementen für Produkte zu verwenden. Auch hier ist wieder wichtig, dass die vergebenen Nummern im Bereich der Themen eindeutig sind. Diese legen wiederum die Reihenfolge der Themen bei der Erzeugung der Produktbeschreibungen fest.

Themen verfügen ebenfalls über Beschreibungen. Verwendet werden die Elemente: *TextbausteinRef*, *Beschreibung* sowie *BeispielProduktgestaltung*. Die Beschreibung entspricht hierbei dem Sinn und Zweck der Produkte und wird analog gehandhabt. Auch die beispielhafte Produktgestaltung und die Textbausteinreferenzen werden analog verwendet.

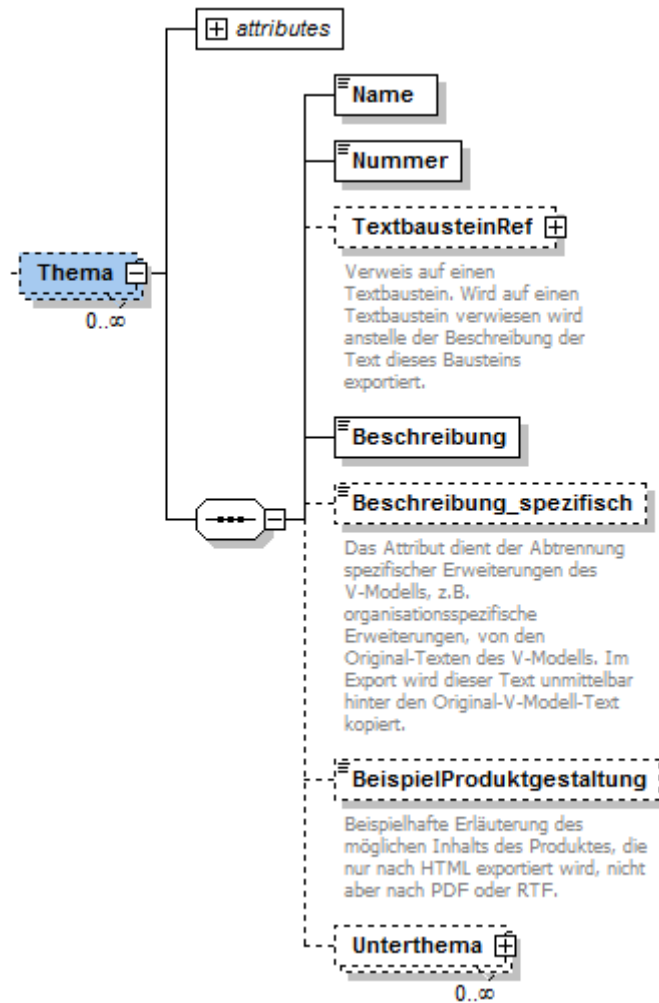


Abbildung 12 Das Produktmodell, Bereich Thema (Sicht: XML-Schema)

**Hinweis:** Anders als in früheren Versionen des V-Modell-Metamodells werden die Zuordnungen von Themen zu Produkten nicht mehr direkt beim Thema vorgenommen, sondern extern durch einen eigenen Beziehungstyp (siehe Abschnitt *Beziehungen*).

**Unterthemen.** Themen können selbst wieder sogenannte Unterthemen enthalten. Diese dienen dazu, Themen analog zu Produkten weiter ausgestalten zu können.

Abbildung 13 zeigt den dafür relevanten Abschnitt des XML-Schemas. Die Struktur entspricht dabei weitgehend der gerade beschriebenen Struktur des Thema-Elements. Unterthemen können selbst auch wieder Unterthemen enthalten.

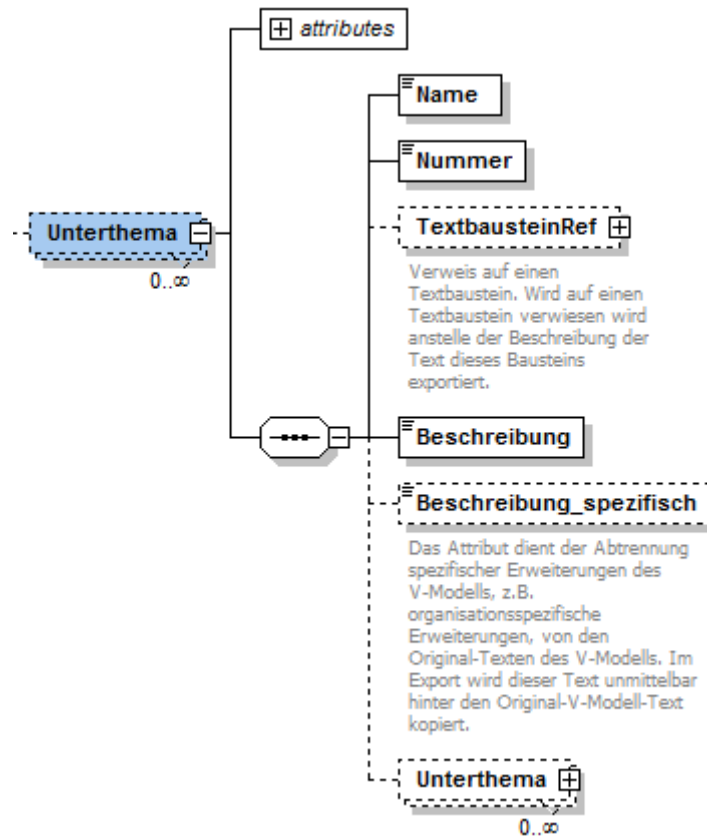


Abbildung 13 Das Produktmodell, Bereich Unterthema (Sicht: XML-Schema)

### 2.3.4 Produktabhängigkeiten

Produkte im V-Modell stehen üblicherweise nicht allein für sich, sondern existieren in einem Abhängigkeitsnetz. Dieses Netz wird durch die Produktabhängigkeiten aufgespannt. Das V-Modell kennt:

- Tailoringabhängigkeiten
- Strukturabhängigkeiten
- Erzeugende Abhängigkeiten
- Inhaltliche Abhängigkeiten

Abbildung 14 stellt alle Produktabhängigkeiten dar. Die Produktabhängigkeiten dienen dabei als „Aufhänger“ für die Zuordnung von Produkten. Diese Zuordnung wird wiederum durch separate Beziehungen hergestellt, siehe Abschnitt 2.3.7, weshalb wir hierauf an dieser Stelle nicht näher eingehen.

*Tailoringabhängigkeiten* werden zwischen Produkten und Vorgehensbausteinen hergestellt. Sie zeigen an, dass bei einer bestimmten Produktkonstellation noch zusätzliche Vorgehensbausteine im Tailoring berücksichtigt werden müssen und welche Produkte dafür ausschlaggebend sind.

*Strukturabhängigkeiten* modellieren u.a. die Systemarchitektur des V-Modell-Systemmodells (vgl. V-Modell XT Referenz Teil 5: Produkte). Diese Abhängigkeit wird zwischen Produkten aufgebaut und kennzeichnet Teil-Ganzes-Beziehungen, z.B. Ein System enthält Segemente.

*Erzeugende Abhängigkeiten* steuern das Verhalten bei der Instanziierung von Produkten. Ausgehend von einem oder mehreren Quellprodukten können beliebig viele Zielprodukte erstellt werden. Die Instanziierung dieser Produktabhängigkeit, z.B. für das Projektmanagement, stellt den Leitfaden für Projektleiter dar, aus dem sie ablesen können, welche Produkte erzeugt werden müssen und warum.

*Inhaltliche Abhängigkeiten* werden zwischen Produkten aufgebaut, die inhaltlich miteinander in Beziehung stehen. Die inhaltlichen Produktabhängigkeiten werden während des Exports der Produktvorlagen ausgewertet und dienen dort als Vorgabe für die Prüfung eines Dokuments.

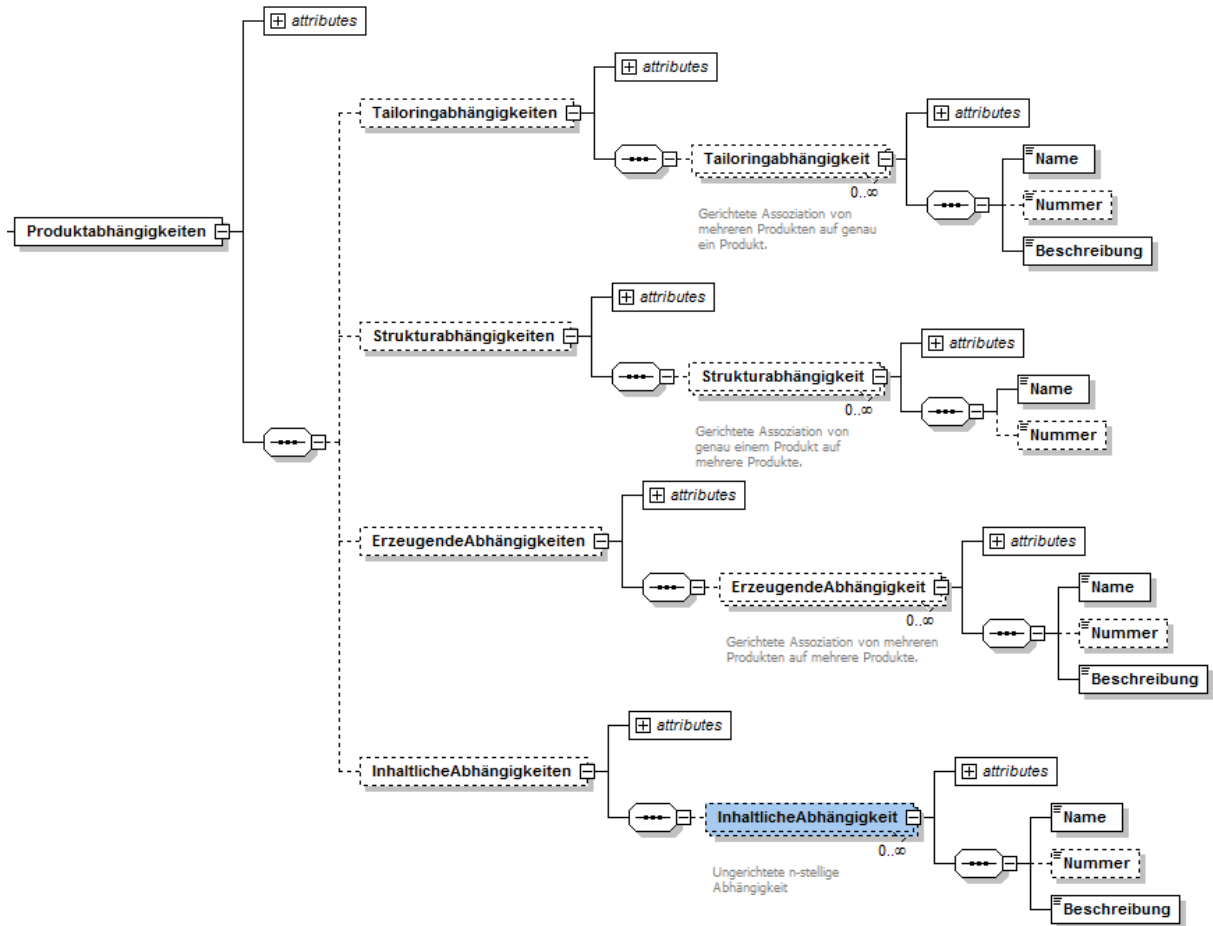


Abbildung 14 Das Produktmodell, Bereich Produktabhängigkeiten (Sicht: XML-Schema)

### 2.3.5 Das Aktivitätsmodell

Das Aktivitätsmodell beschreibt die Abläufe bei der Erstellung eines Produkts. Das Aktivitätsmodell ist ebenfalls in verschiedene Bereiche aufgliedert.

**Aktivitäten.** Abbildung 15 zeigt den Ausschnitt für die *Aktivität*.

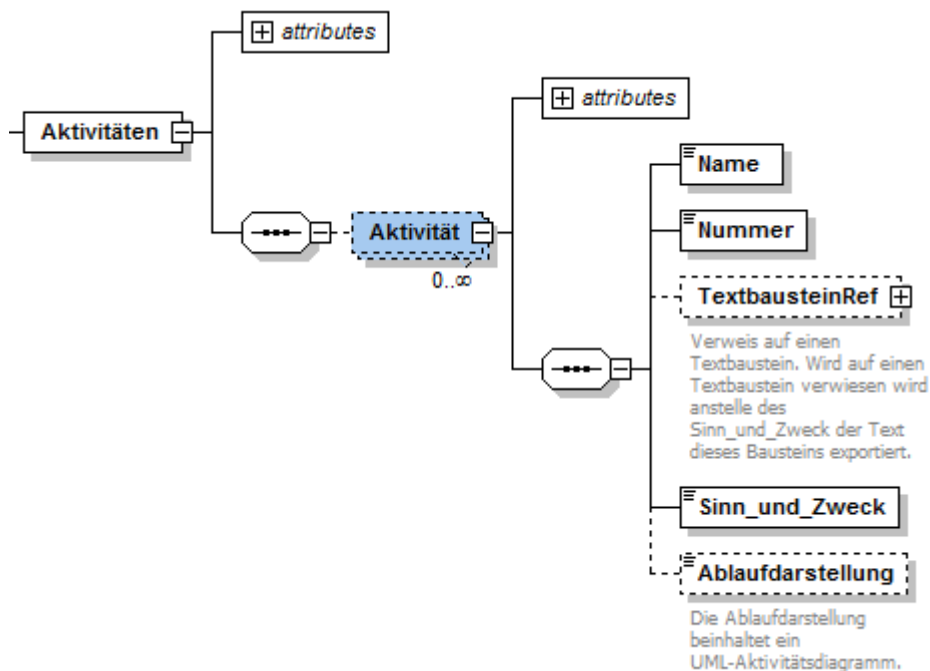


Abbildung 15 Das Aktivitätsmodell, Bereich Aktivität (Sicht: XML-Schema)

Aktivitäten verfügen über die bereits vorgestellte, allgemeine Attributliste (*id* etc.) und über weitere Elemente. *Name* und *Nummer* werden analog zu Produkten und Themen vergeben. Weiterhin gibt es eine Beschreibung, die wiederum entweder über *Sinn\_und\_Zweck* oder *TextbausteinRef* hinterlegt werden kann. Auch kann einer Aktivität eine *Ablaufdarstellung* zugeordnet werden. Diese wird üblicherweise als Bild im Modell hinterlegt.

Anders als in früheren Versionen gibt es keine Teilaktivitäten mehr. Aktivitäten im V-Modell entsprechen nun Containern, die mehrere, an die UML-Aktivitätsdiagramme angelehnte, Modellelemente enthalten können. Diese Modellelemente sind im Wesentlichen *Aktivitätsknoten* und *Aktivitätskanten*.

**Aktivitätsknoten.** Aktivitätsknoten entsprechen im Wesentlichen den aus der UML bekannten Activity Nodes. Abbildung 16 zeigt diesen Bereich des Aktivitätsmodells. Die Knotentypen *Startknoten*, *Endeknoten*, *Parallelisierungsknoten*, *Synchronisationsknoten*, *Verzweigungsknoten* und *Verflechtungsknoten* entsprechen den Strukturierungselementen von UML-Aktivitätsdiagrammen.

Besonders von Interesse ist der Knotentyp *Arbeitsschrittknoten*. Dieser löst die klassische Teilaktivität ab und ist in seiner Struktur im Wesentlichen mit dieser identisch.

*Unteraktivitätsknoten* diener der hierarchischen Strukturierung. Diese Fähigkeit ist auch im Umfeld der UML-Aktivitätsdiagramme zu finden, wo einzelne Aktivitäten wiederum zu komplexen Diagrammen verfeinert werden können. Diese Eigenschaft ist durch Unteraktivitätsknoten auch im V-Modell-Metamodell zu finden. Aktivitäten können somit also prinzipiell auch über mehrere Hierarchieebenen verfeinert und strukturiert werden.

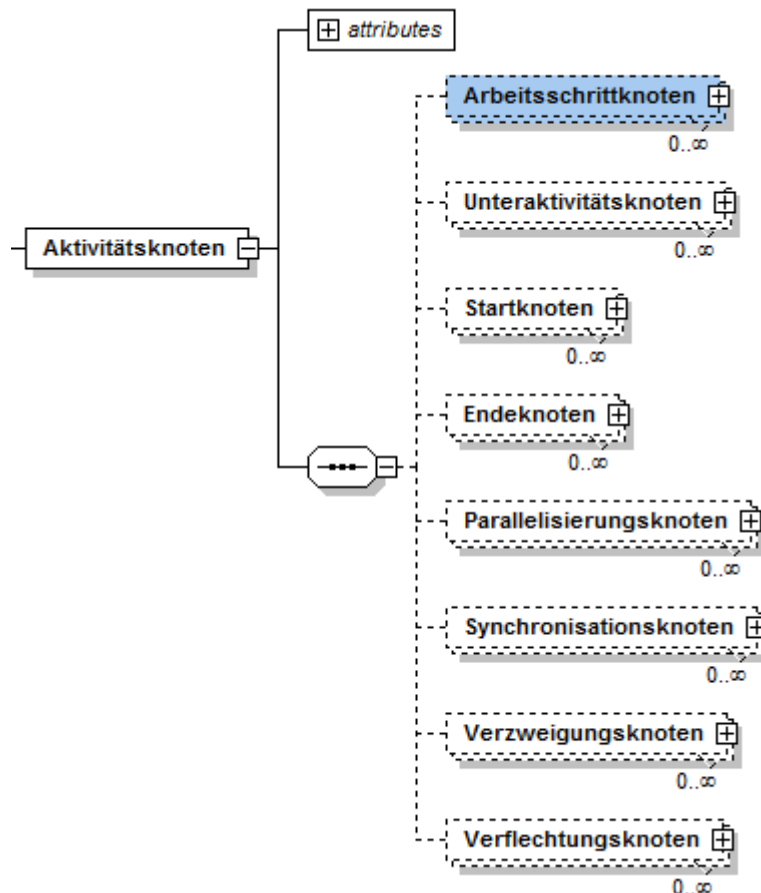


Abbildung 16 Das Aktivitätsmodell, Bereich Aktivitätsknoten (Sicht: XML-Schema)

**Aktivitätskanten.** Die Aktivitätsknoten stellen nur die Knoten im Aktivitätsgraphen dar. Die Kanten zwischen den einzelnen Aktivitätsknoten werden durch die *Aktivitätskanten* modelliert. Abbildung 17 stellt diesen Bereich des Metamodells dar.

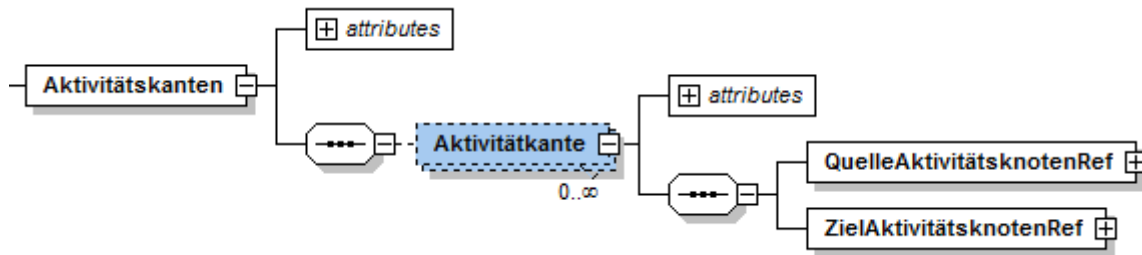


Abbildung 17 Das Aktivitätsmodell, Bereich Aktivitätskanten (Sicht: XML-Schema)

Eine Aktivitätskante wird dabei immer zwischen zwei Aktivitätsknoten gezogen. *QuelleAktivitätsknotenRef* ist dafür der Quellknoten, *ZielAktivitätsknotenRef* kennzeichnet den Zielknoten.

**Hinweis:** Um Aktivitäten und oder Arbeitsschritte zu erstellen und diese Produkten und Themen zuzuordnen, muss keine detaillierte Modellierung der Aktivitäten im Sinne eines UML-Aktivitätsdiagramms erfolgen. Es genügt, um die „klassischen“ V-Modell-Fähigkeiten zu bekommen, die Aktivitätsknoten anzulegen und diese zuzuweisen. Die Modellierung unter Einbeziehung der Aktivitätskanten ist optional.

### 2.3.6 Das Rollenmodell

Das V-Modell definiert ein flaches, nicht-hierarchisches und somit organisationsstrukturunabhängiges Rollenmodell. Rollen als Metamodellelemente sind unmittelbar unter der V-Modellvariante lokalisiert. Anders als in früheren V-Modell-Versionen sind sie somit nicht mehr in Vorgehensbausteinen untergebracht und somit einheitlich für alle Vorgehensbausteine verfügbar. Die Referenzierung von Rollen erfolgt nun nicht mehr durch ein Element innerhalb der Produkte, sondern über spezielle Beziehungstypen (siehe Abschnitt 2.3.7).

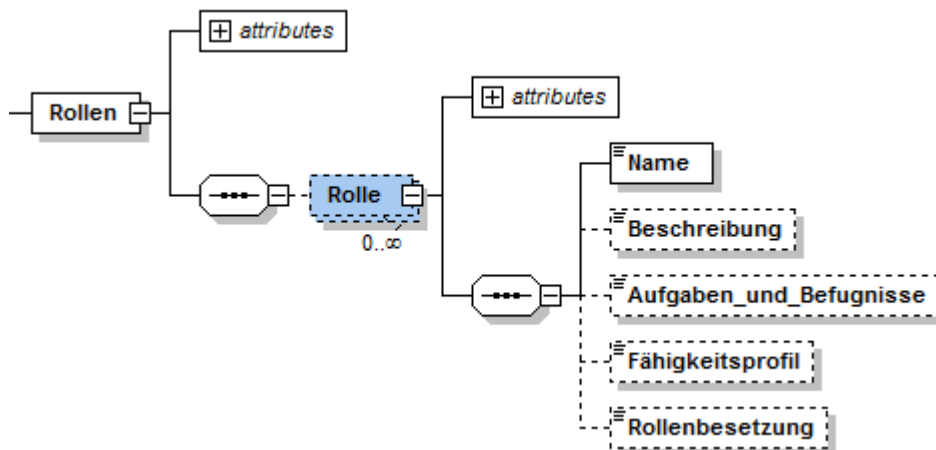


Abbildung 18 Das Rollenmodell (Sicht: XML-Schema)

Abbildung 18 zeigt die Elemente des Rollenmodells. Neben den üblichen Attributen verfügt eine Rolle über weitere Elemente.

Der *Name* wird analog zum Produkt vergeben. Die *Beschreibung* einer Rolle enthält inhaltliche Aussagen und wird durch *Aufgaben\_und\_Befugnisse* sowie ein *Fähigkeitsprofil* ergänzt. Abschließend können im Element *Rollenbesetzung* Empfehlungen für die tatsächliche Besetzung einer Rolle mit personellen Ressourcen hinterlegt werden.

### 2.3.7 Beziehungen

Beziehungen sind im *Vorgehensbaustein* hinterlegt und ordnen die Elemente von Vorgehensbausteinen untereinander zu und stellen somit die Abhängigkeiten her. Viele Beziehungen, die in früheren Versionen des V-Modells über Elemente oder Attribute direkt an den Elementtypen hergestellt wurden, sind nun explizit eigene Beziehungstypen. Dies steigert die Flexibilität des V-Modells insbesondere unter Anpassungsaspekten.



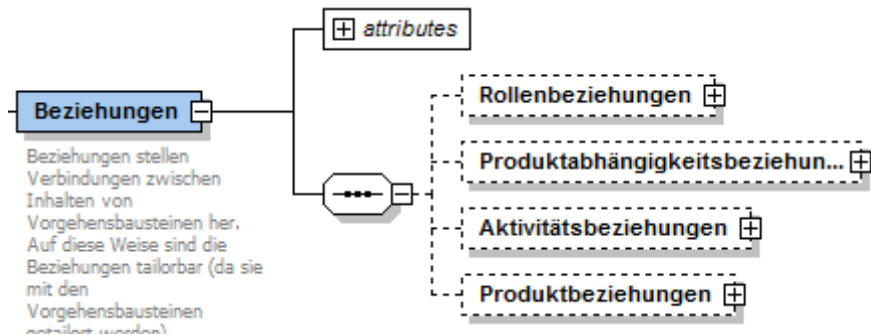


Abbildung 19 Explizite Beziehungstypen im V-Modell-Metamodell

Das V-Modell kennt vier explizite Gruppen von Beziehungstypen, Abbildung 19. *Rollenbeziehungen* fassen alle Beziehungstypen zusammen, die Zusammenhänge zwischen Rollen und weiteren Metamodellelementen herstellen. *Produktabhängigkeitsbeziehungen* enthalten alle Beziehungen, die zur Ausgestaltung von Produktabhängigkeiten erforderlich sind. *Aktivitätsbeziehungen* enthalten alle Beziehungstypen, die zwischen Elementen des Aktivitätsmodells und weiteren Metamodellelementen hergestellt werden können. *Produktbeziehungen* enthalten alle Beziehungstypen zwischen Produkten und weiteren Metamodellelementen. Im Folgenden werden alle Beziehungstypen dargestellt. Eine detaillierte Aufstellung möglicher Zieltypen sowie Kardinalitäten<sup>3</sup> wird im Anschluss gesondert gezeigt.

**Rollenbeziehungen.** Rollen (Abschnitt 2.3.6) sind entweder für die Erstellung eines Produkts *verantwortlich* oder *wirken* bei seiner Erstellung *mit*. In früheren Versionen des V-Modells waren die Rollen direkt in einem Vorgehensbaustein untergebracht und die Verknüpfungen zwischen die Rollen und den referenzierten Produkten wurden durch die Produkte (alt Attribute: *Verantwortlicher* und *Mitwirkend*) selbst hergestellt.

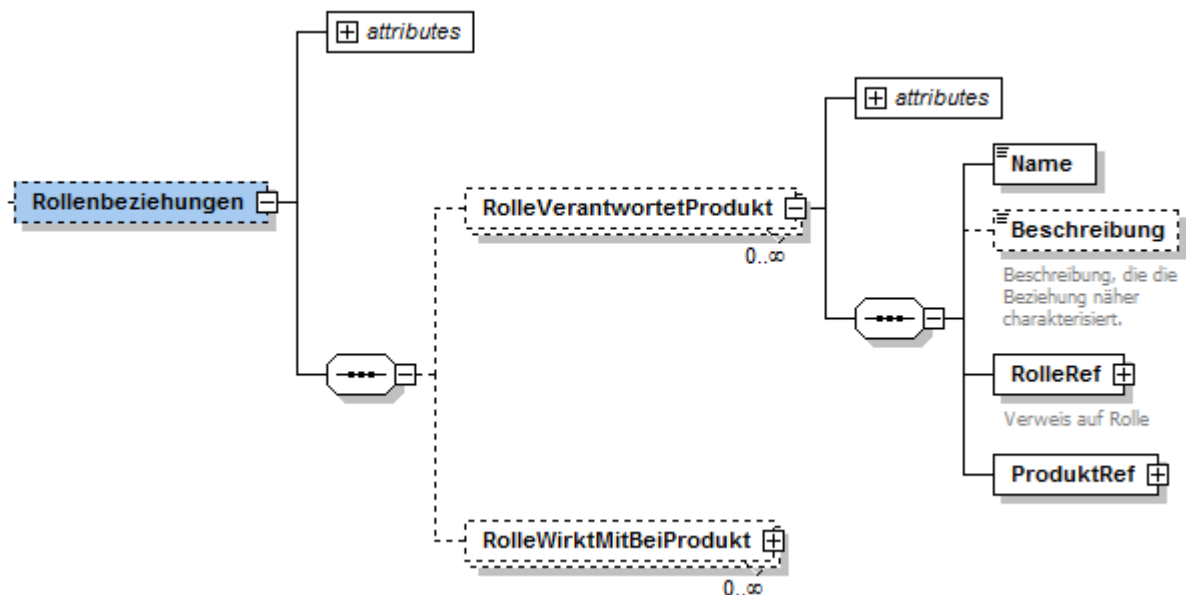


Abbildung 20 Explizite Beziehungstypen, Gruppe: Rollenbeziehungen

Abbildung 20 zeigt das neue Muster zur Herstellung der Beziehungen. Um Produkte und Rollen miteinander zu verknüpfen, müssen:

1. Die jeweiligen Produkte angelegt sein (Vorbedingung).
2. Die benötigten Rollen angelegt sein (Vorbedingung).
3. Beziehungselemente vom Typ *RolleVerantwortetProdukt* und/oder *RolleWirktMitBeiProdukt* angelegt werden, die die existierenden Elemente referenzieren.

<sup>3</sup> Kardinalitäten sind die Wertigkeiten der Kantenenden (vgl. auch UML [6]). Ein Kante zwischen zwei Elementen, die die Wertigkeit 1-1..\* hat, drückt aus, dass das Element am Kantenende mit der Wertigkeit 1 zu mindestens 1 oder beliebig vielen (\*) anderen Elementen passenden Typs am 1..\*-Kantenende in Beziehung stehen kann.

Beziehung	Quellentyp	Zieltyp	Kardinalität
RolleVerantwortetProdukt	Rolle	Produkt	1 – 1
RolleWirktMitBeiProdukt	Rolle	Produkt	1 – 1..*

Die Verantwortlichkeitsbeziehung zwischen Rollen und Produkte nimmt im V-Modell aufgrund seiner Produktorientierung eine spezielle Position ein. Für jedes Produkt ist eine Rolle verantwortlich. Um diese Semantik korrekt zu transportieren, muss jede Verantwortlichkeit separat eingepflegt werden. Für die Mitwirkungen an der Produkterstellung ist ein „Abkürzung“ möglich.

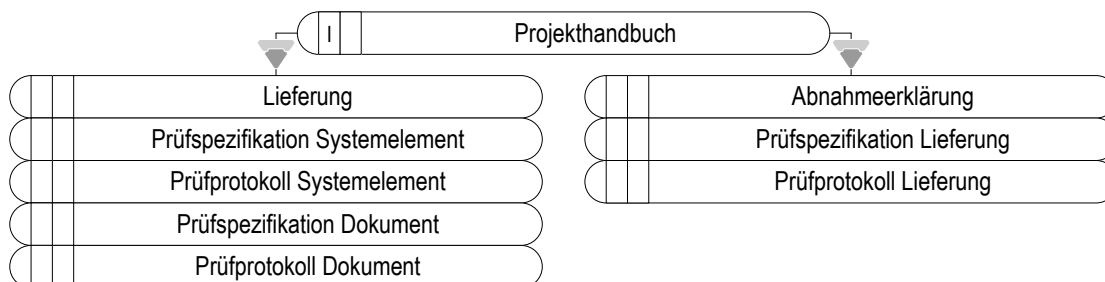


Wie die oben stehende Abbildung zeigt, kann in einer Mitwirkungsbeziehung zu einer Rolle auch mehr als ein Zielprodukt angegeben werden. Dies kann die Anzahl der Element im Modell reduzieren helfen.

Die Beziehung *RolleWirktMitBeiProdukt* verfügt zusätzlich noch über ein Element *IstUnverzichtbar*. Wenn nach dem Tailoring eine Rolle bei einem Produkt mitwirkt und bei dieser Mitwirkung das Element *IstUnverzichtbar=Ja* gesetzt ist, dann wird diese Rolle in das projektspezifische V-Modell übernommen. So kann modelliert werden, dass Rollen aus dem projektspezifischen Modell herausgenommen werden, obwohl sie mitwirkend bei einem Produkt sind.

**Hinweis:** Eine Rolle wird nur dann nach dem Tailoring noch vorhanden sein, wenn sie entweder noch für ein Produkt verantwortlich ist oder bei mindestens einem Produkt mitwirkt und diese Mitwirkung als unverzichtbar gekennzeichnet ist.

**Produktabhängigkeitsbeziehungen.** Die Produktabhängigkeitsbeziehungen dienen der Ausgestaltung der der Produktabhängigkeiten (Abschnitt 2.3.4). Zum besseren Verständnis zeigt Abbildung 21, die eine erzeugende Produktabhängigkeit aus dem V-Modell, genauer die Darstellung aus der Prozessdokumentation. Zu sehen ist ein Quellprodukt (hier das *Projekthandbuch*) und mehrere erzeugte, bzw. Zielprodukte (z.B. die *Lieferung*). Die Produktabhängigkeit stellt die Verbindung zwischen den Quell- und Zielprodukten her.



**Abbildung 21 Beispiel einer erzeugenden Produktabhängigkeit (aus der V-Modell Dokumentation)**

Produktabhängigkeitsbeziehungen im Sinne der Metamodellstrukturen stellen Beziehungen zwischen den Produktabhängigkeiten (Abschnitt 2.3.4) und den relevanten Produkten her. Die Struktur ist daher 2-stufig zu erstellen. Abbildung 22 zeigt im Editor die (erzeugende) Produktabhängigkeit, die in Abbildung 21 teilweise gezeigt ist. Sie besteht aus drei Modellelementen, die in zwei Schritten erstellt werden.

1. Der Produktabhängigkeit (unter dem Knoten *Vorgehensbaustein/Produktabhängigkeiten/Erzeugende Abhängigkeiten*) und
2. Die benötigten Produktabhängigkeitsbeziehungen (hier für Quell- und Zielprodukte), die dann jeweils die benötigten Produkte referenzieren (zu finden unter dem Knoten *Vorgehensbaustein/Beziehungen/Produktabhängigkeitsbeziehungen*).

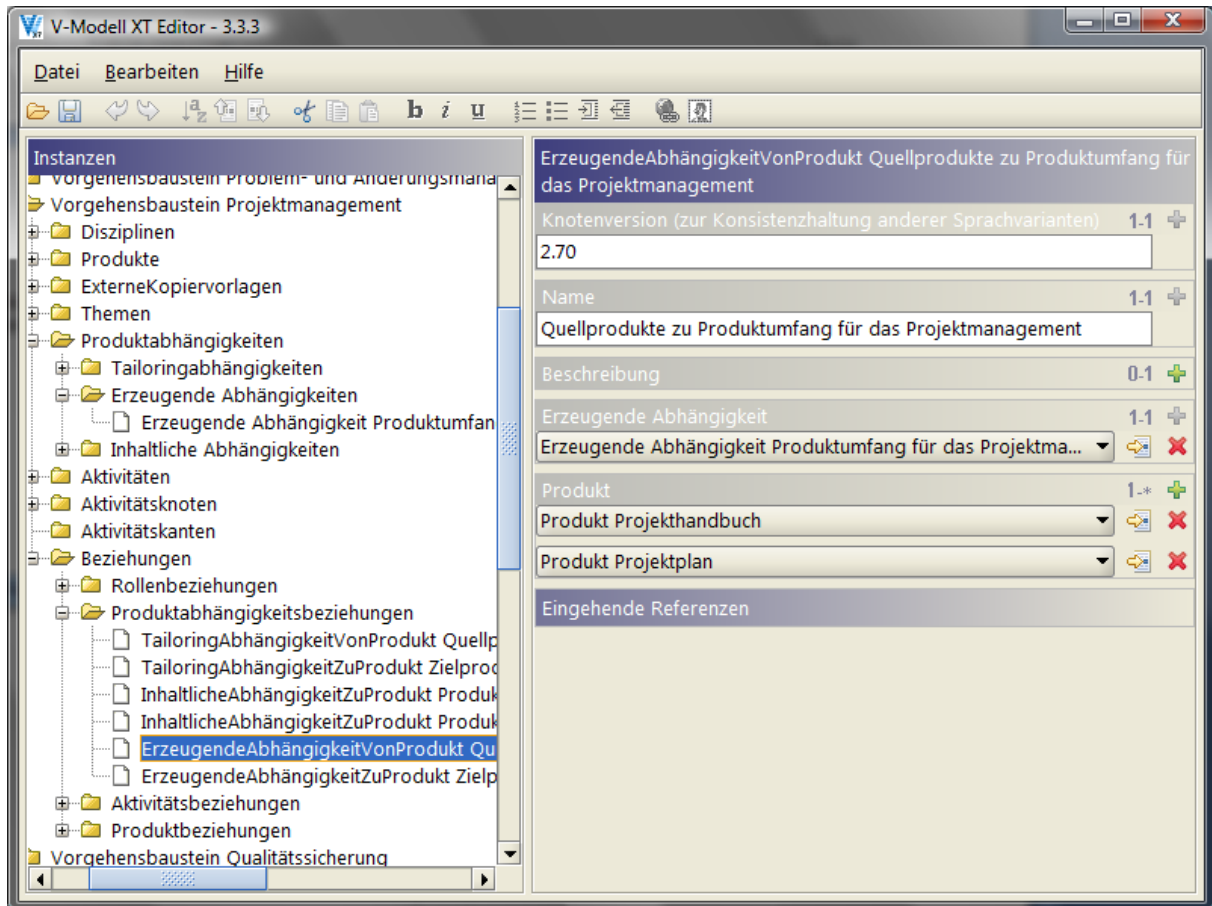


Abbildung 22 2-stufiges Herstellen von Produktabhängigkeiten im Editor

Diese Struktur der Produktabhängigkeiten bewirkt, dass eine Produktabhängigkeit in einem Vorgehensbaustein angelegt werden kann. Andere Vorgehensbausteine können dann weitere Produkte in die Produktabhängigkeiten einbringen. Die Produktabhängigkeiten sind im Kontext einer V-Modell-Variante global sichtbar und können von jeder Stelle im Modell referenziert werden.

**Hinweis:** Dieses Konzept löst die *Produktabhängigkeitserweiterungen* der früheren Versionen des V-Modells ab. Erweiterungen werden nun einfach und homogen über neue Abhängigkeitsbeziehungen eingebracht. Ein zusätzliches Konzept ist nicht mehr erforderlich.

**Gerichtete und ungerichtete Produktabhängigkeiten.** Unterschieden wird im V-Modell zwischen *gerichteten* und *ungerichteten* Produktabhängigkeitsbeziehungen. Abbildung 23 zeigt alle im V-Modell Metamodell definierten Produktabhängigkeitsbeziehungen.

Bei den gerichteten Produktabhängigkeitsbeziehungen ist im Namen jeweils ein *Richtungsattribut* (siehe auch Abbildung 22) zu sehen. Dieses weist aus, ob es sich bei den Produkten, ausgehend von der Produktabhängigkeit, um Quell- oder um Zielprodukte handelt. Das Namensmuster *<Name der Beziehung>VonProdukt* verweist (außer bei strukturellen Produktabhängigkeiten) von der Abhängigkeit auf die Quellprodukte, Beziehungen mit dem Muster *<Name der Beziehung>ZuProdukt* verweisen auf die Zielprodukte einer Produktabhängigkeit. Bei den ungerichteten Produktabhängigkeiten (hier nur die inhaltlichen Produktabhängigkeiten) gibt es nur die *<Name der Beziehung>ZuProdukt*-Beziehung.

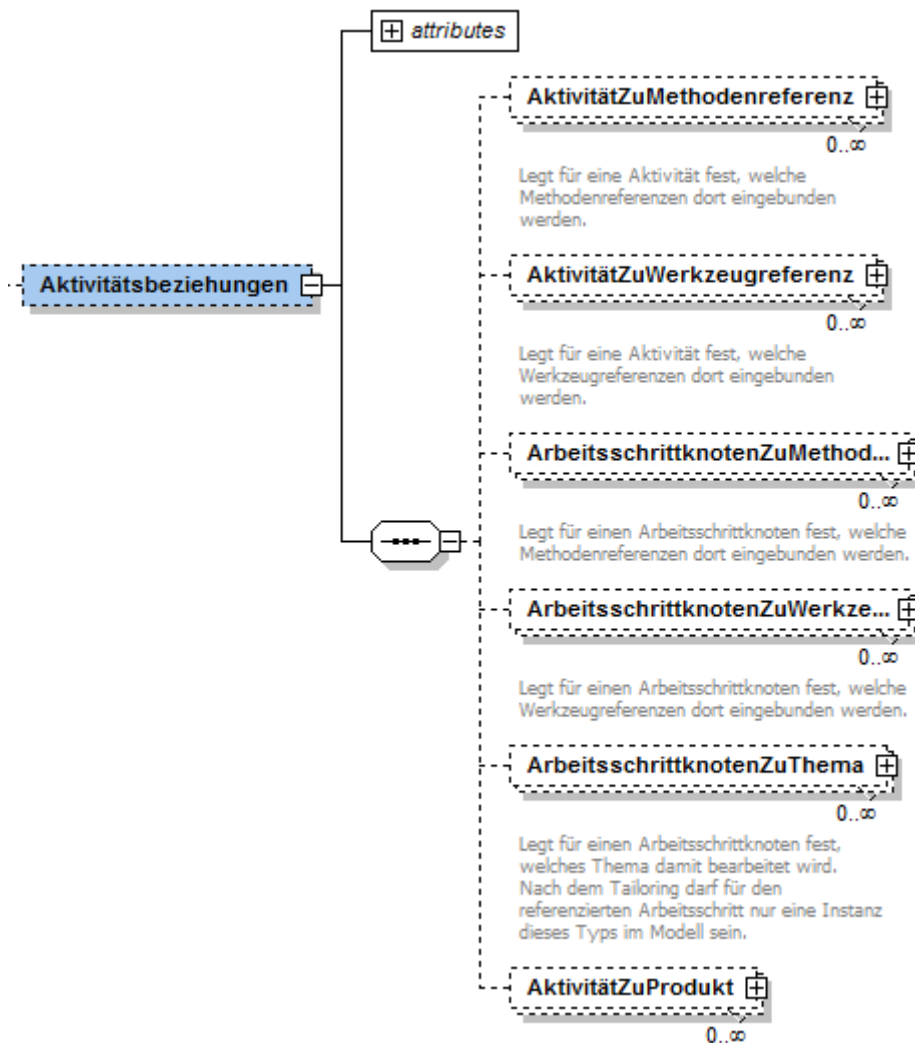
**Besonderheiten der strukturellen Produktabhängigkeit.** Bei der strukturellen Produktabhängigkeit wird vom Benennungsschema abgewichen, da es sich hierbei nicht nur um das in Beziehung setzen von gleichberechtigten Produkten handelt, sondern *hierarchische Produktstrukturen* (z.B. im Rahmen des sog. *Systemmodells*) modelliert werden können. Die Benennung richtet sich somit nach dem *Teil-Ganzes* Prinzip. Ein Produkt (das Ganze) besteht aus mehreren anderen Produkten (die Teile des Ganzen). Im Teil 5 der V-Modell-Dokumentation ist dies für die Produkte *System*, *Segment*, *Komponente*, *Einheit* und *Module* detailliert aufgeschlüsselt. Die Produktabhängigkeit hierzu ist im Vorgehensbaustein *Systemerstellung* hinterlegt. Erweiterungen sind z.B. in den Vorgehensbausteinen *SW-Entwicklung* oder *HW-Entwicklung* zu finden.



Abbildung 23 Explizite Beziehungstypen, Gruppe: Produktabhängigkeitsbeziehungen

Beziehung	Quelltyp	Zieltyp	Kardinalität
TailoringAbhängigkeitVonProdukt	Tailoringabhängigkeit	Produkt	1 – 1..*
TailoringAbhängigkeitZuProdukt	Tailoringabhängigkeit	Produkt	1 – 1..*
InhaltlicheAbhängigkeitZuProdukt	InhaltlicheAbhängigkeit	Produkt	1 – 1..*
ErzeugendeAbhängigkeitVonProdukt	ErzeugendeAbhängigkeit	Produkt	1 – 1..*
ErzeugendeAbhängigkeitZuProdukt	ErzeugendeAbhängigkeit	Produkt	1 – 1..*
StrukturAbhängigkeitTeilProdukt	Strukturabhängigkeit	Produkt	1 – 1..*
StrukturAbhängigkeitGanzesProdukt	Strukturabhängigkeit	Produkt	1 – 1

**Aktivitätsbeziehungen.** Aktivitätsbeziehungen (Abbildung 24) stellen flexibel die Verknüpfung zwischen Elementen des Aktivitätsmodells und anderen Modellteilen her.



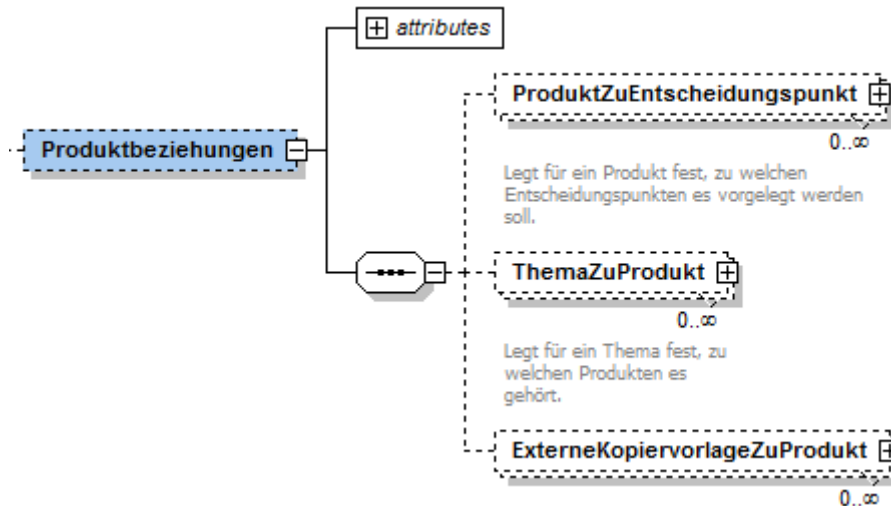
**Abbildung 24 Explizite Beziehungstypen, Gruppe: Aktivitätsbeziehungen**

Die Aktivitätsbeziehungen explizieren die Zuordnungen von Aktivitäten, Arbeitsschritten etc. zu anderen Modellelementen. Dabei werden verschiedene vormals implizite Beziehungen aus den Modellelementen herausgezogen und hier expliziert. Beispielsweise wird in einem Arbeitsschritt (ehemals Teilaktivität) nicht mehr direkt der Bezug zu einem Thema eines Produkts hergestellt. Stattdessen kommt dafür nun die Beziehung *ArbeitsschrittnotenZuThema* zum Einsatz. Analog wird für die Verknüpfung von Aktivitäten und Produkten verfahren. Auch Methoden- und Werkzeugreferenzen referenzieren Aktivitäten und Arbeitsschritte nicht mehr direkt, sondern durch eigene Beziehungstypen.

Beziehung	Quellentyp	Zieltyp	Kardinalität
AktivitätZuMethodenreferenz	Aktivität	Methodenreferenz	1 – 1..*
AktivitätZuWerkzeugreferenz	Aktivität	Werkzeugreferenz	1 – 1..*
ArbeitsschrittnotenZuMethodenreferenz	Arbeitsschrittnoten	Methodenreferenz	1 – 1..*
ArbeitsschrittnotenZuWerkzeugreferenz	Arbeitsschrittnoten	Werkzeugreferenz	1 – 1..*
ArbeitsschrittnotenZuThema	Arbeitsschrittnoten	Thema	1 – 1
AktivitätZuProdukt	Aktivität	Produkt	1 – 1

**Produktbeziehungen.** Die Produktbeziehungen (Abbildung 25) stellen die Beziehungen zwischen Produkten und anderen Elementen des V-Modells her. Im Wesentlichen werden hierbei standardmäßig die Zuordnung von Themen zu Produkten, die Zuordnung von Produkten zu Entscheidungspunkten und die Zuordnung von externen Kopiervorlagen zu Produkten vorgenommen.

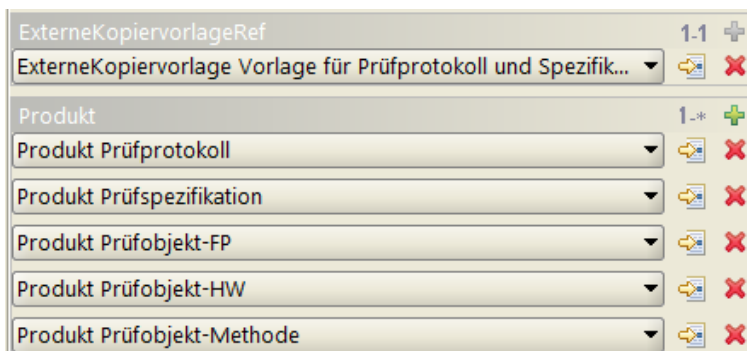
**Hinweis:** Die Zuordnung von Produkten zu Entscheidungspunkten werden nun als eigener Beziehungstyp modelliert, um im Falle einer Erweiterung/Anpassung des V-Modells um neue Entscheidungspunkte, eine Zuordnung von Produkten des Referenzmodells zu den neuen Entscheidungspunkten zu ermöglichen, ohne die vordefinierten Produkte anpassen zu müssen.



**Abbildung 25 Explizite Beziehungstypen, Gruppe: Produktbeziehungen**

Die externen Kopiervorlagen werden analog zu den Produktabhängigkeitsbeziehungen gehandhabt. Das heißt, dass:

1. Eine externe Kopiervorlage unter dem Knoten *Vorgehensbaustein/ExterneKopiervorlagen* angelegt werden muss, die die Kopiervorlage referenziert.
2. Unter dem Knoten *Vorgehensbaustein/Beziehungen/Produktbeziehungen* eine entsprechende Beziehung angelegt werden muss, die die Vorlage mit einem oder mehreren Produkten assoziiert.



Auch diese hat wieder den Vorteil der gesteigerten Flexibilität bei einer Anpassung bzw. Erweiterung eines V-Modells. Eine externe Kopiervorlage, z.B. für eine Prüfspezifikation kann in verschiedenen Vorgehensbausteinen verschiedenen Produkten zugewiesen werden. So kann z.B. eine Vorlage verwendet werden, um mehrere Produkttypen des V-Modells abzudecken (s.o.).

Beziehung	Quellentyp	Zieltyp	Kardinalität
ProduktZuEntscheidungspunkt	Produkt	Entscheidungspunkt	1 – 1..*
ThemaZuProdukt	Thema	Produkt	1 – 1..*
ExterneKopiervorlageZuProdukt	Externe-Kopiervorlage	Produkt	1 – 1..*



## 2.4 Das Paket Dynamik

Das Paket Dynamik beschreibt die Struktur aller Regeln und Elemente zur Konstruktion von Projektdurchführungsstrategien. Damit werden die zeitlichen Vorgaben und Spielräume des Vorgehensmodells erfasst, die die Reihenfolge der zu erstellenden Ergebnisse regeln.

Element	Beschreibung
PDS-Spezifikation	<p>Eine PDS-Spezifikation beschreibt die Anforderungen an eine Projektdurchführungsstrategie hinsichtlich enthaltener Entscheidungspunkte und deren Reihenfolge. Eine PDS-Spezifikation ist an einen Projekttyp gebunden und gibt somit den Rahmen für gültige Projektdurchführungsstrategien dieses Projekttyps vor,</p> <p><b>Achtung:</b> In der Version 1.3 des V-Modells sind die PDS-Spezifikationen zwar bereits im Metamodell enthalten, jedoch sind sie im Modell nicht ausgestaltet.</p>
Ablaufbaustein	<p>Ein Ablaufbaustein stellt das Äquivalent zum Vorgehensbaustein für ablaufbezogene Modellelemente dar. Ein Ablaufbaustein enthält Ablaufpunkte sowie Übergänge, Splits und Joins. Ein einzelner Ablaufbaustein beschreibt einen Teilablauf einer Projektdurchführungsstrategie. Ablaufbausteine können entweder Entscheidungspunkte referenzieren oder hierarchisch andere Ablaufbausteine einbinden.</p> <p>Jeder Ablaufbaustein muss genau einer Ablaufbausteinspezifikation genügen.</p>
Ablaufbausteinspezifikation	<p>Eine Ablaufbausteinspezifikation beschreibt Anforderungen an einen Ablaufbaustein. Diese Anforderungen enthalten Typen von Entscheidungspunkte sowie geforderte Reihenfolgen.</p> <p><b>Achtung:</b> In der Version 1.3 des V-Modells sind die Ablaufbausteinspezifikationen zwar bereits im Metamodell enthalten, jedoch wird der Regelungsanteil nicht angewendet und geprüft.</p>
Projekttypvariante	<p>Eine Projekttypvariante enthält alle Ablaufbausteine, die für die Projektdurchführungsstrategie erforderlich sind. Die finale Projektdurchführungsstrategie steht allerdings erst nach dem Tailoring (Abschnitt 4) fest. Initial bindet eine Projekttypvariante jedoch immer einen Ablaufbaustein ein, der den groben Rahmen für die finale Projektdurchführungsstrategie vorgibt.</p>
Entscheidungspunkt	<p>Ein Entscheidungspunkt ist Modellelement, das einen Meilenstein mit <i>Quality Gate</i> Eigenschaften beschreibt. Entscheidungspunkte werden durch Ablaufbausteine referenziert und durch diese wiederum in einer Projektdurchführungsstrategie verfügbar gemacht. Jedem Entscheidungspunkt sind über explizite Beziehungselemente (Abschnitt 2.3.7) verschiedene Produkte zugeordnet, die im Rahmen der Projektdurchführung zur Qualitätssicherung vorgelegt werden müssen.</p>

In Abbildung 26 ist die Metamodellsicht (UML-Sicht) auf das Paket *Dynamik* zu sehen. Die folgenden Abschnitte gehen anhand der oben stehenden Tabelle durch die einzelnen, zentralen Elemente diesen Pakets und erläutern seine Struktur.

**Hinweis:** Beachtet werden muss hier insbesondere das Element *Projekttypvariante*, das eigentlich dem Paket *Anpassung* zugeordnet ist. Die Projekttypvariante übernimmt im Metamodell die Rolle der *Projektdurchführungsstrategie*, die nicht mehr als explizites Modellelement vorliegt. Für die Ermittlung einer Projektdurchführungsstrategie erfolgt nun nach einem anderen Muster, auf das in Abschnitt 2.5 näher eingegangen wird.

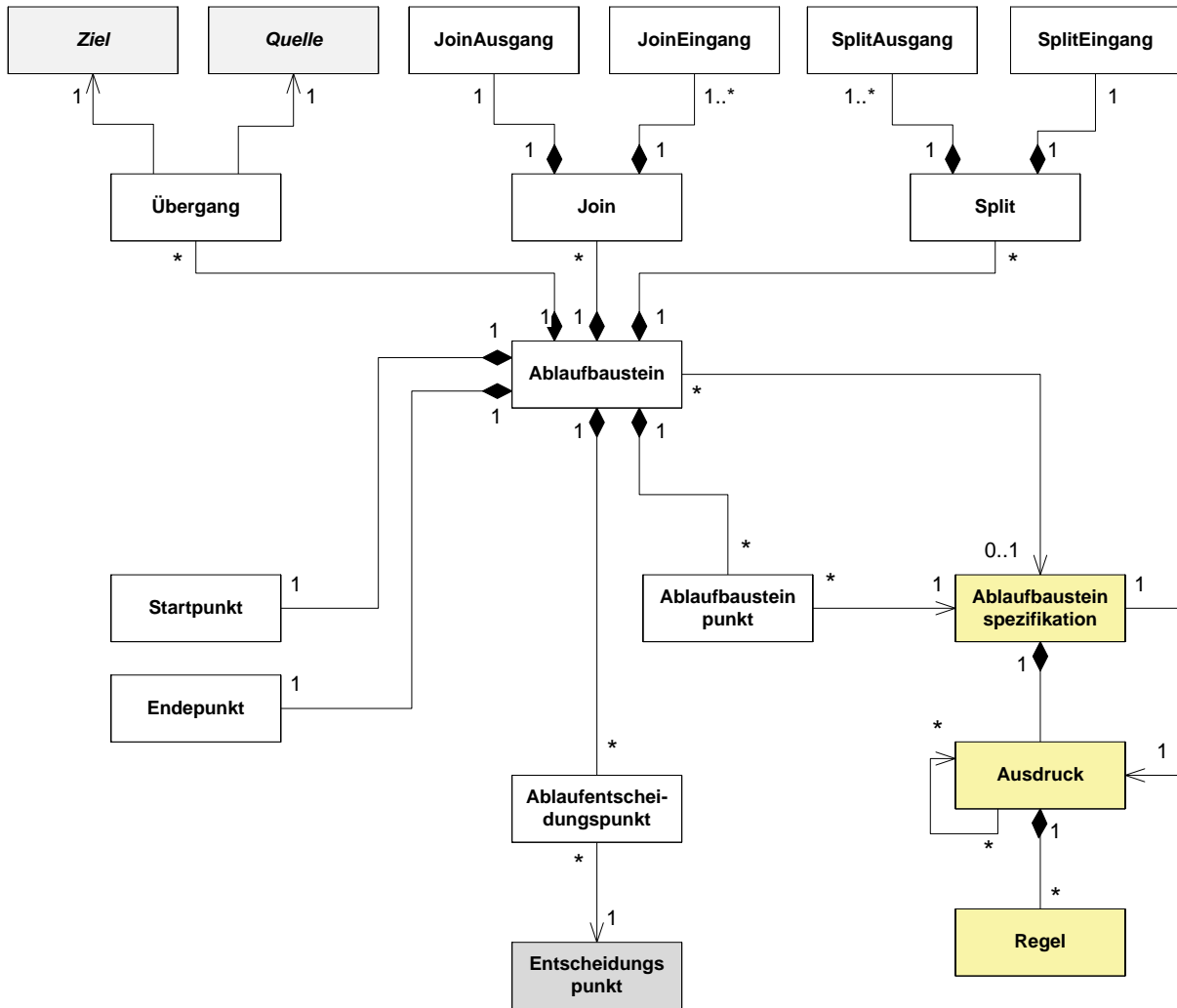


Abbildung 26 Das Paket Dynamik (Metamodellsicht)

### 2.4.1 PDS-Spezifikation

Projektdurchführungsstrategien werden im V-Modell auf der Grundlage eines ausgezeichneten Ablaufbausteins abgeleitet. Dem entsprechend, wird die Spezifikation einer (resultierenden) Projektdurchführungsstrategie auch auf der Grundlage einer ausgezeichneten Ablaufbausteinspezifikation erklärt (Abbildung 27).

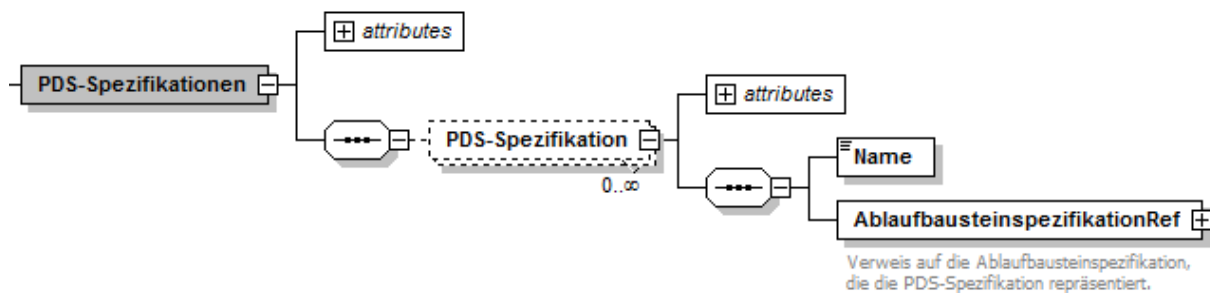


Abbildung 27 PDS-Spezifikationen (Sicht: XML-Schema)

Spezifikationen, sowohl für Projektdurchführungsstrategien als auch für Ablaufbausteine, stellen ein Mittel zur Verfügung, mit dem Regeln definiert werden können, denen modellierte Abläufe des V-Modells im Sinne der V-Modell-Konformität genügen müssen.

### 2.4.2 Ablaufbaustein

Ablaufbausteine entsprechen im Bereich der Abläufe den Vorgehensbausteinen. Ablaufbausteine stellen die modularen Fertigungs- und Tailoring-Einheiten des Dynamikpakets dar. Sie modellieren durch



*Ablauf(entscheidungs/baustein)punkte, Übergänge* sowie weitere Modellelemente die Ablaufstrukturen, aus denen im Rahmen des Tailorings die konkreten Projektdurchführungsstrategien ermittelt werden. Der (strukturelle) Aufbau des Ablaufbausteins (Abbildung 28) entspricht in seiner Komplexität dem Vorgehensbaustein, sodass der Ablaufbaustein im Folgenden anhand seiner Elemente verfeinernd erläutert wird.

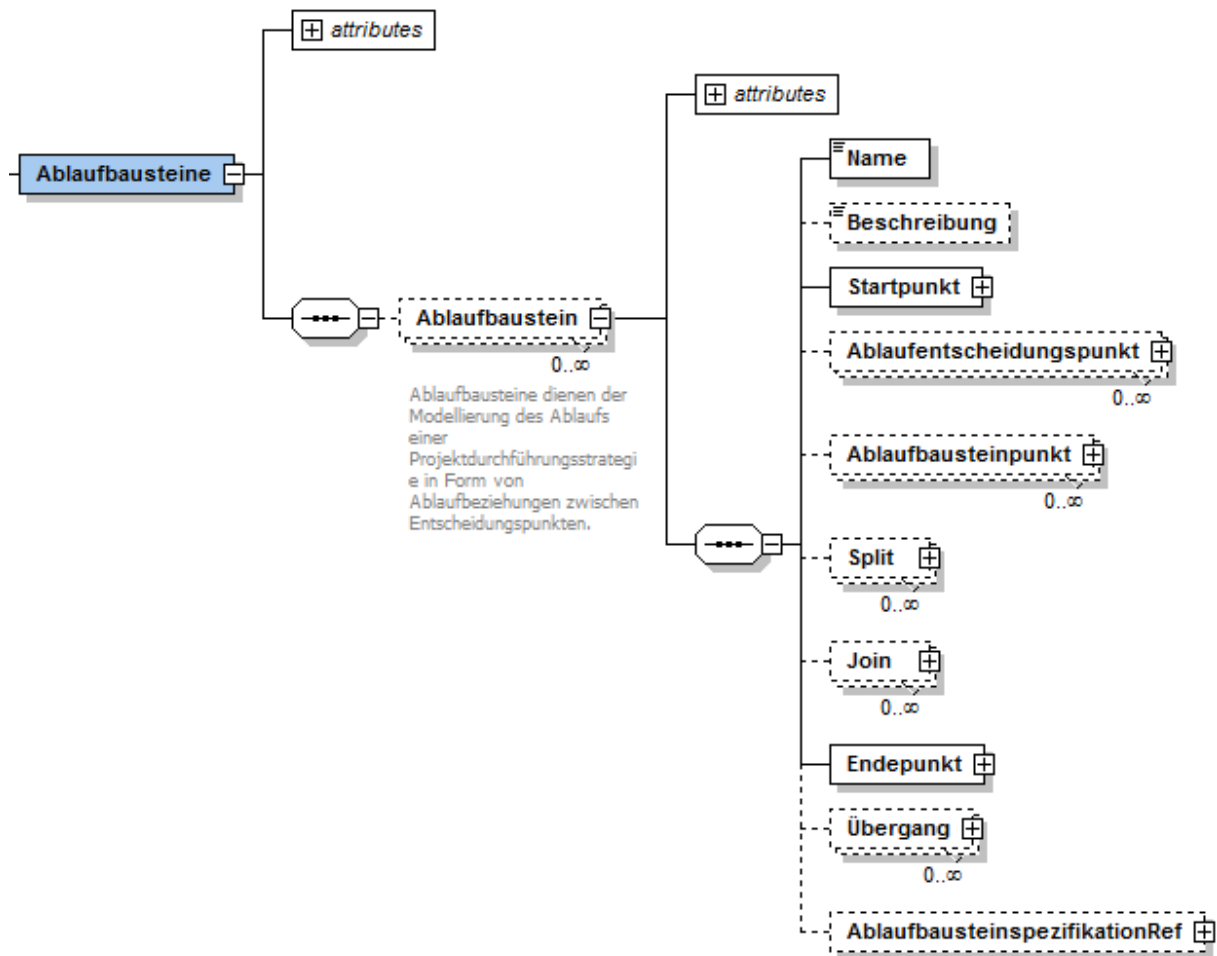


Abbildung 28 Ablaufbausteine (Sicht: XML-Schema)

**Allgemeines zum Ablaufbaustein.** Der grobe Aufbau eines Ablaufbausteins ist in Abbildung 28 zu sehen. Ein Ablaufbaustein besitzt neben den V-Modell-typischen Attributen wie *Name* und *Beschreibung* eine Reihe weiterer Elemente. Er besitzt immer genau einen *Startpunkt* und genau einen *Endepunkt*. Diese beiden speziellen Ablaufpunkte stellen eine notwendige Modellierungstechnik dar, die es gestattet, modulare Strukturen zu modellieren. Ablaufbausteine können *hierarchisch* geschachtelt werden.

Zwischen den *Start-* und *Endepunkten* können verschiedene *Ablaufentscheidungspunkte* bzw. *Ablaufbausteinpunkte* sowie *Splits* und *Joins* erstellt werden. Diese Elemente werden durch *Übergänge* miteinander verbunden. Die Modellierungstechnik dahinter ist an dem Konzept des UML-Aktivitätsdiagramms angelehnt.

Welcher Struktur ein Ablaufbaustein genügt, wird durch eine *Ablaufbausteinspezifikation* angezeigt. Diese entspricht in Analogie zur objektorientierten Programmierung der Schnittstelle des Ablaufbausteins (siehe auch Abschnitt 2.4.3).

Element	Beschreibung
Startpunkt	Der Startpunkt ist die erste Station im Ablauf eines Ablaufbausteins. Er kann entweder einen Entscheidungspunkt oder einen Ablaufbaustein repräsentieren, was während der Zusammenstellung der Projektdurchführungsstrategie errechnet wird.

Element	Beschreibung
Ablaufentscheidungspunkt	Ein Ablaufentscheidungspunkt ist eine Station im Ablauf einer Projektdurchführungsstrategie. Eine solche Station kann entweder einen Entscheidungspunkt oder einen Ablaufbaustein repräsentieren. Eine Reihenfolge wird mittels Übergängen modelliert.
Ablaufbausteinpunkt	
Split	Ein Split stellt eine gerichtete Verbindung zwischen Ablaufpunkten her. Ein Split hat genau einen Eingang und mindestens einen Ausgang. Auf diese Weise lassen sich parallele Abläufe modellieren. Alle Ausgänge eines Splits können in einem Projekt parallel geplant und durchgeführt werden.
Join	Ein Join stellt eine gerichtete Verbindung zwischen Ablaufpunkten her. Ein Join hat genau einen Ausgang und mindestens einen Eingang. Auf diese Weise lassen sich parallele Abläufe modellieren.
Endepunkt	Der Endepunkt ist die letzte Station im Ablauf des Ablaufbausteins. Er kann entweder einen Entscheidungspunkt oder einen Ablaufbaustein repräsentieren, was während der Zusammenstellung der Projektdurchführungsstrategie errechnet wird.
Übergang	Ein <i>NormalerÜbergang</i> stellt eine einfache gerichtete Verbindung zwischen zwei Ablaufpunkten her. Der Übergang kann mit einer Beschreibung versehen werden.

**Ablaufentscheidungspunkt.** Ein Ablaufentscheidungspunkt (Abbildung 29) referenziert einen Entscheidungspunkt, sodass dieser in der Projektdurchführungsstrategie eingebunden wird.

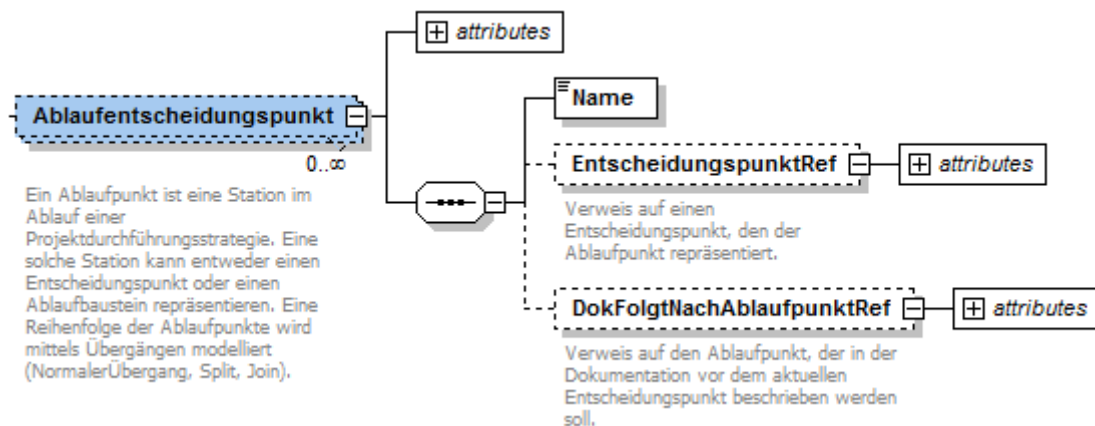


Abbildung 29 Ablaufbausteine, Bereich Ablaufentscheidungspunkt (Sicht: XML-Schema)

Die Dokumentation, die den Pfad zum Erreichen des Entscheidungspunkts beschreibt, ist in den Übergängen zwischen den einzelnen Ablaufbaustein-/Ablaufentscheidungspunkten hinterlegt. Die Position, an die die Dokumentation in der Prozessdokumentation generiert wird, wird im Element *DokFolgtNachAblaufpunktRef* angegeben. Das Element *DokFolgtNachAblaufpunktRef* steuert, an welcher Stelle in der Dokumentation dieser Dokumentationsteil hinterlegt wird. Es ist wichtig zu wissen, dass innerhalb eines Ablaufbausteins genau ein Element an dieser Stelle eine leere Referenz aufweisen muss. Ferner müssen die übrigen Elemente eine *zyklenfreie Liste bilden*, die die Reihenfolge der Ablaufknoten festlegt. Das Element mit der leeren Referenz wird immer als erstes für diesen Ablaufbaustein in die Dokumentation generiert.

**Beispiel:** Betrachten wir zwei Ablaufentscheidungspunkte A und B. A ist gegeben und referenziert den Entscheidungspunkt *Projekt beauftragt*; B wird neu angelegt. In B wird z.B. der Entscheidungspunkt *Iteration geplant* referenziert. Die Dokumentation, die das Erreichen dieses Entscheidungspunkts beschreibt, soll nach der Dokumentation des Ablaufentscheidungspunkts A erstellt werden, also den Weg von *Projekt beauftragt* nach

*Iteration geplant* beschreiben. Demnach ist im Element *DokFolgtNachAblaufpunktRef* das Ablaufentscheidungspunkts B eine Referenz auf A zu setzen.

**Ablaufbausteinpunkt.** Während Ablaufentscheidungspunkte nicht mehr verfeinerbar sind und somit quasi einen konkreten Ablauf beschreiben, referenzieren *Ablaufbausteinpunkte* (Abbildung 30) weitere *Ablaufbausteinspezifikationen*. Da Ablaufbausteine immer einer Ablaufbausteinspezifikation genügen müssen, werden hier also weitere Ablaufbausteine eingebunden. Dies geschieht jedoch erst zum Zeitpunkt des Tailorings, wenn konkrete Ablaufbausteine zu einer Projektdurchführungsstrategie zusammengefasst werden.

Durch die Einbindung weiterer Ablaufbausteine durch einen anderen Ablaufbaustein werden diese hierarchisch strukturiert.

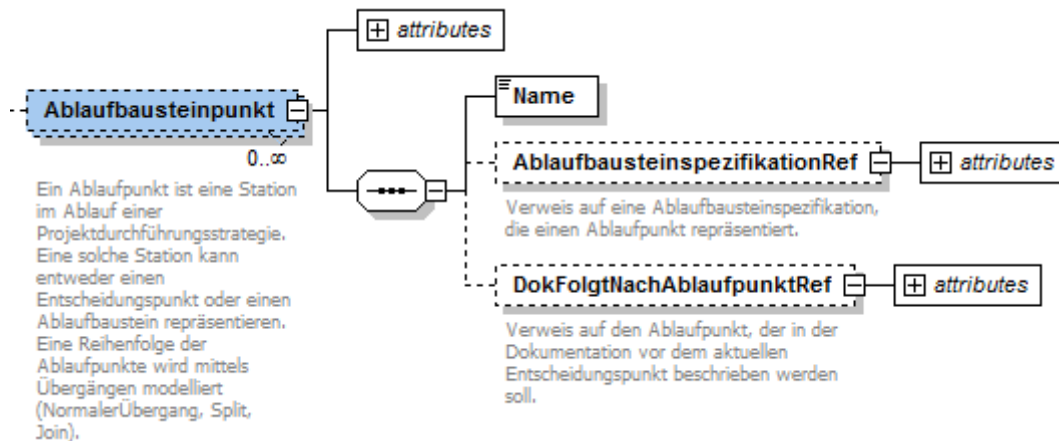
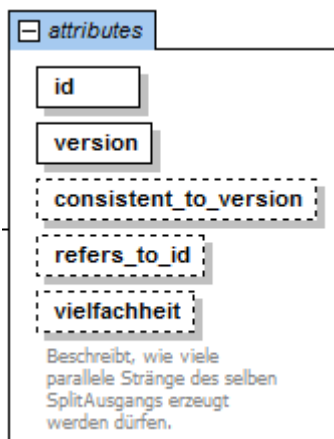


Abbildung 30 Ablaufbausteine, Bereich Ablaufbausteinpunkt (Sicht: XML-Schema)

Die Dokumentation im Kontext eines Ablaufbausteinpunkts wird nach demselben Muster erzeugt, wie bei Ablaufentscheidungspunkten.

**Split.** Splits beschreiben Aufteilungen des Ablaufs als Planungselement. Splits verfügen über genau einen *SplitEingang* und mindestens einen (oder viele) Elemente vom Typ *SplitAusgang*. Jeder Splitausgang kann im Rahmen einer Parallelisierung separat geplant werden. Zusätzlich besitzt der Splitausgang ein Attribut *vielfachheit*, das aussagt, wie viele parallele Stränge für jeden Splitausgang zulässig sind.



**Hinweis:** Das in früheren Versionen des V-Modells gebräuchliche Konzept des Parallelablaufs wird in der vorliegenden Version des V-Modell-Metamodells nicht mehr unterstützt. Die Parallelisierung der Abläufe erfolgt ausschließlich auf Grundlage der Splits und Joins.

Ein überschaubares Beispiel für die Parallelisierung von Abläufen findet sich im Auftraggeber-Projekttyp und dort in der Variante zur Vergabe von Teilprojekten an mehrere Auftragnehmer (Abbildung 31).

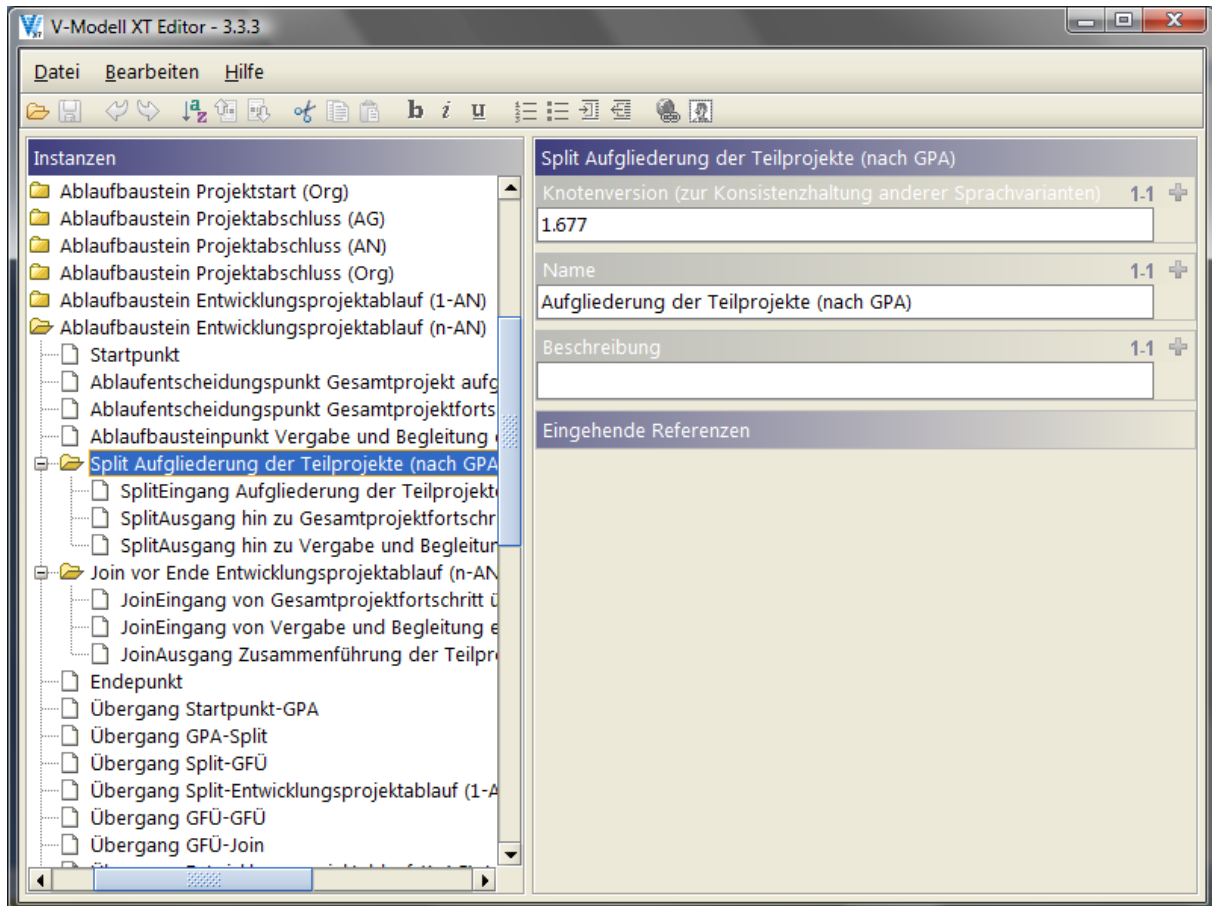
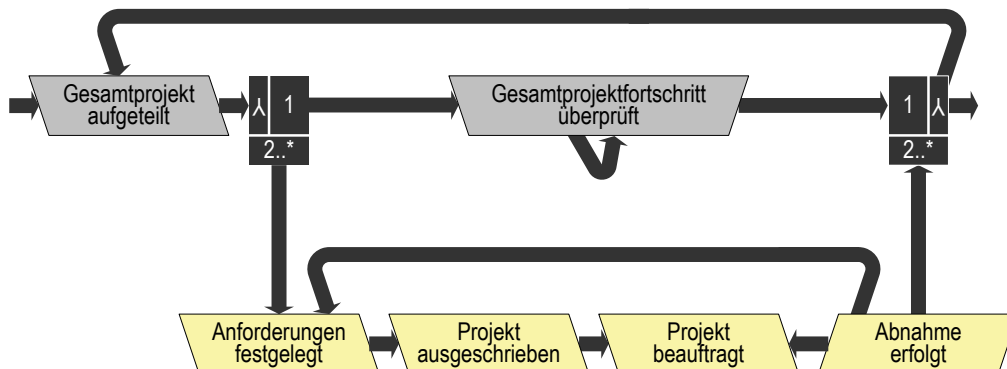


Abbildung 31 Ausschnitt aus dem Entwicklungsablauf für n-Auftragnehmer im AG-Projekt

Hier gibt es neben Start- und Endepunkt zwei Ablaufentscheidungs- und einen Ablaufbausteinpunkt. Der durch diesen Ablaufbaustein modellierte Teilablauf hat folgendes Aussehen:



Die hervorgehobenen Entscheidungspunkte werden durch den referenzierten Ablaufbaustein(punkt) eingebracht, die beiden Entscheidungspunkte werden durch die Ablaufentscheidungs- referenziert. In Abbildung 31 sind der Split und der Join zu sehen, die einmal die Teilprojektaufteilung und die Teilprojektzusammenführung realisieren. Zu sehen sind der Spliteingang und die beiden Splitausgänge, aber auch der Joineingang und die beiden Joineingänge.

Die Verbindungen zwischen diesen Elementen und den Ablaufbaustein- bzw. Ablaufentscheidungs- punkten werden durch die Übergänge hergestellt (Abbildung 31, unten links). Übergänge werden entweder direkt zwischen den Ablaufbaustein- bzw. Ablaufentscheidungs- punkten hergestellt. Übergänge die zu einem Split führen werden zwischen z.B. einem Ablaufentscheidungs- und einem Spliteingang hergestellt und weiterhin zwischen Splitausgängen und den jeweiligen folgenden Ablaufbaustein- bzw. Ablaufentscheidungs- punkten.

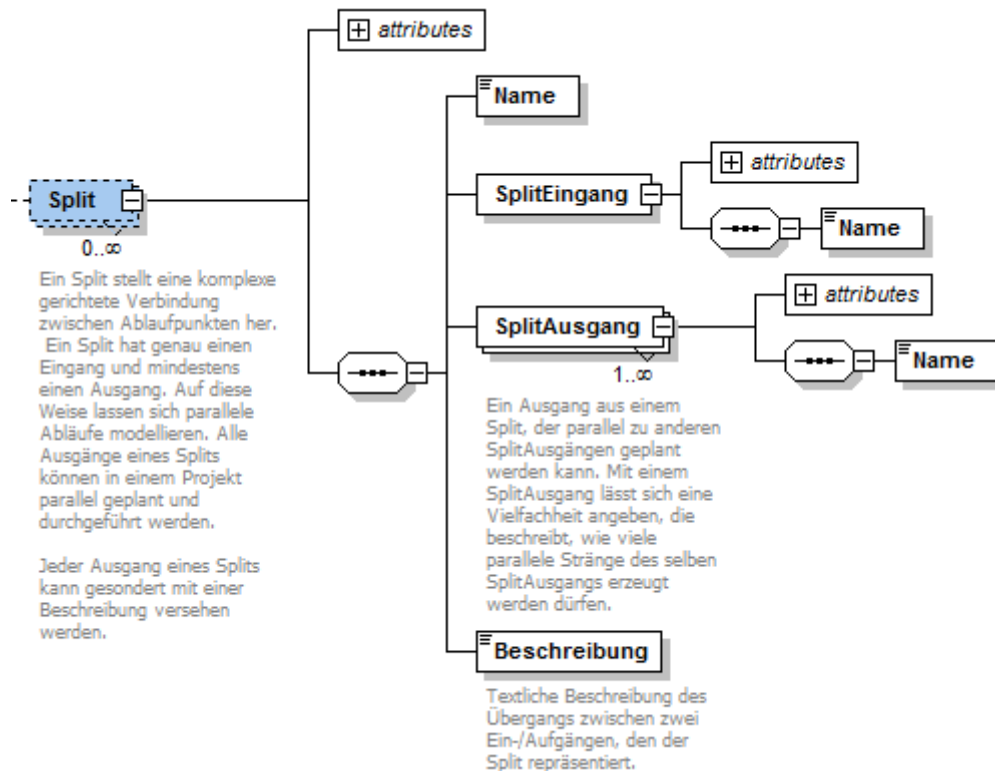


Abbildung 32 Ablaufbausteine, Bereich Split (Sicht: XML-Schema)

Ein Split kann eine Beschreibung enthalten, die den Split an sich und losgelöst von den Übergängen beschreibt. Diese Beschreibung wird im Rahmen der Generierung der Dokumentation der Projektdurchführungsstrategien anhand der Elemente der Ablaufbausteine berücksichtigt.

**Join.** Joins dienen dazu, Abläufe, die durch Splits aufgeteilt wurden, wieder zusammenzuführen (s.o.).

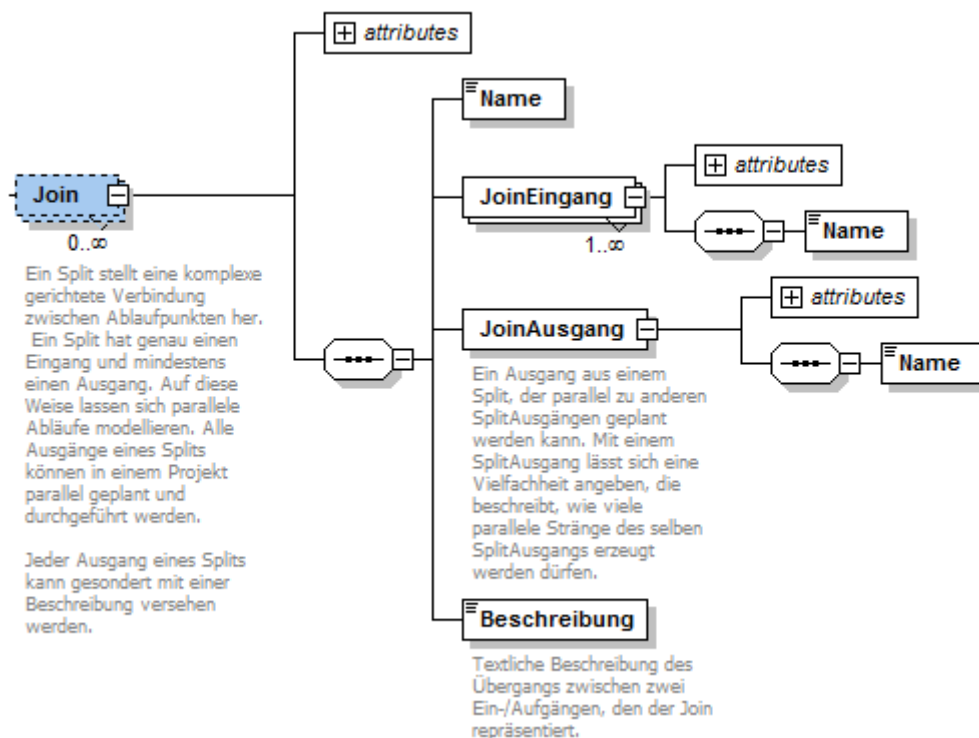


Abbildung 33 Ablaufbausteine, Bereich Join (Sicht: XML-Schema)

Ein Join hat mindestens einen (oder mehrere) Elemente vom Typ *JoinEingang* und genau einen *JoinAusgang*. Analog zum Splitausgang verfügt der Joineingang über der Attribut *vielfachheit*, das aus-

sagt, wie viele Stränge in diesem Eingang zusammengeführt werden dürfen oder müssen. Joins verfügen ebenfalls über ein Beschreibungselement, das analog zum Split behandelt wird.

**Hinweis:** Ein Split modelliert eine Aufteilung eines Ablaufs nach dem 1:n-Muster, also ein Eingang wird auf n Ausgänge (mit  $n \geq 1$ ) gesplittet. Umgekehrt führt der Join eine n:1-Zusammenführung (mit  $n \geq 1$ ) durch. Eine möglicherweise gewünschte Synchronisation, also eine n:m-Modellierung ist durch eine geschickte Join-Split-Kombination ebenfalls möglich.

**Übergang.** Übergänge (Abbildung 34) modellieren die Verknüpfungen von *Ablaufentscheidungspunkten*, *Ablaufbausteinpunkten* sowie *Splits* und *Joins*. Innerhalb eines Ablaufbausteins ist der *Startpunkt* immer die erste Station und der *Endepunkt* immer die letzte Station im Ablauf. Die Pfade zwischen diesen Punkten werden über die *Übergänge* festgelegt.

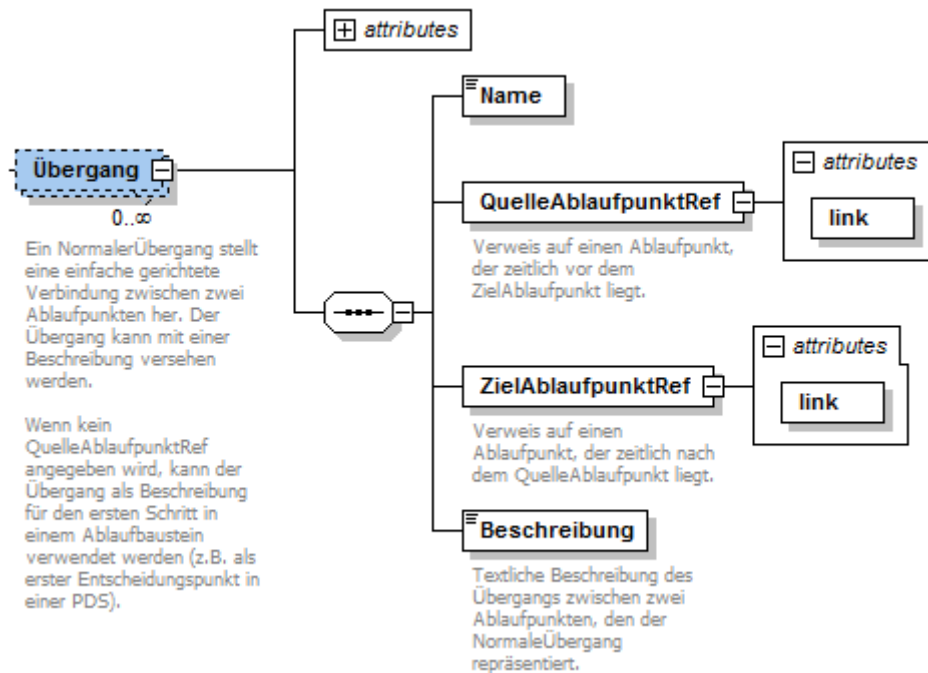


Abbildung 34 Ablaufbausteine, Bereich Übergang (Sicht: XML-Schema)

Übergänge sind einfache Verbindungen, die immer genau zwei Elemente (*Ablaufentscheidungspunkten*, *Ablaufbausteinpunkten* sowie *Splits* und *Joins*) miteinander verknüpfen. Sind Mehrfachverknüpfungen erforderlich, so sind Elemente vom Typ *Split* und *Join* einzuplanen.

Übergänge verfügen über ein Beschreibungselement. Diese Beschreibung wird im Rahmen der Generierung der Dokumentation der Projektdurchführungsstrategien ausgewertet und gemäß der Konfiguration der einzelnen *DokFolgtNachAblaufpunktRef*-Elemente der Ablaufbausteine an die passenden Stellen generiert.

### 2.4.3 Ablaufbausteinspezifikation

Analog zu PDS-Spezifikationen beschreiben Ablaufbausteinspezifikationen (Abbildung 35) die Struktur von Ablaufbausteinen, d.h. welche Entscheidungspunkte enthalten sein sollen und in welcher Reihenfolge sie prinzipiell vorliegen müssen.

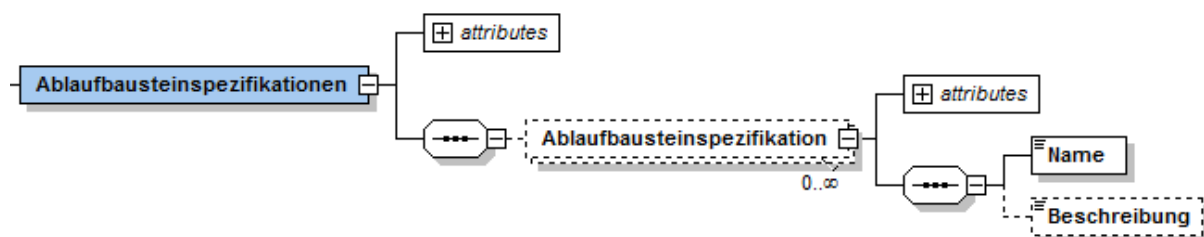


Abbildung 35 Ablaufbausteinspezifikationen (Sicht: XML-Schema)



## 2.4.4 Projekttypvariante

Projekttypvarianten sind ein Bestandteil des Pakets *Anpassung* und werden im Abschnitt 2.5 detailliert beschrieben. Dennoch sind sie auch hier von Interesse, da sie ein Element *Projektdurchführungsstrategie* (Abbildung 36) enthalten. Dieses zeigt auf einen ausgezeichneten *Ablaufbaustein*, der als Aufhänger für die Projektdurchführungsstrategie dient.

**Hinweis:** Anders als in früheren Versionen des V-Modells sind Projektdurchführungsstrategien als Metamodellelemente *nicht mehr explizit modelliert*. Eine Projektdurchführungsstrategie auf der Ebene des Metamodells wird durch einen ausgezeichneten Ablaufbaustein repräsentiert. Dieser dient als Container für weitere Ablaufbausteine, die die Inhalte der Projektdurchführungsstrategie enthalten. Wie im Abschnitt 2.4.2 gezeigt, können Ablaufbausteine hierarchisch strukturiert sein. Diese Fähigkeit nutzt das neue Metamodell und behandelt das Ablaufkonzept nun einheitlich durch Ablaufbausteine. Der Begriff Projektdurchführungsstrategie bleibt jedoch erhalten. Die Projektdurchführungsstrategie wird durch die Projekttypvarianten maßgeblich beeinflusst und im Rahmen des Tailorings abschließend festgelegt.

Weitere Informationen zu Projekttypvarianten sind im Abschnitt 2.5 zu finden. Informationen zur Ableitung einer Projektdurchführungsstrategie sind im Abschnitt 4 zu finden.

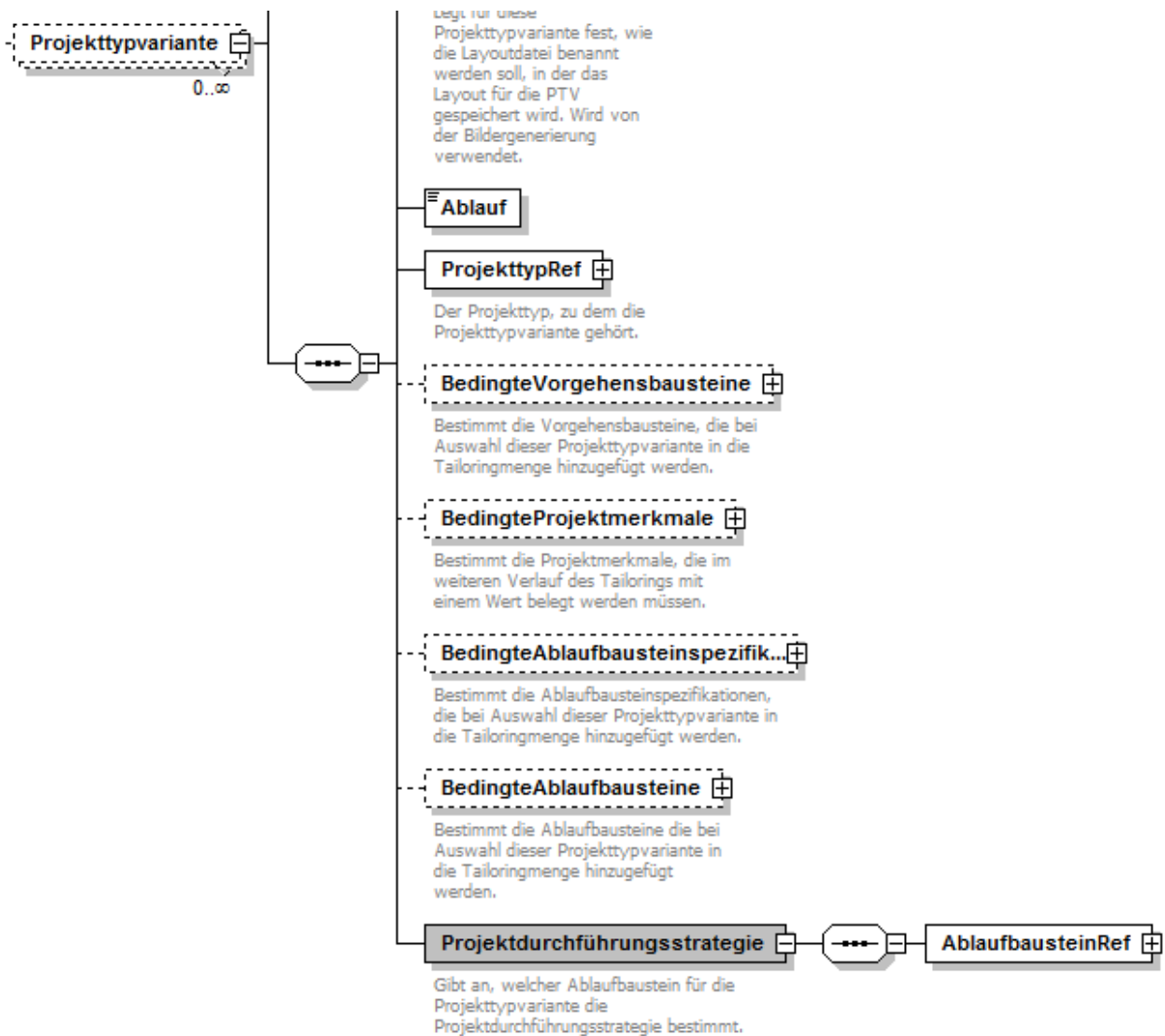


Abbildung 36 Projekttypvarianten, insb. Projektdurchführungsstrategie (Sicht: XML-Schema)

### 2.4.5 Entscheidungspunkt

Entscheidungspunkte dienen im V-Modell dazu, *Quality Gates* zu definieren. Dies sind definierte Meilensteine, zu denen verschiedene Produkte einer qualitätsgesichert vorliegen müssen.

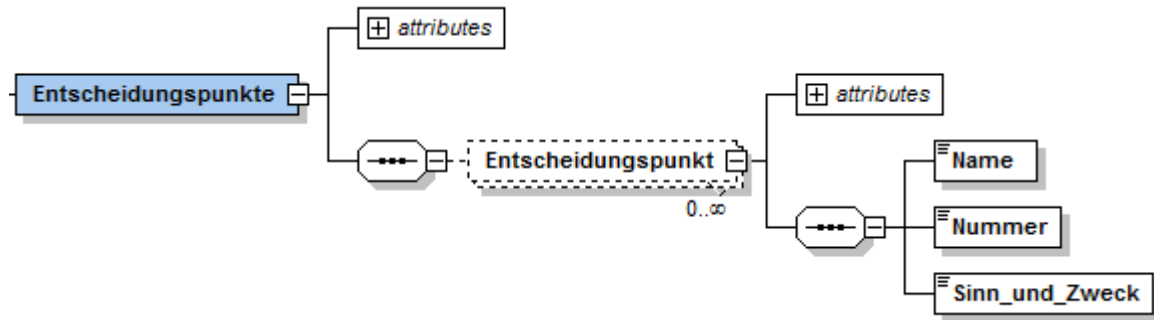


Abbildung 37 Entscheidungspunkte (Sicht: XML-Schema)

Entscheidungspunkte sind im V-Modell unmittelbar unterhalb der *V-Modellvariante* lokalisiert. Sie besitzen den in Abbildung 37 gezeigten strukturellen Aufbau. Entscheidungspunkte verfügen neben den üblichen Attributen über die Elemente *Name* und *Nummer*, die analog zu Produkten und Aktivitäten zu behandeln sind. Das Element *Sinn\_und\_Zweck* enthält eine Beschreibung des Entscheidungspunkts.

Abhängigkeiten zwischen Produkten und Entscheidungspunkten werden über spezielle Beziehungen (Abschnitt 2.3.7) hergestellt. Um in einer Projektdurchführungsstrategie berücksichtigt zu werden, werden die Entscheidungspunkte von Ablaufentscheidungspunkten (Abschnitt 2.4.2) gebunden.

## 2.5 Das Paket Anpassung

Das Paket Anpassung enthält alle Elemente des V-Modell-Metamodells, die für seine Anpassung, insbesondere im Rahmen der projektspezifischen Anpassung (Tailoring), erforderlich sind. Die Abbildung 38 zeigt die UML-Metamodellsicht dieses Pakets mit den Beziehungen, die für die Anpassung relevant sind. Die Elemente, die in diesem Paket enthalten sind, stellen die Grundlage für das Tailoring (Abschnitt 4) dar. Die Strukturen, die hier hinterlegt sind, spiegeln sich unmittelbar im Tailoring-Prozess wider.

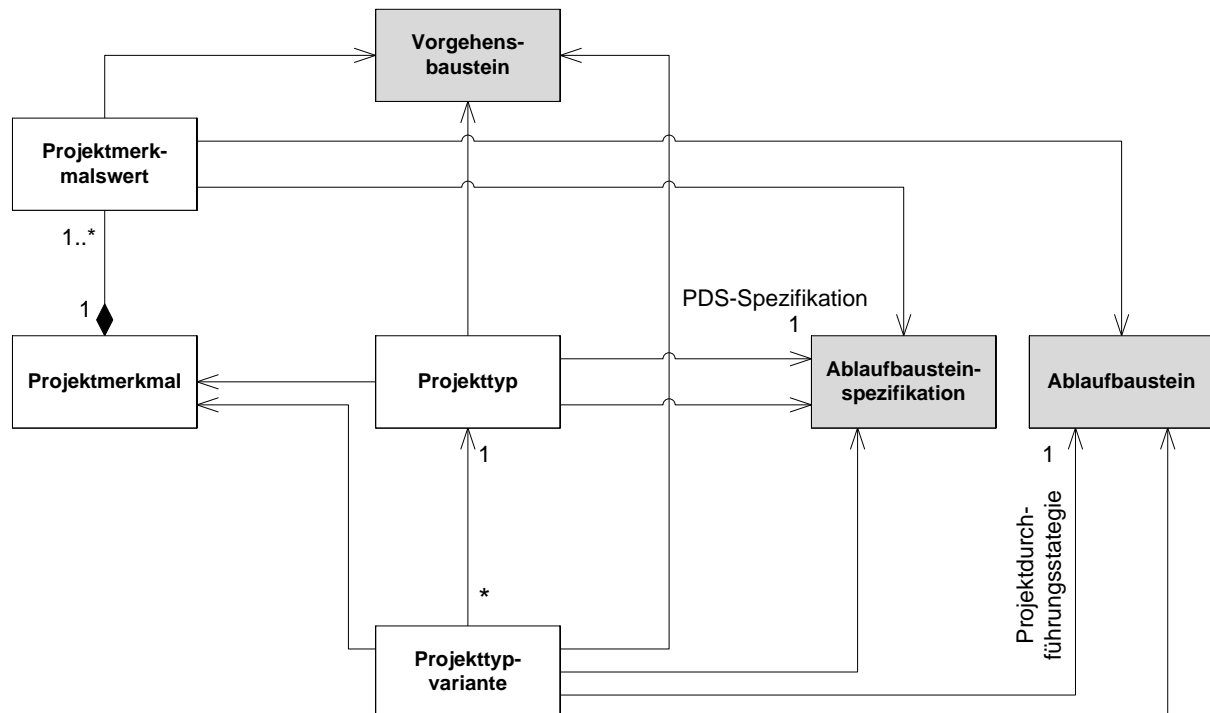


Abbildung 38 Grundlegende Elemente im Paket Anpassung (Metamodellsicht)

Die vier elementaren Elementtypen dieses Pakets sind unmittelbar unter der *V-Modellvariante* lokalisiert. In den folgenden Abschnitten werden diese vier Elemente detailliert vorgestellt. Hinweise zur methodischen Anwendung dieser Elemente finden sich in Abschnitt 4.



Element	Beschreibung
Projekttyp	Ein Projekttyp beschreibt eine Klasse von gleichartigen Projekten. Er macht Vorgaben hinsichtlich der verpflichtend zu verwendenden Vorgehensbausteine, zu belegender Projektmerkmale sowie zu verpflichtend anzuwendenden Ablaufbausteinspezifikationen.
Projekttypvariante	Eine Projekttypvariante ist eine Ausgestaltung eines Projekttyps und genau einem Projekttyp zugeordnet. Sie legt zusätzlich zum Projekttyp verpflichtende Vorgehensbausteine, Ablaufbausteinspezifikationen und Ablaufbausteine sowie zu belegende Projektmerkmale fest. Weiterhin legt die Projekttypvariante den Rahmen für die Projektdurchführungsstrategie.
Projektmerkmal	Ein Projektmerkmal ist eine Kontextinformation, die im Rahmen des Tailorings weitergehende Einschränkungen bzw. Spezialisierungen in Form zusätzlicher Vorgehensbausteine und Abläufe macht. Jedem Projektmerkmal können verschiedene Projektmerkmalswerte zugeordnet werden.
Projektmerkmalswert	Ein Projektmerkmalswert ist genau einem Projektmerkmal zugeordnet. Er kann verschiedene Vorgehensbausteine, Ablaufbausteinspezifikationen und Ablaufbausteine binden und somit das Tailoring-Ergebnis beeinflussen.

### 2.5.1 Projekttypen

Projekttypen (Abbildung 39) beschreiben Klassen gleichartiger Projekte. Sie sind strukturell unter einer *V-Modellvariante* lokalisiert. Neben den üblichen Attributen verfügen sie über die Elemente *Name*, *Nummer* und *Beschreibung*, die analog zu Produkten und Aktivitäten verwendet werden.

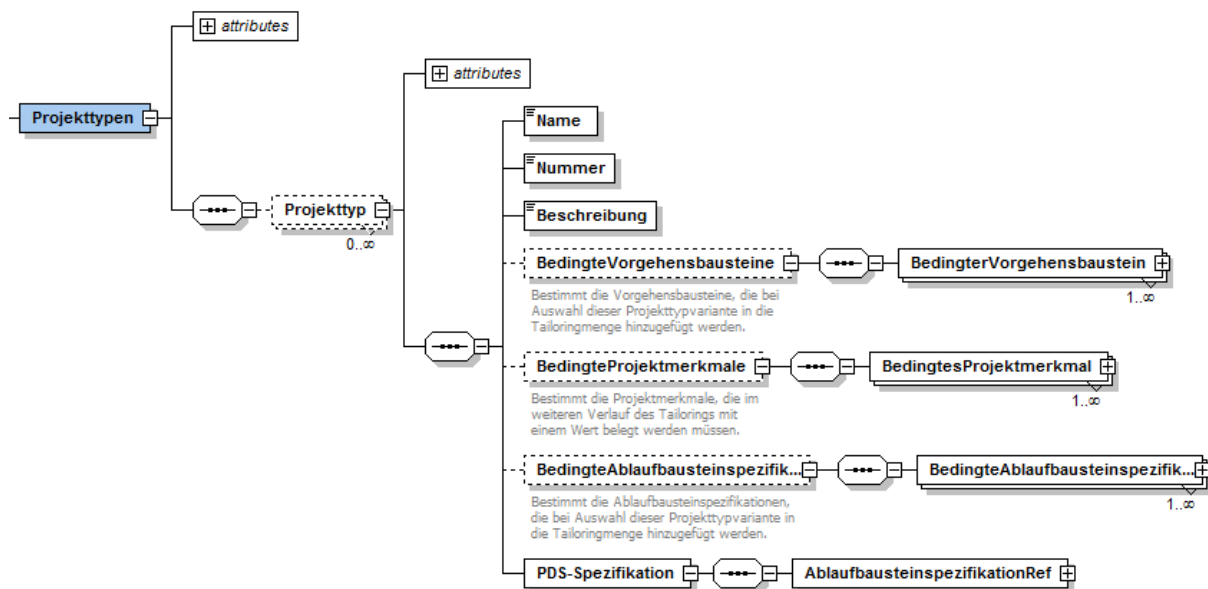


Abbildung 39 Projekttypen (Sicht: XML-Schema)

Darüber hinaus dienen die anderen Elemente dazu, Vorgaben für die abgedeckten Projekte zu machen. Konkret betrifft dies die Elemente:

- *BedingteVorgehensbausteine*, das eine Liste (1 oder viele) von für den Projekttyp verpflichtenden Vorgehensbausteinen enthält. Diese Vorgehensbausteine müssen im Modell bereits hinterlegt sein, um referenziert zu werden.

- *BedingteProjektmerkmale*, das eine Liste (1 oder viele) von Projektmerkmalen aufnimmt. Diese Projektmerkmale dienen zur weitergehenden Spezialisierung eines Projekttyps.
- Darüber hinaus macht ein Projekttyp über die Elemente *BedingteAblaufbausteinspezifikationen* (1 oder viele) und genau eine *PDS-Spezifikation* Vorgaben hinsichtlich der gültigen Abläufe. Für weitere Informationen zu Ablaufbausteinspezifikationen siehe Abschnitt 2.4.3.

## 2.5.2 Projekttypvarianten

Eine Projekttypvariante (Abbildung 40) ist eine Ausgestaltung eines Projekttyps. Sie ist immer genau einem Projekttyp zugeordnet. Neben den üblichen Attributen verfügt sie über die Elemente *Name*, *Nummer* und *Beschreibung*, die analog zu Produkten und Aktivitäten (Abschnitte 2.3.3 und 2.3.5) verwendet werden.

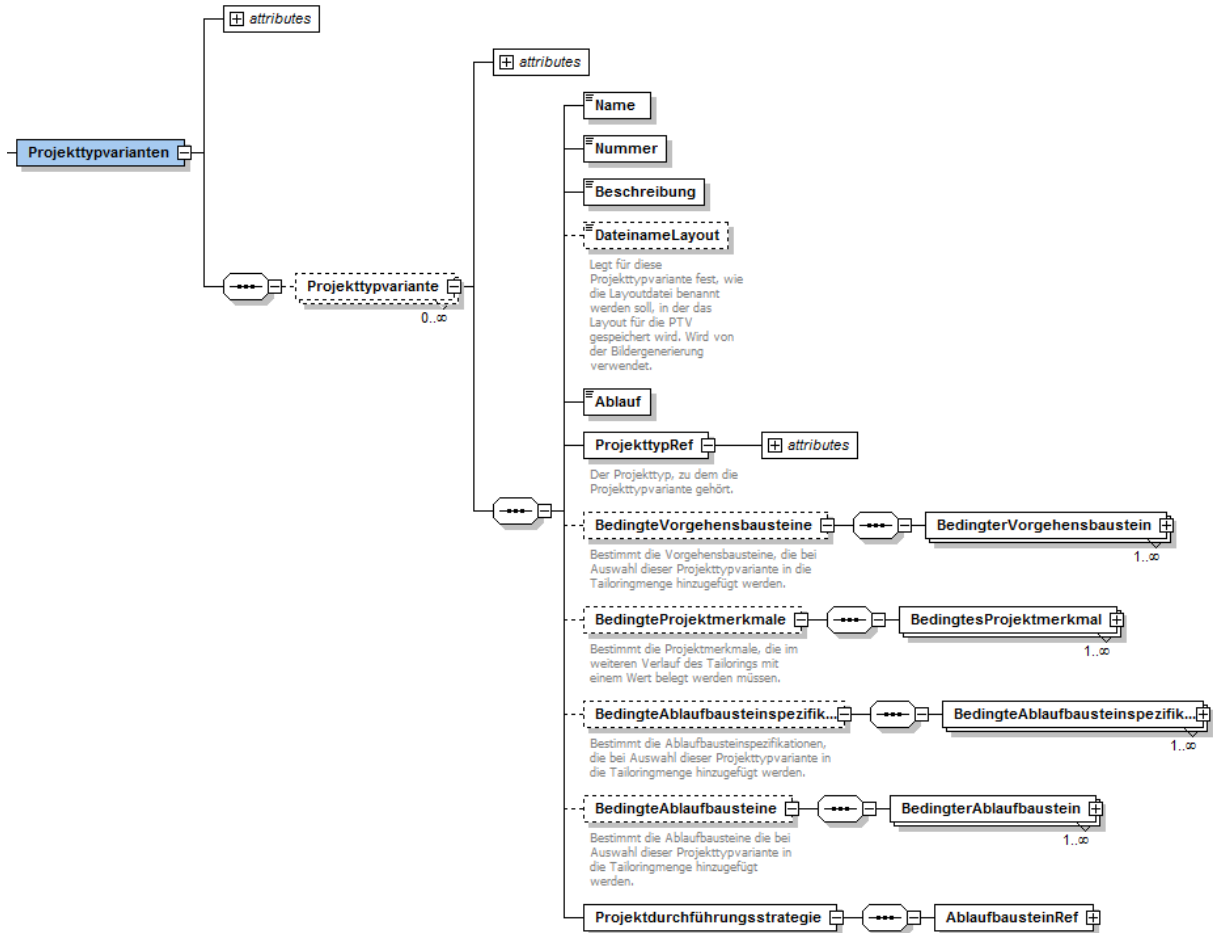


Abbildung 40 Projekttypvarianten (Sicht: XML-Schema)

Das Element *ProjekttypRef* verweist auf den Projekttyp, der durch die Projekttypvariante ausgestaltet wird. Analog zum Projekttyp kann die Projekttypvariante über die Elemente

- *BedingteVorgehensbausteine*,
- *BedingteProjektmerkmale* und
- *BedingteAblaufbausteinspezifikationen*

Vorgaben hinsichtlich Vorgehensbausteinen, Projektmerkmalen und Ablaufspezifikationen machen. Darüber hinaus bindet eine Projekttypvariante über das Element *BedingteAblaufbausteine* konkrete Ablaufbausteine ein, die jedoch den durch den Projekttyp und/oder die Projekttypvariante gebundenen Ablaufbausteinspezifikationen genügen müssen.

Über das Element *Projektdurchführungsstrategie* wird ein ausgezeichneter Ablaufbaustein referenziert. Dieser stellt die Grundlage für die Ermittlung der Projektdurchführungsstrategie im Rahmen des Tailorings dar.

Analog zum Projekttyp werden durch die unmittelbar gebundenen Elemente verbindliche Vorgaben gemacht. Variationsräume, z.B. optionale Vorgehensbausteine, werden ebenfalls durch Projektmerkmale festgelegt.

### 2.5.3 Projektmerkmale und Projektmerkmalswerte

Projektmerkmale und die dazugehörigen Projektmerkmalswerte werden durch Projekttypen und Projekttypvarianten gebunden und stellen zusätzliche Variationsmöglichkeiten bereit.

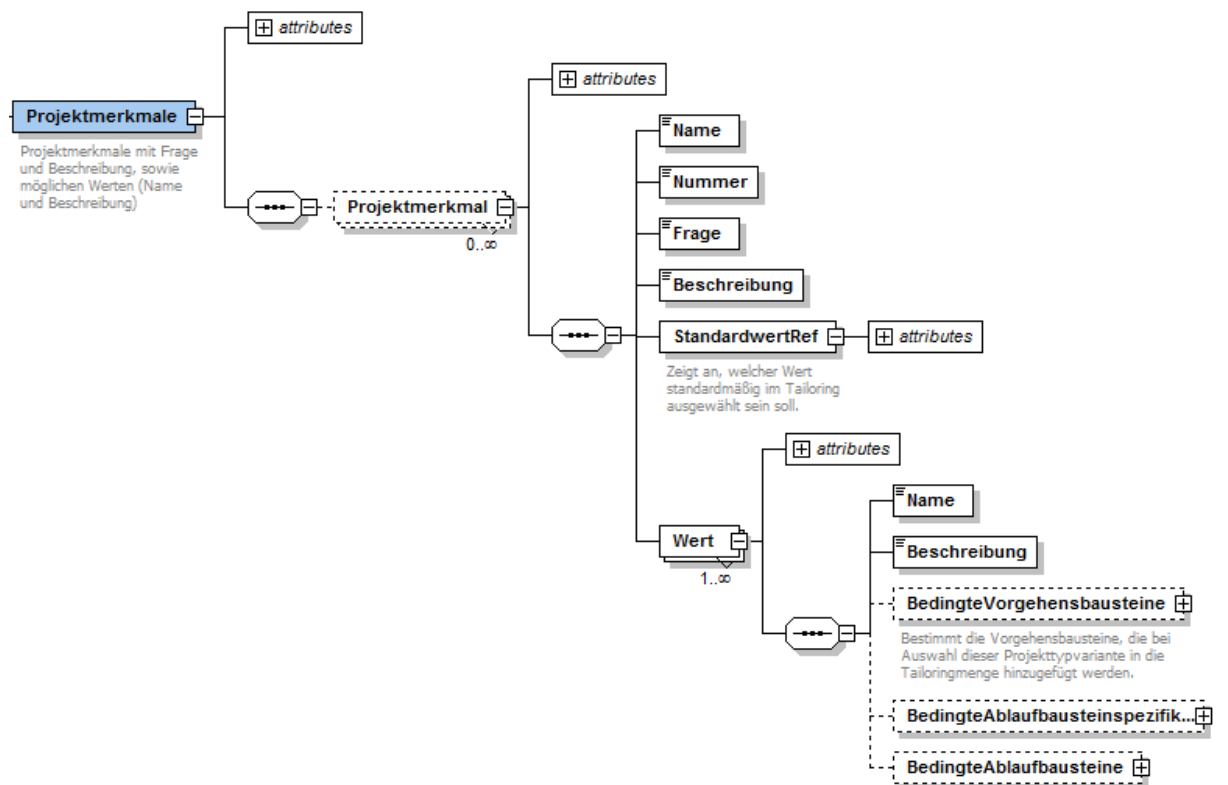
**Hinweis:** Das in früheren Versionen des V-Modells vorhandene Konzept des *optionalen Vorgehensbausteins* wird nicht mehr durch den Projekttyp festgelegt, sondern durch die Wertebelegungen der Projektmerkmale. Alle Vorgaben von Projekttyp und Projekttypvarianten sind stets verpflichtend. Optionen werden ausschließlich über Projektmerkmale modelliert.

Abbildung 41 zeigt die XML-Struktur des *Projektmerkmal*-Elements und die Einbettung der *Projektmerkmalswerte*. Ein Projektmerkmal hat neben den üblichen Attributen und den Elementen *Name*, *Beschreibung* und *Nummer* immer eine Frage, die den Anwender bei der Wertebelegung unterstützen soll.

Einem Projektmerkmal ist eine Menge Projektmerkmalswerte eindeutig zugeordnet. Jeder Projektmerkmalswert kann

- Vorgehensbausteine (Element: *BedingteVorgehensbausteine*),
- Ablaufspezifikationen (Element: *BedingteAblaufbausteinspezifikationen*) und
- Ablaufbausteine (Element: *BedingteAblaufbausteine*)

referenzieren. Mit der Auswahl eines Wertes im Rahmen des Tailorings werden dadurch *optionale* Vorgehensbausteine in Abhängigkeit einer konkreten Wertebelegung eingebunden, oder nicht (selbiges für optionale Abläufe).



**Abbildung 41 Projektmerkmale und Projektmerkmalswerte (Sicht: XML-Schema)**

Zusätzlich ist jedem Projektmerkmal ein Standardwert (Element: *StandardwertRef*) zugeordnet. Dieser Standardwert ist für das Tailoring und dort für die Bereitstellung einer *definierten Grundkonfiguration* erforderlich. Die genauen Zusammenhänge werden jedoch erst im Rahmen des Tailorings deutlich. Daher sei an dieser Stelle auf Abschnitt 4 für weitere Informationen verwiesen.

## 2.6 Das Paket Konventionsabbildung

Konventionsabbildungen (Abbildung 42) erleichtern die Positionierung von Standards und Normen mit dem V-Modell und gestatten Quereinsteigern den leichteren Einstieg in das V-Modell. Eine Konventionsabbildung ist gegliedert durch Bereiche (Element: *Bereich*).

Ein Bereich kann eine Reihe von *Begriffsabbildungen* enthalten. Eine Begriffsabbildung führt den zu erläuternden/abzubildenden Begriff ein, erläutert oder beschreibt ihn und stellt dann über verschiedene Referenzen Abbildungen zu Elementen des V-Modells her.

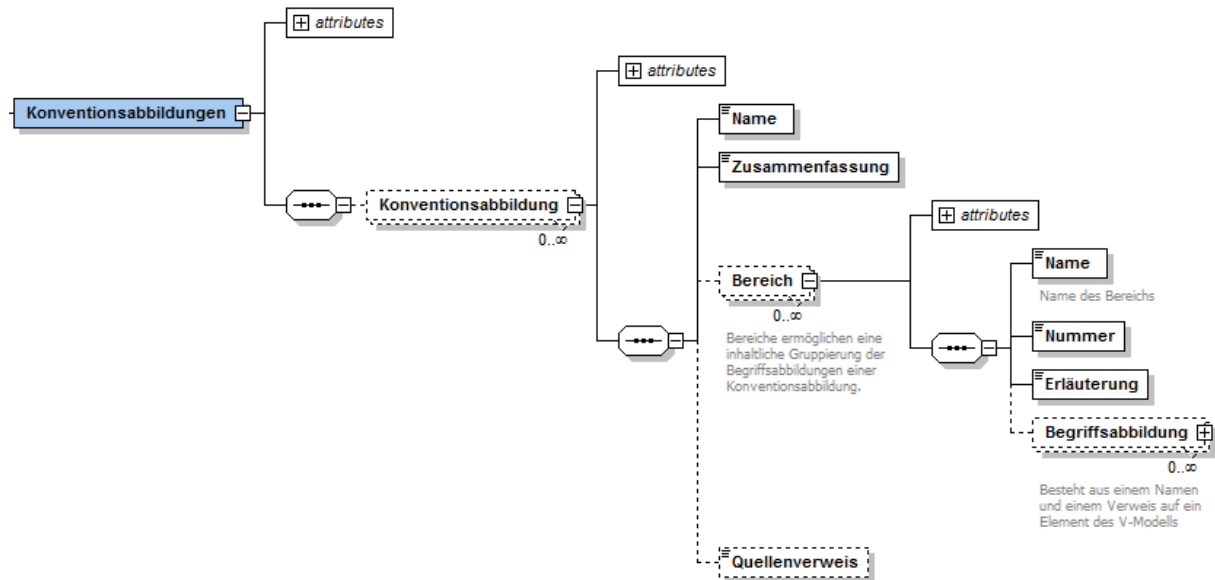


Abbildung 42 Konventionsabbildung (Sicht: XML-Schema)

Das V-Modell kennt folgende Abbildungsoptionen für einen Begriff im Kontext einer Konventionsabbildung:

Element	Beschreibung
wird_abgebildet_durchTeilRef	Ein Begriff wird abgebildet durch einen Teil der V-Modell Prozessdokumentation.
wird_abgebildet_durchKapitelRef	Ein Begriff wird abgebildet durch ein Kapitel in einem Teil der V-Modell Prozessdokumentation.
wird_abgebildet_durchAbschnittRef	Ein Begriff wird abgebildet durch einen Abschnitt in einem Kapitel der V-Modell Prozessdokumentation.
wird_abgebildet_durchVBRRef	Ein Begriff wird abgebildet durch einen vollständigen Vorgehensbaustein des V-Modells.
wird_abgebildet_durchPTVRef	Ein Begriff wird abgebildet durch eine Projekttypvariante des V-Modells – hier genauer durch die durch die Projekttypvariante implizierte Projektdurchführungsstrategie.
wird_abgebildet_durchEntscheidungspunktRef	Ein Begriff wird abgebildet durch einen Entscheidungspunkt.
wird_abgebildet_durchDisziplinRef	Ein Begriff wird abgebildet durch eine Disziplin und somit durch alle enthaltenen Produkte und Aktivitäten.
wird_abgebildet_durchProduktRef	Ein Begriff wird abgebildet durch ein Produkt des V-Modells.

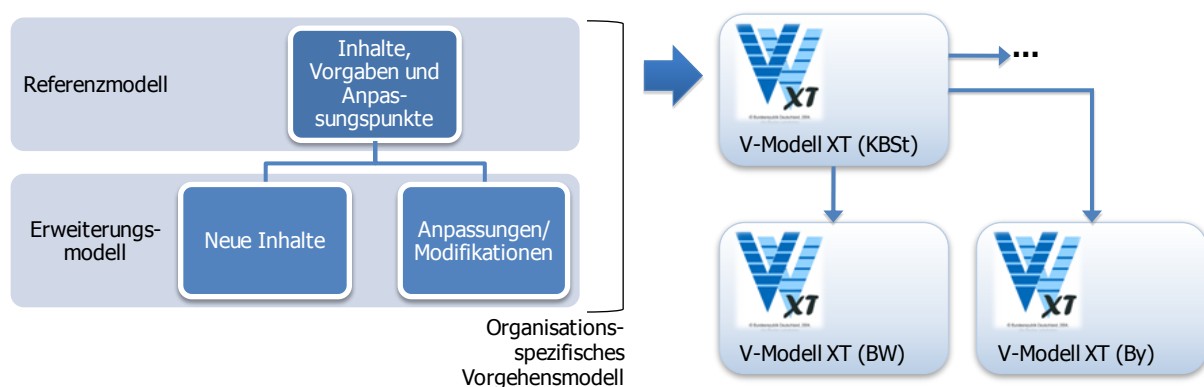
Element	Beschreibung
wird_abgebildet_durchThemaRef	Ein Begriff wird abgebildet durch ein Thema eines Produkts im V-Modell.
wird_abgebildet_durchAktivitätRef	Ein Begriff wird abgebildet durch eine Aktivität im V-Modell.
wird_abgebildet_durchArbeitsschrittRef	Ein Begriff wird abgebildet durch einen Arbeitsschritt im V-Modell.
wird_abgebildet_durchRolleRef	Ein Begriff wird abgebildet durch eine Rolle.
wird_abgebildet_durchMethodenreferenzRef	Ein Begriff wird abgebildet durch eine im V-Modell referenzierte Methode.
wird_abgebildet_durchWerkzeugreferenzRef	Ein Begriff wird abgebildet durch ein im V-Modell referenziertes Werkzeug.
wird_abgebildet_durchKonventionsabbildungRef	Ein Begriff wird abgebildet durch eine andere Konventionsabbildung.

### 3 Anpassungsmöglichkeiten des V-Modell XT

Im Vergleich zu früheren Versionen des V-Modells ist das neue Metamodell stärker auf Variabilität und Anpassbarkeit ausgerichtet. Dabei ist das Metamodell insgesamt auf eine stärker systematisierte Anpassung hin optimiert worden. Zentraler Bestandteil des Anpassungskonzepts ist eine *Aufteilung* des V-Modells in einen sogenannten *Referenzanteil* und einen *organisationsspezifischen Anteil*. Zwischen diesen beiden Teilmodellen sind vielfältige Beziehungen möglich, die der Erweiterung, Anpassung und Veränderung dienen. In diesem Abschnitt werden die dazu notwendigen Grundlagen und Konzepte erläutert.

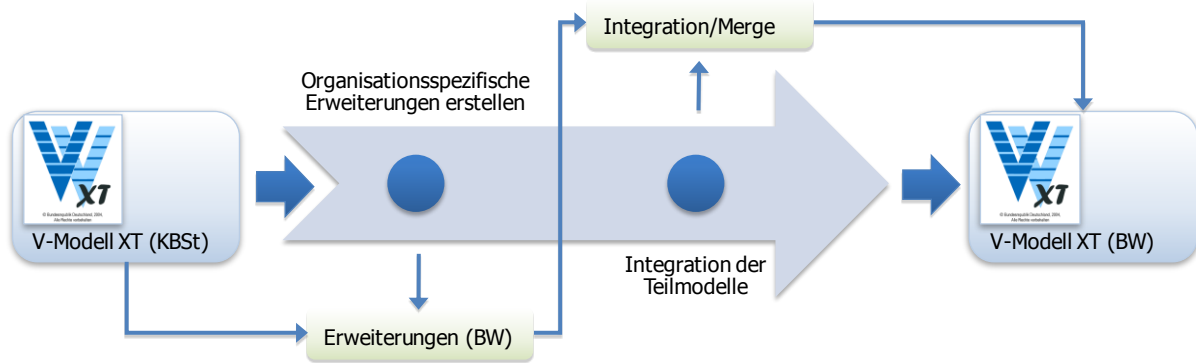
#### 3.1 Aufteilung in Referenz- und Organisationsanteile

Bei der Anpassung des V-Modells kam es in der Vergangenheit immer wieder zu Komplikationen, wenn das Standardmodell im Rahmen seiner Aktualisierungszyklen angepasst und fehlerbereinigt wurde und diese Änderungen in ein angepasstes Modell übernommen werden sollten. Um die Verwaltung entsprechender *Varianten* des V-Modells zu erleichtern und zu systematisieren wurde das V-Modell aufgeteilt.



- Ein *Referenzmodell* enthält alle Standardinhalte des V-Modells. Diese werden zentral gepflegt und entwickelt.
- In einem *organisationsspezifischen Erweiterungsmodell* werden alle Anpassungen des V-Modells hinterlegt, die im Rahmen einer organisationsspezifischen Anpassung erarbeitet wurden. Dieses Erweiterungsmodell liegt nun in physisch separierter Form vor.

Das Referenzmodell und das Erweiterungsmodell werden durch mitgelieferte Werkzeuge des V-Modells zum *organisationsspezifischen V-Modell* zusammengefasst.



Um dies umzusetzen, wurde das Konzept der *V-Modellvariante* eingeführt. Eine V-Modellvariante enthält alle Strukturen und Elemente, die ein V-Modell bilden. Auch das Referenzmodell ist eine (ausgezeichnete) V-Modellvariante.

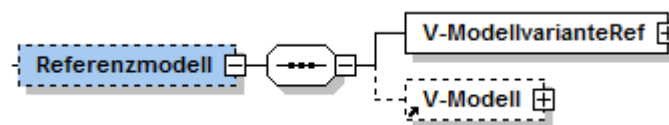


Abbildung 43 Erweiterte V-Modellvariante (Sicht: XML-Schema)

Jede V-Modellvariante enthält ein Element *Referenzmodell* (Abbildung 43), die auf ein weiteres, eigenständiges V-Modell-XML-Dokument verweist. Dieses externe V-Modell stellt das V-Modell dar, das erweitert bzw. angepasst werden soll.

**Das Anlegen einer Erweiterung einer V-Modellvariante im Sinne eines Erweiterungsmodells ist der bevorzugte und empfohlene Weg zur Anpassung des V-Modells.**

Das Referenzmodell ist üblicherweise schreibgeschützt und sollte im Rahmen einer organisationsspezifischen Anpassung *nicht* verändert werden. In den folgenden Abschnitten gehen wir auf die drei möglichen Anpassungsoptionen ein.

### 3.2 Erweiterungen des V-Modell XT

Mit dem Referenzmodell stehen die reichhaltigen Inhalte des V-Modells bereits zur Verfügung. In Situationen, in denen diese Standardinhalte den Anforderungen einer anpassenden Organisation weitgehend genügen, ist es ggf. nur erforderlich, z.B. zusätzliche Produkte oder Themen einzuführen. In diesem Fall handelt es sich um eine einfache *Erweiterung* des V-Modells.

Die Erweiterung ist der einfachste Anwendungsfall, da hier mithilfe der umfangreichen Optionen zur Modellierung von Beziehungen und Abhängigkeiten (Abschnitte 2.3.4 und 2.3.7) lediglich neue Modellelemente eingeführt werden, ohne bereits vorhandene zu verändern. Da auch das Erweiterungsmodell eine vollständige V-Modellvariante ist, stehen sämtliche Metamodellelemente zur Verfügung, um neue Modellelemente zu instanzieren. Standardmäßige bzw. übliche Erweiterungen sind z.B.:

- Neue Vorgehensbausteine
- Neue Rollen
- Neue Ablaufbausteine
- Neue Projekttypen und Projekttypvarianten
- Neue Projektmerkmale
- Neue Konventionsabbildungen
- ...

Diese Modellelemente können im Erweiterungsmodell (der erweiterten Modellvariante) neu angelegt und in das Beziehungsgeflecht des Referenzmodells integriert werden.

**Beispiel:** Ein Standardbeispiel ist das Anlegen einer neuen Projekttypvariante. Diese kann im Erweiterungsmodell angelegt werden und einen Projekttyp im Referenzmodell referenzieren. Im organisationsspezifischen V-Modell steht dann im Tailoring eine zusätzliche Projekttypvariante zur Auswahl bereit.

### 3.3 Anpassungen und Änderungen des V-Modell XT

Das V-Modell-Metamodell ist ab der Version 1.3 stärker auf den Aspekt der organisationsspezifischen Anpassung hin ausgerichtet. Dazu unterstützt es bereits auf der Ebene des Metamodells verschiedene Konzepte, die die Anpassung systematisieren.

Ein Aspekt, die Aufteilung in ein Referenz- und spezifische Erweiterungsmodelle, wurde im letzten Abschnitt bereits dargestellt. Die Aufteilung erfordert jedoch weitere Möglichkeiten, die über die einfache Ergänzung von Inhalten des Referenzmodells hinausgehen. üblicherweise haben anpassende Organisationen z.B. eigene Rollenmodelle oder spezifische Inhalte für bestimmte Produkte. Diesen Tatsachen trägt das V-Modell bereits auf der Ebene des Metamodells durch sogenannte *Änderungsoperationen* Rechnung.

Änderungsoperationen unterstützen die fein granulare Anpassung des V-Modells im Sinne von Änderungen an den Inhalten des Referenzmodells. So können z.B. eigene Rollenmodelle implementiert werden und über Änderungsoperationen Beziehungen zu Produkten aus dem Referenzmodell angepasst werden. In einem Erweiterungsmodell kann somit z.B. eine neue Rolle eingeführt werden, die mithilfe einer Änderungsoperation die Verantwortung für ein gegebenes Produkt aus dem Referenzmodell übernehmen kann. Die Änderungsoperationen sind dabei im Erweiterungsmodell lokalisiert, sodass für diese tiefgreifende Anpassung *keine Eingriffe ins Referenzmodell* erforderlich sind. Änderungsoperationen werden im Rahmen der Erstellung des organisationsspezifischen Vorgehensmodells aus dem Referenz- und dem Erweiterungsmodell automatisch errechnet und konsistent zur Verfügung gestellt.

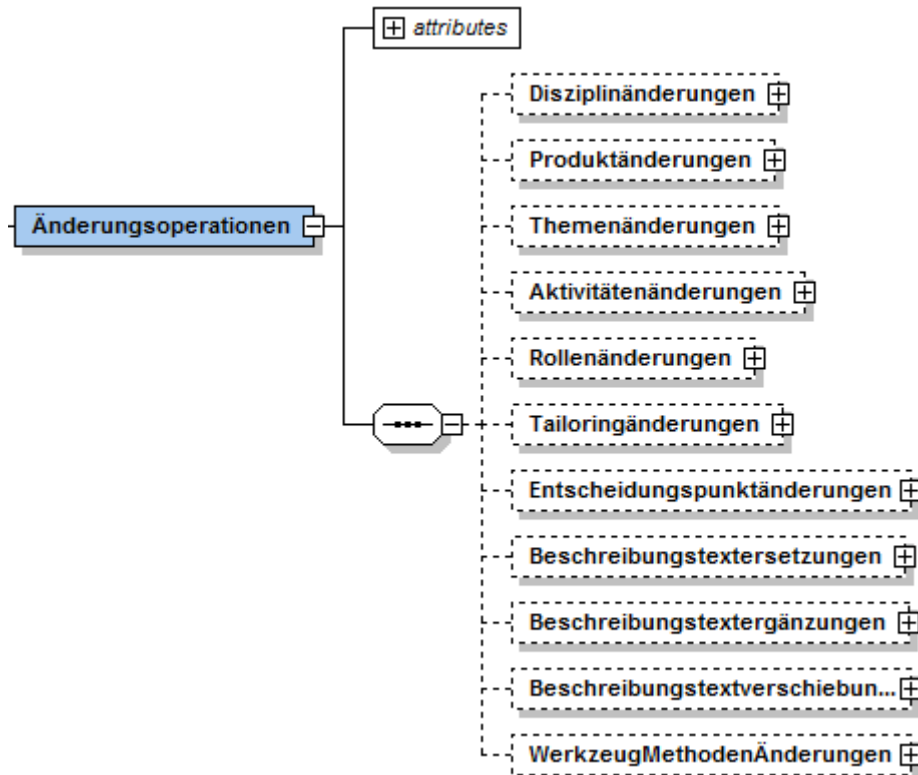


Abbildung 44 Änderungsoperationen des V-Modell XT im Überblick

Das V-Modell definiert bereits eine Reihe von Änderungsoperationen (Abbildung 44), die die häufigsten Anforderungen abdecken. In den folgenden Abschnitten werden die einzelnen Operationen detailliert vorgestellt.

Änderungsoperation	Beschreibung
Disziplinänderung	Operationen zur Änderung von Disziplinen. Hier können Disziplinen umbenannt werden. Weiterhin können die Beschreibungstexte zu einer Disziplin ergänzt werden.



Änderungsoperation	Beschreibung
Produktänderung	Diese Operationen dienen der Anpassung von Produkten des V-Modells. Es können Umbenennungen erfolgen und weiterhin Ergänzungen in den beschreibenden Texten.
Themenänderung	
Aktivitätenänderungen	Aktivitäten können umbenannt werden. Weiterhin können die Beschreibungstexte ergänzt werden.
Rollenänderungen	Operationen zur Änderung von Rollen umfassen die Optionen der Umbenennung sowie der Ergänzung der Rollenbeschreibungen. Weiterhin können hier bereits existierende Verantwortungsbeziehungen angepasst werden, sodass neue bzw. andere Rollen die Verantwortung für ein Produkt des Referenzmodells übernehmen können. Außerdem können Rollen spezialisiert werden.
Tailoringänderungen	Hier können Projektmerkmalswerte vorbelegt werden oder neue Vorgehensbausteine additiv zu bereits existierenden Projekttypvarianten zugeordnet werden.
Entscheidungspunktänderungen	Entscheidungspunkte können umbenannt werden. weiterhin ist eine Ergänzung der Beschreibungstexte möglich.
Beschreibungstextersetzen	Die Ersetzung von Beschreibungstexten bezieht sich auf die V-Modell-Struktur. Es können sowohl komplette Kapitel- als auch Abschnittstexte ersetzt werden.
Beschreibungstextergänzungen	Analog zur Ersetzung können Kapitel- oder Abschnittstexte auch ergänzt werden.
Beschreibungstextverschiebungen	Kapitel und Abschnitte können mit diesen Änderungsoperationen an andere Stellen in der Dokumentation eingefügt werden.
WerkzeugMethodenÄnderungen	Das V-Modell enthält bereits eine Reihe von Beziehungen zwischen Methoden und Werkzeugen zu anderen V-Modell-Elementen. Diese Beziehungen können über die hier zu findenden Operationen entfernt werden.

### 3.3.1 Disziplinänderungen

Disziplinen sind logische Gruppen (die Produkt- und Aktivitätsgruppen ablösen), in denen Produkte und Aktivitäten zusammengefasst werden. Das V-Modell gestattet eine *Umbenennung* und eine *Ergänzung* des Beschreibungstextes einer Disziplin (Abbildung 45).

**Umbenennung.** Im Rahmen einer organisationsspezifischen Anpassung kann die Umbenennung einer Disziplin erforderlich sein. Beispielsweise kann es in einer Organisation eine Ablage „Projektorganisation“ geben, die alle relevanten Produkte des Projektmanagements zusammenfasst. Konsequenterweise soll im Rahmen der Anpassung die Disziplin „Projektmanagement“ des V-Modells in „Projektorganisation“ umbenannt werden. Hierzu muss ein neues Element vom Typ *DisziplinUmbenennen* instanziiert werden.

**Achtung!** Das Element *Name* dieser Operation bezeichnet das neue Element im V-Modell. Der neue Name der zu ändernden Disziplin ist im Element *NeuerName* zu hinterlegen, z.B. „Projektorganisation“. In *DisziplinRef* ist die Referenz auf die Disziplin „Projektmanagement“ des Referenzmodells herzustellen.

Bei der Erzeugung des organisationsspezifischen Vorgehensmodells [3] werden alle referenzierten Bezeichner der Disziplin „Projektmanagement“ in „Projektorganisation“ umbenannt.



**Disziplintexte voran-/hintenanstellen.** Beschreibungstexte von Disziplinen beschreiben üblicherweise die Inhalte einer Disziplin. Werden neue Produkte erstellt und in einer bestehenden Disziplin eingeordnet, müssen die Beschreibungstexte ggf. angepasst werden.

Hierzu muss ein neues Element vom Typ *DisziplintextVoranstellen* oder *DisziplintextHintenanstellen* instanziiert werden, welches den zusätzlichen Beschreibungstextanteil enthält (Element *ZuErgänzenderText*). In *DisziplinRef* ist die Referenz auf die zu ergänzende Disziplin herzustellen. Je nach Operation wird der Ergänzungstext vor oder nach dem bereits existierenden Text eingefügt.

Bei der Erzeugung des organisationsspezifischen Vorgehensmodells werden die Beschreibungstexte konkateniert.

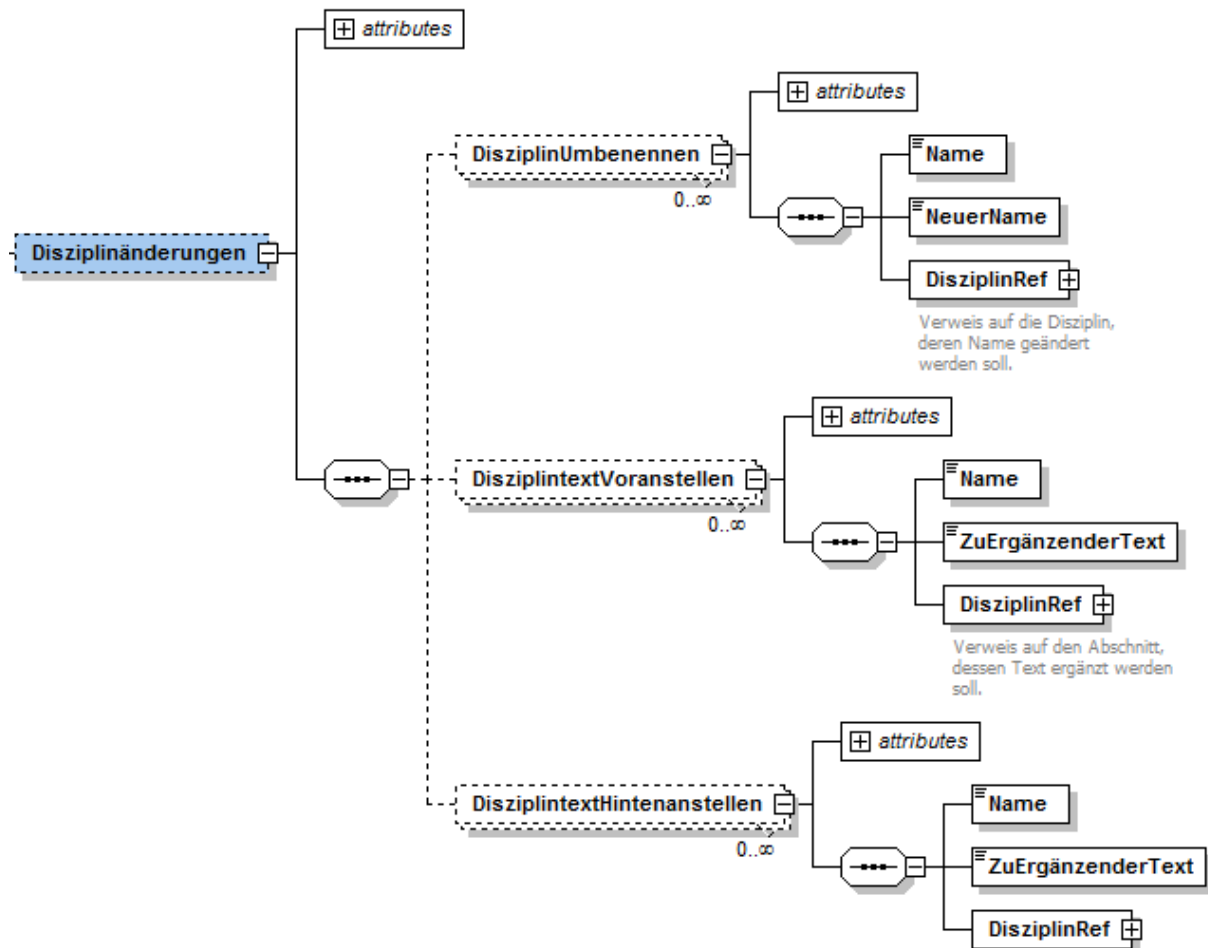


Abbildung 45 Änderungsoperationen, Bereich Disziplinänderungen (Sicht: XML-Schema)

### 3.3.2 Produktänderungen

Produkte stellen die Ergebnistypen des V-Modells dar. Im Rahmen einer organisationsspezifischen Anpassung kann festgestellt werden, dass ein in der Organisation bereits etabliertes Produkt einem Produkt des V-Modells (inhaltlich) entspricht, jedoch anders heißt, z.B. „Projektleitfaden“ anstelle von „Projekthandbuch“. In diesem Fall ist eine Umbenennung erforderlich. Auch Ergänzungen sind denkbar, da Organisationen spezifischere Inhalte für verschiedene Produkte verfügbar haben können. Siehe Abbildung 46.

**Umbenennung.** Die Umbenennung von Produkten bezieht sich auf den Namen des Produkts, z.B. „Projektleitfaden“ anstelle von „Projekthandbuch“. Um ein Produkt umzubenennen muss ein Element vom Typ *ProduktUmbenennen* instanziiert werden.

**Achtung!** Das Element *Name* bezeichnet die Instanz von *ProduktUmbenennen*. Der neue Name des umzubenennenden Produkts wird im Element *NeuerName* hinterlegt. Das Element *ProduktRef* stellt die Verbindung zum umzubenennenden Produkt her.

**Produkttexte voran-/hintenanstellen.** Sind Vorgaben oder Inhalte verfügbar, die spezifischer als die Standardinhalte des V-Modells sind, können die entsprechenden Beschreibungstexte ergänzt werden.

Dazu muss ein neues Element vom Typ *ProdukttextVorstellen* oder *ProdukttextHintenanstellen* instanziiert werden. Der zu ergänzende Text ist im Element *ZuErgänzenderText* zu hinterlegen. Das Produkt, dessen Beschreibungstext ergänzt werden soll, ist durch das Element *ProduktRef* zu referenzieren.

Bei der Erzeugung des organisationsspezifischen Vorgehensmodells werden z.B. alle referenzierten Bezeichner des Produkts „Projekthandbuch“ in „Projektleitfaden“ umbenannt. Die Zuordnung zur Disziplin ändert sich dabei nicht. Auch alle Referenzen zu Entscheidungspunkten, Rollen, Themen etc. bleiben erhalten.

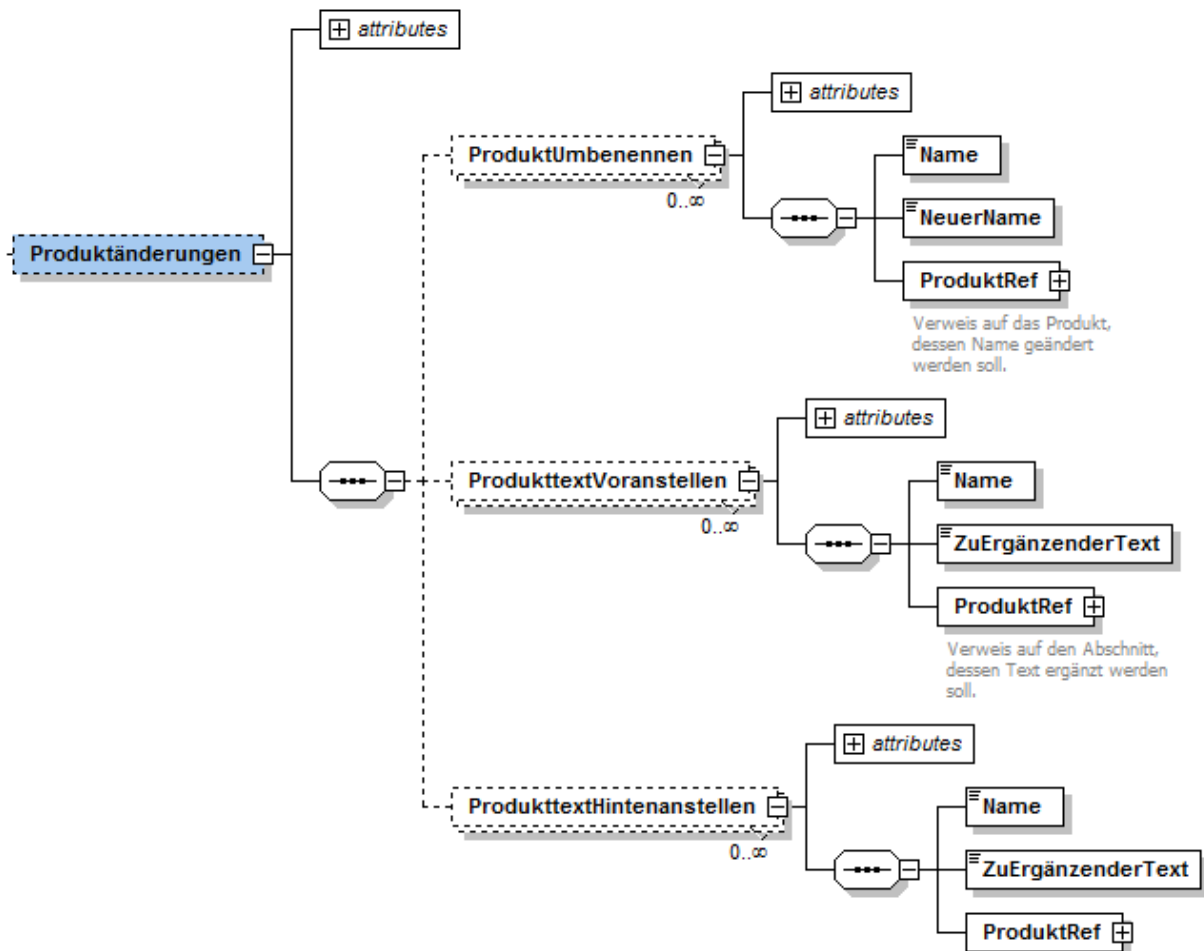


Abbildung 46 Änderungsoperationen, Bereich Produktänderungen (Sicht: XML-Schema)

### 3.3.3 Themenänderungen

Analog zu Produkten können auch Themen des Referenzmodells angepasst werden. Sowohl die Funktionalität als auch die Vorgehensweise dazu ist identisch zu Produkten und Disziplinen. Siehe Abbildung 47 und Abschnitt 3.3.2, auf die wir an dieser Stelle verweisen.

**Unterthemen einordnen.** Zusätzlich zu den Umbenennungs- und Textanpassungsoperationen können auch *Unterthemen* neu eingeordnet werden, bzw. sofern neue Unterthemen definiert sind, diese zu existierenden Themen zugeordnet werden. Dazu muss eine Operation vom Typ *UnterthemaEinordnen* instanziiert werden.

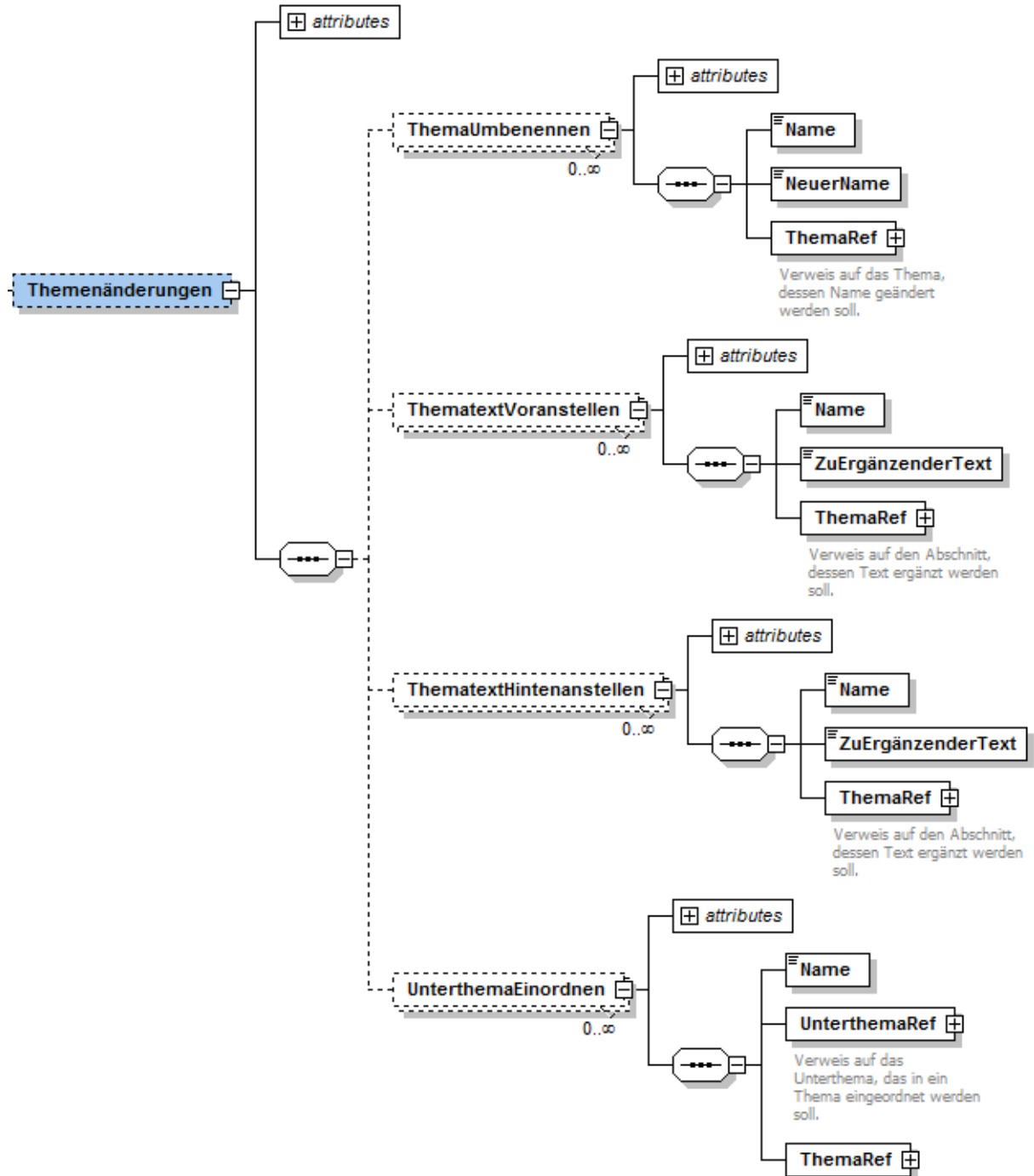


Abbildung 47 Änderungsoperationen, Bereich Themenänderungen (Sicht: XML-Schema)

### 3.3.4 Aktivitätenänderungen

Analog zu Produkten des Referenzmodells können auch Aktivitäten angepasst werden. Sowohl die Funktionalität als auch die Vorgehensweise dazu ist identisch zu Produkt oder Disziplinen. Siehe Abbildung 48 und Abschnitt 3.3.2, auf die wir an dieser Stelle verweisen.

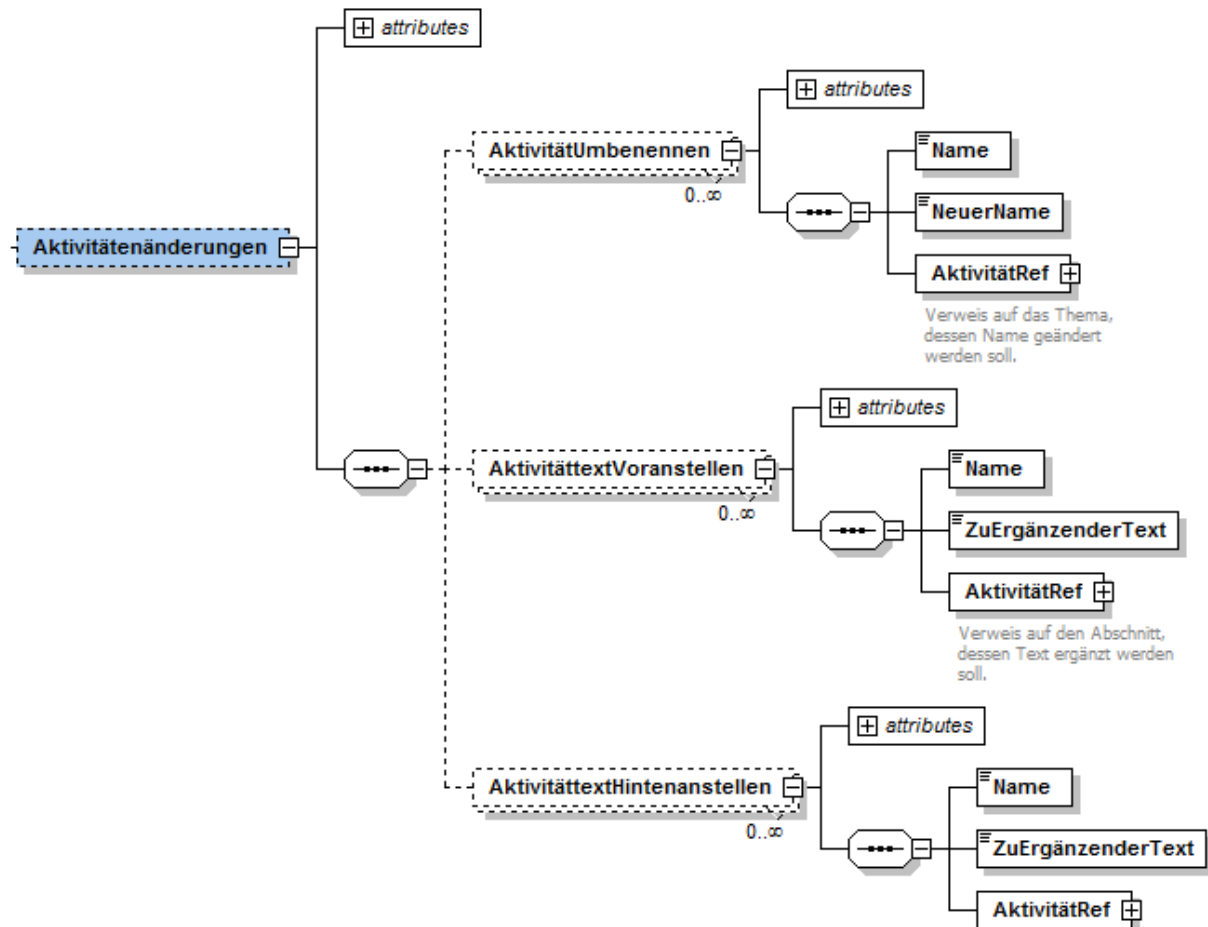


Abbildung 48 Änderungsoperationen, Bereich Aktivitätenänderungen (Sicht: XML-Schema)

### 3.3.5 Rollenänderungen

Im Rahmen der organisationsspezifischen Anpassung ist die Berücksichtigung von Organisationsstrukturen, insbesondere von Rollen, essenziell. Das V-Modell bietet daher weit reichende Möglichkeiten der Anpassung von Rollen (Abbildung 49). Die Vorgehensweise zur Anpassung von Rollen entspricht im Wesentlichen der von Produkten, siehe Abschnitt 3.3.2.

**Ändern/Entfernen von Verantwortlichkeiten.** Passt das Produktmodell, jedoch nicht das Rollenmodell, sind ggf. Verantwortlichkeiten für einzelne Produkte neu zu bestimmen, z.B. der „Projektmanager“ der Organisation ist für das „Projekthandbuch“ verantwortlich und nicht der „Projektleiter“, den es in dieser Organisation in dieser Form womöglich gar nicht gibt. Verschiedene Verantwortungen sind also neu zu vergeben.

Dazu muss ein Element vom Typ *VerantwortlichkeitÄndern* instanziiert werden. Diese muss die Verantwortungsbeziehung *VerantwortlichkeitsbeziehungRef* (Abschnitt 2.3.7) referenzieren, die geändert werden soll. Weiterhin muss die Rolle *RolleRef* referenziert werden, die anstelle der Standardrolle die Verantwortung übernehmen soll. Soll aufgrund von Erfordernissen des Tailorings eine Verantwortlichkeit nur entfernt werden, so muss stattdessen ein Element vom Typ *VerantwortlichkeitEntfernen* instanziiert werden.

**Umbenennung.** Rollen können analog zu Produkten umbenannt werden, siehe Abschnitt 3.3.2.

**Rollenbeschreibung voran-/hintenanstellen.** Die Beschreibungstexte zu den Rollen können analog zu den Beschreibungstexten von Produkten ergänzt werden, siehe Abschnitt 3.3.2. Da Rollen im Ge-

gensatz zu Produkten jedoch über mehrere Beschreibungstexte verfügen (Beschreibung, Fähigkeitsprofil etc.), können diese auch individuell ergänzt werden.

**Rollen spezialisieren.** Rollen können spezialisiert werden. Bei einer Spezialisierung, wird eine neue Rolle angelegt (inkl. aller Verantwortlichkeiten etc.). Spezialisiert diese neue Rolle nun beispielsweise eine bereits existierende aus dem Referenzmodell, übernimmt sie alle Eigenschaften i.S. der *eingehenden Referenzen*. Bei der Erstellung des organisationsspezifischen Vorgehensmodells wird die spezialisierte Rolle aus dem Modell entfernt und durch die Spezialisierende ersetzt.

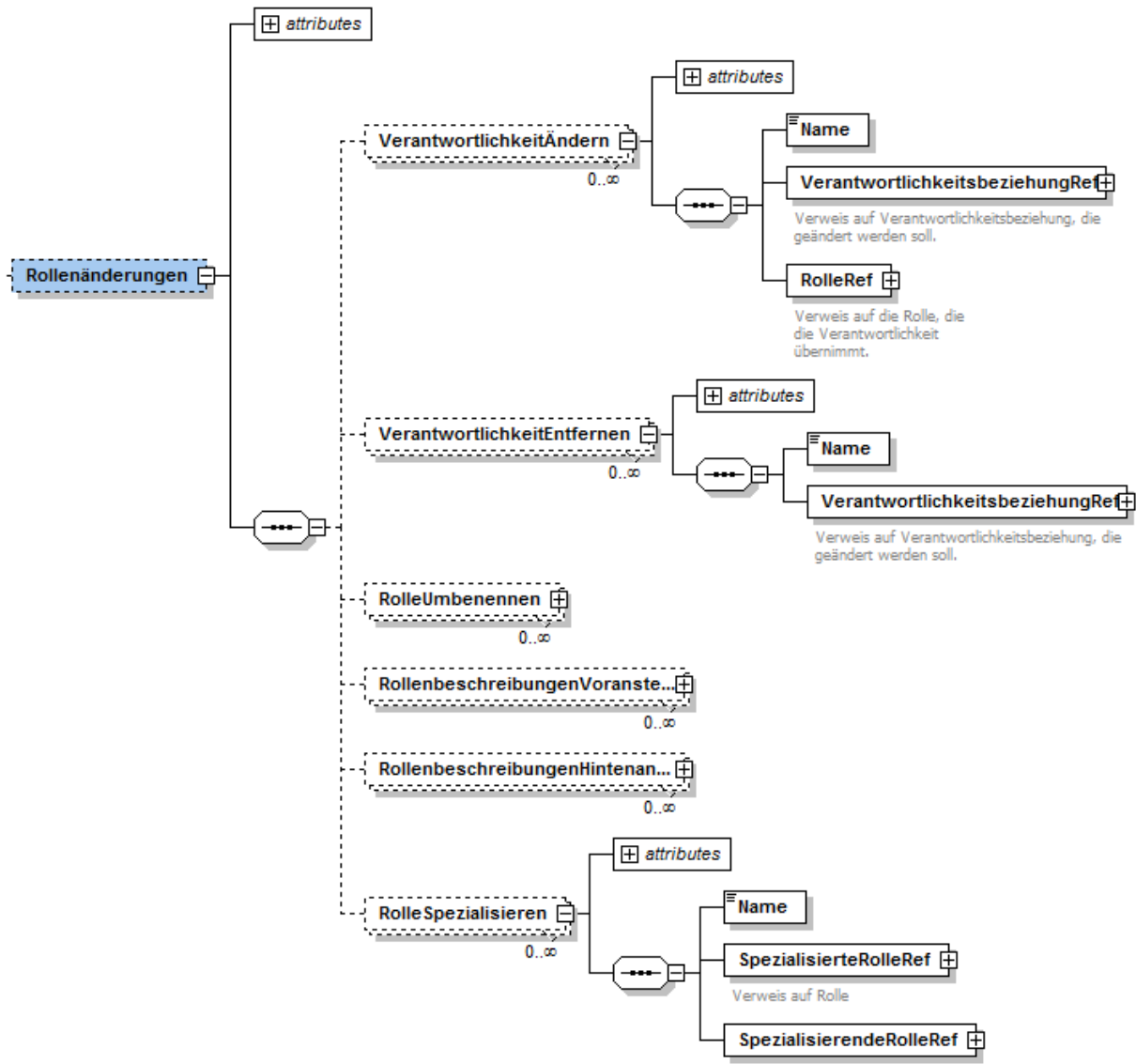


Abbildung 49 Änderungsoperationen, Bereich Rollenänderungen (Sicht: XML-Schema)

### 3.3.6 Tailoringänderungen

Tailoringänderungen dienen der Anpassung des Tailoringverhaltens des V-Modells. Hier können z.B. Projektmerkmale mit neuen, vom Standard abweichenden Defaultwerten belegt werden. Weiterhin können additive Ergänzungen von existierenden Projekttypvarianten durch zusätzliche bedingte Vorgehensbausteine vorgenommen werden.

**Standardwert ändern.** Soll der Standardwert eines Projektmerkmals im Tailoring geändert werden, muss ein Element vom Typ *StandardWertÄndern* instanziiert werden. Dieses muss das Projektmerkmal, auf das es sich bezieht, referenzieren und einen neuen Wert für dieses Merkmal enthalten.

**Vorgehensbaustein ergänzen.** Genügt für eine Anpassung des V-Modells das einfache Hinzufügen eines Vorgehensbausteins, kann dieser als Erweiterung auf bereits existierende Projekttypvarianten angewendet werden. Dazu muss ein Element vom Typ *VorgehensbausteinErgänzen* instanziiert werden. Dieses referenziert den betreffenden Vorgehensbaustein und die Projekttypvariante, die den neuen Baustein als bedingt enthalten soll. Die Zuweisung muss für jede infrage kommende Projekttypva-

rianten individuell erfolgen. Durch die Ergänzung als bedingtes Element ist der ergänzte Vorgehensbaustein für die erweiterte Projekttypvariante *verpflichtend*.

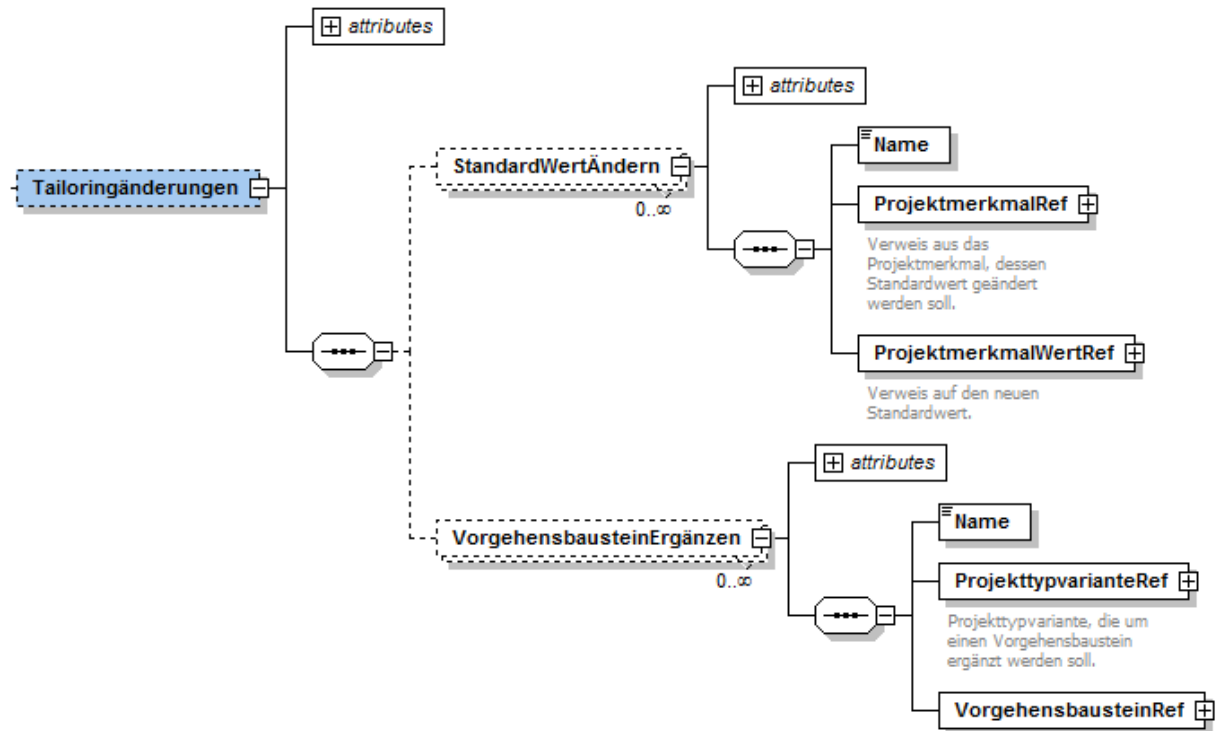


Abbildung 50 Änderungsoperationen, Bereich Tailoringänderungen (Sicht: XML-Schema)

### 3.3.7 Entscheidungspunktänderungen

Analog zu Produkten (Abschnitt 3.3.2) können auch Entscheidungspunkte angepasst werden. Die hierfür bereitgestellten Operationen (Abbildung 51) gestatten wie bei Produkten die Umbenennung und die Ergänzung des Beschreibungstextes.

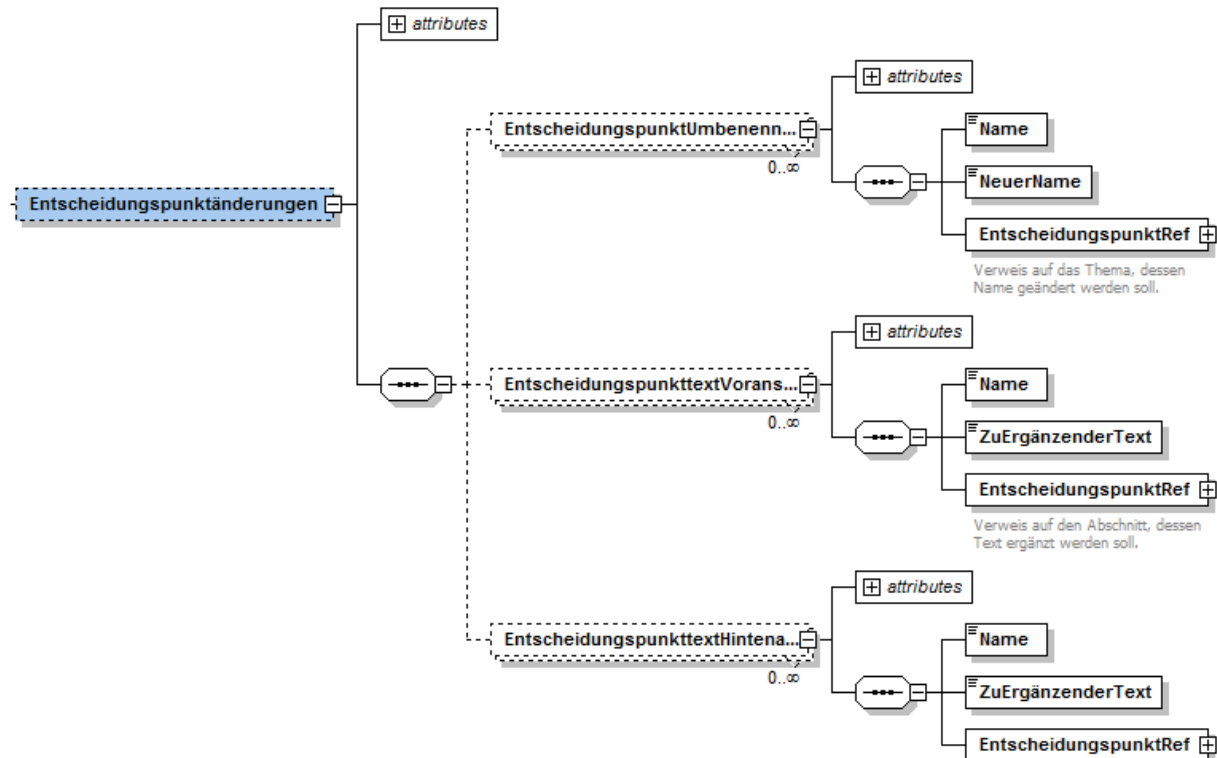


Abbildung 51 Änderungsoperationen, Bereich Entscheidungspunktänderungen (Sicht: XML-Schema)

### 3.3.8 Beschreibungstextersetzungen

Neben den generierten Beschreibungstexten, die durch Produkte, Aktivitäten etc. im V-Modell erzeugt werden, gibt es in der V-Modell-Struktur (Abschnitt 0) auch *Kapitel* und *Abschnitte*, die nicht generierten Text enthalten. Im Rahmen einer organisationsspezifischen Anpassung kann es erforderlich sein, diese Texte entweder vollständig zu ersetzen (dieser Abschnitt, Abbildung 52) oder sie zumindest zu ergänzen (Abschnitt 3.3.9).

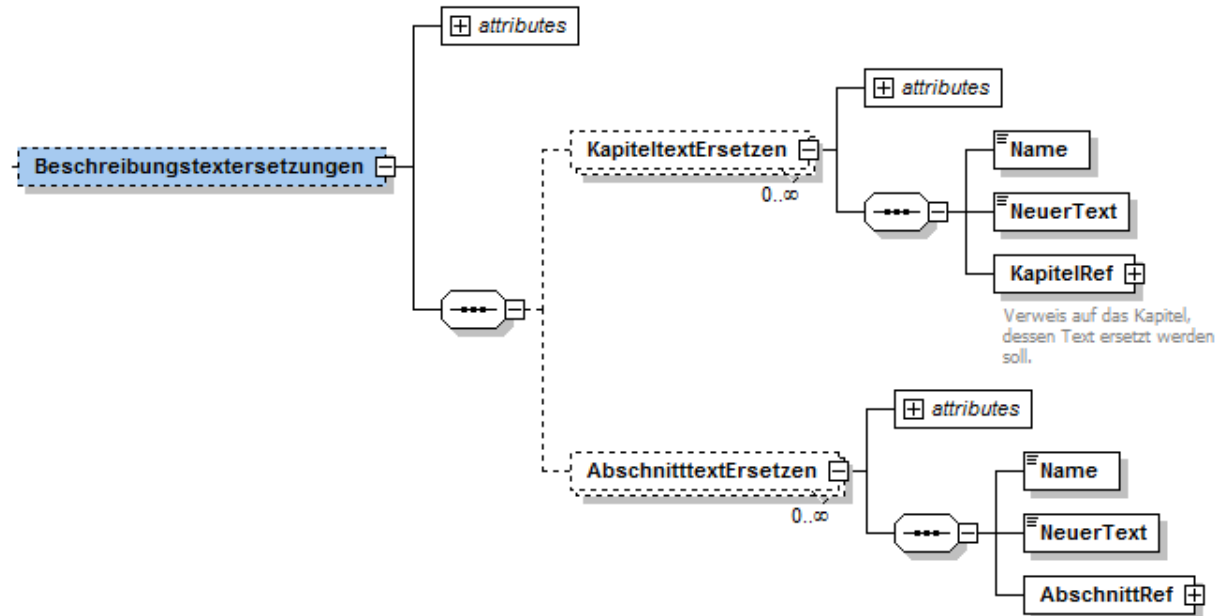


Abbildung 52 Änderungsoperationen, Bereich Beschreibungstextersetzungen (Sicht: XML-Schema)

**Kapiteltexte ersetzen.** Mithilfe der Operation *KapiteltextErsetzen* können die Beschreibungstexte von Kapiteln vollständig ersetzt werden. Das Element *NeuerText* enthält dabei den neuen Text. Über *KapitelRef* wird das Kapitel referenziert, in dem der Text ersetzt werden soll.

**Abschnittstexte ersetzen.** Kapitel sind in Abschnitte gegliedert. Die Beschreibungstexte der Abschnitte können analog zu den Kapiteltexten mithilfe der Operation *AbschnitttextErsetzen* ersetzt werden.

### 3.3.9 Beschreibungstextergänzungen

Sind die Beschreibungstexte weitgehend zutreffend und sollen im Rahmen einer organisationspezifischen Anpassung nur ergänzt werden, sind die Operationen *KapiteltextVoranstellen*, *KapiteltextHintenanstellen*, *AbschnitttextVoranstellen* und *AbschnitttextHintenanstellen* (Abbildung 53) anzuwenden. Syntaktisch sind sie analog zu den Ersetzungsoperationen anzuwenden. Semantisch werden die Textergänzungen bei der Erstellung des organisationspezifischen Vorgehensmodells mit den Standardtexten konkateniert.

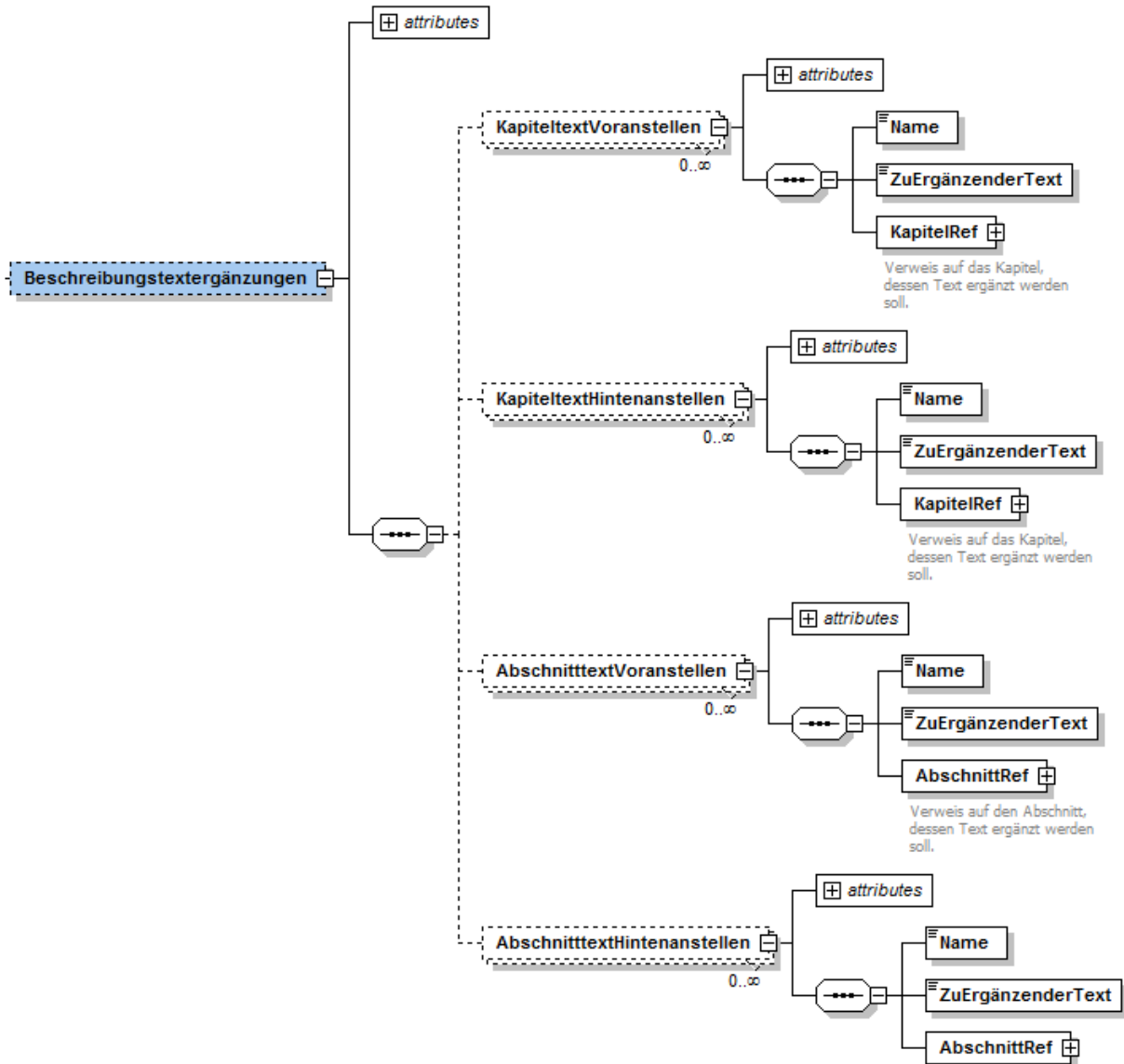


Abbildung 53 Änderungsoperationen, Bereich Beschreibungstextergänzungen (Sicht: XML-Schema)



### 3.3.10 Beschreibungstextverschiebungen

Wenn die Dokumentationsstruktur des V-Modells angepasst oder ergänzt werden soll, kommen Beschreibungstextverschiebungen (Abbildung 54) zum Einsatz. Dabei können einmal Kapitel und Abschnitte neu geordnet werden. Weiterhin können neue Kapitel und Abschnitte auf diese Weise in existierende Strukturen eingeordnet werden.

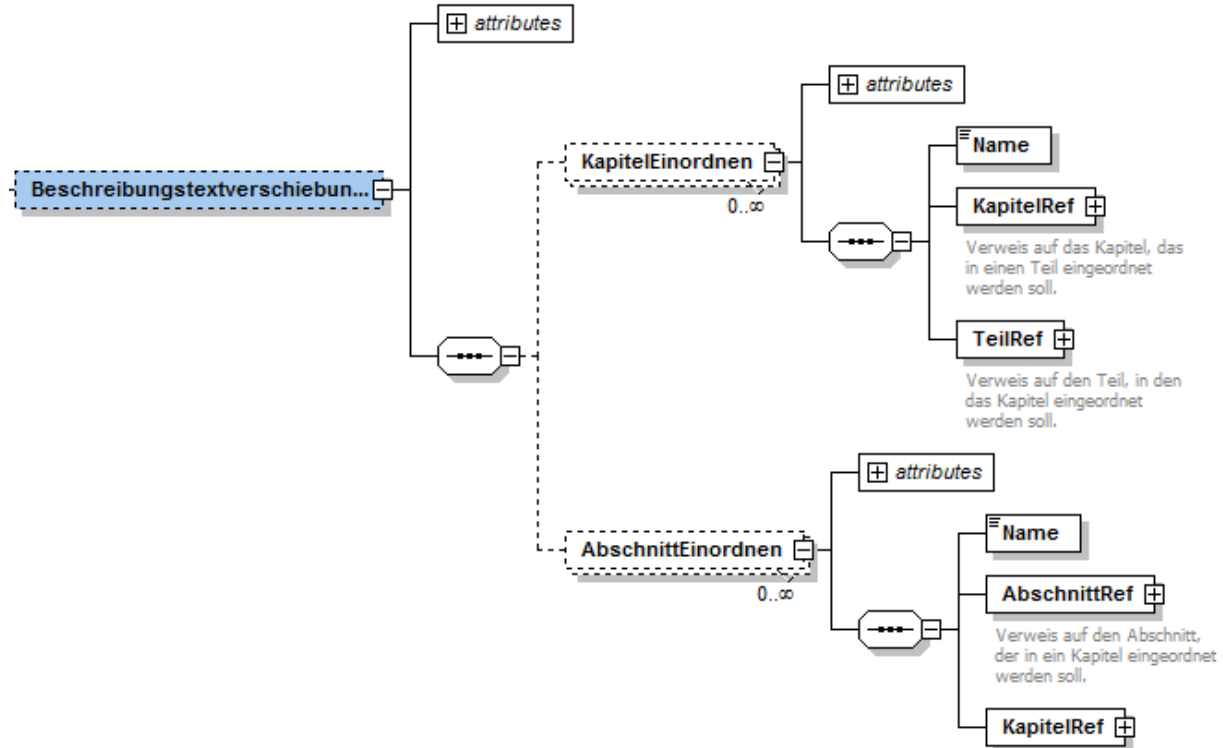


Abbildung 54 Änderungsoperationen, Bereich Beschreibungstextverschiebungen (Sicht: XML-Schema)

**Kapitel einordnen.** Um ein neues Kapitel einzuordnen oder um ein existierendes zu verschieben, muss ein Element vom Typ *KapitelEinordnen* instanziiert werden. Dieses referenziert das betreffende Kapitel und den Teil, zu dem es hinzugefügt werden muss. *Diese Option muss jedoch mit Bedacht eingesetzt werden, da sie ggf. auch Nacharbeiten an den Exporttemplates zur Folge haben kann.*

**Abschnitte einordnen.** Das Einordnen der Abschnitt erfolgt analog zu den Kapiteln.

### 3.3.11 Werkzeug-/Methodenänderungen

Methoden- und Werkzeugreferenzen geben im V-Modell erweiterte Unterstützung bei der Zuordnung geeigneter Methoden und Werkzeugen zu Aktivitäten und Arbeitsschritten. Wird eine Standardvorgabe nicht benötigt oder durch eine organisationspezifische ersetzt, kann die Standardzuordnung entfernt werden (Abbildung 55).

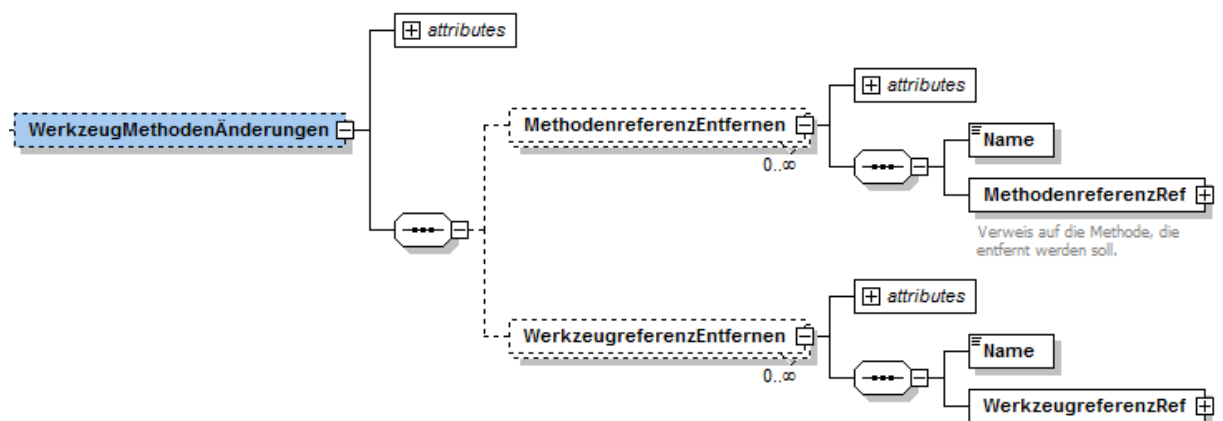


Abbildung 55 Änderungsoperationen, Bereich WerkzeugMethodenÄnderungen (Sicht: XML-Schema)

Mit der Operation *MethodenreferenzEntfernen* kann eine standardmäßig vorgegebene Methodenzuordnung entfernt werden. Analog kann mit der Operation *WerkzeugreferenzEntfernen* kann eine Werkzeugzuordnung entfernt werden. Dazu sind jeweils Instanzen der Operationen anzulegen und die zu entfernenden Zuordnungen zu referenzieren.

### 3.4 Vortailoring

Eine weitere Option im Rahmen der organisationspezifischen Anpassung stellt das *Vortailoring* dar. Das Vortailoring dient dazu „großflächige“ Ausschlüsse im V-Modell vorzunehmen, wenn Inhalte konsequent nicht benötigt werden. Ein mögliches Einsatzszenario ist der Ausschluss eines ganzen Projekttyps, wenn z.B. eine Organisation grundsätzlich nur Aufträge im Sinne eines *Systementwicklungsprojekts (AN)* durchführt und keinen Bedarf an den AG bzw. AG/AN-Projekttypen hat. Die letzteren können dann im Rahmen des Vortailorings konstruktiv und dauerhaft aus dem organisationspezifischen Vorgehensmodell ausgeschlossen werden.

Das Metamodell sieht für das Vortailoring zwei Operationen (Abbildung 56) vor. Ein *Ausschluss* blendet einen oder mehrere Projekttypen (mitsamt aller assoziierten Projekttypvarianten) bzw. einen oder mehrere Projekttypvarianten aus. Ein *SelektiverAusschluss* blendet für den Kontext eines Projekttyps einen bestimmten (oder mehrere) Projektmerkmalswert aus und kann somit für eine (dauerhafte) Vorbelegung von Projektmerkmalen verwendet werden.

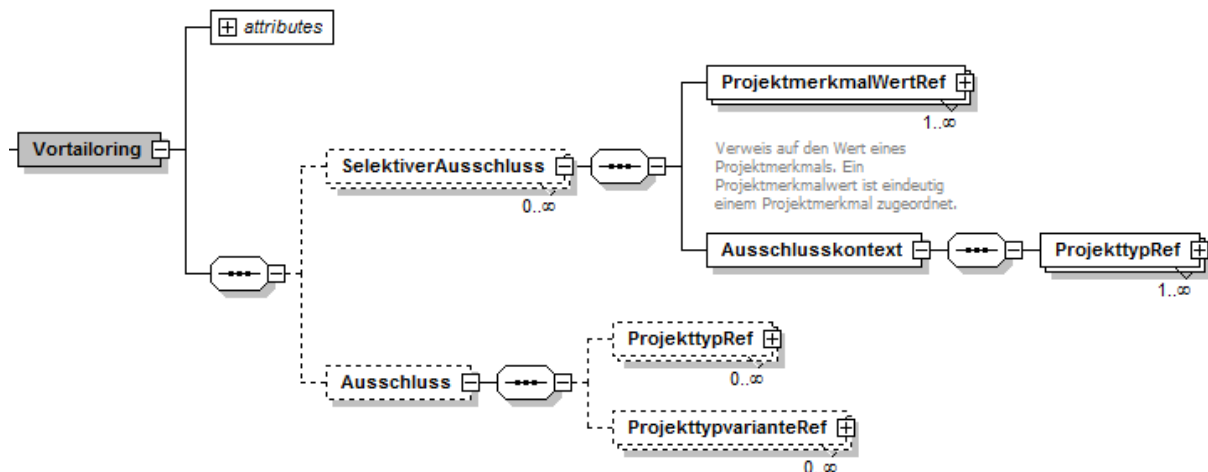


Abbildung 56 Vortailoring im V-Modell XT

## 4 Tailoring

Durch die weitreichenden Anpassungen des V-Modell-Metamodells haben sich auch Änderungen im Tailoring ergeben. Dieser Abschnitt geht auf den Tailoring-Prozess als solchen ein und stellt dabei Zusammenhänge zwischen den im Abschnitt 2 beschriebenen Metamodellkonzepten her.

### 4.1 Der Tailoring-Prozess

Das Tailoring ist im Vergleich zu früheren Versionen des V-Modells einerseits restriktiver und führt den Anwender stärker, gleichzeitig ist es aber flexibler im Hinblick auf Änderungen und Anpassungen des V-Modells. Der Tailoring-Prozess des V-Modells ist in Abbildung 57 dargestellt. Er besteht aus drei Schritten, in denen aus dem Standard- oder organisationspezifischen V-Modell das projektspezifische V-Modell hergeleitet wird:

1. Im ersten Schritt wird der Projekttyp (siehe Abschnitt 2.5.1) festgelegt. Jeder Projekttyp legt dabei die verpflichtenden Vorgehensbausteine sowie eine Reihe von Projektmerkmalen (Abschnitt 2.5.3) fest. Außerdem stellt er über diverse Ablaufbausteinspezifikationen (Abschnitt 2.4.3) die Rahmenbedingungen für die Projektdurchführungsstrategie auf. Jeder Projekttyp enthält mindestens eine Projekttypvariante, die im zweiten Schritt auszuwählen ist.
2. Im zweiten Schritt muss eine Projekttypvariante (siehe Abschnitt 2.5.2) ausgewählt werden. In der Projekttypvariante werden in erster Linie konkrete Abläufe (Ablaufbausteine, Abschnitt 2.4.2) den Spezifikationen des Projekttyps zugeordnet, sodass eine Projektdurchführungsstrategie ermittelbar ist. Weiterhin können zusätzliche Projektmerkmale zur Belegung vorgegeben werden.

- Im dritten Schritt sind alle Projektmerkmale, die durch den Projekttyp und die ausgewählte Projekttypvariante vorgegeben sind, mit Werten zu belegen. Jedes Projektmerkmal hat einen Standardwert, sodass die Änderung dieser Werte optional ist. Durch das Belegen der Projektmerkmale mit Werten, werden ggf. optionale Vorgehensbausteine und zusätzliche Abläufe dem projektspezifischen Vorgehensmodell hinzugefügt.

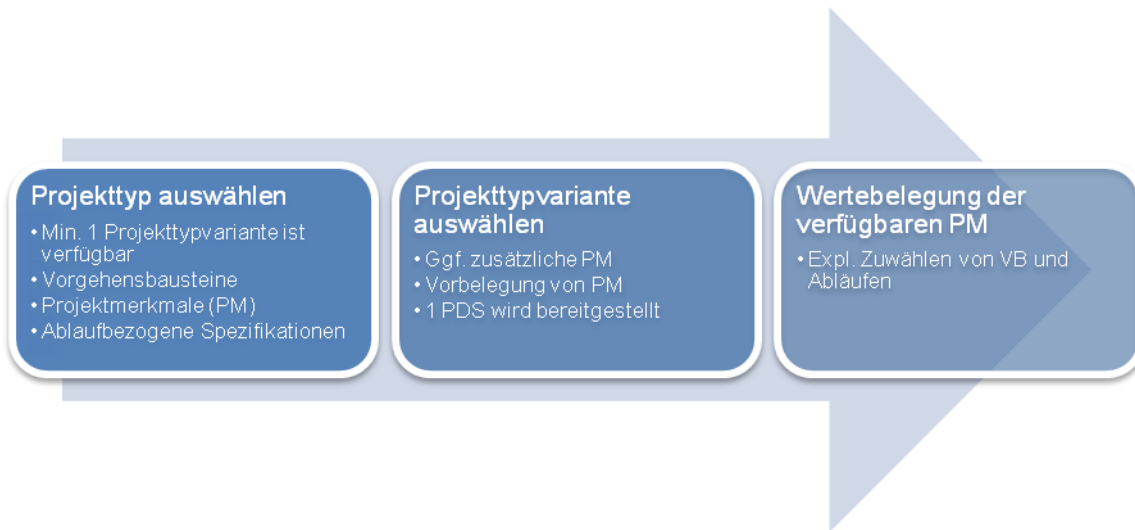


Abbildung 57 Tailoring-Prozess zur projektspezifischen Anpassung

Der Tailoring-Prozess ist für Erweiterungen transparent. Anpassungen und Erweiterungen des V-Modells spiegeln sich hier unmittelbar wieder. Der Projektassistent bietet eine direkte Sicht auf dieses neue Vorgehen:

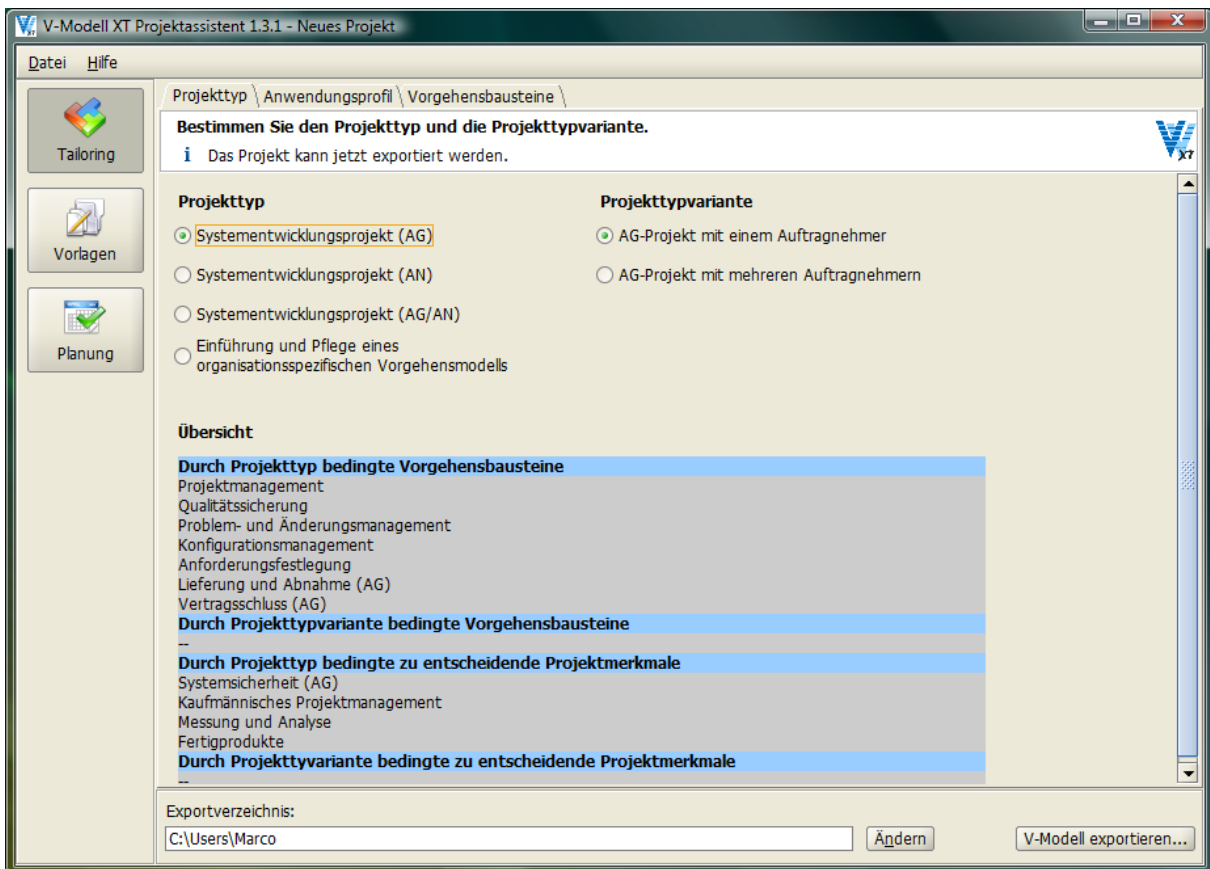


Abbildung 58 Schritte 1 und 2 des Tailorings mit dem Projektassistenten

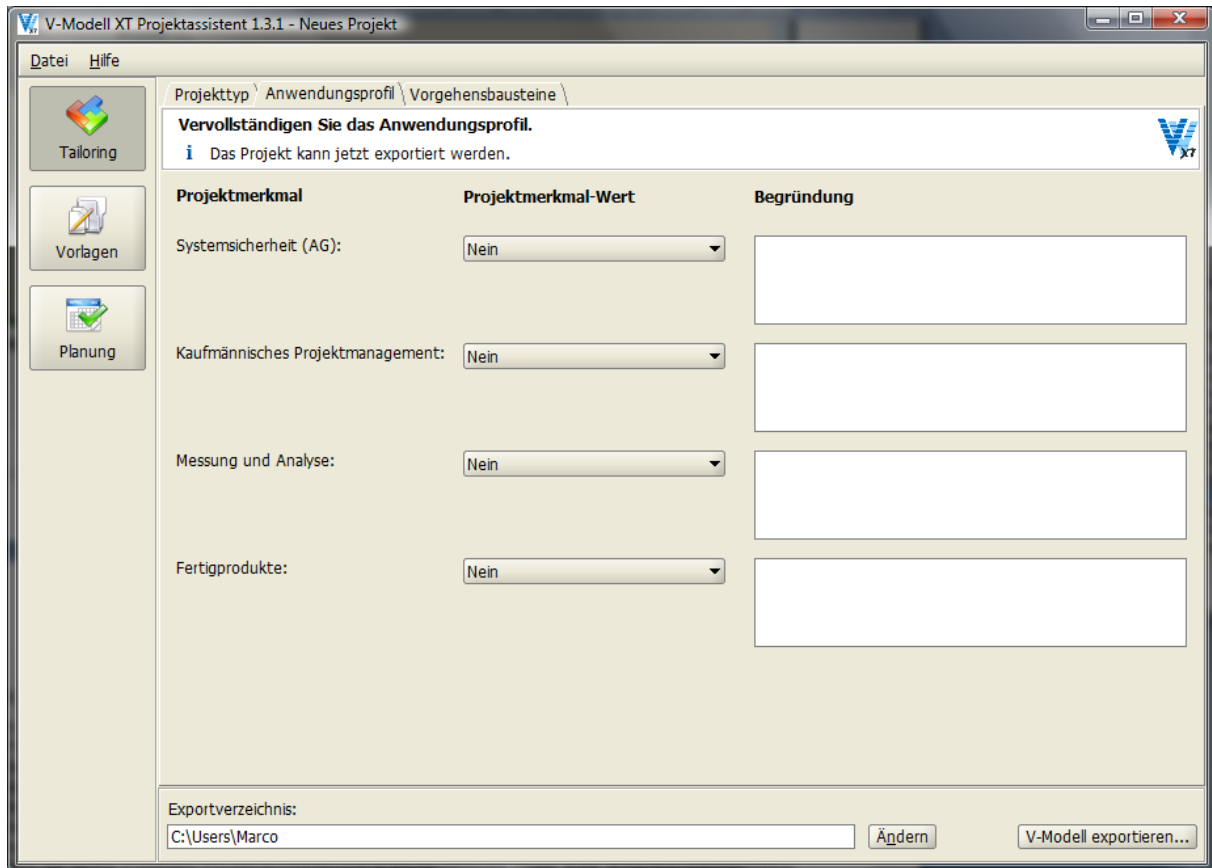


Abbildung 59 Schritt 3 des Tailorings mit dem Projektassistenten

## 4.2 Zusammenhänge Tailoring – Metamodell

In Abbildung 38 sind die Zusammenhänge im Metamodell für das Tailoring bereits dargestellt. Der Projekttyp, der im Tailoring als erstes Auszuwählen ist, bindet die Metamodellelemente

- *Vorgehensbaustein*,
- *Ablaufbausteinspezifikation* und
- *Projektmerkmal*.

Diese Elemente werden im ersten Schritt des Tailorings ausgewählt und gemäß der Modellierung des V-Modells mit Werten belegt. Die durch den Projekttyp referenzierten Vorgehensbausteine sind hierbei die *verpflichtenden* Vorgehensbausteine. Die optionalen Vorgehensbausteine werden durch die Wertebelegungen der bedingten Projektmerkmale bereitgestellt. Die referenzierten Ablaufbausteinspezifikationen legen den Rahmen fest, dem die später ermittelte *Projektdurchführungsstrategie* genügen muss. Eine Ablaufbausteinspezifikation ist dabei ausgezeichnet und dient als Gesamtrahmen für die Projektdurchführungsstrategie, die alle anderen Ablaufbausteinspezifikationen zusammenfügt. Die referenzierten Projektmerkmale enthalten ihrerseits verschiedene *Projektmerkmalswerte*. Diese können jeweils

- *Vorgehensbausteine*,
- *Ablaufbausteinspezifikationen* und
- *Ablaufbausteine* referenzieren.

Die durch Projektmerkmalswerte gebundenen Vorgehensbausteine sind *optional*. Die gebundenen Ablaufbausteine und Ablaufbausteinspezifikationen können den Rahmenablauf erweitern/anpassen.

Die im zweiten Schritt auszuwählende Projekttypvariante bindet

- *Vorgehensbausteine*,
- *Ablaufbausteinspezifikationen* und
- *Ablaufbausteine*.

Analog zum Projekttyp sind die durch die Projekttypvarianten gebundenen Vorgehensbausteine verpflichtend anzuwenden. Ablaufbausteinspezifikationen erweitern den Rahmen des Ablaufs des Projekttyps. Weiterhin werden in der Projekttypvariante durch die Einbindung konkreter Ablaufbausteine die Spezifikationen des Projekttyps erfüllt. Wieder ist ein Ablaufbaustein ausgezeichnet. Dieser stellt den Ausgangspunkt für die Ermittlung der konkreten Projektdurchführungsstrategie dar und dient als Container für alle anderen Ablaufbausteine.

Die Projektmerkmale stellen nach der Auswahl des Projekttyps und der Projekttypvarianten die Variablen dar. Ihre Wertebelegung erfolgt im dritten Schritt des Tailorings. Wie weiter oben beschrieben, stellen sie optionale Elemente zur Verfügung und erweitern ggf. die Abläufe im Rahmen der Projektdurchführungsstrategie.

### 4.3 Konsequenzen für die Modellierung von Erweiterungen

Durch die Umstellung des Tailorings ergeben sich Konsequenzen für die Gestaltung und Erweiterung des V-Modells auf die dieser Abschnitt näher eingeht.

Anders als in früheren Versionen des V-Modells sind die Konfigurationsinformationen für das Tailoring nicht mehr über die verschiedenen, für das Tailoring relevanten Elemente, verstreut. Es sind also keine *Auswahlkriterien*, z.B. für Vorgehensbausteine, mehr zu setzen. Sämtliche Konfigurationseinstellungen sind nun an die Projekttypen, die Projekttypvarianten und die Projektmerkmale gebunden. Diese zeigen explizit auf die Elemente, die sie benötigen bzw. bedingen.

**Beispiel:** Wird im Rahmen einer Anpassung ein Projekttyp als passend angesehen und soll um eine neue Projekttypvariante erweitert werden, die zusätzliche Vorgehensbausteine und Abläufe bereitstellt, sollte wie folgt vorgegangen werden: Es müssen alle relevanten Vorgehensbausteine erstellt werden. Es müssen alle relevanten Abläufe erstellt werden (ggf. mit neuen Ablaufbausteinspezifikationen). Es müssen ggf. weitere Elemente wie Projektmerkmale, Rollen oder Entscheidungspunkte angelegt werden. Abschließend ist im Erweiterungsmodell eine neue Projekttypvariante anzulegen, die auf den passenden Projekttyp verweist und die neuen Elemente entweder direkt (z.B. neue Vorgehensbausteine) oder indirekt über neue Projektmerkmale referenziert. Im Tailoring steht daraufhin eine zusätzliche Projekttypvariante zur Auswahl zur Verfügung.

Für alle Elemente des V-Modells (Vorgehensbausteine etc.) ist somit die Zugehörigkeit zu bestimmten Projekttypen transparent und wird direkt bei den Elementen des Tailorings festgelegt.

**Konsequenzen.** Eine Konsequenz, die sich daraus ergibt, ist, dass Vorgehensbausteine *nicht* mehr direkt ausgewählt werden können, wie das in früheren Versionen möglich war. Das projektspezifische V-Modell wird einzig durch den in Abbildung 57 gezeigten Prozess bestimmt.

**Hinweis:** Es besteht weiterhin die Möglichkeit einen Vorgehensbaustein in ein projektspezifisches V-Modell direkt einzubinden, wenn dieser 1) durch ein Projektmerkmal eingebracht wird und 2) explizit *kein* Wert (Auswahl: --) vorgegeben wird. In diesem Fall kann der betreffende Vorgehensbaustein durch den Anwender im Projektassistenten referenziert werden. Dies wird jedoch nicht empfohlen, da durch die bloße Selektion des Vorgehensbausteins das Abhängigkeitsgeflecht, das durch das Projektmerkmal definiert wird, außer Kraft gesetzt wird. Es kann hier vorkommen, dass die Produkte etc. des Vorgehensbausteins dann zwar im projektspezifischen V-Modell enthalten sind, da aber möglicherweise die ebenfalls durch das Projektmerkmal zur Verfügung stehenden Abläufe nicht eingebunden werden, können die Produkte womöglich nicht eingeplant werden, weil Entscheidungspunkte fehlen.

Für eine Projekttypvariante stehen nicht mehr 1 oder viele Projektdurchführungsstrategien zur Auswahl. Pro Projekttypvariante gibt es nunmehr nur noch eine Projektdurchführungsstrategie, die allerdings, in Abhängigkeit vom Tailoring, verschiedene Ablaufstrukturen anbieten kann. Auch diese Optionen werden durch den in Abbildung 57 gezeigten Prozess bestimmt. Das Tailoring stellt sich dem Anwender nun restriktiver dar, hat aber im Vergleich zu früheren Versionen an Flexibilität gewonnen. Die Komplexität (Festlegung der Abhängigkeitsstrukturen sowie Sicherstellung der Konsistenz der betroffenen Elemente) des Tailorings ist vom Anwender hin zum Prozessingenieur verlagert worden. Es erfordert Mehraufwände in der initialen Modellierung, ist dafür aber einfacher aus der Sicht des Anwenders.

## 5 Konsistenzbedingungen

Dieser Abschnitt beschreibt die Konsistenzbedingungen, die für das V-Modell gelten. Die meisten der Konsistenzbedingungen sind selbstsprechend. Sofern weitere Semantik hinter den Konsistenzbedingungen liegt, wird diese nebenstehend erläutert. Der V-Modell XT Editor prüft diese (semantischen) Konsistenzbedingungen nicht.

Bedingung	Ergänzende Erläuterung
<p>Ein Vorgehensbaustein darf nur Verweise auf Elemente enthalten, die in Vorgehensbausteinen enthalten sind, auf denen er direkt oder indirekt basiert. Das gilt für alle Assoziationen hinsichtlich:</p> <ul style="list-style-type: none"> <li>• Produkt- und Themenzuordnung sowie</li> <li>• Aktivitäts- und Arbeitsschrittzuordnung,</li> </ul> <p>nicht aber für Querverweise im Text. Ebenfalls nicht betroffen von dieser Einschränkung sind Beziehungen zu Rollen, Entscheidungspunkten oder z.B. Werkzeugreferenzen.</p>	
<p>Ein Vorgehensbaustein darf nur auf Vorgehensbausteinen direkt basieren, wenn er jeweils auf mindestens ein Element dieser Vorgehensbausteine auch tatsächlich eine Beziehung aufbaut.</p>	
<p>Die Abhängigkeiten der Vorgehensbausteine dürfen keine Zyklen aufweisen.</p>	
<p>Jedem Produkt, das nicht als extern markiert ist, muss eine Aktivität zugeordnet sein (eine entsprechende Beziehung muss existieren).</p>	
<p>Im organisationsspezifischen V-Modell (nach Anpassungsoperationen und Tailoring) muss für jedes Produkt genau eine Rolle verantwortlich sein.</p>	<p>Im nicht getailorten V-Modell kann es mehrere Verantwortliche für ein Produkt geben.</p>
<p>Jede Rolle muss mindestens entweder an einem Produkt mitwirken oder dafür verantwortlich sein.</p>	<p>Sie wird ansonsten bei der Erstellung des organisationsspezifischen V-Modells wieder entfernt</p>
<p>Stehen ein Arbeitsschritt und ein Thema miteinander in Beziehung, so muss die Aktivität, zu der der Arbeitsschritt gehört, mit dem Produkt in Beziehung stehen, zu dem das Thema gehört.</p>	
<p>Die Vorgehensbausteine des V-Modell-Kerns (Kennzeichnung durch Attribut bei Vorgehensbausteinen) müssen in allen Projekttypen und Projektvarianten als verpflichtende (bedingte) Vorgehensbausteine enthalten sein.</p>	
<p>Verweist ein(e) Produkt/Aktivität/Thema/Arbeitsschritt auf einen Textbaustein müssen die Attribute <i>Beschreibung</i> beziehungsweise <i>Sinn_und_Zweck</i> des betreffenden Elements leer sein.</p>	
<p>Jeder Textbaustein muss mindestens zweimal referenziert werden.</p>	
<p>Tailoringabhängigkeiten verweisen stets auf das Produkt <i>Projekthandbuch</i>.</p>	
<p>Produktabhängigkeiten verweisen nicht mehrfach auf dasselbe Produkt.</p>	
<p>Der in einer Projektdurchführungsstrategie beschriebene Ablauf muss zusammenhängend sein, d.h. ausgehend vom „ersten“</p>	

Bedingung	Ergänzende Erläuterung
<p>Entscheidungspunkt, muss jeder andere Entscheidungspunkt, der Bestandteil der jeweiligen Projektdurchführungsstrategie ist, erreichbar sein.</p>	
<p>Erster Entscheidungspunkt einer Projektdurchführungsstrategie ist stets in einem Ablaufbaustein zu finden, der der Ablaufbausteinspezifikation <i>Projektstart</i> entspricht (i.d.R.: „<i>Projekt genehmigt</i>“). Der letzte Entscheidungspunkt jedes Ablaufs ist stets in einem Ablaufbaustein zu finden, der der Ablaufbausteinspezifikation <i>Projektabschluss</i> entspricht (i.d.R.: „<i>Projekt abgeschlossen</i>“). In diesem Fall ist der Entscheidungspunkt „<i>Projekt abgeschlossen</i>“ also der einzige Entscheidungspunkt ohne Nachfolger.</p>	<p>Alle Projektdurchführungsstrategien setzen sich aus mehreren Ablaufbausteinen zusammen. Diese entsprechen Spezifikationen. Die Ablaufbausteinspezifikationen <i>Projektstart</i> und <i>Projektende</i> stellen dabei die Grenzen für Beginn und Ende einer jeden PDS dar. Die konkrete Ausgestaltung ist abhängig von den Ablaufbausteinen, die der jeweiligen Spezifikation genügen.</p>
<p>Wenn eine Aktivität durch einen Arbeitsschritt repräsentiert wird, darf sie nicht auf ein Produkt verweisen. Wenn eine Aktivität nicht durch einen Arbeitsschritt repräsentiert wird, muss sie auf ein Produkt verweisen.</p>	
<p>Ein/Das Referenzmodell hat keine Referenz auf eine Modellvariante (Element: <i>Referenzmodell</i>).</p>	
<p>Rekursive Einbindung von Ablaufbausteinen sind nicht gestattet über Ablaufbausteine und nicht über Ausdrücke. Ein Zyklus in einem Ablauf muss immer mithilfe von <i>Übergängen</i> modelliert werden.</p>	

## 6 Abkürzungsverzeichnis

Abkürzung	Erklärung
AB	Ablaufbaustein
AB-Spec	Ablaufbausteinspezifikation
EP	Entscheidungspunkt
PAH	Produktabhängigkeit
PM	Projektmerkmal
PMW	Projektmerkmalswert
PT	Projekttyp
PTV	Projekttypvariante
PDS	Projektdurchführungsstrategie
PDS-Spec	PDS-Spezifikation
VB	Vorgehensbaustein



## 7 Abbildungsverzeichnis

Abbildung 1 Überblick über die Paketstruktur des Metamodells.....	2
Abbildung 2 Basisstruktur des V-Modell XT als XML-Baum im V-Modell XT Editor .....	4
Abbildung 3 Das Basis-Paket, Bereich V-Modell-Struktur und Teile .....	5
Abbildung 4 Das Basis-Paket, Bereich Kapitel und Abschnitt.....	5
Abbildung 5 Das Basis-Paket, Bereiche: Glossar, Abkürzungen und Quellen .....	6
Abbildung 6 Das Basis-Paket, Bereich Methodenreferenz .....	7
Abbildung 7 Das Basis-Paket, Bereich Werkzeugreferenz .....	7
Abbildung 8 Struktur der Vorgehensbausteine im Metamodell .....	10
Abbildung 9 Das Produktmodell, Bereich Produkt .....	12
Abbildung 10 Das Produktmodell, Bereich externe Kopiervorlagen .....	13
Abbildung 11 Externe Kopiervorlagen im V-Modell XT Projektassistenten .....	13
Abbildung 12 Das Produktmodell, Bereich Thema) .....	14
Abbildung 13 Das Produktmodell, Bereich Unterthema .....	15
Abbildung 14 Das Produktmodell, Bereich Produktabhängigkeiten .....	16
Abbildung 15 Das Aktivitätsmodell, Bereich Aktivität .....	16
Abbildung 16 Das Aktivitätsmodell, Bereich Aktivitätsknoten .....	17
Abbildung 17 Das Aktivitätsmodell, Bereich Aktivitätskanten .....	18
Abbildung 18 Das Rollenmodell) .....	18
Abbildung 19 Explizite Beziehungstypen im V-Modell-Metamodell .....	19
Abbildung 20 Explizite Beziehungstypen, Gruppe: Rollenbeziehungen .....	19
Abbildung 21 Beispiel einer erzeugenden Produktabhängigkeit.....	20
Abbildung 22 2-stufiges Herstellen von Produktabhängigkeiten im Editor .....	21
Abbildung 23 Explizite Beziehungstypen, Gruppe: Produktabhängigkeitsbeziehungen.....	22
Abbildung 24 Explizite Beziehungstypen, Gruppe: Aktivitätsbeziehungen .....	23
Abbildung 25 Explizite Beziehungstypen, Gruppe: Produktbeziehungen .....	24
Abbildung 26 Das Paket Dynamik .....	26
Abbildung 27 PDS-Spezifikationen .....	26
Abbildung 28 Ablaufbausteine .....	27
Abbildung 29 Ablaufbausteine, Bereich Ablaufentscheidungspunkt .....	28
Abbildung 30 Ablaufbausteine, Bereich Ablaufbausteinpunkt.....	29
Abbildung 31 Ausschnitt aus dem Entwicklungsablauf für n-Auftragnehmer im AG-Projekt .....	30
Abbildung 32 Ablaufbausteine, Bereich Split .....	31
Abbildung 33 Ablaufbausteine, Bereich Join.....	31
Abbildung 34 Ablaufbausteine, Bereich Übergang .....	32
Abbildung 35 Ablaufbausteinspezifikationen.....	32
Abbildung 36 Projekttypvarianten, insb. Projektdurchführungsstrategie .....	33
Abbildung 37 Entscheidungspunkte .....	34
Abbildung 38 Grundlegende Elemente im Paket Anpassung .....	34
Abbildung 39 Projekttypen.....	35
Abbildung 40 Projekttypvarianten.....	36
Abbildung 41 Projektmerkmale und Projektmerkmalswerte.....	37
Abbildung 42 Konventionsabbildung .....	38
Abbildung 43 Erweiterte V-Modellvariante .....	40
Abbildung 44 Änderungsoperationen des V-Modell XT im Überblick .....	41
Abbildung 45 Änderungsoperationen, Bereich Disziplinänderungen .....	43
Abbildung 46 Änderungsoperationen, Bereich Produktänderungen .....	44
Abbildung 47 Änderungsoperationen, Bereich Themenänderungen .....	45
Abbildung 48 Änderungsoperationen, Bereich Aktivitätenänderungen.....	46
Abbildung 49 Änderungsoperationen, Bereich Rollenänderungen .....	47
Abbildung 50 Änderungsoperationen, Bereich Tailoringänderungen .....	48
Abbildung 51 Änderungsoperationen, Bereich Entscheidungspunktänderungen.....	48
Abbildung 52 Änderungsoperationen, Bereich Beschreibungstextersetzungen .....	49
Abbildung 53 Änderungsoperationen, Bereich Beschreibungstextergänzungen.....	50
Abbildung 54 Änderungsoperationen, Bereich Beschreibungstextverschiebungen .....	51
Abbildung 55 Änderungsoperationen, Bereich WerkzeugMethodenÄnderungen .....	51
Abbildung 56 Vortailoring im V-Modell XT.....	52
Abbildung 57 Tailoring-Prozess zur projektspezifischen Anpassung .....	53
Abbildung 58 Schritte 1 und 2 des Tailorings mit dem Projektassistenten .....	53
Abbildung 59 Schritt 3 des Tailorings mit dem Projektassistenten .....	54

## 8 Referenzen

- [1] V-Modell XT Webportal. [www.v-modell-xt.de](http://www.v-modell-xt.de)
- [2] Friedrich, J., Hammerschall, U., Kuhrmann, M., Sihling, M.: Das V-Modell XT. Springer, 2008.
- [3] Anleitung zum V-Modell XT Editor. Enthalten im Lieferumfang des V-Modell XT
- [4] W3C: Extensible Markup Language (XML). <http://www.w3.org/XML/>
- [5] W3C: XML-Schema. <http://www.w3.org/XML/Schema>
- [6] Object Management Group: Unified Modeling Language (UML). <http://www.omg.org/technology/documents/formal/uml.htm>