

# Ein Prozessintegrationsverfahren für das V-Modell XT und den Visual Studio Team Foundation Server

Marco Kuhrmann, Norbert Diernhofer

Technische Universität München,  
Institut für Informatik – Software & Systems Engineering  
Boltzmannstr. 3  
85748 Garching b. München, Germany  
{kuhrmann|diernhof}@in.tum.de

**Abstract:** Im Rahmen des Kooperationsprojekts *CollabXT*<sup>1</sup> wurde ein Verfahren entwickelt und erprobt, das eine (semi-)automatische Übersetzung und Integration eines projektspezifisch angepassten V-Modell XT vornimmt und dieses in die Projektmanagement- und Entwicklungsumgebung Visual Studio Team System mit Team Foundation Server integriert. Zur Anwendung kommen hier ein Meta-Template- und ein Generator-basierter Ansatz, der V-Modell XT Inhalte in die Zielumgebung integriert. Diese Dokumentation stellt die erarbeiteten Konzepte, Methoden und Werkzeuge vor.

## Inhalt

1	Einleitung .....	3
1.1	Das V-Modell XT .....	3
1.2	Inhalt .....	3
2	Abbildung des V-Modell XT .....	3
2.1	Abzubildende V-Modell Elemente .....	4
2.2	Abzubildende Prozesse .....	8
2.3	Weiteres und Besonderheiten .....	9
2.4	Work Item Design .....	10
2.4.1	Work Item: Produkt .....	10
2.4.2	Work Item: Aktivität .....	11
2.4.3	Work Item: Entscheidungspunkt .....	12
2.4.4	Work Item: Arbeitsauftrag .....	13
2.4.5	Work Item: Risiko und Maßnahme .....	14
2.4.6	Work Item: PÄM .....	16
2.4.7	Abfragen für die definierten Work Item Typen .....	18
3	Werkzeug .....	18
3.1	Architektur .....	18
3.2	Abgedeckte Anwendungsfälle und Anwenderunterstützung .....	19
3.3	Installation und Randbedingungen .....	20

---

<sup>1</sup> CollabXT ist ein Kooperationsprojekt zwischen der Technischen Universität München (V-Modell XT, Grundlagen, Konzept), Microsoft Deutschland GmbH (Werkzeughersteller) und der Zühlke Engineering GmbH (Entwicklung und Anwendung), in dem das V-Modell XT mithilfe der Werkzeuglandschaft von Microsoft in die Anwendung überführt werden soll.

3.4	Anwendung des Generators .....	21
3.4.1	Tailoring des V-Modell XT für CollabXT.....	21
3.4.2	Generierung des Process Templates .....	27
3.4.3	Installation im Team Foundation Server.....	29
3.4.4	Instanziierung und Verknüpfung der Default Work Items.....	33
4	Zusammenfassung .....	33
	Anhang – Referenzprofile.....	34
	Referenzprofil Auftragnehmer.....	34
	Inkrementelle Systementwicklung (AN) .....	37
	Agile Systementwicklung (AN).....	37
	Komponentenorientierte Systementwicklung (AN).....	38
	Wartung und Pflege von Systemen (AN) .....	39
	Referenzprofil Auftraggeber/Auftragnehmer .....	39
	Inkrementelle Systementwicklung (AG/AN) .....	42
	Agile Systementwicklung (AN).....	42
	Komponentenorientierte Systementwicklung (AN).....	43
	Wartung und Pflege von Systemen (AN) .....	44

# Ein Prozessintegrationsverfahren für das V-Modell XT und den Visual Studio Team Foundation Server

Marco Kuhrmann, Norbert Diernhofer

Technische Universität München,  
Institut für Informatik – Software & Systems Engineering  
Boltzmannstr. 3  
85748 Garching b. München, Germany  
{kuhrmann|diernhof}@in.tum.de

**Abstract:** Im Rahmen des Kooperationsprojekts *CollabXT*<sup>2</sup> wurde ein Verfahren entwickelt und erprobt, das eine (semi-)automatische Übersetzung und Integration eines projektspezifisch angepassten V-Modell XT vornimmt und dieses in die Projektmanagement- und Entwicklungsumgebung Visual Studio Team System mit Team Foundation Server integriert. Zur Anwendung kommen hier ein Meta-Template- und ein Generator-basierter Ansatz, der V-Modell XT Inhalte in die Zielumgebung integriert. Diese Dokumentation stellt die erarbeiteten Konzepte, Methoden und Werkzeuge vor.

## 1 Einleitung

Vorgehensmodellen sind oft umfangreiche Werkzeuglandschaften beiseite gestellt, die verschiedene Bereiche der Erstellung und Anwendung von Vorgehensmodellen unterstützen. CollabXT adressiert dies für den Kontext V-Modell XT und Visual Studio Team Foundation Server. Im Rahmen eines Teilprojekts dieses Vorhabens entstand in einer Kooperation ein Generatorsystem, das ein projektspezifisch angepasstes V-Modell XT so anpasst, dass es mit dem Team Foundation Server als Prozess betreibbar ist.

### 1.1 Das V-Modell XT

Das V-Modell<sup>®</sup>XT (kurz V-Modell) ist als verbindlicher deutscher Entwicklungsstandard für IT-Vorhaben in der öffentlichen Hand ein solches Vorgehensmodell. Es fasst moderne Konzepte und bewährtes Wissen in einem integrierten, generischen Standard zusammen, der sowohl für umsetzende Institutionen/Organisationen als auch auf für Projekte noch weit gehend angepasst werden kann. Weiterführende Literatur zum V-Modell ist unter [www.v-modell-xt.de](http://www.v-modell-xt.de) zu finden. Weiterhin ist auf den Webseiten der TU München unter <http://www4.in.tum.de/publ/html.php?e=990> ein technischer Bericht zu finden, in dem die grundlegenden Ideen und Konzepte zum Projekt CollabXT motiviert wurden.

### 1.2 Inhalt

Mit dem vorliegenden Dokument<sup>3</sup> liegt eine Dokumentation der entstandenen Lösung vor. Die Rahmenbedingungen auf der Seite des V-Modells erläutern wir ebenso wie die Abbildung auf die entsprechenden Strukturen des Team Foundation Servers. Weiterhin bieten wir einen kurzen Abriss der Anwendung der Software an.

## 2 Abbildung des V-Modell XT

Dies ist nur eine mögliche Realisierung der Idee hinter dem Projekt CollabXT. Hierfür sind im Vorfeld eine Reihe von Überlegungen und Vorarbeiten zu leisten, um eine gültige Struktur zu erhalten. In diesem Kapitel fokussieren wir alle relevanten Anteile des V-Modells, die für die Abbildung erforderlich sind. Es sind

---

<sup>2</sup> CollabXT ist ein Kooperationsprojekt zwischen der Technischen Universität München (V-Modell XT, Grundlagen, Konzept), Microsoft Deutschland GmbH (Werkzeughersteller) und der Zühlke Engineering GmbH (Entwicklung und Anwendung), in dem das V-Modell XT mithilfe der Werkzeuglandschaft von Microsoft in die Anwendung überführt werden soll.

<sup>3</sup> Dieses Dokument ist eine kompakte und anwenderorientiert gestaffte Version eines Technisches Berichts der TU München, welcher noch im 4. Quartal 2007 erscheint. Adresse für weitere Informationen hierzu: <http://www4.in.tum.de/~kuhrmann>.

sehr detaillierte Betrachtungen zu führen, da beiden Ansätzen zunächst grundsätzlich unterschiedliche Philosophien zugrunde liegen. Somit können nicht alle Elemente 1:1 abgebildet werden. Weiterhin sind bereits im Vorfeld Entscheidungen zu treffen, wie sich die Unterstützung des V-Modells durch eine integrierte Werkzeugumgebung in der Konkretisierung des Prozesses niederschlägt. Um hier eine grundlegende Vereinbarung hinsichtlich der Abbildungsvolumina zu treffen, legen wir zunächst die abzubildenden Elemente fest. Im Anschluss unterlegen wir diese Elemente mit einigen wesentlichen Prozessen für die Projektplanung und -führung. Auf Besonderheiten bei der Modellierung bzw. Komplikationen und nicht triviale Abbildungen gehen wir im Anschluss ein und entwerfen darauf hin die TFS Work Items in einer für das V-Modell geeigneten, minimalen Version.

## 2.1 Abzubildende V-Modell Elemente

Die im Rahmen der Abbildung relevanten Prozesselemente des V-Modells klassifizieren wir zunächst in die Kategorien *Steuerung* und *Inhalt*. Steuerungselemente finden wir dort, wo wir Projektfortschritte erfassen, messen und bewerten müssen. Inhaltselemente sind insbesondere im Bereich der Projektergebnisse zu finden. Bezogen auf das V-Modell finden wir Steuerungselemente in den Entscheidungspunkten; Inhaltselemente finden wir beispielsweise in Teilprozessen. Da das V-Modell vollständig durch ein formales Metamodelle beschrieben ist, ist dieses der Ausgangspunkt für die Ermittlung der Elemente. Jedoch sind nicht alle Elemente erforderlich, da bestimmte (technische) Aspekte durch die Zielumgebungen abgedeckt werden. Wir reduzieren die Abbildungselemente anhand der o.a. Klassifizierung und finden:

- Steuerungselemente
  - o Entscheidungspunkte
  - o Produkte (Status)
  - o Aktivitäten (Planung)
- Inhaltselemente (Teilprozesse):
  - o Arbeitsauftrag (Prozess *Task Management*)
  - o Risiko (Prozess *Risikomanagement*)
  - o PÄM (Prozess *Problem- und Änderungsmanagement*)

Produkte ordnen wir in unserer Modellierung nicht den Inhaltselementen zu. In unserer Modellierung sind Produkt-Work Items Repräsentanten von Produkten in einer Produktbibliothek. Im Kontext des TFS ist für das *Work Item Tracking* System eine Messgröße für Produkte einzuführen, damit der Status im Rahmen der Projektfortschrittsbestimmung erfasst werden kann. Hierzu wird für jedes Produktexemplar ein Work Item im System gepflegt, das im Wesentlichen die Statusmaschine des V-Modell-Produktsystems abbildet. Auf das Work Item Design und die Spezialitäten für Produkte gehen wir im Abschnitt 2.4 detailliert ein.

*Entscheidungspunkte*, *Produkte* und *Aktivitäten* beziehen wir in unserer Modellierung für die Projektsteuerung mit ein. Jedes dieser Metamodellelemente wird durch einen eigenen *Work Item Typ* realisiert. Wir definieren daher Daten und Workflows, die Informationen aus dem V-Modell ableiten und ggf. ergänzen. Inhaltlich stellen wir *kompakte Basisprozesse* des V-Modells bereit und integrieren *Risiko*-, *Aufgaben*- sowie *Problem- und Änderungsmanagement*. Das V-Modell stellt hierfür verschiedene Produkttypen bereit, die diese Prozesse „versteckt“ enthalten und in den einzelnen Produkt- und Aktivitätsbeschreibungen ansatzweise ausgestalten. Das V-Modell stellt an einigen Stellen so genannte *beispielhafte Produktgestaltungen* zur Verfügung, die wir bei der Modellierung in geeigneter Weise berücksichtigen. Wir explizieren sie dann und stellen sie entsprechend der Zielumgebung zur Verfügung.

Weitere Elemente, die wir in unsere Betrachtung mit aufnehmen sind die Prozessdokumentation und die erzeugte Produktbibliothek<sup>4</sup>. Wir bestimmen auf dieser Ebene nur die wesentlichen Typen für die Abbildung/Überführung. Im Kontext der Projektierung lagen zwei Referenzprofile (siehe Anhang – Referenzprofile) vor, die eine grobe Abschätzung zu erwartender Ergebnisse gestatten.

In Konsequenz aus diesen Betrachtungen werden folgende Elemente/Teile des V-Modells für die Abbildung herangezogen:

Element	Beschreibung
Entscheidungspunkt	Entscheidungspunkte werden als Planungs- und Messgrößen für die Projektsteuerung herangezogen. Entscheidungspunkte sind als Metamodellelemente des V-Modells definiert. Sie sind in einem Tailoring-Profil projektypspezifisch (umfangsmäßig) reduziert.

<sup>4</sup> Dies setzt ein vorhandenes Konfigurations- und Dokumentenmanagementsystem auf der Seite der Zielumgebung voraus.

Element	Beschreibung
	Entscheidungspunkte liegen für die Ausgestaltung bereits teilweise in instanzierter Form vor. Der V-Modell XT Projektassistent liefert im Rahmen der initialen Planung eine entsprechende, konsistente und geordnete Menge. Weitere Entscheidungspunktestellen sind ggf. im laufenden Projekt zu definieren.
Produkt	<p>Das Produkt als zentrales V-Modell Element ist als Metamodellelement definiert und wird als Repräsentant für ein Produktexemplar im Kontext der Modellierung erfasst. Die Messgrößen werden über das Produktstatusmodell erfasst, gemessen und ausgewertet.</p> <p>Bestimmte Produkttypen werden im Kontext dieser Anpassung in Form von Work Item Typen modelliert.</p>
Aktivität	Die Aktivität ist als Metamodellelement hinterlegt und muss im Rahmen der Modellierung als Planungsgröße herangezogen werden. Die Menge der Aktivitäten ist durch das Tailoring bestimmt. Initial sind maximal so viele Aktivitätstypen wie Produkttypen definiert. Aktivitäten, Produkte werden im Kontext der Entscheidungspunkte miteinander verknüpft.
Arbeitsauftrag	Wird auf Basis des Produkts <i>Arbeitsauftrag</i> und der assoziierten Aktivitäten erstellt und als Prozess abgebildet. → Abschnitt 2.2
Risiko	Wird auf der Basis des Produkts <i>Risikoliste</i> und der assoziierten Aktivitäten erstellt und als Prozess abgebildet. → Abschnitt 2.2
PÄM	Wird auf der Basis der Produkte <i>Problemmeldung/Änderungsforderung, -bewertung, -statusliste</i> sowie <i>Änderungsentscheidung</i> und der assoziierten Aktivitäten erstellt, integriert und zu einem Prozess integriert. → Abschnitt 2.2
Prozessdokumentation	<p>Die Prozessdokumentation ist auf der Modellebene im V-Modell selbst hinterlegt. Ein Export z.B. in das HTML-Format stellt nur eine spezialisierte Sicht zur Verfügung. Die Prozessdokumentation kann in verschiedener Weise verwendet werden. In unserem Kontext wird die Prozessdokumentation nach Anpassung und Export mit den V-Modell-Standardwerkzeugen 1:1 übernommen und als Hilfe integriert (Process Guidance).</p> <p>Weitere Anpassungen sind dann nicht mehr vorgesehen.</p>
Produktbibliothek	<p>Die Produktbibliothek wird ebenfalls standardmäßig exportiert. Sie wird dabei gleichzeitig spezialisiert und (dynamisch) angepasst.</p> <p>Sie kann, unter der Voraussetzung, dass ein entsprechendes Konfigurationsmanagementsystem verfügbar ist, 1:1 überführt werden. Weitere Anpassungen sind hierbei nicht vorgesehen. Die Produktbibliothek liefert auf der Grundlage des Tailorings die für das Projekt relevanten Vorlagentypen. Diese werden in das KM-System eingespeist und dort zweckabhängig verwendet. Unterschieden werden muss dabei zwischen initialen und abhängigen Produkten. Sie unterscheiden sich hinsichtlich des späteren Instanzierungsverhaltens bei und nach der Generierung.</p>

Tabelle 1 Elemente zur Abbildung der V-Modell XT Strukturen

Die in der Tabelle 1 aufgeführten Elemente finden sich zum Teil bereits in den Metamodellelemente des V-Modells wieder. Die Elemente *Produkt*, *Aktivität* und *Entscheidungspunkt* entstammen dabei den V-Modell Sub-Metamodellen: Produkt-, Aktivitäts- und Ablaufmodell. Diese Sub-Metamodelle besprechen wir im Anschluss.

**Produktmodell des V-Modells.** Das V-Modell XT ist ein *produktorientiertes* Vorgehensmodell, d.h. nicht Aktivitäten, sondern Projektergebnisse (in Form von Produkten) stehen im Zentrum des Prozesses. Abbildung 1 zeigt den betreffenden Ausschnitt des Metamodells für das Produktmodell des V-Modells. Das Produkt als solches besitzt nur zwei für uns relevante Beziehungstypen zu anderen Elementen. Einmal ist dies die *Rolle*, also ein Verweis der die Verantwortung für ein Produkt beschreibt. Der zweite Aspekt wird durch den *Entscheidungspunkt* gegeben, der Vereinbarungen hinsichtlich der Fertigstellung von Produkten trifft. Der Entscheidungspunkt ist dabei gleichzeitig Teil des Ablaufmodells.

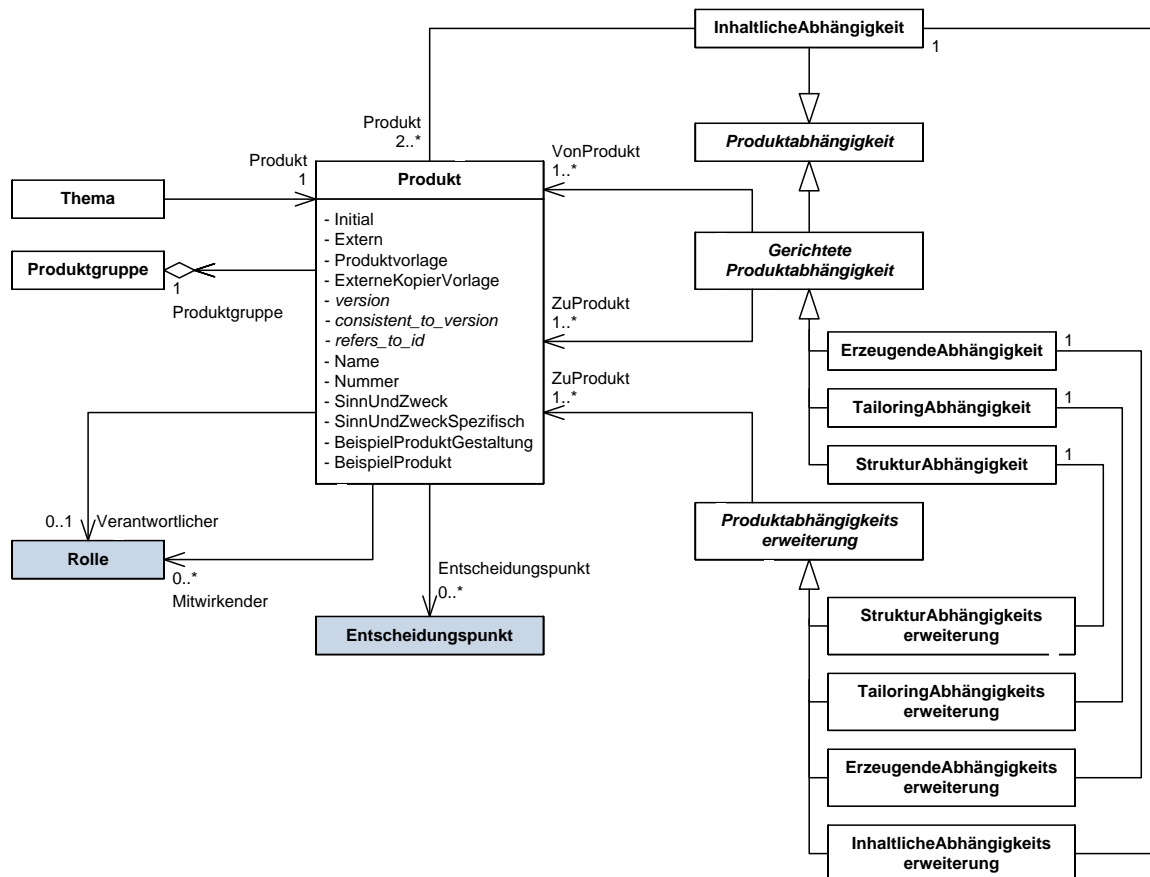


Abbildung 1 Ausschnitt des V-Modell XT Metamodells (Sicht: Produktmodell)

Weitere Elemente, die noch zu berücksichtigen sind, sind die Produktabhängigkeiten. Diese dienen im V-Modell dazu, Beziehungen und Abhängigkeiten zwischen Produkten auszudrücken. Für die Überführung auf TFS spielen diese Abhängigkeitstypen nur eine untergeordnete Rolle. Trotzdem stellen sie ein wesentliches Mittel des Prozesses dar, um Erzeugnisstrukturen des V-Modells verstehen und umsetzen zu können. Da das V-Modell seine Ergebnisse nicht durch eine Sequenz von Aktivitäten und deren Ausgaben beschreibt, dienen Produktabhängigkeiten dazu, insbesondere die erzeugenden Produktabhängigkeiten, die Notwendigkeit der Erstellung eines Produktexemplars zu regeln. Strukturelle Produktabhängigkeiten bilden die Basis der Systemstruktur im Entwicklungsanteil des V-Modells. Hier werden *Teile-Ganzes*-Beziehungen zwischen System, Teilsystemen und sonstigen Komponenten und Modulen hergestellt. Diese Strukturen finden sich im V-Modell auch bei den *Projektdurchführungsstrategien* wieder. Dort – insbesondere in den Entwicklungsanteilen der AN- und AG/AN-Strategien – sind durch die Entscheidungspunkte Ergebnisse angefordert, die sich passend in der Systemstruktur wieder finden.

*Obwohl Produktabhängigkeiten im V-Modell eine wesentliche Rolle spielen, berücksichtigen wir sie im Weiteren nicht. Im Rahmen der Abbildung belassen wir es dabei, sie im Rahmen der Dokumentation zu erhalten, wo sie durch die Bereitstellung eines Hyperlink-Systems einen wertvollen Beitrag leisten. Für die Abbildung auf den TFS spielen sie hingegen keine primäre Rolle.*

**Aktivitätsmodell des V-Modell XT.** Da das V-Modell ein produktzentriertes Vorgehensmodell ist, ist das Aktivitätsmodell vergleichsweise undetailliert ausgearbeitet. Vielmehr findet sich in den Aktivitäten eine eher allgemein gehaltene, grobgranulare Beschreibung von Aktionen, die zur Produkterstellung durchgeführt werden müssen. Aktivitäten im V-Modell dienen nur der Fertigstellung eines einzigen Produkts (siehe Abbildung 2, Assoziation *stellt\_fertig* 0..1 – 1). Die kontinuierliche Bearbeitung wird durch das V-Modell nicht beschrieben. Im Rahmen der Abbildung auf das aktivitätsorientierte TFS-System muss das Aktivitätsmodell daher erweitert und ausgestaltet werden.

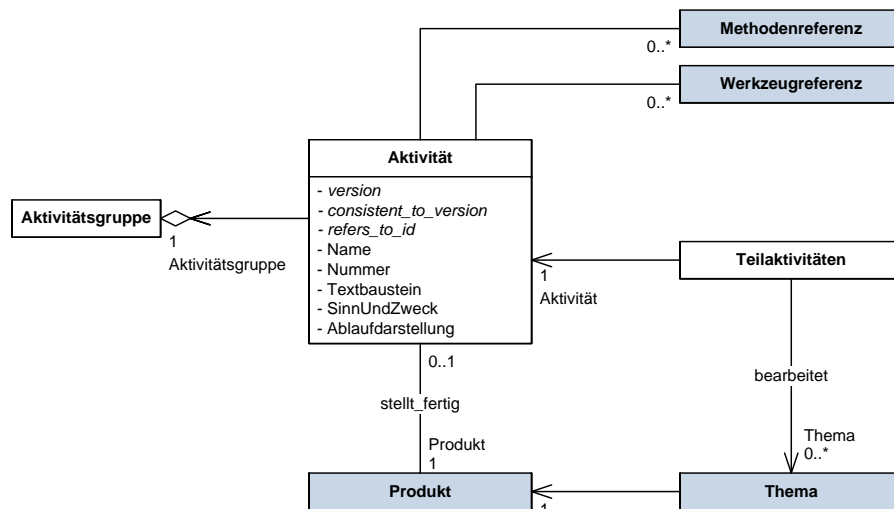


Abbildung 2 Ausschnitt des V-Modell XT Metamodells (Sicht: Aktivitätsmodell)

Unter der Aktivität ist auch noch die Größe *Teilaktivität* zu finden. Diese strukturiert eine Aktivität weiter. Ähnlich zu Produkten definiert eine Teilaktivität somit quasi Anforderungen an eine Aktivität. Anders als bei den Produkten, ist es hier sinnvoll, die Teilaktivitäten weiter zu berücksichtigen. Da Aktivitäten nur grobgranular beschrieben sind, haben wir hier die Möglichkeit durch eine Menge von Teilaktivitäten eine entsprechende Verfeinerung zu erhalten, mit deren Hilfe wir sogar eine Integration weiterer Prozesse und Methoden realisieren können.

*Aktivitäten betrachten wir im Folgenden als Planungsgrößen. Dies hat zur Folge, dass wir sie mir entsprechenden Informationen anreichern müssen, um sie unmittelbar für Planungswerkzeuge verfügbar zu machen. Wir nehmen diese Erweiterung im Rahmen der Generierung des passenden Work Item Typs im Kapitel 2.4 vor.*

**Ablaufmodell des V-Modell XT.** Das Ablaufmodell des V-Modell XT (Abbildung 3) dient der groben Strukturierung des Projekts und dessen Einteilung in verschiedene Projektfortschrittsstufen. Zentrale Elemente sind Entscheidungspunkte, die *Projektdurchführungsstrategien* und wieder die Produkte.

Der in Abbildung 3 gezeigte Ausschnitt des Metamodells zeigt die Einordnung dieser Elemente im V-Modell Metamodel. Über diesen Modellausschnitt wird aber nicht nur die (logische, zeitliche) Projektstrukturierung vorgenommen, sondern gleichzeitig die Planungskomponente des *V-Modell XT Projektassistenten* unterstützt. Diese greift auf die vorliegenden Daten zu und nutzt sie zur Erstellung des initialen Projektplans. Dabei verwendet er im Wesentlichen folgendes Verfahren:

- Eine Projektdurchführungsstrategie fasst mehrere Entscheidungspunkte zusammen und bringt diese in eine mögliche Ordnung.
- Bei der initialen Planung des Projekts werden die Entscheidungspunkte dann gemäß den Vorgaben verschieden oft instanziiert und entsprechend der Projektdurchführungsstrategie und der durch sie definierten Pfade zwischen den einzelnen Entscheidungspunkten geordnet.
- Entscheidungspunkte wiederum werden durch eine Reihe von Produkten referenziert, was nichts anderes heißt, als dass ein Produkt, das einen Entscheidungspunkt referenziert, zu einer Instanz dieses Entscheidungspunktes fertig gestellt und zur Qualitätssicherung vorgelegt werden muss.

Entscheidungspunkte regeln somit die Ergebnismengen in bestimmten Projektfortschrittsstufen. Sie verweisen weiterhin auf so genannte *Ablaufentscheidungspunkte*, die dann für Planungen herangezogen werden. Ein Ablaufentscheidungspunkt referenziert einen *Ablaufbaustein* und kann über eine Nachfolgerbeziehung weiter verzweigen (auch in Parallelabläufe). Der Ablaufbaustein mit Ablaufentscheidungspunkten und Parallelabläufen regelt somit die Erstellungsreihenfolge der Ergebnisse. Ein Ablaufbaustein ist dann wiederum einer Projektdurchführungsstrategie zugeordnet.

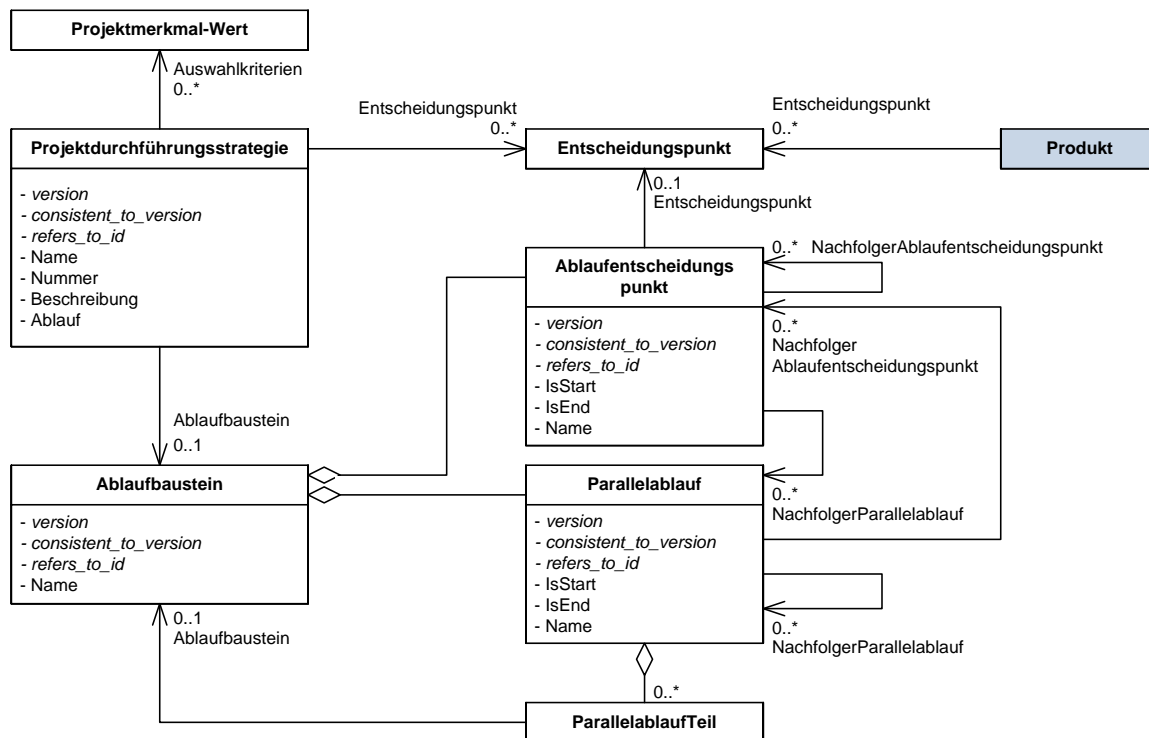


Abbildung 3 Ausschnitt des V-Modell XT Metamodells (Sicht: Ablaufmodell)

*Entscheidungspunkte werden im Projektplan üblicherweise als Meilensteine abgebildet, zu denen bestimmte Vorgänge fällig werden. Diese Umsetzung wird dem Entscheidungspunkt nicht vollständig gerecht, da er nicht nur ein Zieldatum für Aktivitäten/Vorgänge definiert, sondern gleichzeitig zu prüfende Zwischenergebnisse. Er enthält somit in der notwendigen Ausgestaltung sowohl Eigenschaften eines Meilensteins als auch die eines Quality Gates.*

## 2.2 Abzubildende Prozesse

Das V-Modell XT ist ein Vorgehensmodell, das neben der Produktzentrierung gleichzeitig versucht, eine hohe Wiederverwendbarkeit zu erreichen, indem es verschiedene Verfahren beschreibt und für die Anwendung empfiehlt, jedoch keine konkreten Methoden vorgibt. Das V-Modell ist *methodenneutral*. Für die Anwendung im Kontext TFS ist es jedoch erforderlich, an einigen Stellen Methoden auszugestalten, vorzufertigen und verbindlich vorzugeben. Diese vorgefertigten Methoden (ausgestaltete Prozesse) dienen dann im Weiteren als Schnittstellen für die bereits existierenden Prozesse der Zielumgebung, die passend und sinnvoll mit dem V-Modell integriert werden müssen. Aufgrund des Zielsystems TFS können wir Prozesse in Grenzen durch einen entsprechenden Entwurf von Work Items vorfertigen. Tabelle 1 gibt hierzu bereits drei passende Element-/Prozessdefinitionen an:

- Arbeitsauftrag (Prozess *Aufgaben-* bzw. *Taskmanagement*)
- Risiko (Prozess *Risikomanagement*)
- PÄM (Prozess *Problem- und Änderungsmanagement*)

**Aufgabenmanagement.** Das Aufgabenmanagement ist ein elementarer Prozess, der notwendig ist, um den Prozess V-Modell „zum Leben zu erwecken“. Elementare Aufgaben können somit bereits im Projekt vergeben und verfolgt werden. Eine einfache Statusüberwachung von Aufgaben ist möglich. Unter Berücksichtigung der Modellierung der Metamodellelemente Produkt, Aktivität und Entscheidungspunkt sind mit Hinzunahme des Aufgabenmanagements bereits alle elementaren Bestandteile für eine einfache Projektsteuerung vorhanden. Abgrenzen müssen wir die Konzepte *Arbeitsauftrag* und *Aktivität*, die wir im Kontext dieser Abbildung parallel vorhalten. Aktivitäten sind messbare Repräsentanten des Metamodelllkonzepts aus dem V-Modell. Für konkrete Aufgaben, bzw. für eine konkrete Leistungserbringung, sind Aktivitäten zu grobgranular, außerdem können sie kontinuierliche Arbeiten an einem Produkt nicht abdecken. Aktivitäten sind weiterhin ausschließlich einem Produkt zu geordnet. Diese Beschränkungen gelten nicht für den Arbeitsauftrag. Dieser ist ein Konzept, das eine konkrete Handlung verursacht. Aus diesem Grund modellieren wir bspw. keine *Teilaktivitäten* als Work Item Typ, sondern erfassen dieses Konzept sofort mit dem Arbeitsauftrag.



*Wir bilden auf der Konzeptebene die Kombination Aktivität/Teilaktivität auf Aktivität/Arbeitsauftrag ab.*

**Risikomanagement.** Das Risikomanagement im Rahmen des V-Modells wird umgesetzt durch das Produkt *Risikoliste* und die korrespondierende Aktivität. Das Produkt *Risikoliste* definiert als Themenstruktur die identifizierten Risiken und einen Maßnahmenplan zur Minderung und Behandlung von Risiken. Wir nehmen auch hier eine Ersetzung des V-Modell Produkts vor und bilden es auf eine durch TFS unmittelbar unterstützte Konstruktion ab. Zusätzlich müssen wir hier einen Workflow definieren, der wesentlich konkreter ist als die Vorgaben des V-Modells.

**Problem- und Änderungsmanagement.** Das Problem- und Änderungsmanagement ist ebenfalls wie Aufgaben- und Risikomanagement eine der zu unterstützenden Kerndisziplinen des V-Modells. Anders bei den anderen beiden betrachteten Disziplinen betrachten wir hier nicht nur einen Produkttyp, der in einem der Kernbausteine integriert ist, sondern einen auch im V-Modell separat hinterlegten Teilprozess. Diesen müssen wir komplett fertigen und sinnvoll auf einen Work Item Typ abbilden. Im Abschnitt 2.4.6 gehen wir darauf detailliert ein.

## 2.3 Weiteres und Besonderheiten

Es gibt eine Reihe von Disziplinen und Elementen im V-Modell, die geeignet wären, als Work Items definiert zu werden. Andere Elemente im V-Modell entfallen bei genauerer Betrachtung der Zielplattform und werden entsprechend der Gegebenheiten abgebildet. Beispielhaft sei hier das Konfigurationsmanagement (KM) zu nennen, das in wesentlichen Teilen durch die Infrastruktur des TFS erbracht wird. Das V-Modell bietet einige Besonderheiten, die das Work Item Design erschweren. Die Prozesse, die im V-Modell enthalten sind, sind sehr generisch hinterlegt und können weit interpretiert werden. Wir geben mit der Version 1.0 des CollabXT-TFS Generators *nur eine von vielen möglichen Interpretationen* an.

**Anpassung des V-Modells für Organisationen.** Der letzte Punkt muss noch einmal tiefer gehend betrachtet werden. Wir definieren unser Konzept/unsere Abbildung auf der Basis des Standard V-Modells 1.2.1. Üblicherweise wird das V-Modell in dieser reinen Form nicht verwendet, sondern organisationsspezifisch angepasst. Diese Anpassungsspezifika erfassen wir nicht, sondern definieren eine Abbildung, die das V-Modell trotz Integration in eine konkrete Entwicklungsplattform, noch weitgehend allgemein hält. Die organisationsspezifische Anpassung sehen wir daher nicht nur als Option an, sondern als *Notwendigkeit*. Zusätzlich zur Anpassung des V-Modells kommen hier ggf. weitere Anpassungen im Kontext des TFS Templates hinzu.

Zunächst ist der Standardentwicklungszyklus des V-Modells zu nennen, in dem das V-Modell ausgehend vom Standard organisationsspezifisch angepasst wird. Diese Anpassung findet mit den Standardwerkzeugen statt. Der der aktuell gültige<sup>5</sup> Anpassungsprozess für das V-Modell XT orientiert sich an den Werkzeugen der Standardwerkzeugkette des V-Modells und den Vorgaben, die das V-Modell durch das Metamodell macht. Ausgegangen wird vom Standard V-Modell XT, das durch einen Prozessingenieur an die jeweilige Organisation angepasst wird. Dieser Vorgang ist durch den V-Modell XT Editor werkzeugunterstützt. Als Ergebnis liegt ein *organisationsspezifisches V-Modell* vor, das weiter gepflegt oder dem V-Modell XT Projektassistent als Eingabe zugeführt werden kann. Der Projektassistent nimmt das organisationsspezifische V-Modell, mischt Mustertexte bei und stellt ein Mastertemplate für die Vorlagengenerierung bereit. Mit diesen Eingaben führt er das Tailoring durch. Als Ausgabe produziert der Projektassistent das projektspezifische V-Modell, einen initialen Projektplan und eine initiale Produktbibliothek, die die generierten Produktvorlagen enthält.

*Diese Ausgaben werden mithilfe der Werkzeuge aus CollabXT für die konkrete Zielplattform transformiert.*

Hier ist das der TFS als Ziel, der selbst auch eine Werkzeugunterstützung anbietet. Ein Prozessingenieur muss zunächst alle relevanten Inhalte zusammentragen und diese dann mithilfe des *Process Template Editor* sinnvoll zusammenstellen. Diese Aufgabe übernehmen die Werkzeuge aus CollabXT, die die Inhalte aus dem V-Modell XT entnehmen und auf eine gültige Templatestruktur übersetzen. Das Ergebnis muss in einen TFS übertragen werden. Hierzu dient der *Process Template Manager*, der neben der Bereitstellung der Templates gleichzeitig auch einen Export für weitere Anpassungen gestattet.

*Mit CollabXT werden diese beiden Lebenszyklusmodelle miteinander kombiniert. Problematisch an dieser Kombination ist, dass Entwicklungen parallel verlaufen können und beide Lebenszyklusmodelle entsprechende Anpassungen vorsehen. Die nachlaufende Anpassung eines aus einem V-Modell generierten Pro-*

---

<sup>5</sup> Ein verbindlicher, prozesslinienorientierter Prozess zur standardkonformen Entwicklung und Anpassung des V-Modell XT ist zurzeit noch nicht verfügbar und wird im Rahmen des Projekts WEIT-IV/2 und V-Bench entwickelt.

cess Templates stellt hier einen besonders kritischen Punkt dar, da Anpassungen im Template nicht auf das ursprünglich verwendete V-Modell wirken. Das Ergebnis einer solchen Anpassung ist durch den Generator nicht wiederherstellbar, womit die beiden Prozesse nicht mehr konsistent sind. Im schlimmsten Fall geht die Konformität zum V-Modell XT verloren. Pauschal kann hierzu jedoch keine Aussage getroffen werden – dies ist im Zweifelsfall durch ein individuelles Assessment zu klären.

## 2.4 Work Item Design

In diesem Abschnitt stellen wir das Work Item Design für den TFS vor. Mit den identifizierten Work Item Typen decken wir einen **grundlegenden** Umfang an zu überwachenden Anteilen des V-Modells ab.

### 2.4.1 Work Item: Produkt

Der Work Item Typ *Produkt* entspricht dem Metamodellelement Produkt des V-Modell XT Metamodells (vgl. Abbildung 1). Es dient als Repräsentant für ein Produktexemplar, das in der Dokumentenverwaltung des TFS (genauer des hinter dem TFS liegenden SharePoint Servers) aufbewahrt wird. Gleichzeitig bildet dieser Work Item Typ das *Produktzustandsmodell* des V-Modell Produktmodells ab. Produkte sind somit nach den Regeln des V-Modells mit verschiedenen Stati versehen und dem entsprechend überwachbar. Änderungen am Status sind ebenso einfach nach zu verfolgen.

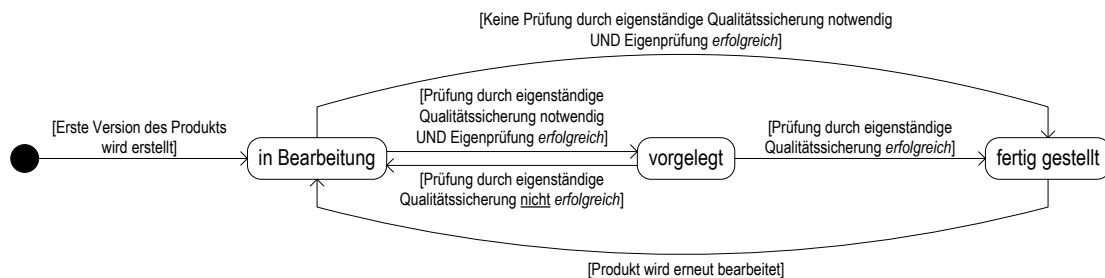


Abbildung 4 Produktzustandsautomat des V-Modell XT

Das Zustandsmodell für den Produktstatus hat einige Konsequenzen für die Work Items der Typen *Produkt* und *Aktivität*. Einerseits muss das Zustandsmodell für den Aktivitäts-Work Item Typ anhand dieses Modells entworfen werden, da die Aktivitäten im Kontext des V-Modells primär dazu dienen, Produkte fertig zu stellen. Diese Aufgabe muss mindestens beim Entwurf berücksichtigt werden. In der Modellierung des passenden Zustandsautomaten sind wir jedoch vergleichsweise frei, da das V-Modell hierzu keine Angaben macht. Andererseits ist das Zustandsmodell entscheidend für das Handling der einzelnen Produkttypen im Kontext der Qualitätssicherung. Technisch lässt sich die Zuordnung zur eigenständigen Qualitätssicherung in Form von Abfragen über den Work Items eines Projekts realisieren.

*Der Zustandsautomat des Produkts muss durch einen Work Item Workflow abgebildet werden.*

V-Modell Produkte besitzen ein Erzeugungsattribut, welches aussagt, wie Exemplare eines Produkts im Rahmen eines Projekts erstellt werden. Das V-Modell unterscheidet *initiale*, *abhängige* und *externe* Produkte. Abhängig und extern erzeugte Produkte sind unspektakulär, da sie nicht unmittelbar erzeugt werden, sondern zunächst nur Planungsgrößen sind und dann bei „Bedarf“ erstellt, bzw. dem Projekt von außen zugeführt werden.

Initiale Produkte werden hingegen genau einmal in einem Projekt erzeugt. Im Kontext eines Work Items heißt das, dass es nur genau eine Instanz eines solchen Produkt Work Items gibt, auf das sich alle geplanten Aktivitäten beziehen. So z.B. ist in verschiedenen Entscheidungspunkten das Projekthandbuch vorzulegen, weshalb hier ggf. Aktivitäten eingeplant sind. Alle diese Aktivitäten beziehen sich immer auf dasselbe Projekthandbuch. Folgendes Datenmodell legen wir dem Work Item Typ *Produkt* zugrunde:

Name	Ref.-Name	Beschreibung
Titel	System.Title	Bezeichnung des Work Items.
Zustand	System.State	Workflow und Status des Work Items.
Grund	System.Reason	Der Grund, warum dieses Work Item in diesem Zustand ist.
Iterationspfad	System.IterationPath	Der Zeitabschnitt, dem dieses Work Item zugeordnet ist.
Zugewiesen an	System.AssignedTo	Der Verantwortliche für dieses Work Item.
Geschlossen von	Microsoft.VSTS.Common.ClosedBy	Name desjenigen, der dieses Work Item geschlossen hat.
Schließungsdatum	Microsoft.VSTS.Common.ClosedDate	Datum, an dem dieses Work Item geschlossen wurde.
Beschreibung	System.Description	Beschreibung des Work Items.
Verlauf	System.History	Verlauf und historische Informationen.
Bereichspfad	System.AreaPath	Der logische Bereich, dem dieses Work Item zugeordnet wurde.
Produkttyp (VMXT)	VMXT.ProduktTyp	Produkttyp, zu dem dieses Produkt gehört.
Erzeugung (VMXT)	VMXT.Erzeugung	Die Art der Produkterzeugung
vmxt_ref (VMXT)	VMXT.id	
Bemerkungen (VMXT)	VMXT.Log	Feld für generelle Anmerkungen und Diskussionen.
Prüfung (VMXT)	VMXT.Assessment	Dokumentation der Prüfungsergebnisse.
Verbleibende Arbeit	Microsoft.VSTS.Scheduling.RemainingWork	Schätzung noch zu erbringender Aufwände (in Stunden).
Startdatum	Microsoft.VSTS.Scheduling.StartDate	
Abschlussdatum	Microsoft.VSTS.Scheduling.FinishDate	Geplantes Datum, zu dem dieses Work Item abgeschlossen sein soll.

Tabelle 2 Datenfelder für Produkte

Für das Feld *VMXT.Erzeugung* greift der Wertebereich des V-Modell XT Produktmodells: {initial, extern, abhängig}, um die Art der Produkterzeugung zu zeigen. Folgende Modellierung ergibt sich für den Workflow des Produkt Work Item Typs auf der Grundlage des Produktzustandsmodells des V-Modells:

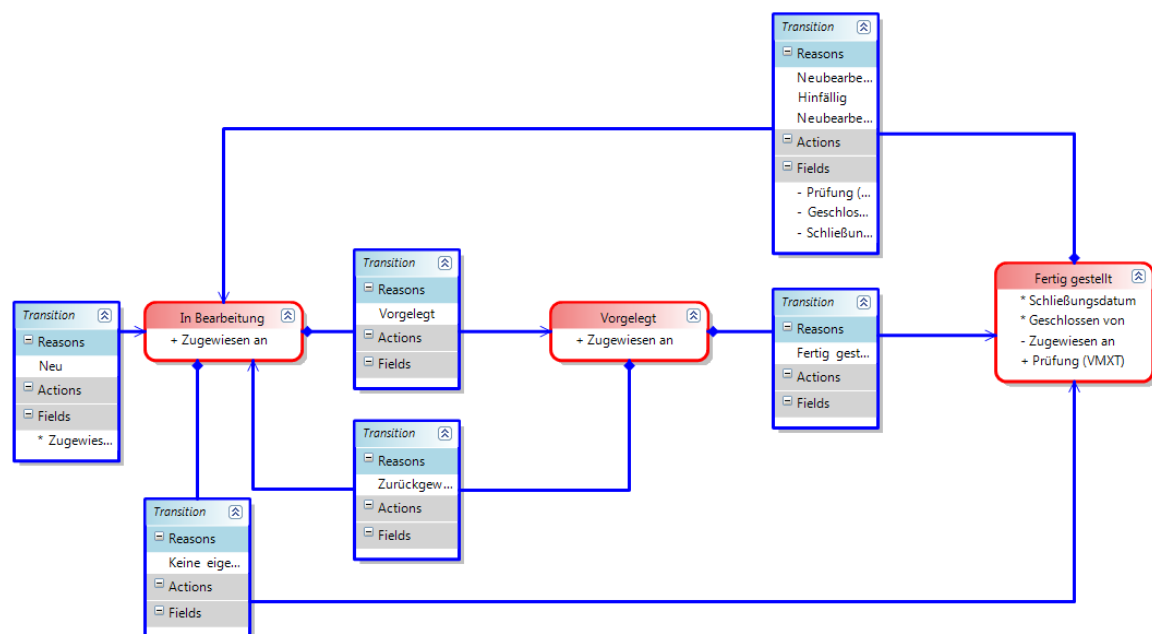


Abbildung 5 Workflow für Produkte

**Konsistenzbedingungen.** Ein Produkt darf nur in den Zustand *fertig gestellt* überführt werden,

- wenn die Aktivität, die das Produkt erstellt abgeschlossen ist und
- alle definierten QS-Maßnahmen erfolgreich ausgeführt wurden.

## 2.4.2 Work Item: Aktivität

Aktivitäten sind im Metamodell des V-Modells als Elemente enthalten, jedoch sind sie im V-Modell vergleichsweise undetailliert ausgearbeitet. Sind Aktivitäten definiert, so beziehen sie sich immer nur auf genau ein Produkt, können jedoch beliebig viele Teileaktivitäten enthalten (Teileaktivitäten verweisen auf eine Aktivität zu der sie gehören, Abbildung 2). Aktivitäten verfügen im V-Modell nur über deskriptiven Charakter. Sie enthalten keinerlei planungsrelevanten Informationen (Daten, Plan, Ist, Aufwände etc.). Im Rahmen des Tailorings werden die Aktivitäten instanziiert und mit Endedaten und Vorgänger-/Nachfolgerbeziehungen angereichert.

Weitere Informationen erhalten Aktivitäten jedoch nicht, sodass wir sowohl Daten als auch Verhalten ergänzen bzw. neu definieren müssen. Folgende Daten liegen der Modellierung der Aktivitäten zugrunde (Ergänzungen sind hervorgehoben):

Name	Ref.-Name	Beschreibung
Titel	System.Title	Bezeichnung des Work Items.
Zustand	System.State	Workflow und Status des Work Items.
Grund	System.Reason	Der Grund, warum dieses Work Item in diesem Zustand ist.
Iterationspfad	System.IterationPath	Der Zeitabschnitt, dem dieses Work Item zugeordnet ist.
Zugewiesen an	System.AssignedTo	Der Verantwortliche für dieses Work Item.
Geschlossen von	Microsoft.VSTS.Common.ClosedBy	Name desjenigen, der dieses Work Item geschlossen hat.
Schließungsdatum	Microsoft.VSTS.Common.ClosedDate	Datum, an dem dieses Work Item geschlossen wurde.
Beschreibung	System.Description	Beschreibung des Work Items.
Verlauf	System.History	Verlauf und historische Informationen.
Bereichspfad	System.AreaPath	Der logische Bereich, dem dieses Work Item zugeordnet wurde.
vmxt_ref (VMXT)	VMXT.id	Die V-Modell GUID des Elements
ep_ref (VMXT)	VMXT.EPRef	Entscheidungspunkt Referenz
p_ref (VMXT)	VMXT.PRef	Produktreferenz
Bemerkungen (VMXT)	VMXT.Log	Feld für generelle Anmerkungen und Diskussionen.
Verbleibende Arbeit	Microsoft.VSTS.Scheduling.RemainingWork	Schätzung noch zu erbringender Aufwände (in Stunden).
Startdatum	Microsoft.VSTS.Scheduling.StartDate	
Abschlussdatum	Microsoft.VSTS.Scheduling.FinishDate	Geplantes Datum, zu dem dieses Work Item abgeschlossen sein soll.

Tabelle 3 Datenfelder für Aktivitäten

Folgende Modellierung liegt der Definition des Workflows zugrunde:

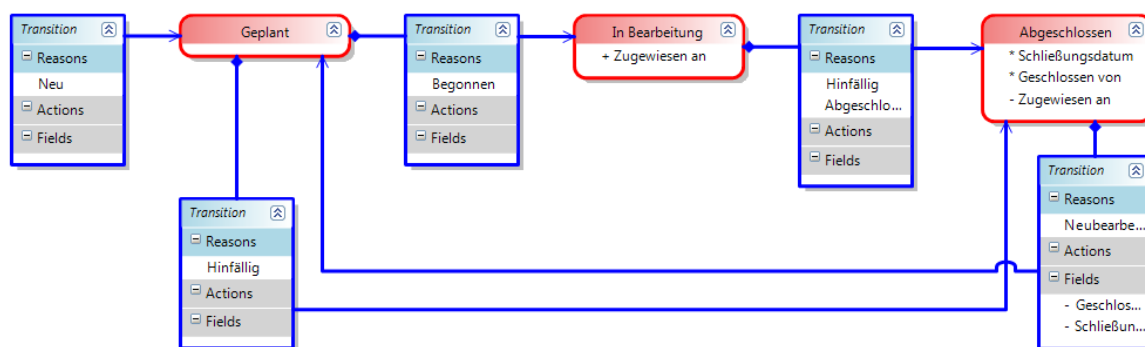


Abbildung 6 Workflow für Aktivitäten

**Konsistenzbedingungen.** Eine Aktivität darf nur abgeschlossen werden,

- wenn alle enthaltenen Arbeitsaufträgen oder Aktivitäten abgeschlossen sind.
- Wird eine Aktivität abgeschlossen, muss sich der Zustand referenzierter Produkte ändern.
- Eine Aktivität referenziert max. ein Produkt.

### 2.4.3 Work Item: Entscheidungspunkt

Der Entscheidungspunkt hat im V-Modell die Aufgabe, Produkte zusammenzufassen, die im Rahmen eines Quality Gates eine Bestimmung des Projektfortschritts zulassen. Im Entscheidungspunkt sind alle vorzulegenden Produkte im Zustand *fertig gestellt* vorzuhalten. Dies hat zur Folge, dass alle hier geplanten Aktivitäten abgeschlossen sein müssen. Dies zeigt den Meilensteincharakter des Entscheidungspunkts. Im Projektplan wird er deshalb als solcher instanziiert. Für einen Entscheidungspunkt definiert das V-Modell fast keine im Rahmen eines Work Item Trackings verwertbaren Daten. Diese und ein entsprechendes Zustandsmodell müssen im Rahmen der Modellierung ergänzt werden. Folgende Daten legen wir der Modellierung des Work Item Typs *Entscheidungspunkt* zugrunde:

Name	Ref.-Name	Beschreibung
Titel	System.Title	Bezeichnung des Work Items.
Zustand	System.State	Workflow und Status des Work Items.
Grund	System.Reason	Der Grund, warum das Work Item in diesem Zustand ist.
Iterationspfad	System.IterationPath	Der Zeitabschnitt, dem dieses Work Item zugeordnet ist.
Zugewiesen an	System.AssignedTo	Der Verantwortliche für dieses Work Item.
Geschlossen von	Microsoft.VSTS.Common.ClosedBy	Name desjenigen, der das Work Item geschlossen hat.
Schließungsdatum	Microsoft.VSTS.Common.ClosedDate	Datum, an dem das Work Item geschlossen wurde.
Beschreibung	System.Description	Beschreibung des Work Items.
Verlauf	System.History	Verlauf und historische Informationen.
Bereichspfad	System.AreaPath	Der logische Bereich, dem dieses Work Item zugeordnet wurde.
Entscheidungspunkttyp (VMXT)	VMXT.Entscheidungspunkttyp	Typ des Entscheidungspunkts.
vmxt_ref (VMXT)	VMXT.id	
Prüfung (VMXT)	VMXT.Assessment	Dokumentation der Prüfungsergebnisse.
Bemerkungen (VMXT)	VMXT.Log	Feld für generelle Anmerkungen und Diskussionen.
Verbleibende Arbeit	Microsoft.VSTS.Scheduling.RemainingWork	Schätzung noch zu erbringender Aufwände (in Stunden).
Startdatum	Microsoft.VSTS.Scheduling.StartDate	
Abschlussdatum	Microsoft.VSTS.Scheduling.FinishDate	Geplantes Datum, zu dem dieses Work Item abgeschlossen sein soll.

Tabelle 4 Datenfelder für Entscheidungspunkte

Das Feld *VMXT.Entscheidungspunkttyp* beinhaltet als vorgegebene Werte die Bezeichner der durch das V-Modell XT vorgegebenen Entscheidungspunkte:

- Abnahme erfolgt
- Anforderungen festgelegt
- Angebot abgegeben
- Feinentwurf abgeschlossen
- Gesamtprojekt aufgeteilt
- Gesamtprojektfortschritt überprüft
- Iteration geplant
- Lieferung durchgeführt
- Projekt abgeschlossen
- Projekt ausgeschrieben
- Projekt beauftragt
- Projekt definiert
- Projektfortschritt überprüft
- Projekt genehmigt
- Systemelemente realisiert
- System entworfen
- System integriert
- System spezifiziert
- Nicht zugeordnet (entspricht dem *freien Meilenstein* des V-Modell XT Projektassistenten)

Folgende Modellierung legen wir dem Workflow des Work Item Typs *Entscheidungspunkt* zugrunde:

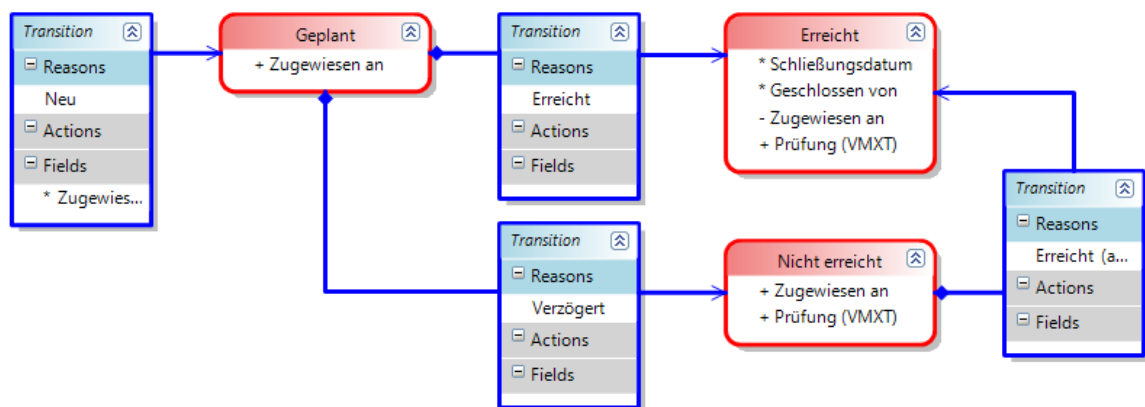


Abbildung 7 Workflow für Entscheidungspunkte

**Konsistenzbedingungen.** Ein Entscheidungspunkt darf nur dann erfolgreich abgeschlossen (erreicht) werden, wenn:

- Alle Produkte, die zum Entscheidungspunkt geplant sind, im Zustand fertig gestellt vorliegen.
- Alle Aktivitäten, die im Entscheidungspunkt mit Fälligkeit geplant sind vorliegen.

#### 2.4.4 Work Item: Arbeitsauftrag

Der Work Item Typ *Arbeitsauftrag* ist ein fachlicher Work Item Typ, der keine V-Modell-bezogenen Statusinformationen enthält, sondern einen im V-Modell integrieren Prozess abdeckt. Das Aufgabenmanagement wird im V-Modell durch ein Produkt vom Typ *Arbeitsauftrag* repräsentiert. Die korrespondierende Aktivität ist *Arbeitsauftrag erteilen*. Im Rahmen unserer Abbildung auf TFS stellen wir einen Aufgabenmanagementprozess auf der Basis des Arbeitsauftrags bereit, der gleichzeitig die Teilaktivitäten ersetzen soll.

Eine Aktivität nach V-Modell-Interpretation für den TFS ist somit ein Container für spezifische Arbeitsaufträge. Arbeitsaufträge können auch ohne Bezug zu einer Aktivität im Projekt vergeben werden, wodurch sie nicht unter „Produktzwang“ stehen und daher relativ einfach das Projekttageschäft unterstützen können. Der Modellierung des Work Item Typs *Arbeitsauftrag* legen wir folgende Daten zugrunde<sup>6</sup>:

<sup>6</sup> Das V-Modell definiert zu verschiedenen Produkten so genannte beispielhafte Produktgestaltungen. Wir orientieren uns weitgehend an diesen, stellen jedoch im Zweifelsfall die Einfachheit und Klarheit voran. Weitere Ausgestaltungen, insbesondere im Rahmen von organisationspezifischen Anpassungen sind aber möglich und durch das V-Modell erwünscht. In wie fern eine stark abweichende, inhaltlich Ausgestaltung den Regeln der V-Modell Konformität widerspricht, muss ggf. durch ein separates Assessment geklärt werden. Sofern die Anpassung/Ausgestaltung im Rahmen einer organisationspezifischen Anpassung nach V-Modell XT stattgefunden

Name	Ref.-Name	Beschreibung
Titel	System.Title	Bezeichnung des Work Items.
Zugewiesen an	System.AssignedTo	Der Verantwortliche für dieses Work Item.
Verlauf	System.History	Verlauf und historische Informationen.
Bereichspfad	System.AreaPath	Der logische Bereich, dem dieses Work Item zugeordnet wurde.
Iterationspfad	System.IterationPath	Der Zeitabschnitt, dem dieses Work Item zugeordnet ist.
Zustand	System.State	Workflow und Status des Work Items.
Grund	System.Reason	Der Grund, warum das Work Item in diesem Zustand ist.
Beschreibung	System.Description	Beschreibung des Work Items.
Schließungsdatum	Microsoft.VSTS.Common.ClosedDate	Datum, an dem das Work Item geschlossen wurde.
Geschlossen von	Microsoft.VSTS.Common.ClosedBy	Name desjenigen, der das Work Item geschlossen hat.
Priorität	Microsoft.VSTS.Common.Priority	Priorität des Arbeitsauftrags zur Fertigstellung.
Schweregrad	Microsoft.VSTS.Common.Severity	Gewichtung der Auswirkungen und Effekte des Arbeitsauftrags für das Projekt.
Disziplin (VMXT)	VMXT.Discipline	Die Disziplin, zu der der Arbeitsauftrag gehört.
Schätzung (VMXT)	VMXT.Estimate	Aufwandsschätzung für einen Arbeitsauftrag (in Stunden)
Blockiert (VMXT)	VMXT.Blocked	Ist die Weiterbearbeitung des Work Items blockiert?
Verbleibende Arbeit	Microsoft.VSTS.Scheduling.RemainingWork	Schätzung noch zu erbringender Aufwände (in Stunden).
Abgeschlossene Arbeit	Microsoft.VSTS.Scheduling.CompletedWork	Bereits geleistete Arbeit in einem Arbeitsauftrag (in Stunden).
Startdatum	Microsoft.VSTS.Scheduling.StartDate	
Abschlussdatum	Microsoft.VSTS.Scheduling.FinishDate	
Aufgabenhierarchie	Microsoft.VSTS.Scheduling.TaskHierarchy	Ein String der den Microsoft Project Kontext für diesen Arbeitsauftrag angibt.
Prüfung (VMXT)	VMXT.Assessment	Dokumentation der Prüfungsergebnisse.
Bemerkungen (VMXT)	VMXT.Log	Feld für generelle Anmerkungen und Diskussionen.

Tabelle 5 Datenfelder für Arbeitsaufträge

Im Arbeitsauftrag sind verschiedene Felder mit definierten Wertebereichen hinterlegt.

- Priorität (*Microsoft.VSTS.Common.Priority*) hat den Wertebereich: {1, 2, 3}.
- Schweregrad (*Microsoft.VSTS.Common.Severity*) hat den Wertebereich: {kritisch, hoch, mittel, niedrig}.
- Disziplin (VMXT) (*VMXT.Discipline*) hat den Wertebereich:
  - o Entwicklung
  - o Qualitätssicherung
  - o Projektmanagement
  - o Anforderungsfestlegung
  - o Architektur, Entwurf, Spezifikation
  - o Konfigurationsmanagement
  - o Ausschreibungs- und Vertragswesen
  - o Problem- und Änderungsmanagement, um eine nähere Klassifikation eines Arbeitsauftrags durchzuführen.
- Blockiert (VMXT) (*VMXT.Blocked*) hat den Wertebereich: {ja, nein}.

Folgende Modellierung liegt dem Workflow des Work Item Typs *Arbeitsauftrag* zugrunde:

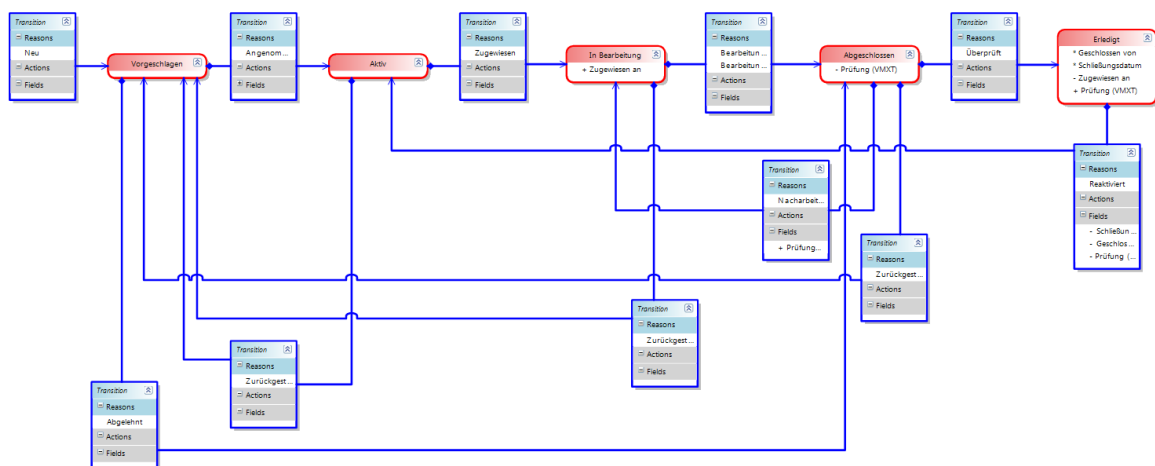


Abbildung 8 Workflow für Arbeitsaufträge

## 2.4.5 Work Item: Risiko und Maßnahme

Die Work Item Typen *Risiko* und *Maßnahme* sind ebenfalls fachliche Typen, die einen komplexen Teilprozess des Projektmanagements – das Risikomanagement – adressieren. Das V-Modell berücksichtigt dies

hat, ist jedoch nicht mit Problemen zu rechnen. Ggf. muss hier ein angepasster Work Item Typ erstellt werden. **Diese Aussage gilt stellvertretend für alle inhaltlichen Work Item Typen.**

durch den Produkttyp *Risikoliste*, der alle erkannten Risiken und einen korrespondierenden Maßnahmenplan enthält. Analog zum Arbeitsauftrag sind hier verschiedene Ausgestaltungen möglich. Folgende Datenstruktur legen wir dem Work Item Typ *Risiko* zugrunde:

Name	Ref.-Name	Beschreibung
Titel	System.Title	Bezeichnung des Work Items.
Zustand	System.State	Workflow und Status des Work Items.
Grund	System.Reason	Der Grund, warum dieses Work Item in diesem Zustand ist.
Zugewiesen an	System.AssignedTo	Der Verantwortliche für dieses Work Item.
Geschlossen von	Microsoft.VSTS.Common.ClosedBy	Datum, an dem dieses Work Item geschlossen wurde.
Schließungsdatum	Microsoft.VSTS.Common.ClosedDate	Name desjenigen, der dieses Work Item geschlossen hat.
Erstellungsdatum	System.CreatedDate	Datum, an dem dieses Work Item erzeugt wurde.
Beschreibung	System.Description	Beschreibung des Work Items.
Verlauf	System.History	Verlauf und historische Informationen.
Wahrscheinlichkeit (VMXT)	VMXT.Likelihood	Eintrittswahrscheinlichkeit des Risikos in %.
Auswirkung auf Zeit (VMXT)	VMXT.ImpactTime	Qualitative Auswirkung des Risikos auf die Zeit.
Auswirkung auf Qualität (VMXT)	VMXT.ImpactQuality	Qualitative Auswirkung des Risikos auf die Qualität.
Auswirkung auf Budget (VMXT)	VMXT.ImpactBudget	Qualitative Auswirkung des Risikos auf das Budget.
Auswirkung auf Umfang (VMXT)	VMXT.ImpactScope	Qualitative Auswirkung des Risikos auf den Funktionsumfang.
Bereichspfad	System.AreaPath	Der logische Bereich, dem dieses Work Item zugeordnet wurde.
Risikoschaden (VMXT)	VMXT.Severity	Der geschätzte Schaden, der im Schadensfall mit einem Risiko im Projekt verbunden ist.
Strategietyp (VMXT)	VMXT.StrategyType	Typ der Risikostrategie.
Strategiebeschreibung (VMXT)	VMXT.StrategyDescription	Beschreibung der Risikostrategie.
Prüfung (VMXT)	VMXT.Assessment	Dokumentation der Prüfungsergebnisse.
Maßnahmenbeschreibung (VMXT)	VMXT.MeasureComment	Beschreibung der Links zu den Maßnahmen.
Risikomaß (VMXT)	VMXT.RiskImpact	Risikomaß = Risikowahrscheinlichkeit * Risikoschaden
Risikoklasse (VMXT)	VMXT.RiskClass	Qualitative Einstufung des Risikos anhand des Risikomasses.
Iterationspfad	System.IterationPath	Der Zeitabschnitt, dem dieses Work Item zugeordnet ist.
Autor (VMXT)	VMXT.Autor	Urheber dieses Work Items.
Bemerkungen (VMXT)	VMXT.Log	Feld für generelle Anmerkungen und Diskussionen.

Tabelle 6 Datenfelder für Risiken

Auch für das Projektrisiko Work Item sind verschiedene Standardwertebereiche definiert:

- Wahrscheinlichkeit (VMXT) (*VMXT.Likelihood*) hat den Wertebereich: {10, 30, 50, 70, 90} zur groben Einstufung der Eintrittswahrscheinlichkeiten.
- Die Felder Auswirkung auf Zeit (VMXT) (*VMXT.ImpactTime*), Auswirkung auf Qualität (VMXT) (*VMXT.ImpactQuality*), Auswirkung auf Budget (VMXT) (*VMXT.ImpactBudget*), Auswirkung auf Umfang (VMXT) (*VMXT.ImpactScope*) haben jeweils die Wertebereiche: {kritisch, hoch, mittel, gering, keine} zur genaueren Einstufung und Bewertung von Risiken.
- Strategietyp (VMXT) (*VMXT.StrategyType*) hat den Wertebereich: {vermeidung, transfer, akzeptanz}.
- Risikoklasse (VMXT) (*VMXT.RiskClass*) hat den Wertebereich: {katastrophal, kritisch, unerwünscht, tolerierbar}.

Folgende Modellierung liegt dem Workflow des Work Item Typs *Risiko* zugrunde:

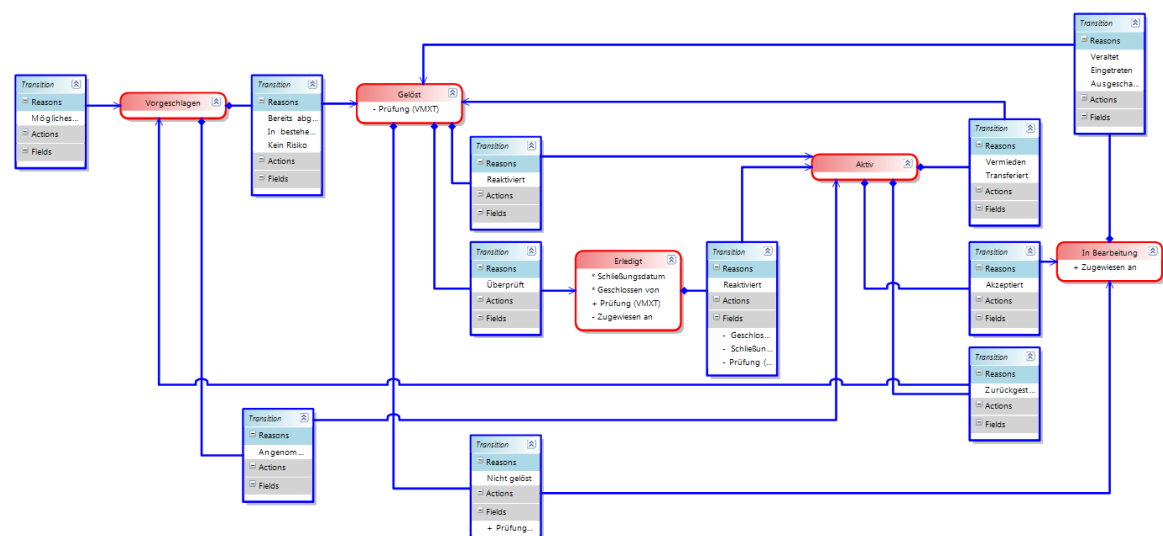


Abbildung 9 Workflow für Risiken

**Mögliche Verknüpfungen und Strukturierungen.** Es ist sinnvoll, Risiken mit anderen Work Item Typen zu verknüpfen. Dies ist nicht nur sinnvoll sondern sogar erforderlich, wenn Risiken schwer wiegend und intensiv zu bewerten sind. Dann ist es erforderlich, *Arbeitsaufträge* zu vergeben, um bspw. Risiken zu bewerten oder Maßnahmen zu deren Minderung einzuleiten. Risiken können weiterhin Bezug zu *Produkten*,



Aktivitäten und Entscheidungspunkten haben, sofern diese durch das Eintreten eines Risikos direkt betroffen wären. Im Rahmen der Abbildung ist hier insbesondere für Maßnahmen ein eigener Work Item Typ *Maßnahme* als beispielhafte Ausgestaltung vorgesehen. Folgende Datenstruktur legen wir dem Work Item Typ *Maßnahme* zugrunde:

Name	Ref.-Name	Beschreibung
Titel	System.Title	Bezeichnung des Work Items.
Beschreibung	System.Description	Beschreibung des Work Items.
Zustand	System.State	Workflow und Status des Work Items.
Grund	System.Reason	Der Grund, warum dieses Work Item in diesem Zustand ist.
Zugewiesen an	System.AssignedTo	Der Verantwortliche für dieses Work Item.
Verlauf	System.History	Verlauf und historische Informationen.
Schließungsdatum	Microsoft.VSTS.Common.ClosedDate	Datum, an dem dieses Work Item geschlossen wurde.
Geschlossen von	Microsoft.VSTS.Common.ClosedBy	Name desjenigen, der dieses Work Item geschlossen hat.
Bereichspfad	System.AreaPath	Der logische Bereich, dem dieses Work Item zugeordnet wurde.
Prüfung (VMXT)	VMXT.Assessment	Dokumentation der Prüfungsergebnisse.
Bemerkungen (VMXT)	VMXT.Log	Feld für generelle Anmerkungen und Diskussionen.
Iterationspfad	System.IterationPath	Der Zeitabschnitt, dem dieses Work Item zugeordnet ist.
Abschlussdatum	Microsoft.VSTS.Scheduling.FinishDate	Geplantes Datum, zu dem dieses Work Item abgeschlossen sein soll.
Trigger	VMXT.Trigger	Auslöser zur Durchführung der Maßnahme, wenn die Maßnahme nicht sofort durchgeführt wird.

Tabelle 7 Datenfelder für Maßnahmen

Folgende Modellierung liegt dem Workflow des Work Item Typs *Maßnahme* zugrunde:

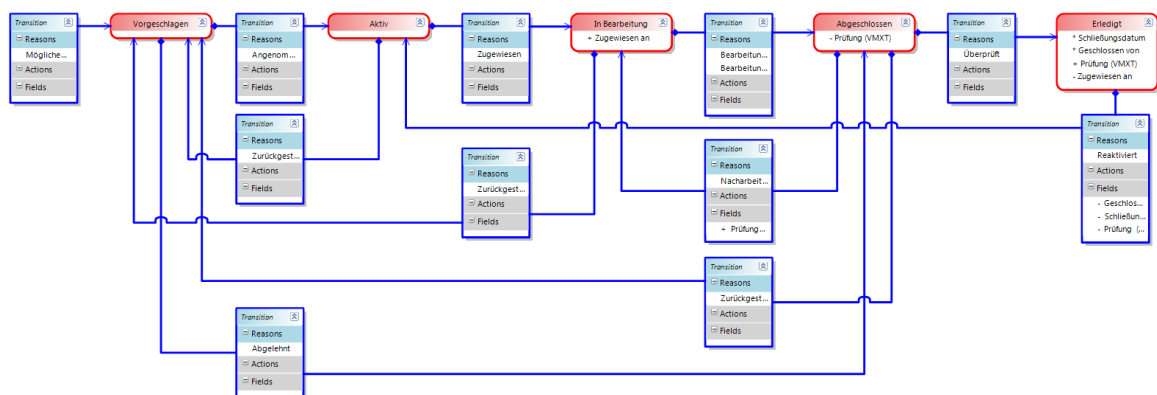


Abbildung 10 Workflow für Maßnahmen

## 2.4.6 Work Item: PÄM

Das Problem- und Änderungsmanagement ist ein komplexer Prozess den wir im Kontext dieser Interpretation so weit wie möglich vereinfacht haben. Auch hier sind selbstverständlich Anpassungen möglich. Ausgestaltungen sind insbesondere bei diesem Work Item Typ notwendig, da wir mit unserer Überführung des V-Modells nur den Managementrahmen, jedoch keinen Software Entwicklungsanteile berücksichtigen. Diese Abbildung leistet das *nicht*, um den generischen Rahmen des V-Modells zu erhalten.

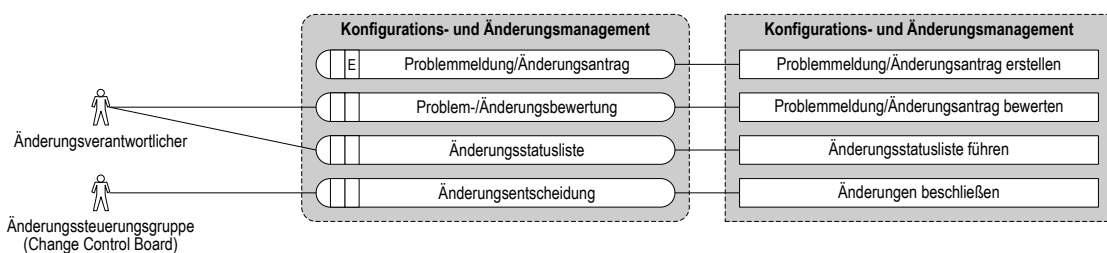


Abbildung 11 Vorgehensbaustein Problem- und Änderungsmanagement des V-Modell XT

Problem- und Änderungsmanagement ist im V-Modell als eine der Kerndisziplinen in einem eigenen Vorgehensbaustein untergebracht. Dieser Vorgehensbaustein *Problem- und Änderungsmanagement* definiert eine Reihe von Produkten und korrespondierenden Aktivitäten, die wir geeignet zu einem integrierten und kompakten Prozess zusammenfassen. Kernelement unseres Prozesses ist eine integrierte *Problem- und Änderungsmeldung* (PÄM), die wir als Work Item Typ modellieren. Wir orientieren uns hier jedoch nicht weiter an den Daten und Datenstrukturen die das V-Modell im Rahmen seiner Dokumentation bereitstellt, sondern bauen auf dem Änderungsmanagementsystem des V-Modell XT Entwicklungsprojekts WEIT auf. Das System setzt auf Mantis<sup>7</sup> auf, einem Open Source Bug Trackingsystem, das über die Webseiten der

<sup>7</sup> Quelle zu Mantis: <http://www.mantisbt.org>



KBSt (V-Modell XT Portal) öffentlich zugänglich ist und somit auch den Anwendern des V-Modells offen steht, um Fehler am V-Modell zu melden, Änderungsforderungen einzustellen oder Feature Requests zu starten.

In diesem Vorgehensbaustein sind aber gleichzeitig auch einige Aktivitäten definiert, die wir in die Modellierung des Workflows einfließen lassen müssen. Hier gehen wir einen gänzlich anderen Weg in der Modellierung und orientieren uns an einem existierenden Workflow aus dem MSF for CMMI. Der Hintergrund ist die angestrebte Einfachheit des PÄM-Prozesses. Der hinter dem Änderungssystem des V-Modells liegende Prozess ist dafür zu komplex und vielschichtig. Die in den problem- und änderungsbezogenen Aktivitäten beschriebenen Pfade sind zu allgemein und wenig integriert. Die Definitionen des MSF for CMMI bilden hingegen eine schlanke und elegante Grundlage, auf der wir aufbauen. Folgende Datenstruktur legen wir für die Modellierung des Work Item Typs *PÄM* zugrunde:

Name	Ref.-Name	Beschreibung
Titel	System.Title	Bezeichnung des Work Items.
Zugewiesen an	System.AssignedTo	Der Verantwortliche für dieses Work Item.
Verlauf	System.History	Verlauf und historische Informationen.
Bereichspfad	System.AreaPath	Der logische Bereich, dem dieses Work Item zugeordnet wurde.
Iterationspfad	System.IterationPath	Der Zeitabschnitt, dem dieses Work Item zugeordnet ist.
Zustand	System.State	Workflow und Status des Work Items.
Grund	System.Reason	Der Grund, warum das Work Item in diesem Zustand ist.
Beschreibung	System.Description	Beschreibung des Work Items.
Schließungsdatum	Microsoft.VSTS.Common.ClosedDate	Datum, an dem das Work Item geschlossen wurde.
Geschlossen von	Microsoft.VSTS.Common.ClosedBy	Name desjenigen, der das Work Item geschlossen hat.
Analyse (VMXT)	VMXT.Analysis	Analyse (inkl. Grund) und Auflistung von Optionen und potenziellen Risiken.
Auswirkungen auf Projektziel (VMXT)	VMXT.ImpactOnProjectPromise	Auswirkung auf das Projekt/Tragweite.
Tatsächliche Lösung (VMXT)	VMXT.CorrectiveActionActualResolution	Dokumentation der tatsächlichen vorgeschlagenen Lösung.
Gewünschter Fertigstellungszeitpunkt (VMXT)	VMXT.TargetDate	Datum, an dem die PÄM umgesetzt sein soll.
Kategorie (VMXT)	VMXT.Kategorie	Die Kategorie der PÄM
Reproduzierbar (VMXT)	VMXT.Reproduzierbar	Flag, das anzeigt, ob die PÄM reproduzierbar auftritt.
Autor (VMXT)	VMXT.Autor	Urheber dieses Work Items.
Dringlichkeit (VMXT)	VMXT.Urgency	Dringlichkeit des Work Items.
Gegenstand (VMXT)	VMXT.Item	Gegenstand auf den sich das Work Item bezieht.
Version (VMXT)	VMXT.ItemVersion	Version des Gegenstands.
Prüfung (VMXT)	VMXT.Assessment	Dokumentation der Prüfungsergebnisse.
Bemerkungen (VMXT)	VMXT.Log	Feld für generelle Anmerkungen und Diskussionen.
Empfehlung (VMXT)	VMXT.Recommendation	Entscheidungsempfehlung für dieses Work Item.
Entscheid (VMXT)	VMXT.Decision	Beschreibung der Entscheidung zu dieser PÄM
Verbleibende Arbeit	Microsoft.VSTS.Scheduling.RemainingWork	Schätzung noch zu erbringender Aufwände (in Stunden).
Startdatum	Microsoft.VSTS.Scheduling.StartDate	
Abschlussdatum	Microsoft.VSTS.Scheduling.FinishDate	Geplantes Datum, zu dem dieses Work Item abgeschlossen sein soll.

Tabelle 8 Datenfelder für PÄMs

Das PÄM Work Item legt für einige Felder Standardwertebereiche fest:

- Auswirkungen auf Projektziel (VMXT) (*VMXT.ImpactOnProjectPromise*) hat den Wertebereich: {kritisch, hoch, mittel, niedrig}.
- Kategorie (VMXT) (*VMXT.Kategorie*) hat den Wertebereich: {änderungsantrag, problemmeldung}.
- Reproduzierbar (VMXT) (*VMXT.Reproduzierbar*) hat den Wertebereich: {ja, nein, nicht relevant}.
- Dringlichkeit (VMXT) (*VMXT.Urgency*) hat den Wertebereich: {kritisch, sehr wichtig, wichtig, wünschenswert}.
- Empfehlung (VMXT) (*VMXT.Recommendation*) hat den Wertebereich: {ablehnen, annehmen, zurückstellen}.

Folgende Modellierung liegt dem Workflow des Work Item Typs *PÄM* zugrunde:

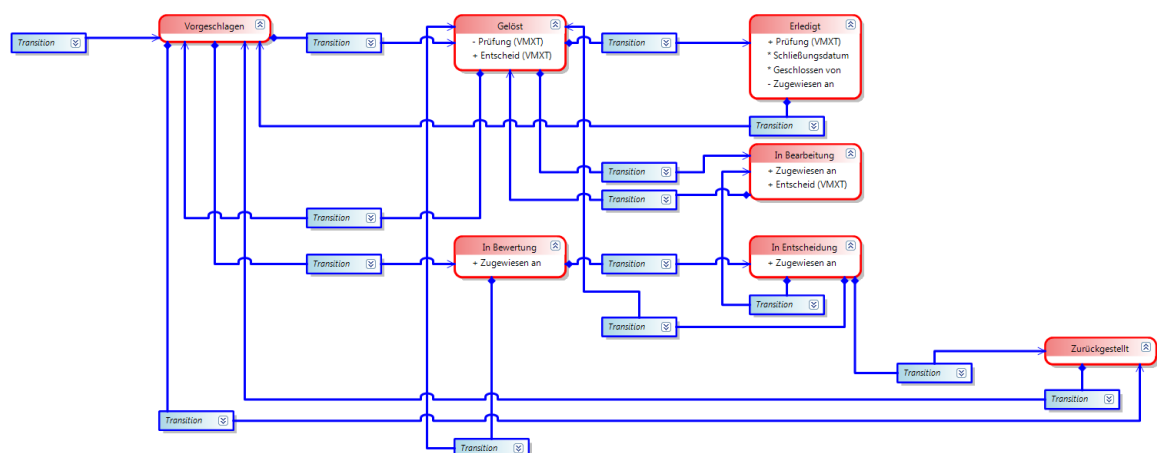


Abbildung 12 Workflow für PÄMs

**Mögliche Verknüpfungen und Strukturierungen.** Auch für Problemmeldungen ist es erforderlich, andere Work Items sinnvoll anzubinden und zu verknüpfen. Insbesondere betrifft das wieder Arbeitsaufträge, die ausgelöst werden, um PÄMs zu bewerten, einzuplanen oder umzusetzen. Weiterhin sind auch Risiken zu berücksichtigen, die durch auftretende PÄMs motiviert oder ausgelöst werden können. Dem entsprechend sind auch Produkte zu betrachten, die durch PÄMs direkt oder indirekt betroffen sind.

#### 2.4.7 Abfragen für die definierten Work Item Typen

Im Rahmen eines Process Template sind auch einfache Abfragemechanismen umsetzbar<sup>8</sup>. Diese können dazu verwendet werden, schnell einen Überblick über das Work Item Tracking System des TFS erhalten. In einem TFS-Projekt gibt es verschiedene Bereiche für die Abfragen, nämlich so genannte *Teamabfragen* (Team Queries) und *Eigene Abfragen* (My Queries). Bei der Generierung eines Process Templates aus einem projektspezifisch angepassten V-Modell können die Teamabfragen bereits für die neuen V-Modell-bezogenen Work Item Typen vorgefertigt werden. Die Abfragen beziehen sich somit auf:

- Produkte
- Aktivitäten
- Entscheidungspunkte
- Arbeitsaufträge
- Risiken
- PÄMs

Wir geben diese Abfragen im Folgenden an:

- Query: Alle
- Query: Alle Entscheidungspunkte
- Query: Alle Produkte
- Query: Alle Aktivitäten
- Query: Alle PÄM
- Query: Alle Arbeitsaufträge
- Query: Alle Risiken
- Query: Alle offenen Produkte
- Query: Alle offenen Aktivitäten
- Query: Alle offenen PÄM
- Query: Alle offenen Arbeitsaufträge
- Query: Alle offenen Risiken

### 3 Werkzeug

Für CollabXT wurde über die Konzeption von Prozess-/Toolintegrationsverfahren ebenfalls eine Werkzeugentwicklung durchgeführt. Wir konzentrieren uns hier auf die Entwicklungen für das Visual Studio – respektive den dahinter liegenden Team Foundation Server.

#### 3.1 Architektur

Die Anforderungen an das Werkzeug beschreiben ein generatorbasiertes System, das die Kette der Referenzwerkzeuge des V-Modells weiterführt und eine weitere Stufe so einführt, dass Ergebnisse des Tailorings direkt in andere Umgebungen zu überführen und zu integrieren. Für TFS ist dafür ein template-, generatorbasiertes System gewählt worden, das einerseits auf den Eingaben des V-Modells basiert, andererseits jedoch auch Daten mit einer zeitlich höheren Stabilität in einem Meta-Process Template ablegt.

In Abbildung 6 ist eine Skizze der Systemarchitektur des Generatorsystems zu sehen. Sie zeigt die Eingaben des Systems, die aus dem Meta-Process Template und den V-Modell Inhalten aus dem Tailoring bestehen. Der Generator mischt also das Meta-Process Template mit den V-Modell Inhalten zu einem validen TFS-kompatiblen Process Template für ein V-Modell Projekt.

**Achtung!** An dieser Stelle ist es wichtig, noch einmal sprachlich präzise zu werden und das Ergebnis des Generators genauer zu charakterisieren. Genau genommen, erzeugt der Generator ein projektspezifisches Template, sondern ein *projekttypspezifisches* Template. Damit handelt es sich noch nicht um ein Projektmodell, sondern genau genommen immer noch um ein organisationsspezifisches Vorgehensmodell. Das V-

---

<sup>8</sup> Damit sich nicht die Reports gemeint, die auf den SQL-Reporting Services aufsetzen.

Modell beschreibt mit seinem Variantenraum lediglich eine diskrete Einschränkung des Prozesses sowohl in Umfang als auch Struktur. Durch den CollabXT-Generator werden die Inhalte weitgehend übernommen und spezifisch interpretiert. Das ändert jedoch nichts daran, dass auch weiterhin ein Rest Generik erhalten bleibt, da es sich hier im Wesentlichen um eine Modelltransformation auf einer Abstraktionsebene handelt. Durch die Aufbewahrung im Process Template Manager finden wir hier ein Konzept, das sich nicht sehr von einem V-Modell *Projekttyp* unterscheidet. *Präzise gesprochen wird also aus einem TFS Process Template ein V-Modell-basiertes Projekt vom Typ des Templates instanziiert.*

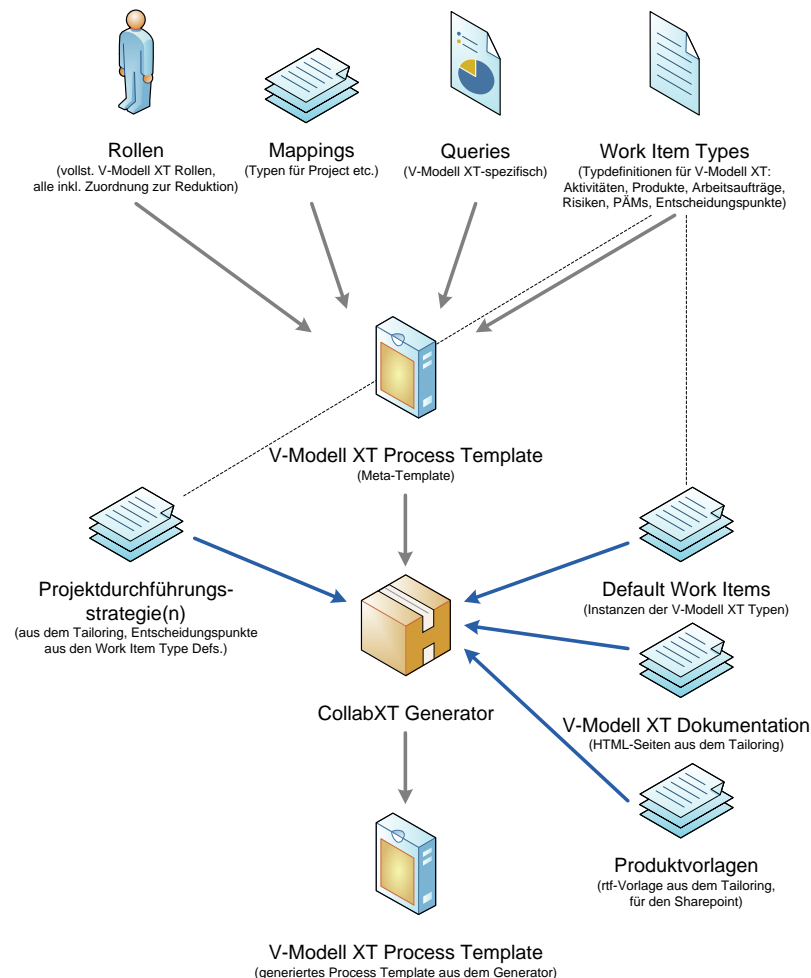


Abbildung 13 Systemarchitekturskizze des Generatorsystems für den TFS

### 3.2 Abgedeckte Anwendungsfälle und Anwenderunterstützung

Der CollabXT-Generator ist ein Werkzeug, das von Prozessingenieuren und Projektleitern eingesetzt werden kann. Erstere können mithilfe dieses Werkzeugs ein V-Modell-basiertes Vorgehen in eine TFS-Struktur überführen und dort ggf. Anpassungen und Ausgestaltungen vornehmen. Letztere können mithilfe des Werkzeugs ein projektspezifisch angepasstes V-Modell unmittelbar in den TFS einspielen und sofort instanziierten. Dabei werden die Anwender jeweils durch ein grafisches Werkzeug Abbildung 14 unterstützt, das in seiner ersten Stufe die Prozessstruktur anpasst und weiterhin in einer zweiten Stufe die Verknüpfung<sup>9</sup> der Elemente des im TFS instanziierten Projekts vornimmt.

<sup>9</sup> Dabei handelt es sich um ein Zusatzwerkzeug, das Zühlke als sinnvolles Add-On gefertigt hat.

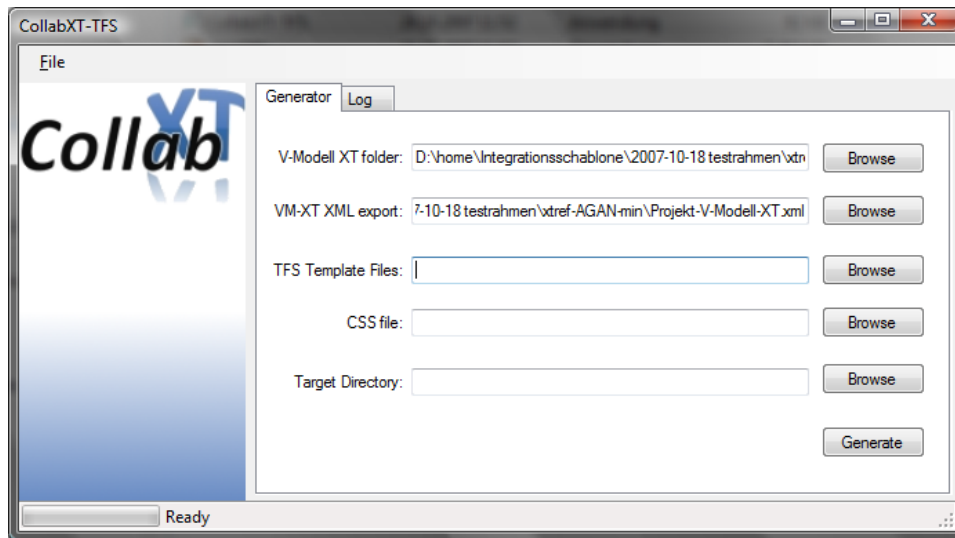


Abbildung 14 Benutzerschnittstelle des CollabXT-Generators

Aufgrund der Ausrichtung auf das Visual Studio als Zielplattform gibt es jedoch Beschränkungen hinsichtlich der unterstützten Einsatzszenarios. Der CollabXT-Generator für TFS unterstützt nur Projekte mit Entwicklungsanteilen. Das sind V-Modell Projekte, die auf den Projekttypen *Auftragnehmer* (AN) oder *Auftraggeber/Auftragnehmer* (AG/AN) basieren. Reine Auftraggeber (AG) Projekte oder Projekte zur Einführung und Anpassung eines Vorgehensmodells (Org) werden **explizit nicht** unterstützt. Der CollabXT-Generator unterstützt somit:

- Systementwicklungsprojekte AN
- Systementwicklungsprojekte AG/AN
- Angepasste V-Modell XT Derivate, die AN bzw. AG/AN erweitern.

Der Generator arbeitet für inhaltliche Anpassungen des V-Modells ebenso und unterstützt somit die Variationen des Tailorings transparent. Im Anhang – Referenzprofile befinden sich die Referenzprofile für das Tailoring des Standard V-Modell XT, die durch den Generator unterstützt werden mitsamt der Ergebnisstrukturen für den Vergleich und die Abstimmung/Feststellung der Konformität des Generators zum originalen V-Modell.

### 3.3 Installation und Randbedingungen

Für die Installation der Software sind im Wesentlichen die Anforderungen der beteiligten Komponenten maßgebend. Prinzipiell ist das Gesamtsystem so entworfen, dass keine tiefen Eingriffe in bestehende Infrastrukturen notwendig sind. Folgende Software wird benötigt:

- Die V-Modell XT Werkzeuge (erhältlich via [www.v-modell-xt.de](http://www.v-modell-xt.de)) erfordern eine Java Laufzeitumgebung (1.5 oder besser).
- Für die CollabXT Werkzeuge ist (erhältlich via <http://www.codeplex.com/VModellXTTFS>) eine .NET Laufzeitumgebung (2.0 oder besser) erforderlich.
- Ein Team Foundation Server ist installiert und im Netzwerk verfügbar.

Für den Team Foundation Server gelten folgende Anforderungen:

- Sprache des Systems: Deutsch
- Installationsstatus: Neu

Die aktuell vorliegende Version der CollabXT Werkzeuge adressiert ausschließlich Szenarios in denen ein deutschsprachiges V-Modell XT-Derivat in Verbindung mit deutschsprachigen TFS-Installationen. Die Verwendung anderer Sprachversionen ist möglich, erfordert ggf. aber Anpassungen hinsichtlich Im- und Exportskripten z.B. für die Installation des Sharepoint-Templates.

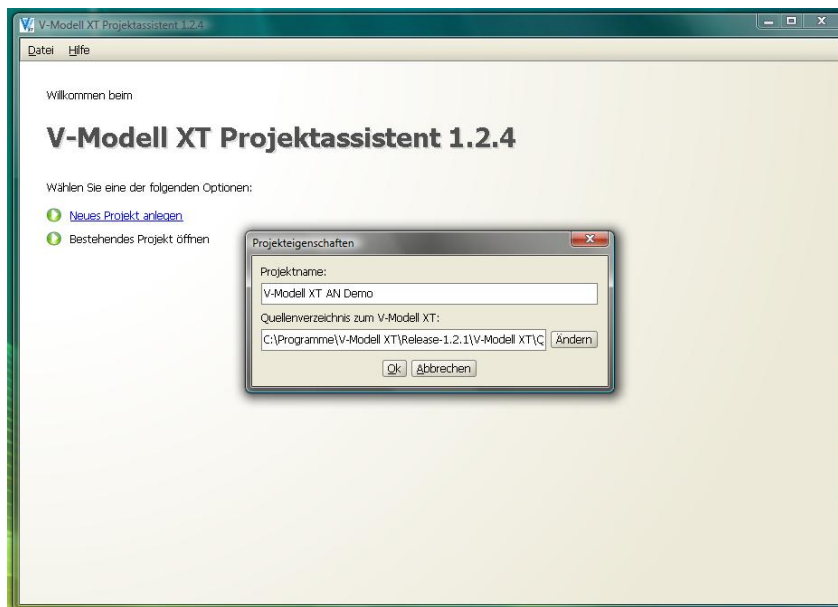
**Achtung!** Um Schwierigkeiten mit bereits etablierten Work Item Typen aus dem Weg zu gehen, empfehlen wir einen „frischen“ Server zu installieren.

### 3.4 Anwendung des Generators

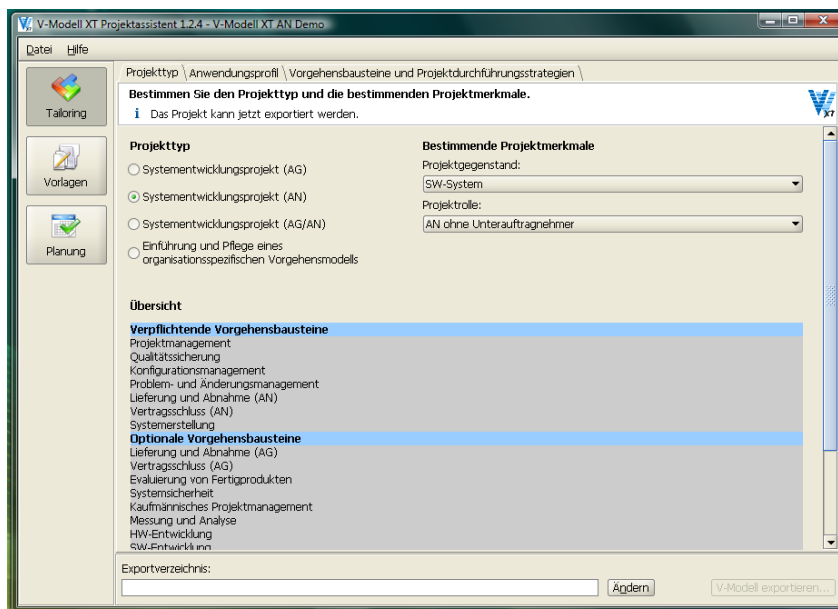
Die Anwendung des CollabXT-Generators erweitert die Anwendung der V-Modell XT Referenzwerkzeuge. Wir stellen in diesem Abschnitt die Anwendung der Referenzwerkzeuge und des Generators an einem kleinen Beispiel vor.

#### 3.4.1 Tailoring des V-Modell XT für CollabXT

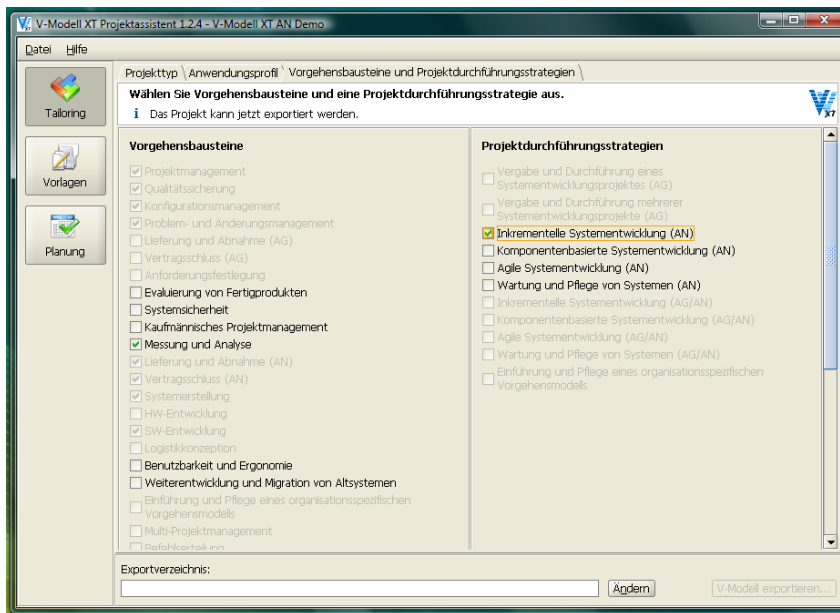
Der erste Schritt für die Erstellung eines TFS-Projekts ist das Erstellen eines projektspezifischen V-Modells mithilfe des V-Modell XT Projektassistenten. Wir gehen kurz auf alle notwendigen Schritte ein.



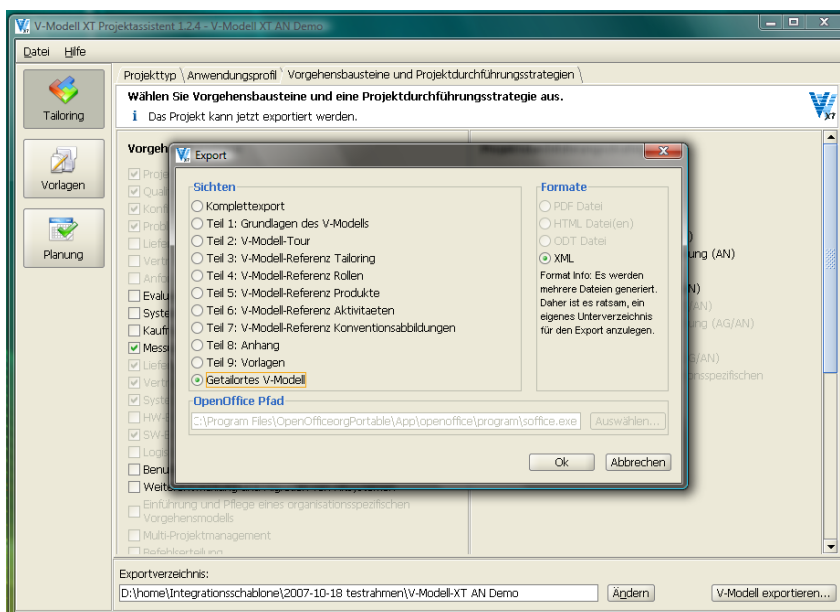
Zuerst wird ein neues Projekt angelegt. Zu beachten ist hierbei, dass CollabXT nur V-Modell Derivate geprüft ist, die auf der Metamodellversion 1.2.1 basieren.



Als nächstes zeigt der Projektassistent die Eingabe und Auswahlmasken für das Tailoring an. In diesem Beispiel erstellen wir ein reines Auftragnehmerprojekt ohne die Einbindung von Unterauftragnehmern. Weiterhin wählen wir als Projektgegenstand den Wert: „SW-System“ aus. Im unteren Teil zeigt der Projektassistent die verpflichtenden und optionalen Vorgehensbausteine sowie die möglichen Projektdurchführungsstrategien an.



Für dieses Projekt wählen wir nur die Projektdurchführungsstrategie „Inkrementelle Systementwicklung (AN)“ aus. Sie gestattet eine einfache Projektorganisation in mehreren Iterationen. Weiterhin können wir im Rahmen des Tailorings noch optionale Vorgehensbausteine einbinden. Wir tun dies exemplarisch für den Vorgehensbaustein „Messung und Analyse“, der die Möglichkeit zur Erfassung von Projektkennzahlen anbietet.



Mit diesen Einstellungen kann das projektspezifische V-Modell jetzt exportiert werden. Der Projektassistent bietet dazu mehrere Exportformate an. Zunächst benötigen wir das angepasste (getailortes) V-Modell im XML-Format. Dieses wird durch den CollabXT-Generator weiter verarbeitet.

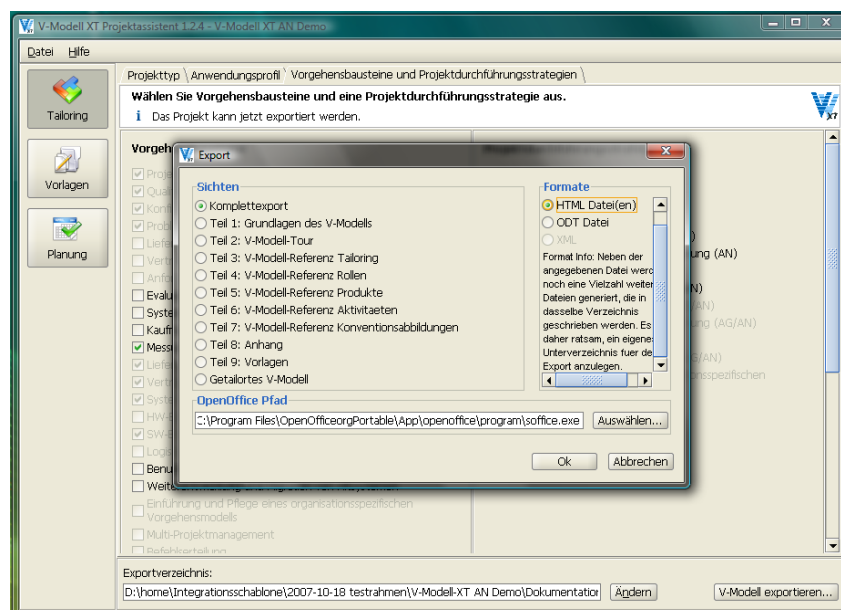
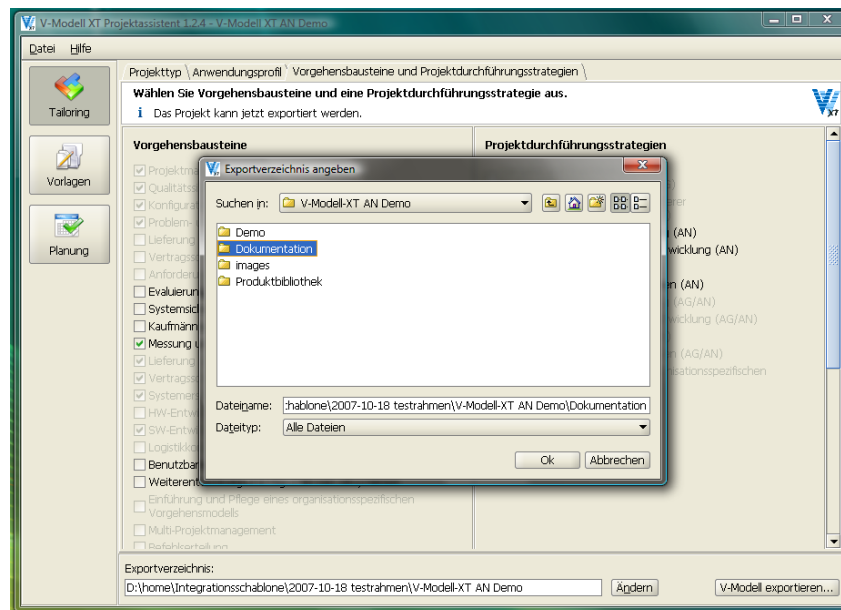
Um das projektspezifische V-Modell gesammelt verarbeiten zu können, haben wir eine Verzeichnisexportstruktur definiert, die der Generator voraussetzt:

```
Projektverzeichnis\
  Dokumentation\
  Produktbibliothek\
  Projekt-V-Modell-XT.xml
```

Unter dem Verzeichnis *Projektverzeichnis* erwartet der Generator die Unterverzeichnisse *Dokumentation* und *Produktbibliothek*. In Ersterem wird der HTML-Export der Prozessdokumentation des projektspezifischen V-Modells abgelegt. Im anderen Verzeichnis wird die Produktvorlagenbibliothek des V-Modells exportiert. Die Inhalte dieser beiden Verzeichnisse übernimmt der Generator für die Erstellung der Inhalte

für den TFS. Weiterhin *empfehlen* wir unter *Projektverzeichnis* auch den XML-Export des projektspezifischen V-Modells anzulegen. Neben der XML-Datei erzeugt der Projektassistent daraufhin auch alle notwendigen Schemainformationen und kopiert alle benötigten Bilder in ein entsprechendes Unterverzeichnis.

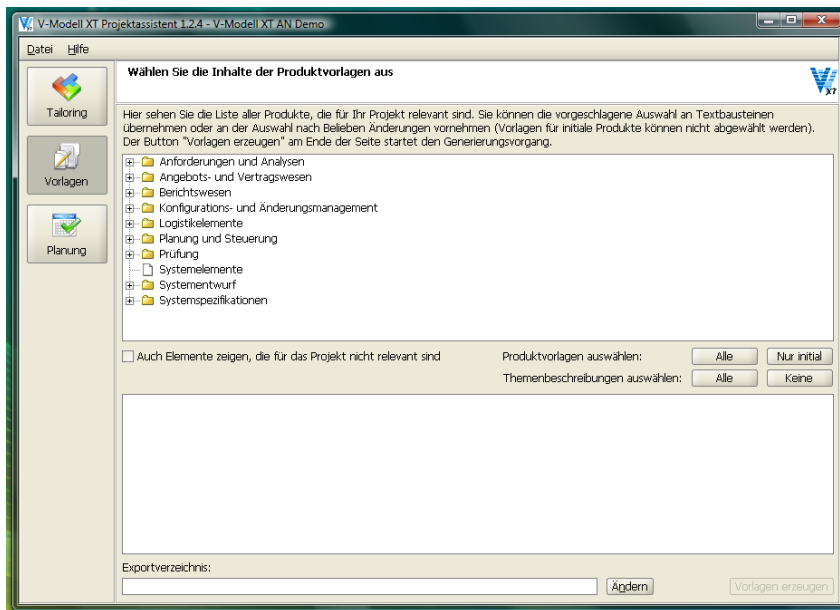
Die Verzeichnisstruktur muss durch den Anwender von Hand angelegt werden. Die Auswertung erfolgt automatisch.



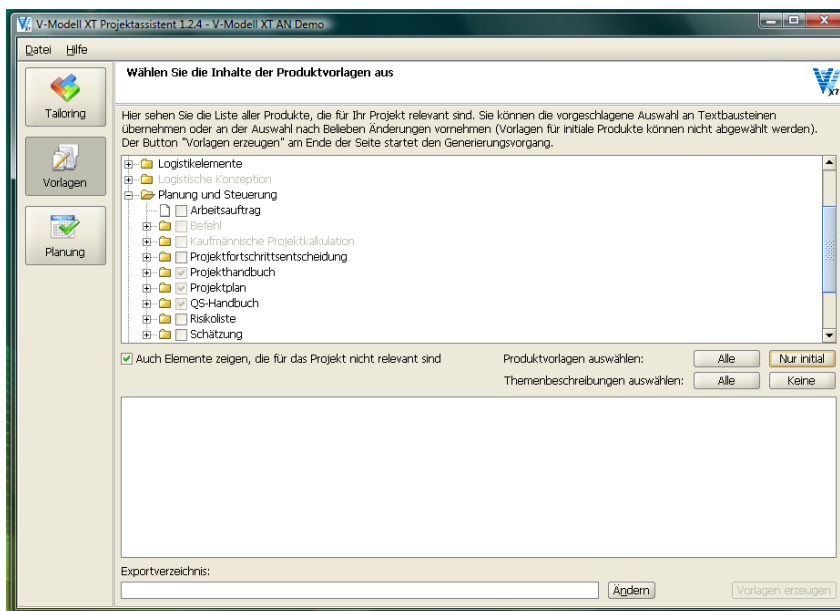
In das Unterverzeichnis *Dokumentation* wird nun die Prozessdokumentation im HTML-Format exportiert. Diese wird durch den Generator in die TFS-Process Guidance überführt.

Wir gehen konsequent weiter und erstellen nun die Produktvorlagen für das projektspezifische V-Modell. Diese sind aufgrund der Tailoringeinstellungen auf die Inhalte des Projekts angepasst. Dies äußert sich z.B. durch projektspezifisch gestaltete Kapitelstrukturen in den Dokumenten.





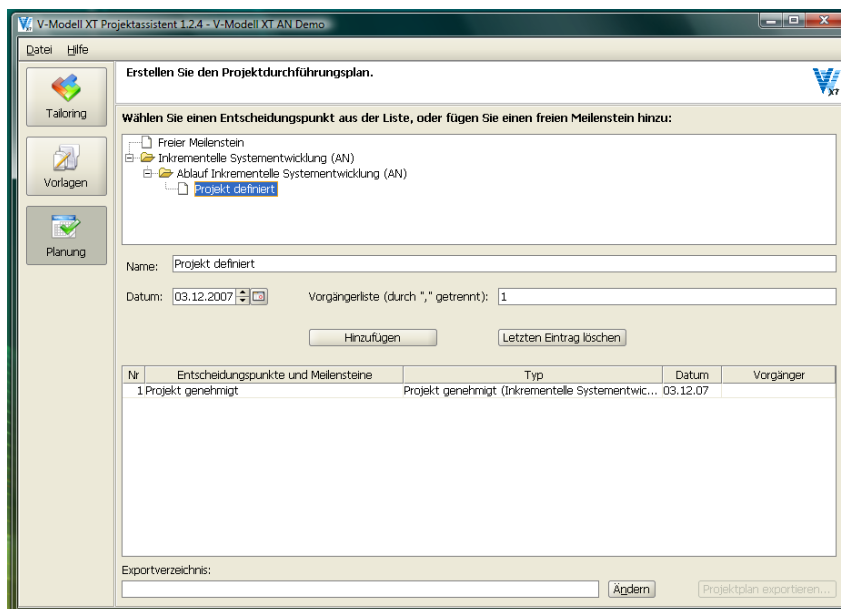
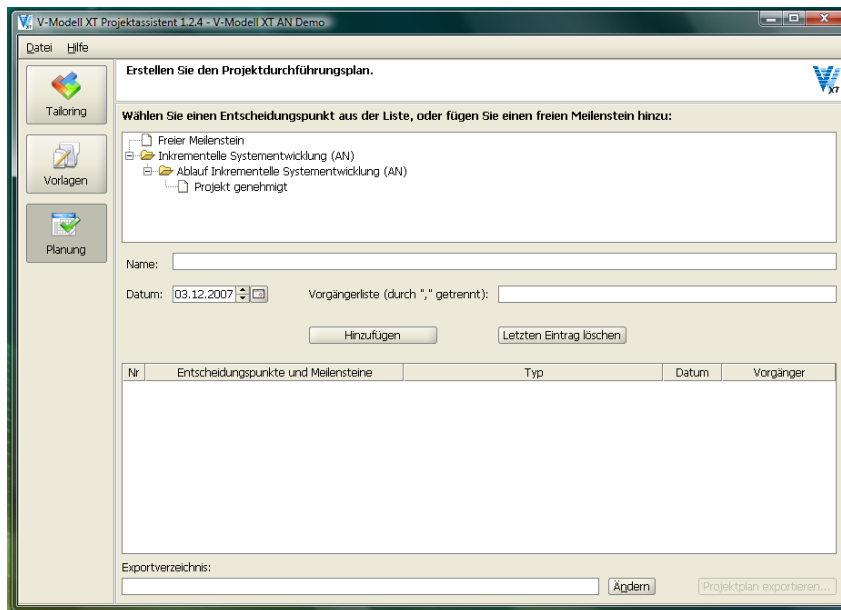
Für dieses Projekt benötigen wir nur einen kleinen Teil der der durch das V-Modell bereitgestellten Umfänge. Diese können aber noch weiter verfeinert und ggf. eingeschränkt werden.



Wir zeigen dies einmal, indem wir die Auswahlbox „Auch Elemente zeigen, die für das Projekt nicht relevant sind“ auswählen. Dies vermittelt einen guten Eindruck von der Weite der Anpassung. Ferner kann der Umfang der Produktvorlagenerzeugung durch die Schalter *Produktvorlagen* und *Themenbeschreibungen* weiter justiert werden. Wir wählen für unser Beispiel die Option *Alle Themenbeschreibungen* und *Nur initiale Produktvorlagen* erzeugen.

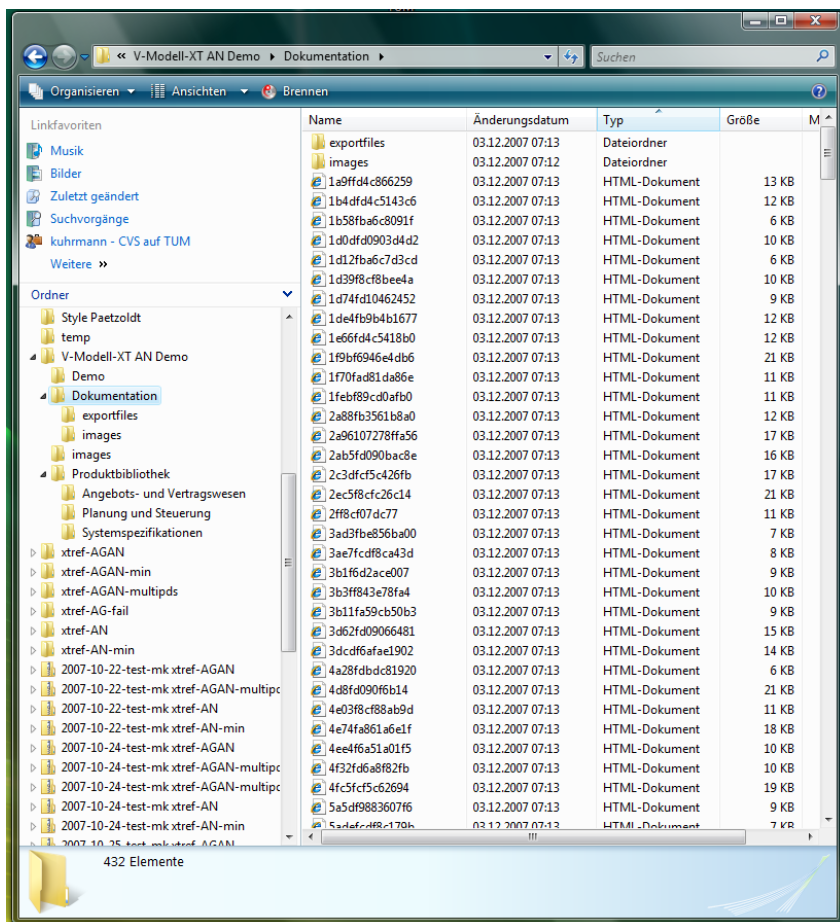
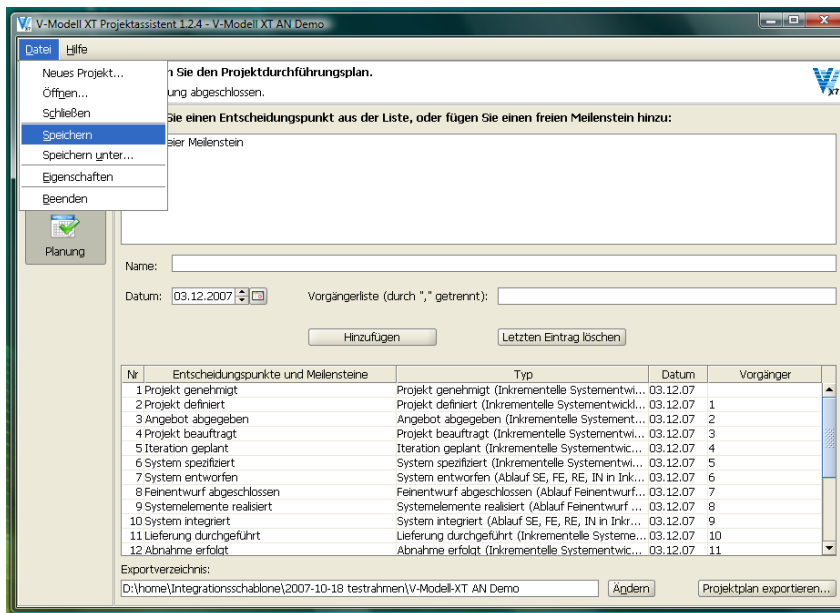
Abschließend führen wir auf der Grundlage der ausgewählten Projektdurchführungsstrategien noch die initiale Projektplanung durch.





Uns steht hier nur die „Inkrementelle Systementwicklung (AN)“ zu Verfügung, die uns aber zur einfachen Demonstration genügt. Die initiale Planung eines Projekts mit dem Projektassistenten ist *nicht* mit einer vollwertigen Projektplanung zu verwechseln. Hier wird lediglich das initiale Set-Up vorgenommen und die grobe Projektstruktur festgelegt (z.B. Entwicklung in zwei Iterationen mit einer initialen Beauftragung). Dies wird festgelegt, indem die zur Verfügung stehenden Projektdurchführungsstrategien durch Einplanung der möglichen Entscheidungspunkte zeitlich geordnet werden. Dazu können die Entscheidungspunkt ausgewählt und mit Planungsdaten versehen werden. Der Projektassistent überwacht dabei die Einhaltung der Vorgaben des V-Modells.

Nachdem sämtliche Planungen abgeschlossen sind, muss die Projektdatei des Projektassistenten gespeichert werden. Diese und das projektspezifische V-Modell im XML-Format sind die Komponenten, die durch den Generator zur Ermittlung des Projektplans benötigt werden. Ein expliziter Export des Projektplans durch den Projektassistenten ist hingegen nicht nötig.



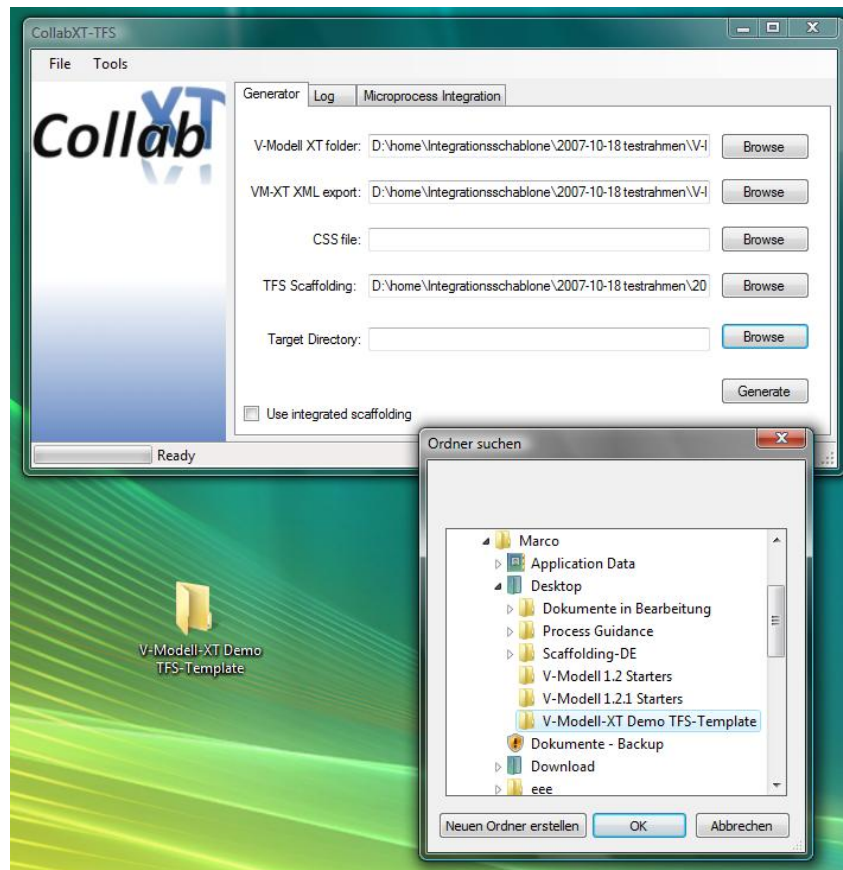
Damit stehen alle wesentlichen Eingaben für den Generator fest. Diese sind noch einmal kurz zusammengefasst:

- Das projektspezifische V-Modell im XML-Format
- Die Prozessdokumentation im HTML-Format im Unterverzeichnis *Projekt\Dokumentation*
- Die gewünschten Produktvorlagen im Unterverzeichnis *Projekt\Produktbibliothek*
- Die V-Modell XT Projektassistent Projektdatei mit allen Planungsinformationen

### 3.4.2 Generierung des Process Templates

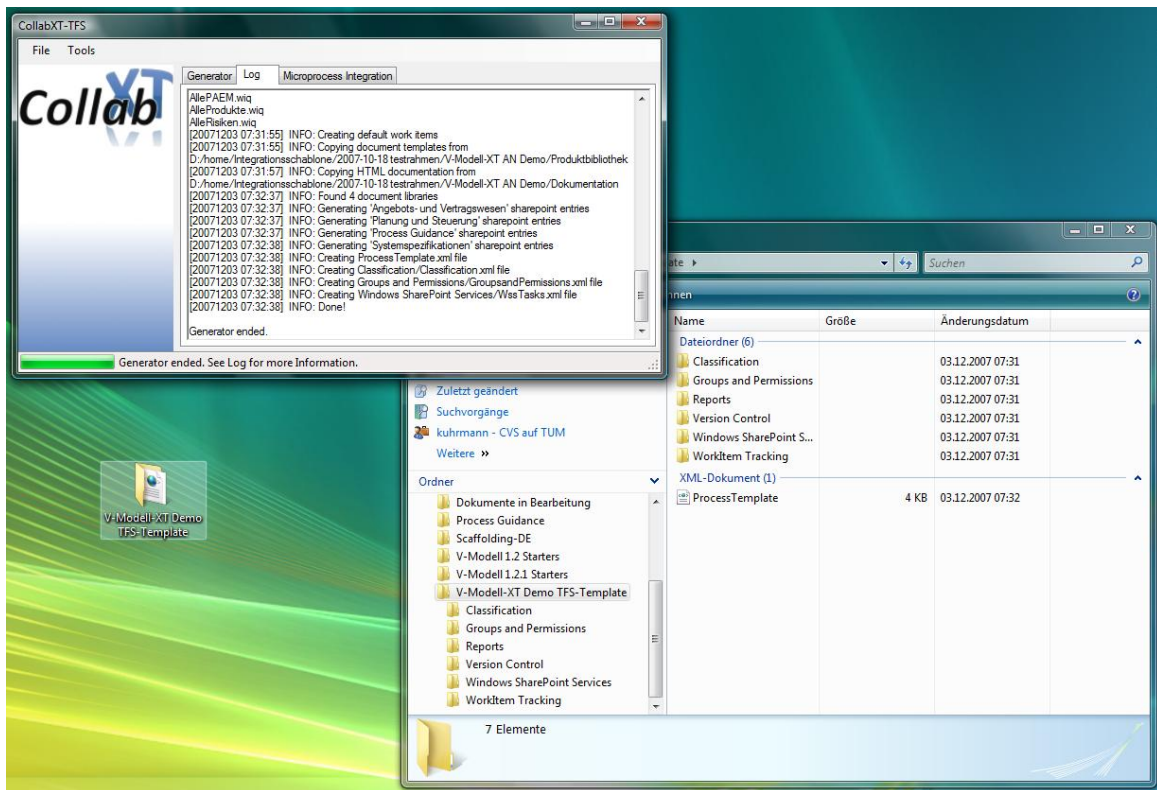
Der Generator verfügt über eine einfache Oberfläche. Im Wesentlichen dient sie dazu, die Pfade zu den Ausgaben des Projektassistenten zu erfassen. Die einzugebenden Felder sind:

- *V-Modell XT folder*: Das ist der Ordner in der oben beschriebenen Struktur.
- *VM-XT XML export*: Das ist das projektspezifische V-Modell im XML-Format
- *CSS file*: Hier besteht die Möglichkeit, das Ausgabeformat für die Prozessdokumentation zu variieren und in Form eines CSS-files zu hinterlegen. Dieses wird dann im TFS für dieses Template mit hinterlegt.
- *TFS Scaffolding*: Hiermit ist das TFS-Metatemplate gemeint, das z.B. die V-Modell XT Work Item Typen enthält.
- *Target Directory*: Das ist das Ausgabeverzeichnis für das resultierende V-Modell/TFS Template.



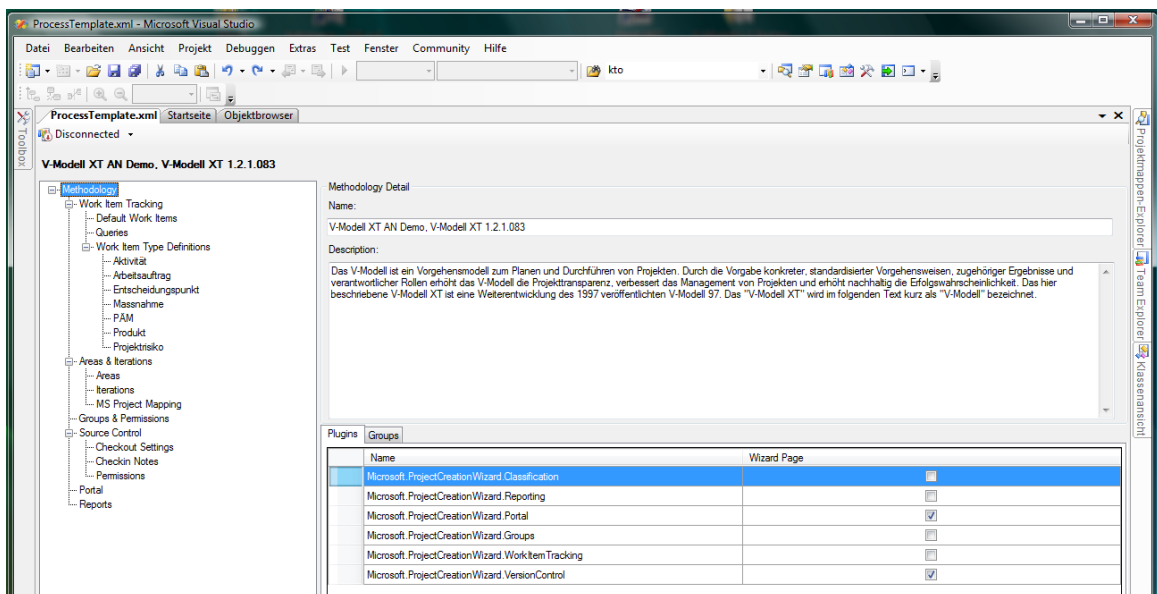
Nachdem alle Informationen und Eingabedaten vorliegen, kann der Generator gestartet werden. Der Vorgang kann ein wenig Zeit in Anspruch nehmen – je nachdem, wie komplex das zu verarbeitende Tailoring ist. Als Ergebnis wird durch den Generator ein valides TFS-Template auf der Basis des projektspezifischen V-Modells erzeugt.

Dieses Template kann im *Process Template Editor* des Visual Studio (als Teil der TFS Power Toys) betrachtet werden. Für einen ausführlicheren Rundgang verweisen wir auf die Demo im Web. Wir betrachten nur kurz die grundlegende Struktur.



Durch den Generator wird das Work Item Tracking System angesteuert. Hierzu werden auf der Basis der durch das Metatemplate bereitgestellten Work Item Types und der verfügbaren Planungsinformationen aus dem V-Modell initiale (Default) Work Items erstellt. Dies umfasst im Wesentlichen:

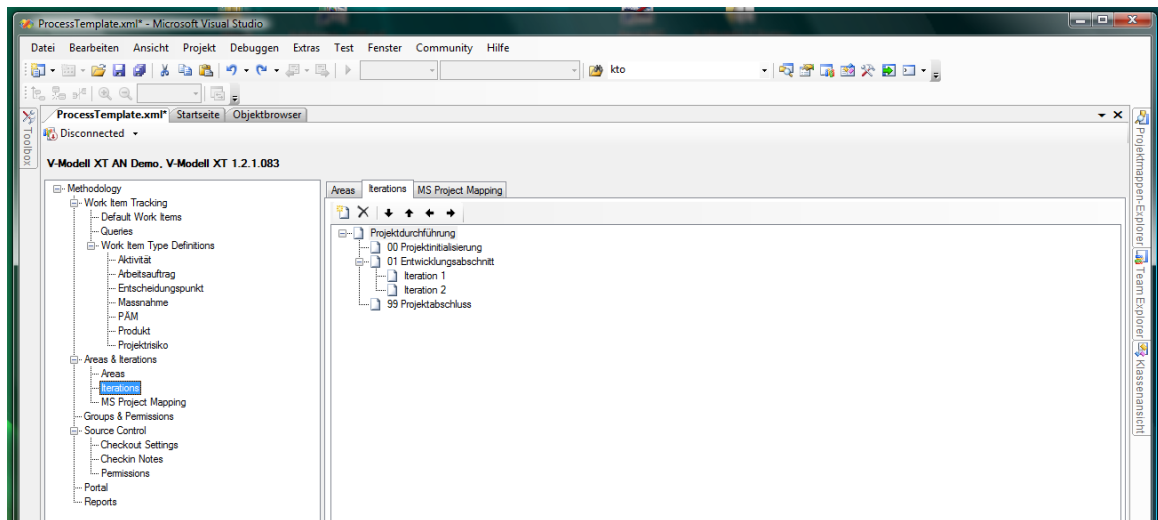
- Entscheidungspunkte
- Aktivitäten
- Produkte (teilw.)



Auf der Basis des initialen Projektplans wird die Projektstruktur im TFS auch entsprechend konfiguriert. Im weiter oben skizzierten Vorgang der Planung mit dem Projektassistenten haben wir eine einfache inkrementelle Vorgehensweise mit zwei Iterationen gewählt. Im *Iterations*-Abschnitt des Templates finden wir daher die Projektstruktur:

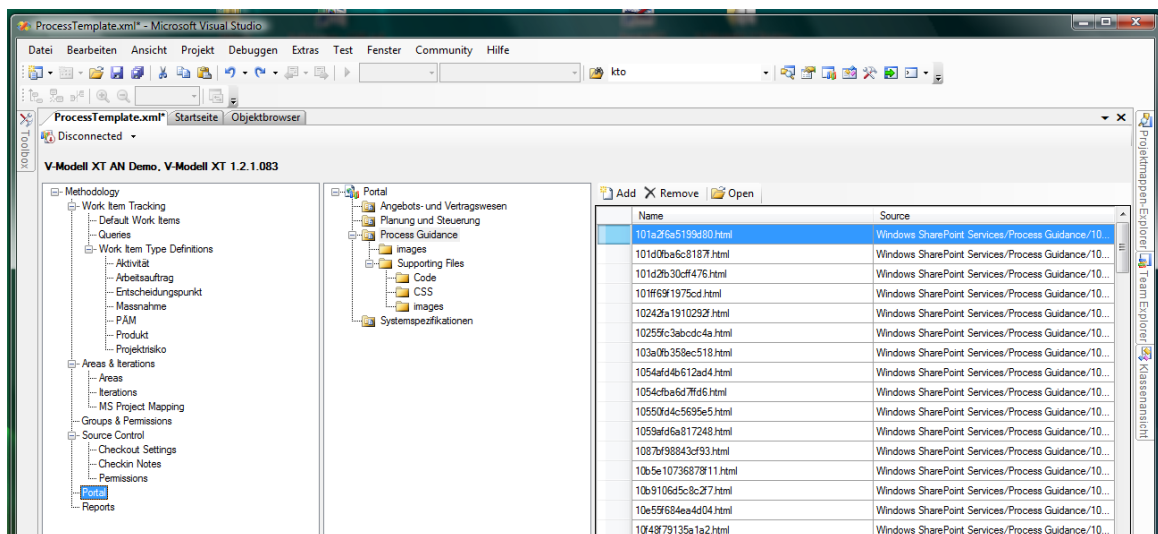
- Projektinitialisierung
- Entwicklungsabschnitt
  - o Iteration 1

- Iteration 2
- Projektabschluss



Im Abschnitt *Groups & Permissions* finden wir die grundlegenden Sicherheitskonfigurationen. Hier werden die durch das V-Modell XT definierten Rollen zunächst flach hinterlegt und beim Import auf dem TFS bekannt gemacht. Der Administrator des Systems/des Projekts kann diese Rollen dann nach Bedarf vergeben.

Unter dem Abschnitt *Portal* befinden sich die vom V-Modell Export übernommenen Daten (Prozessdokumentation und Produktbibliothek). Diese werden bei der Anlage des Projekts in das Sharepoint-Portal übernommen.

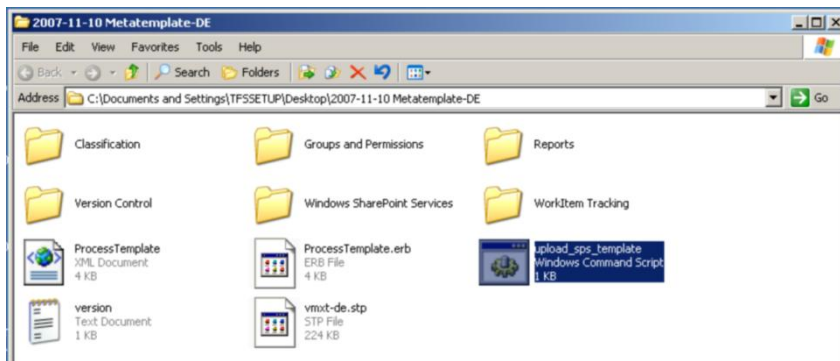


In den verbleibenden Sektionen (Source Control etc.), die wir hier nicht erläutert haben, finden sich dann wieder die üblichen Inhalte des TFS (Reports, Queries etc.).

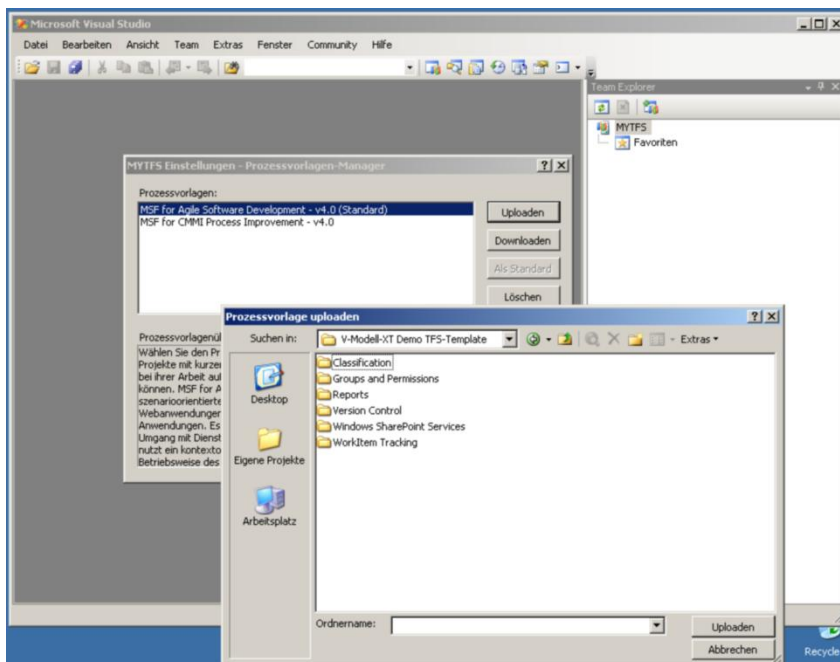
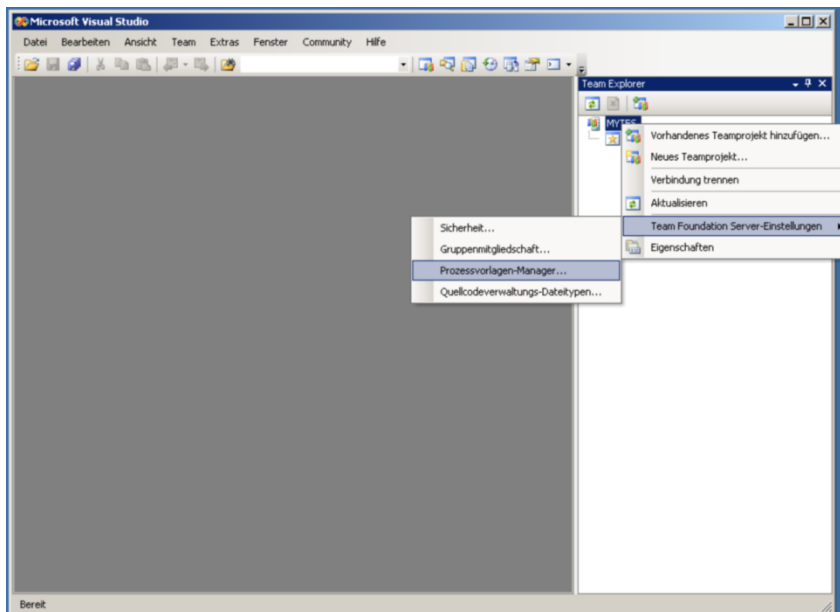
### 3.4.3 Installation im Team Foundation Server

Nach dem das Template fertig erzeugt ist, können wir es auf den TFS überspielen. Zunächst jedoch muss noch ein eigens angepasstes Sharepoint-Template auf dem Server bereitgestellt werden. Dieses Template liegt dem Download-Paket bei. Ein einfaches Skript installiert es:



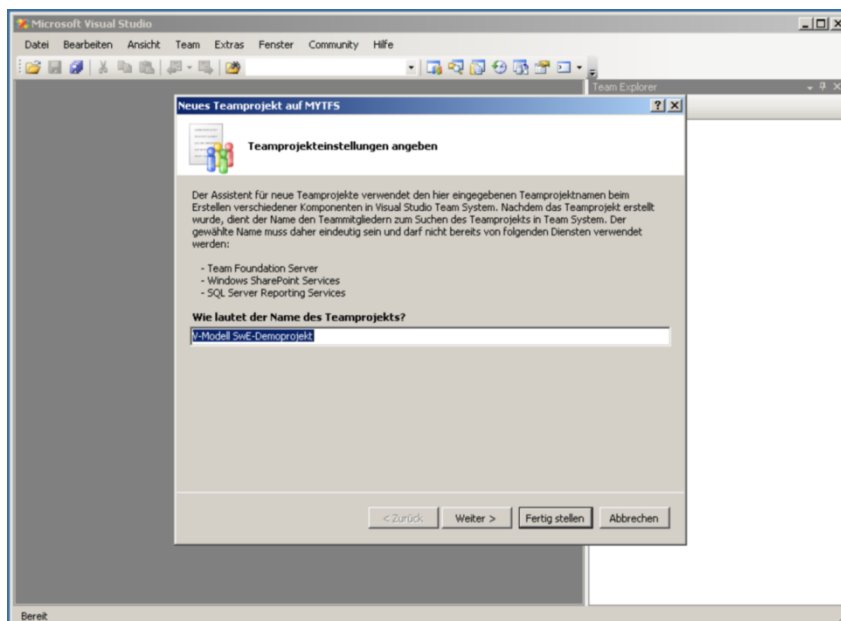
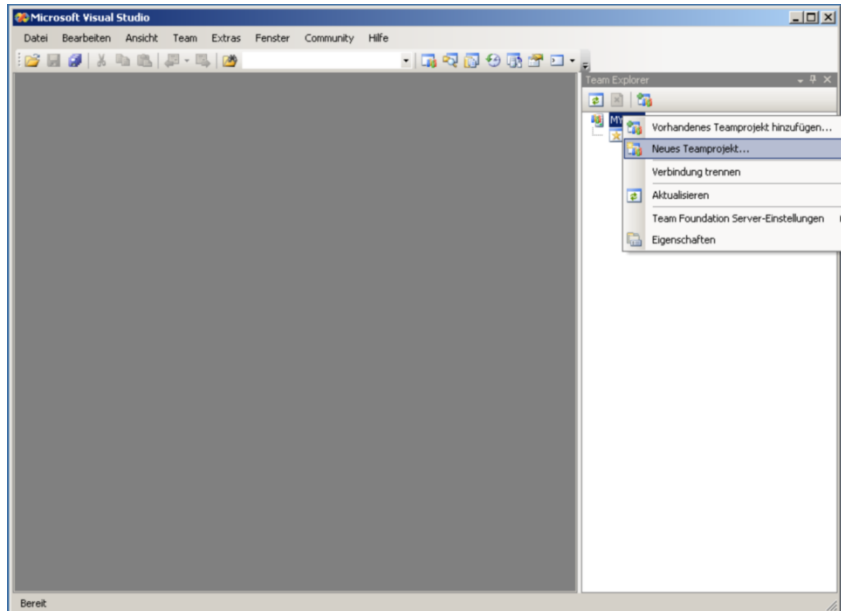


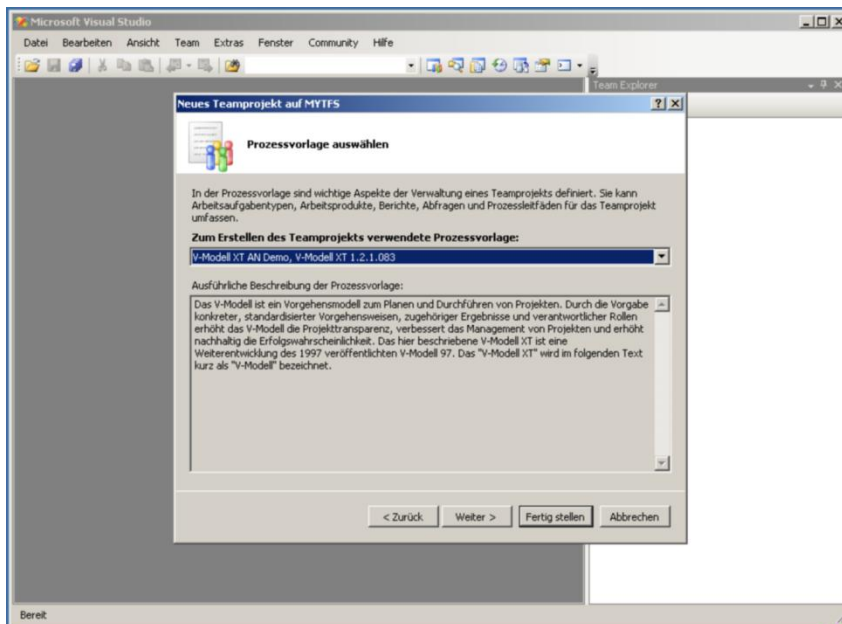
Im Anschluss daran kann das V-Modell/TFS-Template in den TFS importiert werden. Dazu dient der *Prozessvorlagen-Manager*. Er verwaltet alle auf einem TFS verfügbaren Prozessvorlagen, kann existierende zur weiteren Bearbeitung exportieren und neue zur Bereitstellung für neue Projekte importieren.



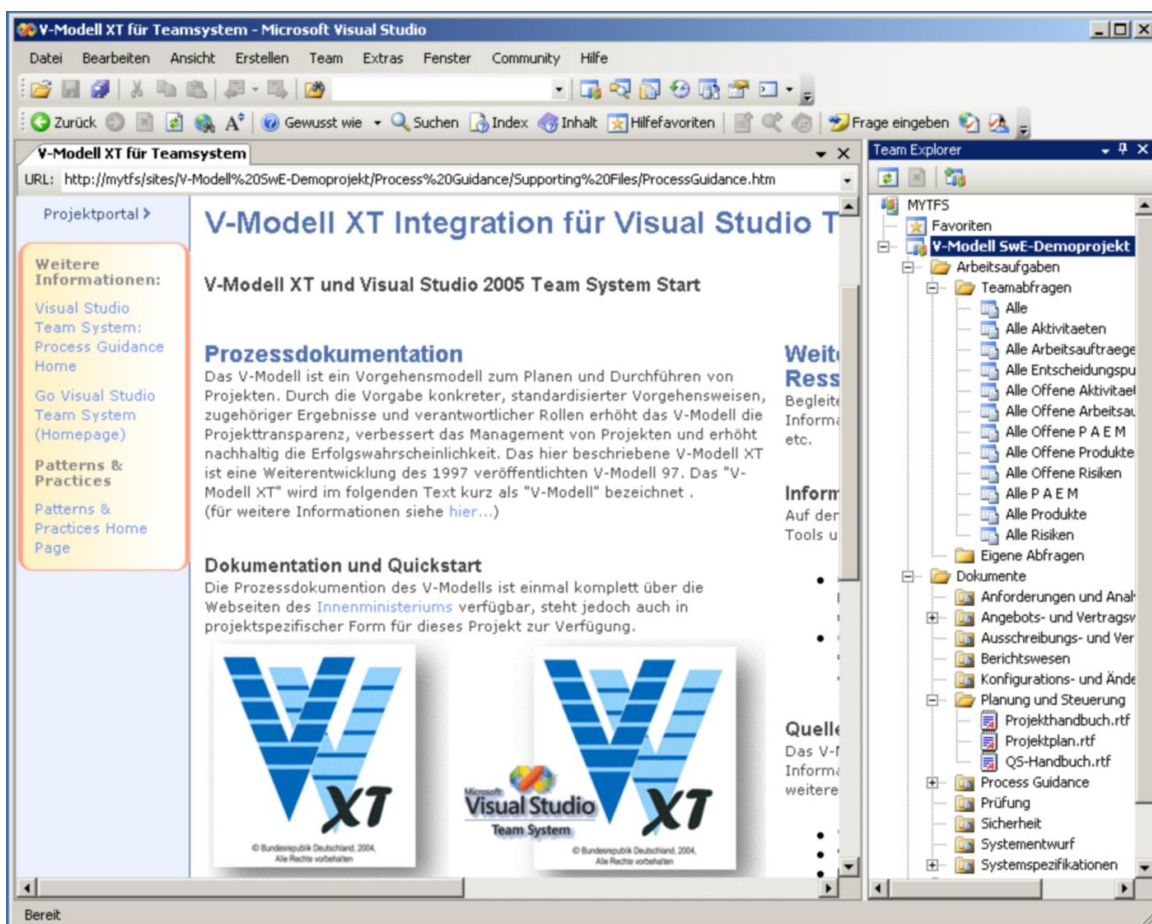
Dem Prozessvorlagen-Manager muss das Verzeichnis übergeben werden, in dem das V-Modell/TFS Template liegt (Ausgabeverzeichnis. Danach verläuft der Import automatisch. Im Anschluss steht das V-Modell/TFS Template als neue Prozessvorlage für das Anlegen neuer Projekte bereit.

Über den Menüpunkt *Neues Teamprojekt...* wird ein neues Projekt im TFS angelegt. Hier hat man nun die Möglichkeit, zusätzlich zu ggf. vorher schon installierten Vorlagen, ein Projekt auf der Basis des gerade eingespielten V-Modell/TFS Templates anzulegen.





Das Anlegen erfolgt wiederum mit den standardmäßigen Schritten des TFS-Assistenten. Als Resultat steht ein neues V-Modell-basiertes Projekt zur Verfügung.

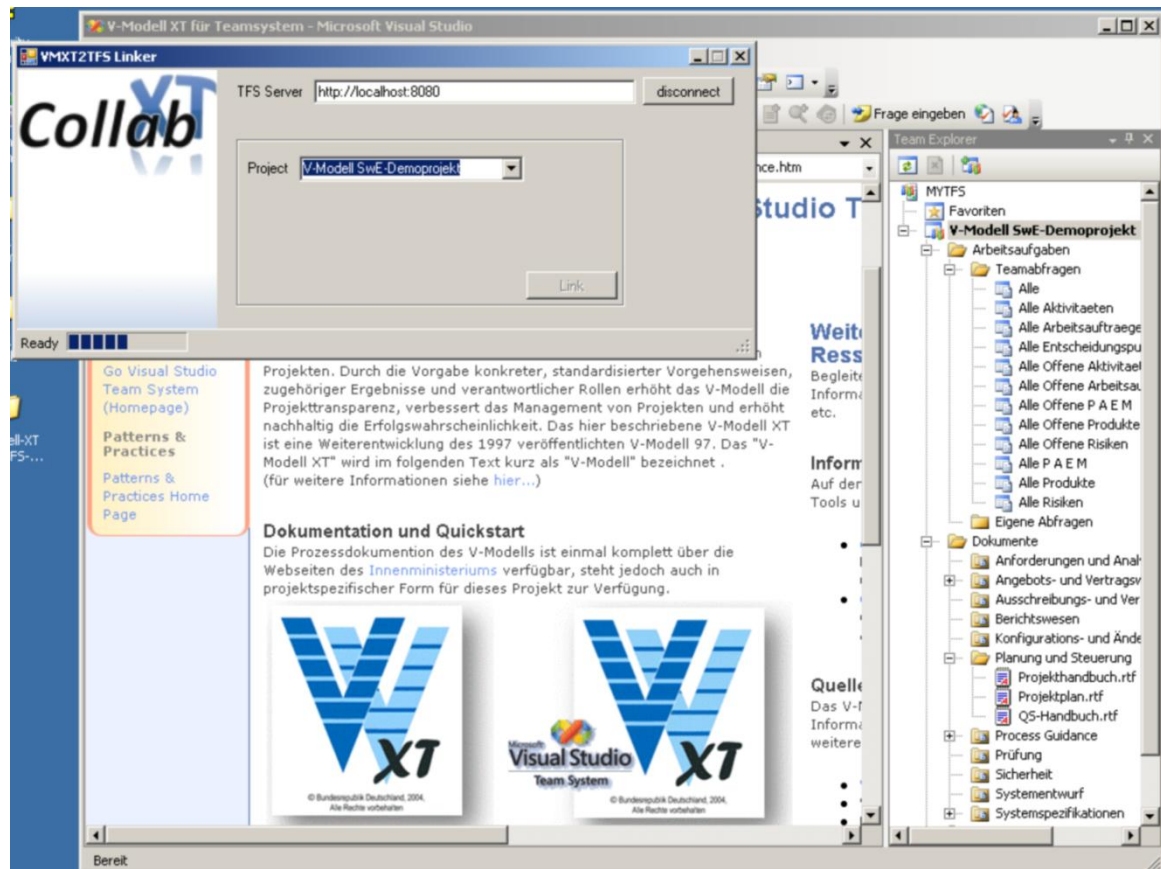


Dieses enthält alle Inhalte, die der Projektassistent erstellt hat und deckt somit die wesentlichen Grundeigenschaften eines V-Modell-Projekts ab. Zusätzlich sind bereits die Planungsinformationen für das Work Item Tracking passende aufbereitet und eingespielt worden, sodass über eine entsprechende Abfragestruktur auch der (zunächst noch) initiale Projektplan aus dem System heraus erzeugt werden kann. Dieser kann weiter bearbeitet werden und wird somit zum Live-Plan.



### 3.4.4 Instanziierung und Verknüpfung der Default Work Items

Bereits aus der rudimentären Grundkonfiguration des V-Modells aus dem Projektassistenten sind mehrere hundert Default Work Items (Entscheidungspunkte, Aktivitäten, Produkte) abzuleiten. Im Rahmen der initialen Planung werden diese bereits instanziiert. Aufgrund technischer Grenzen ist jedoch eine unmittelbare, deklarative Verknüpfung auf der Ebene des generierten Templates dieser Work Items nicht möglich. Die Verknüpfung kann erst bei der Instanziierung des Template in der Live-Datenbank erfolgen. Da das V-Modell jedoch alle relevanten Verknüpfungsinformationen bereitstellt, können diese ausgewertet werden. Alle V-Modell Work Items sind daher mit entsprechenden Referenzdatenfeldern versehen, die beim Export (und nur dort) mit Modellinformationen befüllt werden. Auf dieser Grundlage kann durch ein kleines Zusatzwerkzeug (den VMXT2TFSLinker) die Verknüpfung nach der Instanziierung hergestellt werden. Alle initial erstellten V-Modell Inhalte auf der Ebene von Work Items sind im Anschluss gemäß der Möglichkeiten der Verknüpfung über TFS-Links miteinander in Beziehung gesetzt (z.B. Entscheidungspunkte und Aktivitäten).



## 4 Zusammenfassung

Dieses Dokument hat die wesentlichen Ergebnisse und Arbeiten des Projekts *CollabXT*, insbesondere des Teils, der die Integration von V-Modell XT und Visual Studio Team Foundation Server zu Ziel hat. Dieses Dokument hat die wesentlichen Ziele beschrieben und ist detailliert auf die relevanten und für die Integration modellierten Anteile des V-Modells und deren Abbildung auf Strukturen des Team Foundation Server eingegangen. Insbesondere wurden die Ein- und Ausgabestrukturen eines Generatorwerkzeugs beschrieben. In einem Meta-Process Template wurde die wesentlichen, zeitlich stabilen Anteile des V-Modell XT statisch hinterlegt. Für wichtige Strukturen wurden eigene Work Item Typen erstellt, die wesentliche Anteile des V-Modells in die Welt des TFS überführen:

- Produkt, Aktivitäten und Entscheidungspunkte (Projektverfolgung)
- Arbeitsaufträge, Risiken, Problem- und Änderungsmeldungen (Ausgestaltung und Steuerung)

Zusätzlich wurden hier Anwendungsszenarios beschrieben, die eine sinnvolle Abbildung des V-Modells gestatten. Für die in diesem Dokument beschriebene strukturelle Umgestaltung und Anpassung des V-Modells kann aufgrund der Erhaltung des Inhalts, der Abbildung aller wesentlichen Strukturen sowie der

konsistenten Übernahme der Projektergebnisse eine weit gehende Konformität zum V-Modell Standard konstruktiv sicher gestellt werden. Die vorliegende Beschreibung von Ein- und Ausgaben, sowie der unmittelbare Anschluss an die Referenzwerkzeugkette und die ausschließliche Nutzung der durch diese bereitgestellten Ergebnisse sind dafür die Voraussetzung.

**Achtung!** Weitere Anpassungen sind darüber hinaus natürlich weiterhin möglich. Die Erfüllung aller Anforderungen der V-Modell-Konformität in konkreten Projekten oder weiteren folgenden Anpassungsschritten, die über dieses Dokument hinausgehen, muss jedoch durch ein reguläres Assessment bestätigt werden.

Weiterhin haben wir dem Anwender neben den Konzepten, Modellierungen und Referenzprofilen noch einen kurzen Anwendungsleitfaden bereitgestellt, der alle wesentlichen Schritte vom Tailoring bis zur Instanziierung des V-Modell/TFS Templates zusammenfasst. Dieser Leitfaden dient nur als grobe Orientierung. Auf den Projektwebseiten finden sich weitere Unterlagen und Materialien.

## Anhang – Referenzprofile

In diesem Anhang sind die Referenzprofile für die Abbildung des V-Modell XT auf die Visual Studio Umgebung. Die Referenzprofile umfassen die Tailoring- bzw. Anwendungsprofile für den Projektassistenten. Die Profile beschreiben dabei einen „Maximalausbau“, d.h. Teilmengen von Vorgehensbausteinen und Projektdurchführungsstrategien werden ebenfalls unterstützt.

### Referenzprofil Auftragnehmer

Das Profil für (einfache) Auftragnehmer berücksichtigt ausschließlich Entwicklungsprojekte ohne eigenständige Anforderungsfestlegung. Die Vergabe von Unteraufträgen wird aber durch dieses Profil mit abgedeckt.

Eigenschaft	Wert (e)
Bestimmende Projektmerkmale	<ul style="list-style-type: none"> <li>• Projekttyp: Systementwicklungsprojekt (AN)</li> <li>• Projektgegenstand: SW-System</li> <li>• Projektkontrolle: AN mit Unterauftragnehmern</li> </ul>
Verpflichtende Vorgehensbausteine	<ul style="list-style-type: none"> <li>• Projektmanagement</li> <li>• Qualitätssicherung</li> <li>• Konfigurationsmanagement</li> <li>• Problem- und Änderungsmanagement</li> <li>• Lieferung und Abnahme (AN)</li> <li>• Vertragsschluss (AN)</li> <li>• Systemerstellung</li> </ul>
Optionale Vorgehensbausteine (gewählt)	<ul style="list-style-type: none"> <li>• Lieferung und Abnahme (AG)</li> <li>• Vertragsschluss (AG)</li> <li>• Evaluierung von Fertigprodukten</li> <li>• Systemsicherheit</li> <li>• Kaufmännisches Projektmanagement</li> <li>• Messung und Analyse</li> <li>• SW-Entwicklung</li> <li>• Benutzbarkeit und Ergonomie</li> <li>• Weiterentwicklung und Migration von Altsystemen</li> </ul>
Mögliche Projektdurchführungsstrategien	<ul style="list-style-type: none"> <li>• Inkrementelle Systementwicklung (AN)</li> <li>• Agile Systementwicklung (AN)</li> <li>• Komponentenorientierte Systementwicklung (AN)</li> <li>• Wartung und Pflege von Systemen (AN)</li> </ul>

Das Tailoring erzeugt folgende Produkt- und Aktivitätslisten. Die Produkttypen beschreiben die erwarteten Ergebnistypen des projektspezifischen V-Modells. Die Aktivitätslisten beschreiben die zu erbringenden Vorgängen. Die Produkttypen werden für die Erzeugung der Vorlagen in der Produktbibliothek benötigt; die Aktivitäten zur Erzeugung von Planungsgrößen zur Ableitung der Projektpläne. Die genaue Beschreibung der vorliegenden Produkt- und Aktivitätstypen ist der V-Modell XT Dokumentation zu entnehmen.

Projekttyp AN (mit UAN)	
Produkte	Aktivitäten
Abnahmeerklärung	Abnahmeerklärung ausstellen (AG)
Abnahmeerklärung (von AG)	Abnahmeerklärung erhalten (AN)

<u>Projekttyp AN (mit UAN)</u>	
Produkte	Aktivitäten
Altsystemanalyse	Altsystemanalyse erstellen
Änderungsentscheidung	Änderungen beschließen
Änderungsstatusliste	Änderungsstatusliste führen
Angebot	Angebot abgeben
Angebot (von AN)	Angebote bewerten und auswählen
Angebotsbewertung	Angebote bewerten und auswählen
Anwenderaufgabenanalyse	Anwenderaufgaben analysieren
Arbeitsauftrag	Arbeitsauftrag vergeben
Ausbildungsunterlagen	Ausbildungsunterlagen erstellen
Ausschreibung	Ausschreibung erstellen
Ausschreibung (von AG)	-
Ausschreibungskonzept	Ausschreibungskonzept festlegen
Besprechungsdokument	Besprechung durchführen
Bewertung der Ausschreibung	-
Datenbankentwurf	Datenbankentwurf erstellen
Externe Einheit	Externe Einheit übernehmen
Externe-Einheit-Spezifikation	Externe-Einheit-Spezifikation erstellen
Externes SW-Modul	Externes SW-Modul übernehmen
Externes-SW-Modul-Spezifikation	Externes-SW-Modul-Spezifikation erstellen
Gefährdungs- und Systemsicherheitsanalyse	Gefährdungs- und Systemsicherheitsanalyse durchführen und bewerten
Gesamtsystemspezifikation (Pflichtenheft)	Gesamtsystemspezifikation (Pflichtenheft) erstellen
Implementierungs-, Integrations- und Prüfkonzep SW	Implementierungs-, Integrations- und Prüfkonzep SW erstellen
Implementierungs-, Integrations- und Prüfkonzep System	Implementierungs-, Integrations- und Prüfkonzep System erstellen
Implementierungs-, Integrations- und Prüfkonzep Unterstützungssystem	Implementierungs-, Integrations- und Prüfkonzep Unterstützungssystem erstellen
Kaufmännische Projektkalkulation	Kaufmännische Projektkalkulation durchführen
Kaufmännischer Projektstatusbericht	Kaufmännischen Projektstatusbericht erstellen
Kriterienkatalog für die Angebotsbewertung	Kriterienkatalog für die Angebotsbewertung erstellen
Lieferung	Lieferung erstellen und ausliefern
Lieferung (von AN)	Lieferung prüfen
Logistische Unterstützungsdokumentation	Zur logistischen Unterstützungsdokumentation integrieren
Make-or-Buy-Entscheidung	Make-or-Buy-Entscheidung durchführen
Marktsichtung für Fertigprodukte	Marktsichtung für Fertigprodukte durchführen
Mensch-Maschine-Schnittstelle (Styleguide)	Styleguide für die Mensch-Maschine-Schnittstelle erstellen
Messdaten	Messdaten erfassen
Metrikauswertung	Metrik berechnen und auswerten
Migrationskonzept	Migrationskonzept erstellen
Nachweisakte	Nachweisakte führen
Nutzungsdocumentation	Nutzungsdocumentation erstellen
Problem-/Änderungsbewertung	Problemmeldung/Änderungsantrag bewerten
Problemmeldung/Änderungsantrag	Problemmeldung/Änderungsantrag erstellen

<u>Projekttyp AN (mit UAN)</u>	
Produkte	Aktivitäten
Produktbibliothek	Produktbibliothek einrichten und pflegen
Produktkonfiguration	Produktkonfiguration verwalten
Projektabschlussbericht	Projekt abschließen
Projektabschlussbericht (von AN)	-
Projektfortschrittsentscheidung	Projektfortschrittsentscheidung herbeiführen
Projekthandbuch	Projekthandbuch erstellen
Projektmanagement-Infrastruktur	Projektmanagement-Infrastruktur einrichten und pflegen
Projektplan	Projekt planen
Projektstatusbericht	Projektstatusbericht erstellen
Projektstatusbericht (von AN)	-
Projekttagebuch	Projekttagebuch führen
Prüfprotokoll Benutzbarkeit	Benutzbarkeit prüfen
Prüfprotokoll Dokument	Dokument prüfen
Prüfprotokoll Lieferung	Lieferung prüfen
Prüfprotokoll Produktkonfiguration	Produktkonfiguration prüfen
Prüfprotokoll Prozess	Prozess prüfen
Prüfprotokoll Systemelement	Systemelement prüfen
Prüfprozedur Systemelement	Prüfprozedur Systemelement realisieren
Prüfspezifikation Benutzbarkeit	Prüfspezifikation Benutzbarkeit erstellen
Prüfspezifikation Dokument	Prüfspezifikation Dokument erstellen
Prüfspezifikation Lieferung	Prüfspezifikation Lieferung erstellen
Prüfspezifikation Produktkonfiguration	Prüfspezifikation Produktkonfiguration erstellen
Prüfspezifikation Prozess	Prüfspezifikation Prozess erstellen
Prüfspezifikation Systemelement	Prüfspezifikation Systemelement erstellen
QS-Bericht	QS-Bericht erstellen
QS-Handbuch	QS-Handbuch erstellen
Risikoliste	Risiken managen
Schätzung	Schätzung durchführen
Segment	Zum Segment integrieren
SW-Architektur	SW-Architektur erstellen
SW-Einheit	Zur SW-Einheit integrieren
SW-Komponente	Zur SW-Komponente integrieren
SW-Modul	SW-Modul realisieren
SW-Spezifikation	SW-Spezifikation erstellen
System	Zum System integrieren
Systemarchitektur	Systemarchitektur erstellen
Systemspezifikation	Systemspezifikation erstellen
Unterstützungssystem	Zum Unterstützungssystem integrieren
Unterstützungs-Systemarchitektur	Unterstützungs-Systemarchitektur erstellen
Vertrag	Vertrag abschließen (AG)
Vertrag (von AG)	Vertrag abschließen (AN)

Projekttyp AN (mit UAN)	
Produkte	Aktivitäten
Vertragszusatz	Vertragszusatz abschließen (AG)
Vertragszusatz (von AG)	Vertragszusatz abschließen (AN)

Neben den Vorgehensbausteinen und den daraus resultierenden Produkt- und Aktivitätslisten sind folgende Projektdurchführungsstrategien möglich, wobei drei davon ausgezeichnete Entwicklungsstrategien für die Neuentwicklung von Softwaresystemen sind:

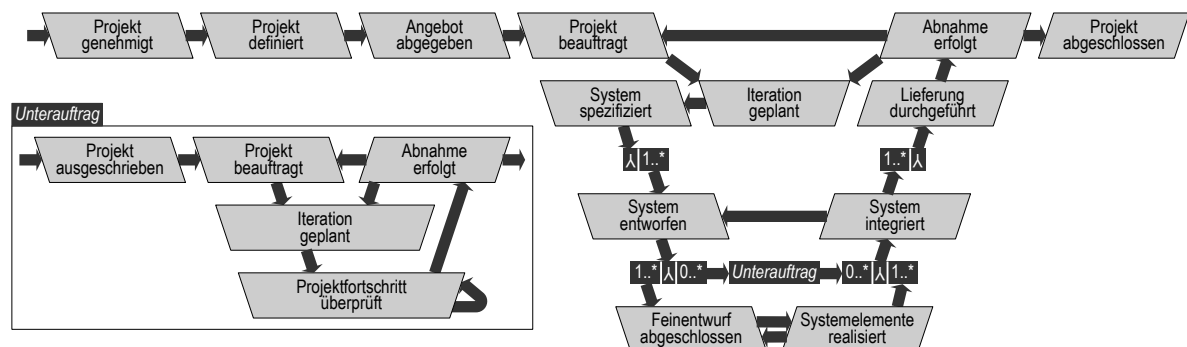
- Inkrementelle Systementwicklung (AN)
- Agile Systementwicklung (AN)
- Komponentenorientierte Systementwicklung (AN)
- Wartung und Pflege von Systemen (AN)

Die Projektdurchführungsstrategien für den Projekttyp dieses Szenarios sind miteinander kombinierbar. Die Kopplung erfolgt über einen „gemeinsamen“ Entscheidungspunkt *Iteration geplant*, der eine definierte Schnittstelle zwischen den verschiedenen Vorgehensweisen beschreibt.

### Inkrementelle Systementwicklung (AN)

Die inkrementelle Systementwicklung ist das bereits aus dem V-Modell 97 bekannte Standardvorgehen. Ausgehend von einer Iterationsplanung wird über schrittweise Verfeinerung ein Gesamtsystem über mehrere Kompositionsstufen auf seine Komponenten und Module herunter gebrochen. Das Projekt strukturiert sich hier in die drei Phasen:

- Initialisierung, Angebot und Beauftragung
- Systementwicklung
- Projektabschluss



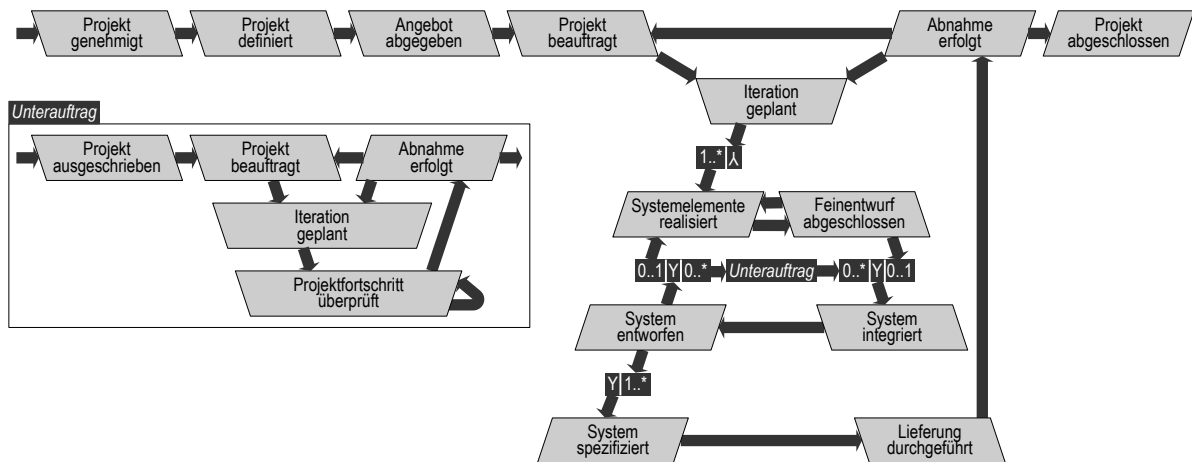
Die inkrementelle Systementwicklung sieht die Vergabe von Unteraufträgen vor. Diese können entweder auf ausgelagerte Entwicklungsprojekte oder sonstige Einkaufs- und Dienstleistungsanteile abgebildet werden. Die Möglichkeit zur Unterbeauftragung ist in allen verfügbaren Projektdurchführungsstrategien gegeben und durch das spezifizierte Szenario unterstützt.

### Agile Systementwicklung (AN)

Die agile Projektdurchführungsstrategie bildet einen Prototyping-basierten Ansatz ab. Das Projekt strukturiert sich auch hier in die drei Phasen:

- Initialisierung, Angebot und Beauftragung
- Systementwicklung
- Projektabschluss

Diese Projektdurchführungsstrategie eignet sich bspw. für UI-lastige Anwendungen, in denen Anwender schnell Feedback geben können, ohne dass seitens der Entwicklung unverhältnismäßig hohe Dokumentations- oder Spezifikationsaufwände entstehen.



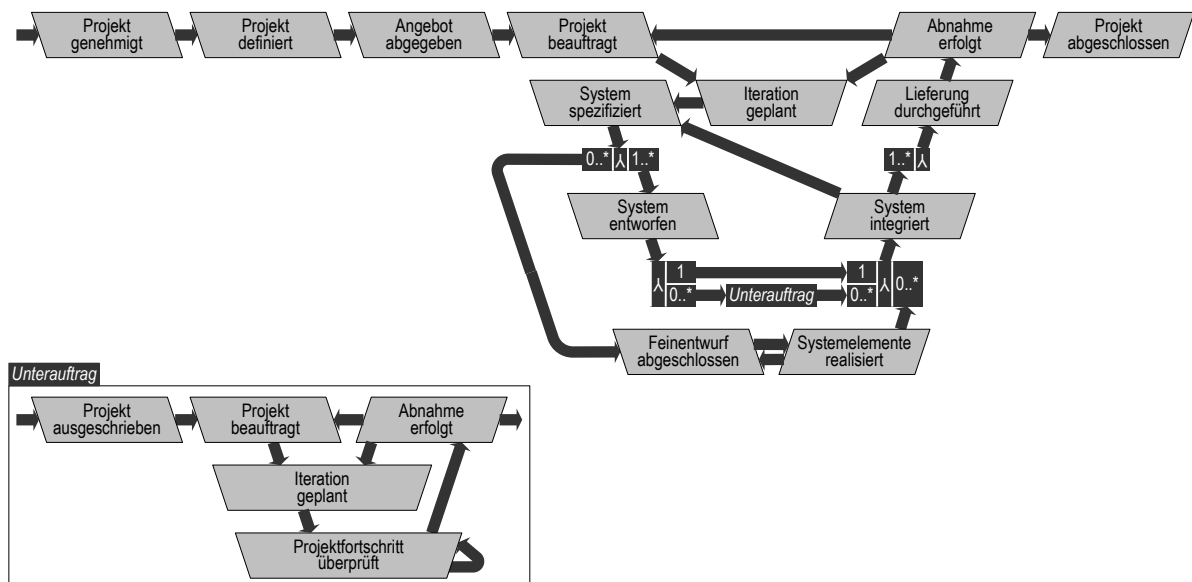
Die agile Systementwicklung sieht die Vergabe von Unteraufträgen vor. Diese können entweder auf ausgelagerte Entwicklungsprojekte oder sonstige Einkaufs- und Dienstleistungsanteile abgebildet werden. Die Möglichkeit zur Unterbeauftragung ist in allen verfügbaren Projektdurchführungsstrategien gegeben und durch das spezifizierte Szenario unterstützt.

### Komponentenorientierte Systementwicklung (AN)

Die komponentenorientierte Systementwicklung entspricht von der Grundstruktur im Wesentlichen der *Inkrementelle Systementwicklung (AN)*. Das Projekt strukturiert sich auch hier in die drei Phasen:

- Initialisierung, Angebot und Beauftragung
- Systementwicklung
- Projektabschluss

Anders als die inkrementelle Projektdurchführungsstrategie wird hier die explizite Entwicklung – das Coden – i.d.R. auf das Erstellen von Gluecode für vorhandene Komponenten abgebildet. Das Vorgehen setzt auf das Vorhandensein von Komponenten, die zu einem System integriert werden. Komponenten sind hier grob granulare Einheiten, die entweder schon erstellt und in einem Repository vorhanden sind oder durch den Zukauf von Fertigkomponenten im System verfügbar gemacht werden.

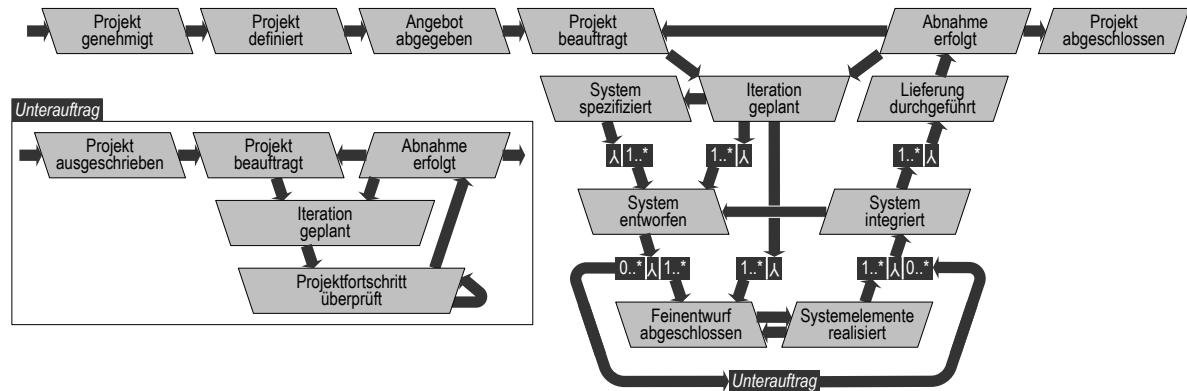


Die komponentenorientierte Systementwicklung sieht die Vergabe von Unteraufträgen vor. Diese können entweder auf ausgelagerte Entwicklungsprojekte oder sonstige Einkaufs- und Dienstleistungsanteile abgebildet werden. Die Möglichkeit zur Unterbeauftragung ist in allen verfügbaren Projektdurchführungsstrategien gegeben und durch das spezifizierte Szenario unterstützt.

## Wartung und Pflege von Systemen (AN)

Wartung und Pflege ist nicht den eigentlichen Entwicklungsvorgehensweisen zuzuordnen. Hier sind nach der Iterationsplanung verschiedene Aufsetzpunkte möglich, je nach Größe, Umfang oder Tragweite der umzusetzenden Änderung. Das Projekt strukturiert sich auch hier in die drei Phasen:

- Initialisierung, Angebot und Beauftragung
- (punktuelle, änderungsgetriebene) Systementwicklung
- Projektabschluss



Wartung und Pflege sieht die Vergabe von Unteraufträgen vor. Diese können entweder auf ausgelagerte Entwicklungsprojekte oder sonstige Einkaufs- und Dienstleistungsanteile abgebildet werden. Die Möglichkeit zur Unterbeauftragung ist in allen verfügbaren Projektdurchführungsstrategien gegeben und durch das spezifizierte Szenario unterstützt.

## Referenzprofil Auftraggeber/Auftragnehmer

Das Profil für Auftraggeber/Auftragnehmerprojekttypen deckt analog zum reinen Auftragnehmerprojekt im Wesentlichen Entwicklungsprojekte ab. Anders als beim reinen Auftragnehmerprojekt sind hier jedoch keine Ausschreibungs- und Vergabeprozesse zu durchlaufen. Dafür findet eine integrierte Anforderungsfeststellung statt. Auch hier ist die Einbindung von Unterauftragnehmern vorgesehen.

Eigenschaft	Wert (e)
Bestimmende Projektmerkmale	<ul style="list-style-type: none"> <li>• Projekttyp: Systementwicklungsprojekt (AN)</li> <li>• Projektgegenstand: SW-System</li> <li>• Projektrolle: AG/AN mit Unterauftragnehmern</li> </ul>
Verpflichtende Vorgehensbausteine	<ul style="list-style-type: none"> <li>• Projektmanagement</li> <li>• Qualitätssicherung</li> <li>• Konfigurationsmanagement</li> <li>• Problem- und Änderungsmanagement</li> <li>• Lieferung und Abnahme (AN)</li> <li>• Lieferung und Abnahme (AG)</li> <li>• Anforderungsfestlegung</li> <li>• Systemerstellung</li> </ul>
Optionale Vorgehensbausteine (gewählt)	<ul style="list-style-type: none"> <li>• Vertragsschluss (AG)</li> <li>• Evaluierung von Fertigprodukten</li> <li>• Systemsicherheit</li> <li>• Kaufmännisches Projektmanagement</li> <li>• Messung und Analyse</li> <li>• SW-Entwicklung</li> <li>• Benutzbarkeit und Ergonomie</li> <li>• Weiterentwicklung und Migration von Altsystemen</li> </ul>
Mögliche Projektdurchführungsstrategien	<ul style="list-style-type: none"> <li>• Inkrementelle Systementwicklung (AG/AN)</li> <li>• Agile Systementwicklung (AG/AN)</li> <li>• Komponentenorientierte Systementwicklung (AG/AN)</li> <li>• Wartung und Pflege von Systemen (AG/AN)</li> </ul>

Das Tailoring erzeugt folgende Produkt- und Aktivitätslisten. Die Produkttypen beschreiben die erwarteten Ergebnistypen des projektspezifischen V-Modells. Die Aktivitätslisten beschreiben die zu erbringenden



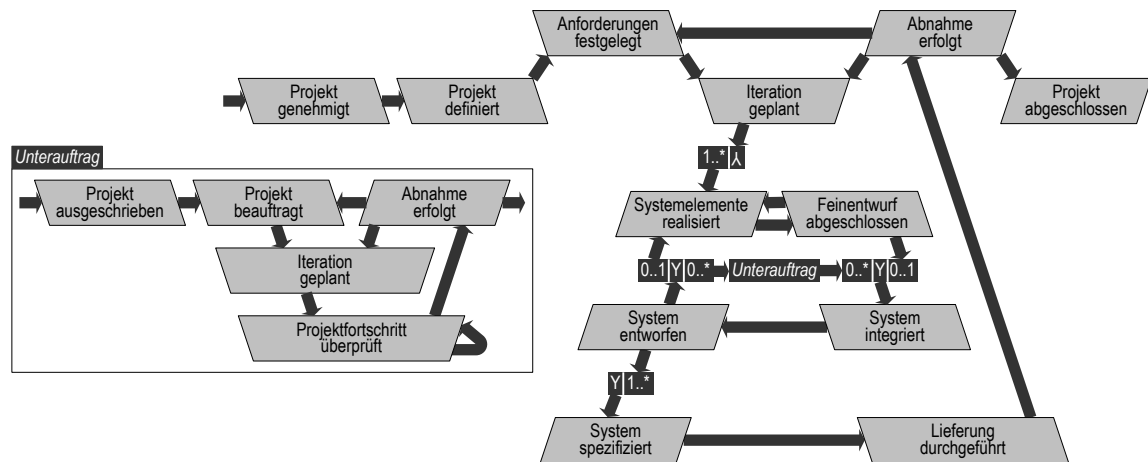
Vorgängen. Die Produkttypen werden für die Erzeugung der Vorlagen in der Produktbibliothek benötigt; die Aktivitäten zur Erzeugung von Planungsgrößen zur Ableitung der Projektpläne. Die genaue Beschreibung der vorliegenden Produkt- und Aktivitätstypen ist der V-Modell XT Dokumentation zu entnehmen.

<b>Projekttyp AG/AN (mit UAN)</b>	
<b>Produkte</b>	<b>Aktivitäten</b>
Abnahmeerklärung	Abnahmeerklärung ausstellen (AG)
Altsystemanalyse	Altsystemanalyse erstellen
Änderungsentscheidung	Änderungen beschließen
Änderungsstatusliste	Änderungsstatusliste führen
Anforderungen (Lastenheft)	Anforderungen festlegen
Anforderungsbewertung	Anforderungsbewertung erstellen
Angebot (von AN)	-
Angebotsbewertung	Angebote bewerten und auswählen
Anwenderaufgabenanalyse	Anwenderaufgaben analysieren
Arbeitsauftrag	Arbeitsauftrag vergeben
Ausbildungsunterlagen	Ausbildungsunterlagen erstellen
Ausschreibung	Ausschreibung erstellen
Ausschreibungskonzept	Ausschreibungskonzept festlegen
Besprechungsdokument	Besprechung durchführen
Datenbankentwurf	Datenbankentwurf erstellen
Externe Einheit	Externe Einheit übernehmen
Externe-Einheit-Spezifikation	Externe-Einheit-Spezifikation erstellen
Externes SW-Modul	Externes SW-Modul übernehmen
Externes-SW-Modul-Spezifikation	Externes-SW-Modul-Spezifikation erstellen
Gefährdungs- und Systemsicherheitsanalyse	Gefährdungs- und Systemsicherheitsanalyse durchführen und bewerten
Gesamtsystemspezifikation (Pflichtenheft)	Gesamtsystemspezifikation (Pflichtenheft) erstellen
Implementierungs-, Integrations- und Prüfkonzept SW	Implementierungs-, Integrations- und Prüfkonzept SW erstellen
Implementierungs-, Integrations- und Prüfkonzept System	Implementierungs-, Integrations- und Prüfkonzept System erstellen
Implementierungs-, Integrations- und Prüfkonzept Unterstützungssystem	Implementierungs-, Integrations- und Prüfkonzept Unterstützungssystem erstellen
Kaufmännische Projektkalkulation	Kaufmännischen Projektstatusbericht erstellen
Kaufmännischer Projektstatusbericht	Kaufmännische Projektkalkulation durchführen
Kriterienkatalog für die Angebotsbewertung	Kriterienkatalog für die Angebotsbewertung erstellen
Lieferung	Lieferung erstellen und ausliefern
Lieferung (von AN)	-
Logistische Unterstützungsdokumentation	Zur logistischen Unterstützungsdokumentation integrieren
Make-or-Buy-Entscheidung	Make-or-Buy-Entscheidung durchführen
Marktsichtung für Fertigprodukte	Marktsichtung für Fertigprodukte durchführen
Mensch-Maschine-Schnittstelle (Styleguide)	Styleguide für die Mensch-Maschine-Schnittstelle erstellen
Messdaten	Messdaten erfassen
Metrikauswertung	Metrik berechnen und auswerten
Migrationskonzept	Migrationskonzept erstellen
Nachweisakte	Nachweisakte führen
Nutzungsdokumentation	Nutzungsdokumentation erstellen



<u>Projekttyp AG/AN (mit UAN)</u>	
Produkte	Aktivitäten
Problem-/Änderungsbewertung	Problemmeldung/Änderungsantrag bewerten
Problemmeldung/Änderungsantrag	Problemmeldung/Änderungsantrag erstellen
Produktbibliothek	Produktbibliothek einrichten und pflegen
Produktkonfiguration	Produktkonfiguration verwalten
Projektabschlussbericht	Projekt abschließen
Projektabschlussbericht (von AN)	-
Projektfortschrittsentscheidung	Projektfortschrittsentscheidung herbeiführen
Projekthandbuch	Projekthandbuch erstellen
Projektmanagement-Infrastruktur	Projektmanagement-Infrastruktur einrichten und pflegen
Projektplan	Projekt planen
Projektstatusbericht	Projektstatusbericht erstellen
Projektstatusbericht (von AN)	-
Projekttagebuch	Projekttagebuch führen
Projektvorschlag	-
Prüfprotokoll Benutzbarkeit	Benutzbarkeit prüfen
Prüfprotokoll Dokument	Dokument prüfen
Prüfprotokoll Lieferung	Lieferung prüfen
Prüfprotokoll Produktkonfiguration	Produktkonfiguration prüfen
Prüfprotokoll Prozess	Prozess prüfen
Prüfprotokoll Systemelement	Systemelement prüfen
Prüfprozedur Systemelement	Prüfprozedur Systemelement realisieren
Prüfspezifikation Benutzbarkeit	Prüfspezifikation Benutzbarkeit erstellen
Prüfspezifikation Dokument	Prüfspezifikation Dokument erstellen
Prüfspezifikation Lieferung	Prüfspezifikation Lieferung erstellen
Prüfspezifikation Produktkonfiguration	Prüfspezifikation Produktkonfiguration erstellen
Prüfspezifikation Prozess	Prüfspezifikation Prozess erstellen
Prüfspezifikation Systemelement	Prüfspezifikation Systemelement erstellen
QS-Bericht	QS-Bericht erstellen
QS-Handbuch	QS-Handbuch erstellen
Risikoliste	Risiken managen
Schätzung	Schätzung durchführen
Segment	Zum Segment integrieren
SW-Architektur	SW-Architektur erstellen
SW-Einheit	Zur SW-Einheit integrieren
SW-Komponente	Zur SW-Komponente integrieren
SW-Modul	SW-Modul realisieren
SW-Spezifikation	SW-Spezifikation erstellen
System	Zum System integrieren
Systemarchitektur	Systemarchitektur erstellen
Systemspezifikation	Systemspezifikation erstellen
Unterstützungssystem	Zum Unterstützungssystem integrieren





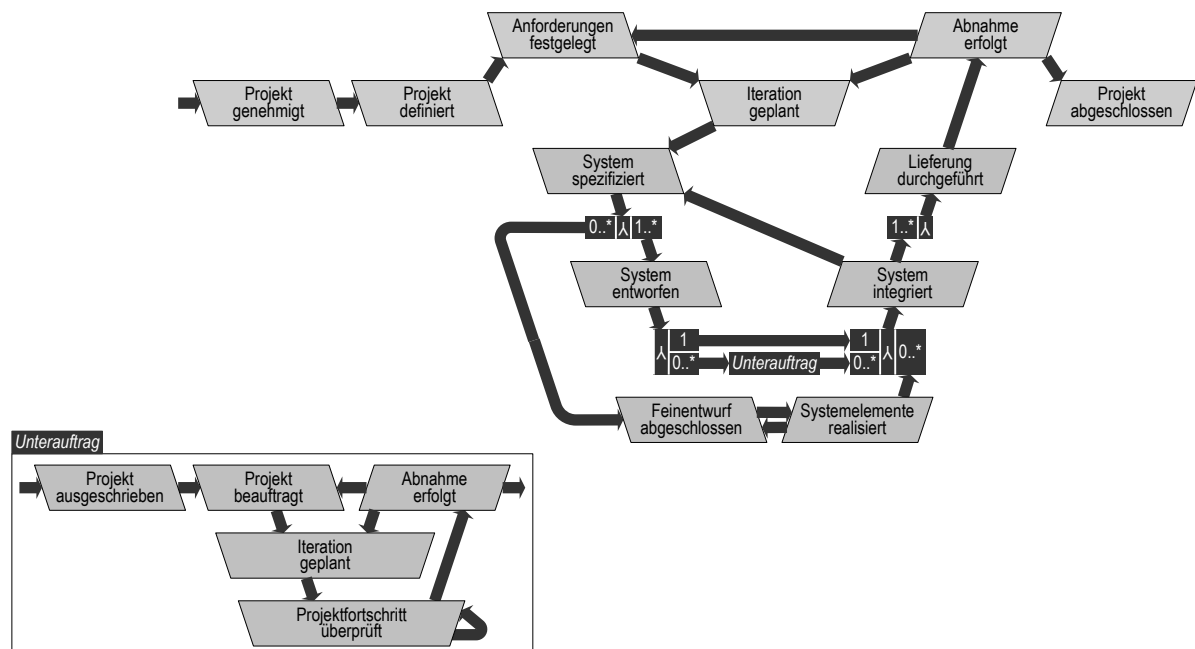
Die agile Systementwicklung sieht die Vergabe von Unteraufträgen vor. Diese können entweder auf ausgelagerte Entwicklungsprojekte oder sonstige Einkaufs- und Dienstleistungsanteile abgebildet werden. Die Möglichkeit zur Unterbeauftragung ist in allen verfügbaren Projektdurchführungsstrategien gegeben und durch das spezifizierte Szenario unterstützt.

### Komponentenorientierte Systementwicklung (AN)

Die komponentenorientierte Systementwicklung entspricht von der Grundstruktur im Wesentlichen der *Inkrementelle Systementwicklung (AG/AN)*. Das Projekt strukturiert sich auch hier in die drei Phasen:

- Initialisierung und Anforderungsfestlegung
- Systementwicklung
- Projektabschluss

Anders als die inkrementelle Projektdurchführungsstrategie wird hier die explizite Entwicklung – das Coden – i.d.R. auf das Erstellen von Gluecode für vorhandene Komponenten abgebildet. Das Vorgehen setzt auf das Vorhandensein von Komponenten, die zu einem System integriert werden. Komponenten sind hier grob granulare Einheiten, die entweder schon erstellt und in einem Repository vorhanden sind oder durch den Zukauf von Fertigteilkomponenten im System verfügbar gemacht werden.

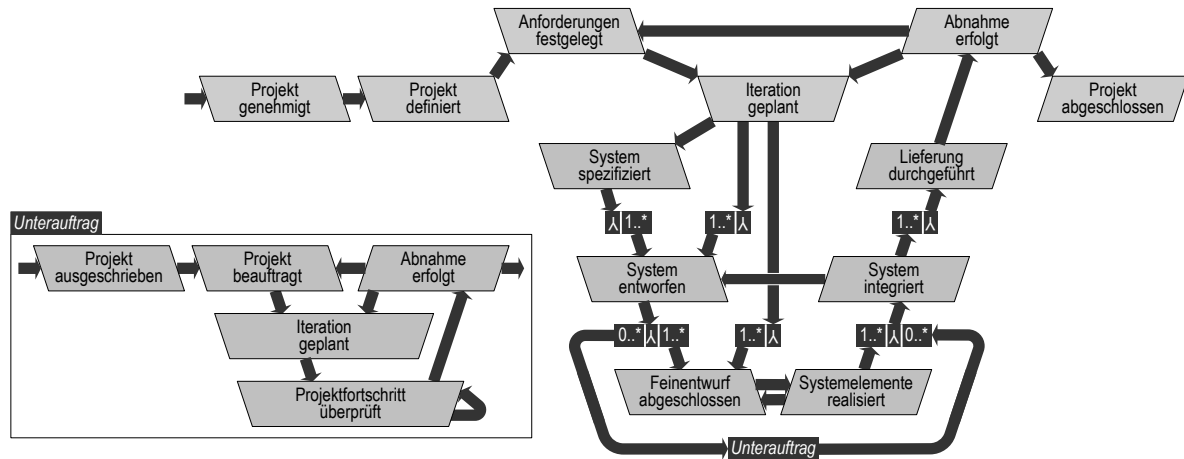


Die komponentenorientierte Systementwicklung sieht die Vergabe von Unteraufträgen vor. Diese können entweder auf ausgelagerte Entwicklungsprojekte oder sonstige Einkaufs- und Dienstleistungsanteile abgebildet werden. Die Möglichkeit zur Unterbeauftragung ist in allen verfügbaren Projektdurchführungsstrategien gegeben und durch das spezifizierte Szenario unterstützt.

## Wartung und Pflege von Systemen (AN)

Wartung und Pflege ist nicht den eigentlichen Entwicklungsvorgehensweisen zuzuordnen. Hier sind nach der Iterationsplanung verschiedene Aufsetzpunkte möglich, je nach Größe, Umfang oder Tragweite der umzusetzenden Änderung. Das Projekt strukturiert sich auch hier in die drei Phasen:

- Initialisierung und Anforderungsfestlegung
- (punktuelle, änderungsgetriebene) Systementwicklung
- Projektabschluss



Wartung und Pflege sieht die Vergabe von Unteraufträgen vor. Diese können entweder auf ausgelagerte Entwicklungsprojekte oder sonstige Einkaufs- und Dienstleistungsanteile abgebildet werden. Die Möglichkeit zur Unterbeauftragung ist in allen verfügbaren Projektdurchführungsstrategien gegeben und durch das spezifizierte Szenario unterstützt.