

Referenzprofil

# IT Configuration Coordinator

**Armin Saalman**

Dieses Referenzprofil wurde im Rahmen des bmb+f geförderten Projekts „Arbeitsprozess-orientierte Weiterbildung in der IT-Branche“ erarbeitet von:

**Fraunhofer**   
Institut  
Software- und  
Systemtechnik

Fraunhofer ISST

**Rational** software

Rational Software  
IBM Software Group

Unternehmenspartner

## Danksagung

---

Diese Profilbeschreibung entstand auf der Basis mehrerer Praxisprojekte der *Firma Rational*, deren Projektleiter Jörg Svenßon wir herzlich für seine fachkundige und umfassende Hilfe danken. Ohne seine Hilfe hätte dieses Dokument nicht entstehen können.

# Inhalt

---

<b>1</b>	<b>EINFÜHRUNG: REFERENZPROZESSE ALS CURRICULA.....</b>	<b>4</b>
1.1	EREIGNIS-PROZESS-KETTEN: SYMBOLIK.....	5
1.2	REFERENZPROZESS UND TEILPROZESSE .....	7
<b>2</b>	<b>DAS PROFIL: IT CONFIGURATION COORDINATOR (IT-KONFIGURATIONS-KOORDINATOR/IN).....</b>	<b>9</b>
2.1	TÄTIGKEITSBESCHREIBUNG .....	9
2.2	PROFILTYPISCHE ARBEITSPROZESSE .....	10
2.3	PROFILPRÄGENDE KOMPETENZFELDER .....	11
2.4	QUALIFIKATIONSERFORDERNISSE .....	12
2.5	EINORDNUNG INS SYSTEM UND KARRIEREPFADE.....	12
<b>3</b>	<b>REFERENZPROZESS .....</b>	<b>13</b>
3.1	KERNTÄTIGKEITEN DES IT CONFIGURATION COORDINATOR .....	13
3.1.1	Referenzprozess IT Configuration Coordinator .....	15
3.1.2	Das Beispielprojekt: .....	17
3.1.3	Prozesskompass IT Configuration Coordinator .....	17
3.1.4	Teilprozesse des IT Configuration Coordinator .....	17
3.1.4.1	Erstellen des Konfigurationsmanagement-Plans.....	18
3.1.4.2	Konfigurieren des CMS nach Konfigurationsmanagement-Plan .....	22
3.1.4.3	Importieren von vorhandenem Material .....	25
3.1.4.4	Erstellen des Konfigurationsplans .....	28
3.1.4.5	Bereitstellen und Übernehmen von Configuration Items .....	31
3.1.4.6	Zuordnen des Change Request zu Configuration Items .....	34
3.1.4.7	Pflegen der Metadaten, Ableiten notwendiger Änderungen .....	37
3.1.4.8	Erstellen von Statusberichten und Reports .....	40
3.1.4.9	Zusammenstellen von Komponenten zu Tests.....	42
3.1.4.10	Einpfelegen von Testergebnissen (Integrationstest) .....	44
3.1.4.11	Zusammenstellen von getesteten Derived Objects zum Gesamtsystem .....	46
3.1.4.12	Vorschlagen von Änderungen .....	48
3.1.4.13	Einpfelegen von Testergebnissen (Systemtest) .....	51

# 1 Einführung: Referenzprozesse als Curricula

---

Das Referenzprojekt des IT Configuration Coordinator verdeutlicht paradigmatisch die diesem Tätigkeitsfeld zugrunde liegenden Arbeitsprozesse, die mit ihnen verbundenen Ansprüche sowie die daraus resultierenden Anforderungen an Inhalt und Durchführung einer qualitativ hochwertigen Weiterbildung.

Das Referenzprojekt erfüllt mehrere Funktionen:

## **Aus der Praxis für die Praxis**

Als Abstraktion tatsächlich stattgefundener Projekte und Prozesse bieten die Referenzprozesse eine realistische und leicht nachvollziehbare Abbildung dessen, was die Tätigkeiten eines IT Configuration Coordinator sind.

## **Prozessorientierung als innovatives „Curriculum“**

Als vollständige Darstellung aller wichtigen Arbeitsprozesse sowie der dazugehörigen Qualifikationen, Tätigkeiten und Werkzeuge bieten die Referenzprozesse die Grundlage für die Weiterbildung zum IT Configuration Coordinator. Alle diese Prozesse müssen - entsprechend den Vorgaben - einmal oder mehrfach durchlaufen werden und ermöglichen dadurch den Weiterzubildenden den arbeitsplatznahen, integrativen Erwerb von relevanten Kompetenzen. Durch den Verbleib im Arbeitsprozess wird nicht nur für die Weiterzubildenden eine hohe Motivation (Arbeit an echten Projekten/Aufgaben) und Nachhaltigkeit erreicht, sondern auch - aus Sicht des Unternehmens - die Kontinuität und Qualität der laufenden Arbeiten gesichert (keine Ausfallzeit durch Seminartage, kein mühsamer Transfer).

## **Qualitätsstandard für die Weiterbildung**

Als Referenz bieten insbesondere die Teilprozesse und die mit ihnen verbundenen Tätigkeits- und Qualifikationsziele einen Qualitätsmaßstab für die arbeitsprozessorientierte Weiterbildung und die resultierenden Abschlüsse. Vollständige Transparenz und klare Zielvorgaben ermöglichen die qualitativ hochwertige Absicherung auch komplexer Kompetenzen sowie den systematischen Erwerb des notwendigen Erfahrungswissens.

## **Transferprozesse**

Die Generalisierung des Referenzprojekts aus der Praxis und seine didaktische Anreicherung ermöglichen eine leichte Auswahl angemessener Transferprozesse, deren Bearbeitung die Grundlage der Weiterbildung ist. Transferprozesse sind reale Prozesse, die Referenzprojekte in einer lernförderlichen Umgebung abbilden. Abgeschlossene Transferprozesse auf Basis der hier dargestellten Anforderungen und Qualitätsmaßstäbe sind nicht nur Qualifikationsnachweis des Einzelnen, sondern bilden auch die Basis eines angemesseneren und zielgerichteteren Umgangs mit Geschäfts- und Arbeitsprozessen im Unternehmen.

## 1.1 Ereignis-Prozess-Ketten: Symbolik

Die Darstellung der Referenzprozesse in Form von Ereignis-Prozess-Ketten<sup>1</sup> ermöglicht einen schnellen Überblick. Vollständigkeit kann leicht überprüft werden, Anpassungen und Modifikationen in Hinblick auf das eigene Unternehmen sind problemlos möglich und Anknüpfungspunkte an andere Prozesse, aber auch zu weiterführenden Informationen ergeben sich automatisch.

Die bei der Darstellung der Referenz- und Teilprozesse verwendete Modellierungssprache stellt eine Anpassung und Weiterentwicklung der klassischen EPK-Modellierung dar:

- Referenz- wie Teilprozesse sind aus der Sicht des jeweiligen Spezialisten, also als Arbeitsprozesse einer Person dargestellt.
- Referenz- wie Teilprozesse stellen in der Regel keinen Geschäftsprozess dar.

Die EPK-Symbole werden hier wie folgt verwendet:

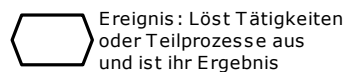
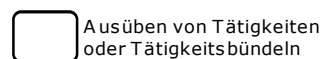
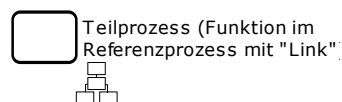


Abbildung A: Grundlegende Symbole der Referenz- und Teilprozessmodelle.


Die wichtigsten Symbole sind:


- die Tätigkeiten bzw. Tätigkeitsbündel oder Teilprozesse, die mit dem Funktionssymbol dargestellt werden
- die Ereignisse, die Tätigkeiten bzw. Teilprozesse auslösen und Ergebnisse von Teilprozessen sind

Grundsätzlich gilt:

- Auf ein Ereignis folgt immer ein Teilprozess bzw. eine Tätigkeit.
- Ergebnisse von Tätigkeiten sind sehr oft Dokumente; diese werden dann zusätzlich durch das Dokumentsymbol dargestellt.

<sup>1</sup> Vgl. A.-W. Scheer, *Wirtschaftsinformatik*, Springer 1998.

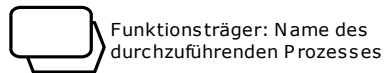
 UND-Verknüpfung

 XOR-Verknüpfung

 ODER-Verknüpfung

*Abbildung B: Konnektoren.*

Wenn Alternativmöglichkeiten bestehen, werden Ereignisse und Teilprozesse/Tätigkeiten über Konnektoren (AND, OR, XOR) verbunden. Dabei steht AND für ein verbindendes „Und“, OR für ein „Oder“, das alle Möglichkeiten offen lässt, und XOR für ein „ausschließendes Oder“, welches nur einen der angegebenen Pfade ermöglicht.



*Abbildung C: Schnittstelle.*

Da die Prozesse aus der Sicht des jeweiligen Spezialisten formuliert werden, sind Schnittstellen zu Prozessen anderer Spezialisten oder zu Entscheidungsprozessen auf höherer Ebene notwendig. Dazu wird das Schnittstellensymbol verwendet. Es steht für Prozesse, die der Spezialist nicht selber durchführt, auf deren Durchführung er aber angewiesen ist. Parallel zu jeder Schnittstelle wird die Tätigkeit dargestellt, die der Spezialist selbst in diesem Zusammenhang ausübt, wie „Beraten bei ...“, „Unterstützen bei ...“ oder „Informieren des ...“.

Alle Prozesse werden durch die Verwendung dieser Symbole klar und einfach strukturiert dargestellt und sind offen für die Übertragung in konkrete Transferprozesse.

## 1.2 Referenzprozess und Teilprozesse

Der hier vorgestellte Referenzprozess und seine Teilprozesse stellen das Curriculum des Spezialistenprofils IT Configuration Coordinator dar.

Der Referenzprozess erhebt nicht den Anspruch eines Vorgehensmodells, sondern bildet beispielhaft den möglichen Arbeitsprozess und Verlauf eines Projekts auf Spezialistenebene ab.

Er bildet die Grundlage für Weiterbildungen und damit einen Qualitäts-, Niveau- und Komplexitätsmaßstab. Die zugehörigen Teilprozesse sind hier beispielhaft modelliert und stellen eine Möglichkeit der Durchführung dar. Einzelheiten zu den unverzichtbaren Prozessen und Kompetenzfeldern sind hier im Referenzprojekt festgelegt. Die Reihenfolge und die Inhalte der Teilprozesse sind abhängig vom jeweils auszuwählenden Transferprojekt und werden in diesem Zusammenhang festgelegt.

Die Darstellung der Prozesse erfolgt systematisch:

Jeder Prozess wird mithilfe von Ereignis-Prozess-Ketten dargestellt. Einem auslösenden Ereignis folgt eine Funktion, die wiederum ein oder mehrere Ereignisse als Ergebnis hat. Ereignisse und Funktionen können mit AND, OR oder XOR, den Konnektoren, verbunden sein.

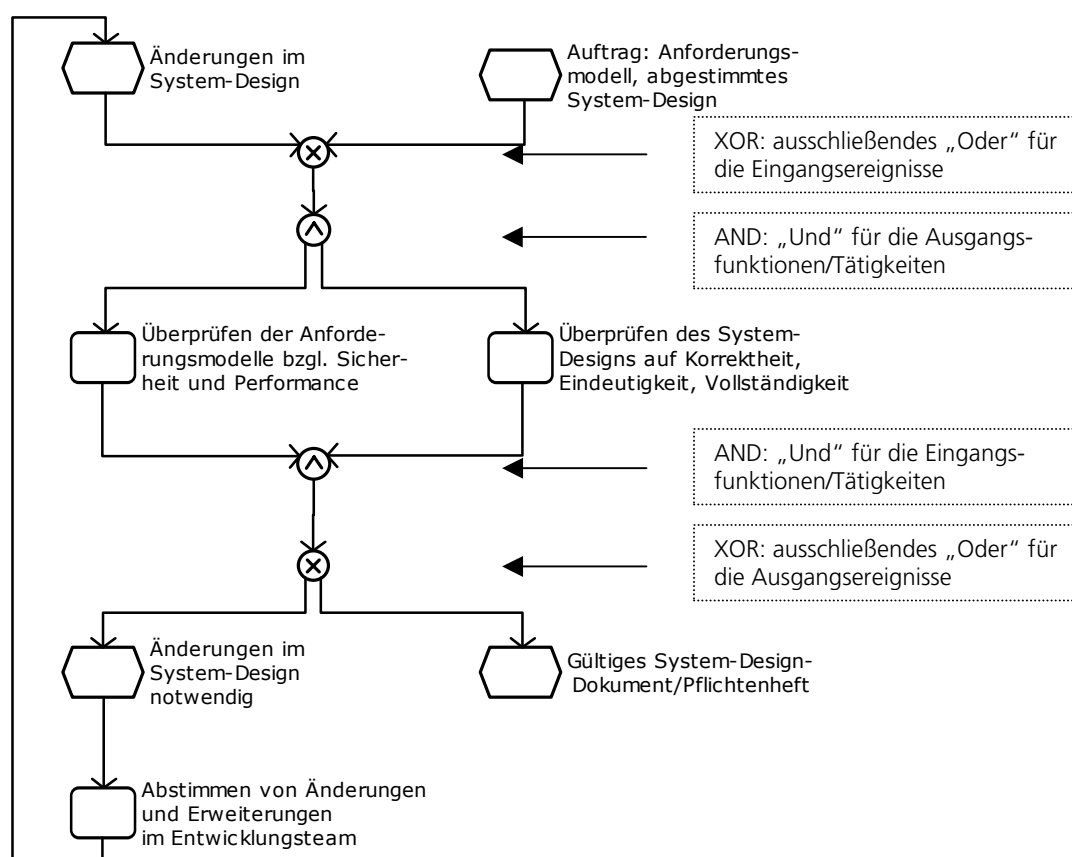


Abbildung D: Beispielprozess (Teilprozess "Überprüfen des Systemdesigns") mit unterschiedlicher Verwendung von Konnektoren.

Die Verbindung von Referenzprozess und Teilprozessen erfolgt über die Funktionen des Referenzprozesses:

Jede Funktion im Referenzprozess steht für einen Teilprozess.

Ereignisse, die dem jeweiligen Teilprozess direkt vor- oder nachgeordnet sind, sind Anfangs- und Endereignisse der jeweiligen Teilprozesse. Damit stellen die Teilprozesse die

Funktionen des Referenzprozesses ausführlich dar, und ein Hin- und Herbewegen zwischen Referenz- und Teilprozessen ist jederzeit problemlos möglich.

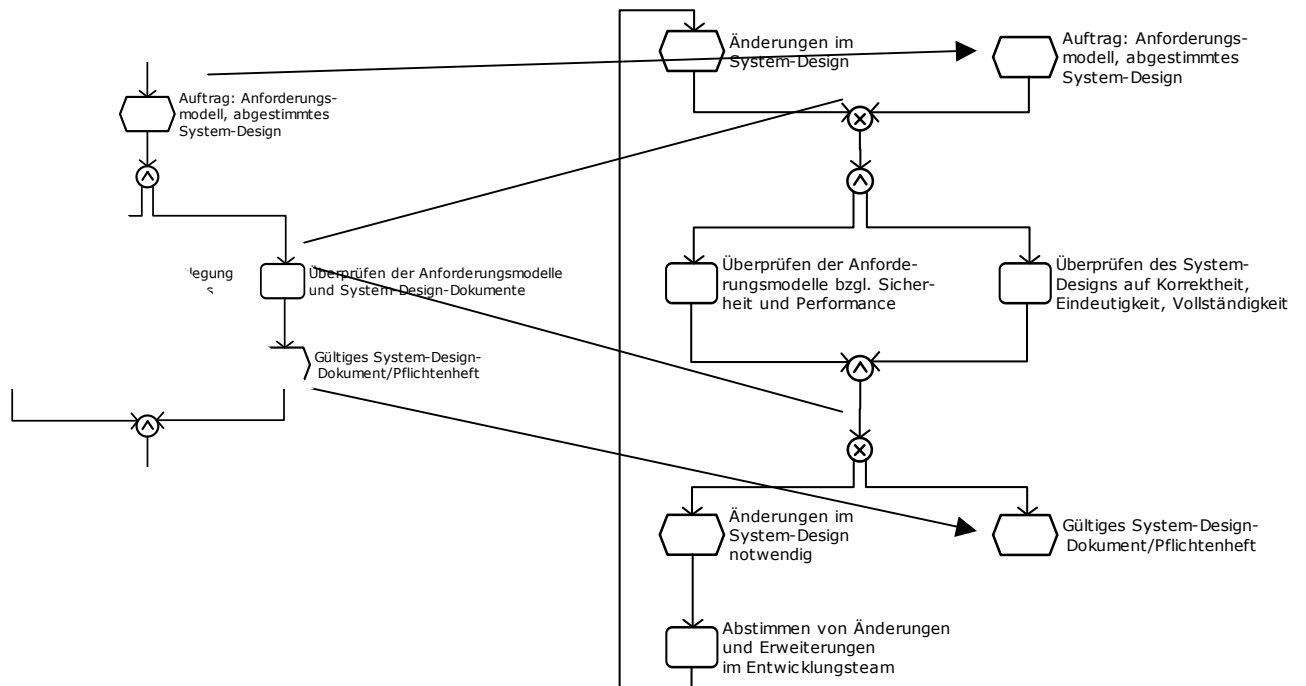


Abbildung E: Ausschnitt aus dem Referenzprozess des Database Developer (links) und Teilprozess des Database Developer "Überprüfen des Systemdesigns" (rechts).

Die Teilprozesse stellen so die wesentlichen Teile eines Projekts dar und lassen sich entsprechend auf Transferprojekte übertragen. Den Teilprozessen sind die jeweils wesentlichen Tätigkeiten und Kompetenzfelder zugeordnet.



## 2 Das Profil: IT Configuration Coordinator (IT-Konfigurationskoordinator/in)

---

IT Configuration Coordinator<sup>2</sup> organisieren das Konfigurations- und Änderungsmanagement, indem sie Software-Entwicklungsprozesse und -Ergebnisse strukturieren, verwalten und dokumentieren. Sie stellen Testumgebungen für die verschiedenen Tests (Modul-, Integrations-, Systemtests) zur Verfügung, indem sie die entsprechenden Versionen der Module mithilfe des Konfigurationsmanagement-Werkzeugs in die Testumgebung kopieren.

### 2.1 Tätigkeitsbeschreibung

---

IT Configuration Coordinator begleiten komplexe Software-Entwicklungsprozesse. Sie verwalten den gesamten Programmcode, die zugrunde liegenden Programmbibliotheken und alle im Entwicklungsprozess anfallenden Dokumente mithilfe eines Konfigurations- und Änderungsmanagement-Systems (Configuration and Change Management System, CCMS). Sie richten den Entwicklern und Testern Arbeitsumgebungen ein und stellen wechselseitig verträgliche Software-Komponenten zu konsistenten Builds und Releases zusammen.

IT Configuration Coordinator sind Mitglieder in einem so genannten Change Control Board. In dieser Funktion sind sie verantwortlich für die Verwaltung und Überwachung der Stati von Änderungsanforderungen (Change Orders) und Fehlermeldungen, bezogen auf ausgelieferte bzw. zum Akzeptanztest freigegebene IT-Systeme (so genannte Problem Notifications).

IT Configuration Coordinator arbeiten an der Schnittstelle zwischen Projektleitern, IT Test Coordinator und Spezialisten aus dem Bereich Software-Entwicklung und sind in ihrer Funktion Mitglieder der entsprechenden Entwicklungsteams. Sie sichern durch ihre Arbeit die Transparenz des Entwicklungsprozesses für alle Beteiligten.

---

<sup>2</sup> Kapitel 2: „Das Profil: IT Configuration Coordinator (IT-Konfigurationskoordinator/in)“ gibt den offiziellen Text der „Vereinbarung über die Spezialistenprofile im Rahmen des Verfahrens zur Ordnung der IT-Weiterbildung“ vom 25.05.2002 (Bundesanzeiger 105, ausgegeben am 12.06.2002) wieder.

## 2.2 Profiltypische Arbeitsprozesse

---

Die im Folgenden beschriebenen Teilprozesse dokumentieren den gesamten profiltypischen Arbeitsprozess des IT Configuration Coordinator. Die Beherrschung dieses Arbeitsprozesses in Verbindung mit den Kompetenzen in den jeweiligen Kompetenzfeldern und der Berufserfahrung bildet die Grundlage für die berufliche Handlungskompetenz.

1. Erstellen des Konfigurationsmanagement-Plans; dieser enthält Verfahren und Richtlinien zur Identifikation, Aufbewahrung und Dokumentation von Entwicklungsgegenständen (Configuration Items), Verfahren und Richtlinien zur Kontrolle von Änderungen sowie Regeln und Zuständigkeiten für das Change Control Board
2. Anlegen einer CCMS-Umgebung und Bereitstellen individueller CCM-Arbeitsbereiche für jeden Entwickler sowie spezieller Integrations- und Systemarbeitsbereiche
3. Bereitstellen (Check-Out) und Übernahme (Check-In) von Configuration Items
4. Erstellen von neuen Releases und Baselines im Integrations- bzw. Systemarbeitsbereich, Vergabe der entsprechenden Versionsnummern
5. Zusammenstellen separat entwickelter Komponenten zu Builds, Erstellen und Verwalten der dazugehörigen Stücklisten
6. Verwalten der aus den Builds abgeleiteten Derived Objects, Erstellen und Verwalten der dazugehörigen Stücklisten
7. Weitergeben von Builds zu System- und Integrationstests
8. Einpflegen von Fehlermeldungen und Änderungswünschen in das CCMS, Vorschlagen von Software-Änderungen im Change Control Board
9. Erstellen von Statusberichten und Reports über den Stand der Software-Entwicklung

## 2.3 Profilprägende Kompetenzfelder

---

Die Beherrschung der profiltypischen Arbeitsprozesse setzt Kompetenzen unterschiedlicher Reichweite in den nachstehend aufgeführten beruflichen Kompetenzfeldern<sup>3</sup> voraus. Den Kompetenzfeldern sind Wissen und Fähigkeiten sowie typische Methoden und Werkzeuge unterschiedlicher Breite und Tiefe zugeordnet.

Grundlegend zu beherrschende, gemeinsame Kompetenzfelder<sup>4</sup>:

- Unternehmensziele und Kundeninteressen
- Problemanalyse, -lösung
- Kommunikation, Präsentation
- Konflikterkennung, -lösung
- fremdsprachliche Kommunikation (englisch)
- Projektorganisation, -kooperation
- Zeitmanagement, Aufgabenplanung und -priorisierung
- wirtschaftliches Handeln
- Selbstlernen, Lernorganisation
- Innovationspotenziale
- Datenschutz, -sicherheit
- Dokumentation, -standards
- Qualitätssicherung

Fundiert zu beherrschende, gruppenspezifische Kompetenzfelder:

- Methoden und Werkzeuge der Software-Entwicklung
- Engineering-Prozesse
- Entwicklungsstandards (Leistungsfähigkeit, Sicherheit, Verfügbarkeit, Innovation)
- Projektplanung und -management
- Qualitätsstandards
- Moderation
- Konfliktbewältigung

---

<sup>3</sup> Die Kompetenzfelder werden in der nachfolgenden Auflistung jeweils durch ein zusammenfassendes Stichwort benannt. Da die Weiterbildung zum Spezialisten auf die erfolgreiche Bewältigung zunehmend offener beruflicher Handlungssituationen sowie ganzheitlichen Kompetenzerwerb abzielt, bildet der Kompetenzerwerb einen integralen Bestandteil der Arbeits- und Weiterbildungsprozesse und lässt sich nur im Zusammenhang mit diesen operationalisieren (vgl. dazu die Abschnitte „Kompetenzfelder“ in den Kapiteln 3.1.4.1ff.).

<sup>4</sup> Jeder Spezialist muss in den in diesem Abschnitt genannten „weichen“ Kompetenzfeldern wie „Kommunikation, Präsentation“, „Konflikterkennung, -lösung“ usw. ein Niveau erreichen, das über dem einer Fachkraft liegt. Das heißt, er muss auch in diesen Feldern zu eigenständigem Handeln in der Lage sein und zum Erreichen des Ziels in dem jeweiligen Feld ggf. über den Rahmen bekannter Verfahren und Lösungen hinausgehen können.

Routiniert zu beherrschende, profilspezifische Kompetenzfelder:

- Designmethoden, -strategien
- Systemarchitekturen
- Systemintegration und -anpassung
- Konfigurationsmanagement-Systeme
- Datenbanken, Betriebssysteme
- Programmbibliotheken
- Programmiersprachen

## **2.4 Qualifikationserfordernisse**

---

Im Regelfall wird ein hinreichendes Qualifikationsniveau auf der Basis einschlägiger Berufsausbildung oder Berufserfahrung vorausgesetzt.

## **2.5 Einordnung ins System und Karrierepfade**

---

Das neue IT-Weiterbildungssystem gibt auf Basis der vier neuen IT-Ausbildungsberufe drei Ebenen für die Weiterqualifizierung vor: Spezialisten, wie z. B. der IT Configuration Coordinator einer ist, sowie operative und strategische Professionals. Auf der Ebene der Spezialisten existiert eine Reihe verwandter Profile, und selbstverständlich kann sich auch der IT Configuration Coordinator zu einem Professional weiterqualifizieren.

### **Verwandte Profile**

Der IT Configuration Coordinator weist eine Verwandtschaft zu zwei Profilgruppen auf: zum einen zu den anderen Koordinatoren/Entwicklungsprozessbegleitern, insbesondere zum Quality Management Coordinator; zum anderen zu den Entwicklern, da ein Schwerpunkt der Tätigkeiten die Verwaltung der Configuration Items und die Überwachung der Systemarchitektur darstellt.

### **Aufstiegsqualifizierung**

Die am nächsten liegende Aufstiegsmöglichkeit stellt der IT Systems Manager dar, der ähnlich wie der IT Configuration Coordinator einen Überblick über den gesamten Entwicklungsprozess haben muss. Ebenfalls möglich ist der Aufstieg zum IT Business Manager als Projektmanager. Eine weitere Entwicklungsmöglichkeit für den IT Configuration Coordinator außerhalb des IT-Weiterbildungssystems stellt der Berater für Konfigurationsmanagement dar.

### 3 Referenzprozess

---

Der Referenzprozess lässt sich in vier Teilgebiete zerlegen, die den Kern der Aufgaben eines IT Configuration Coordinator beinhalten. Diese vier Teile sind:

- Erstellen des Konfigurationsmanagement-Plans und Erstellen der initialen Konfiguration
- Verwaltung der Configuration Items
- Bereitstellen der Configuration Items
- Mitarbeit im Change Control Board

Diese vier Tätigkeitsfelder finden sich im Referenzprozess – zum Teil in mehrere Teilprozesse aufgesplittet – wieder. Die Tätigkeitsfelder lassen sich leicht den Phasen des Software-Entwicklungsprozesses und des Software Life Cycle zuordnen.

#### 3.1 Kerntätigkeiten des IT Configuration Coordinator

---

- Erstellen und Führen des Konfigurations- und Konfigurationsmanagement-Plans
- Erstellen der Konfiguration im CMS
- Bereitstellen und Übernehmen von Configuration Items
- Pflege der Metadaten der Configuration Items
- Übernahme der Testergebnisse ins CMS
- Erstellen von Deliveries (Testumgebungen, Releases, Bug-Fixes)

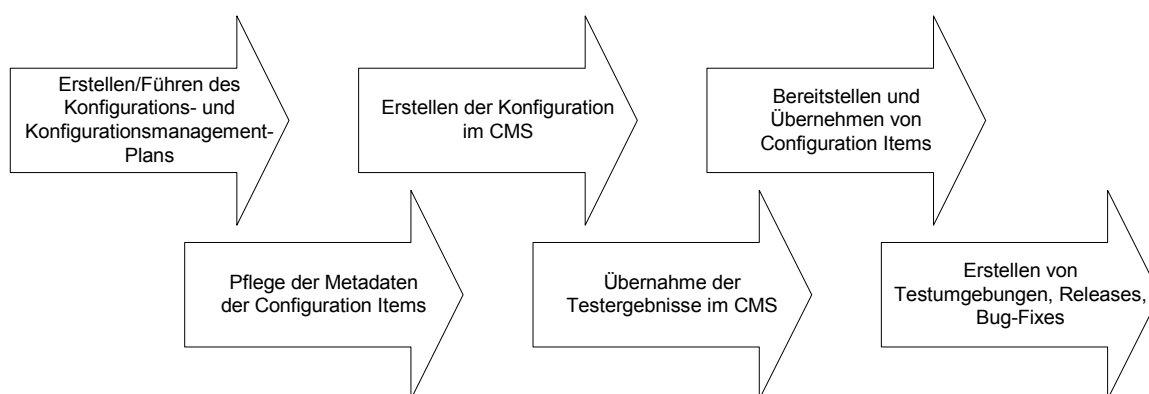


Abbildung F: Zusammenfassung der Kerntätigkeiten des IT Configuration Coordinator.

Diese Arbeitsschritte werden in den folgenden Teilprozessen ausführlich dargestellt:

Der Referenzprozess gibt die gesamten Aktivitäten des IT Configuration Coordinator auf hohem Abstraktionsniveau wieder und ermöglicht so einen Überblick.

Mit den Teilprozessen wird in den Referenzprozess hineingezoomt. Die Teilprozesse entsprechen damit in etwa der Abbildung von Arbeitsprozessen; sie stellen einen konkreten Tätigkeitsverlauf dar, einschließlich auslösenden Ereignis und Ergebnis.

Die zur Durchführung der Teilprozesse notwendigen Tätigkeiten und Kompetenzfelder werden jeweils in einem separaten Abschnitt aufgelistet.

Das Praxisprojekt dient als Beispiel zur Konkretisierung und Veranschaulichung. Es ist ein echtes, bereits durchgeführtes Projekt, auf dessen Grundlage die hier dargestellten Referenz- und Teilprozesse entwickelt wurden.

### 3.1.1 Referenzprozess IT Configuration Coordinator

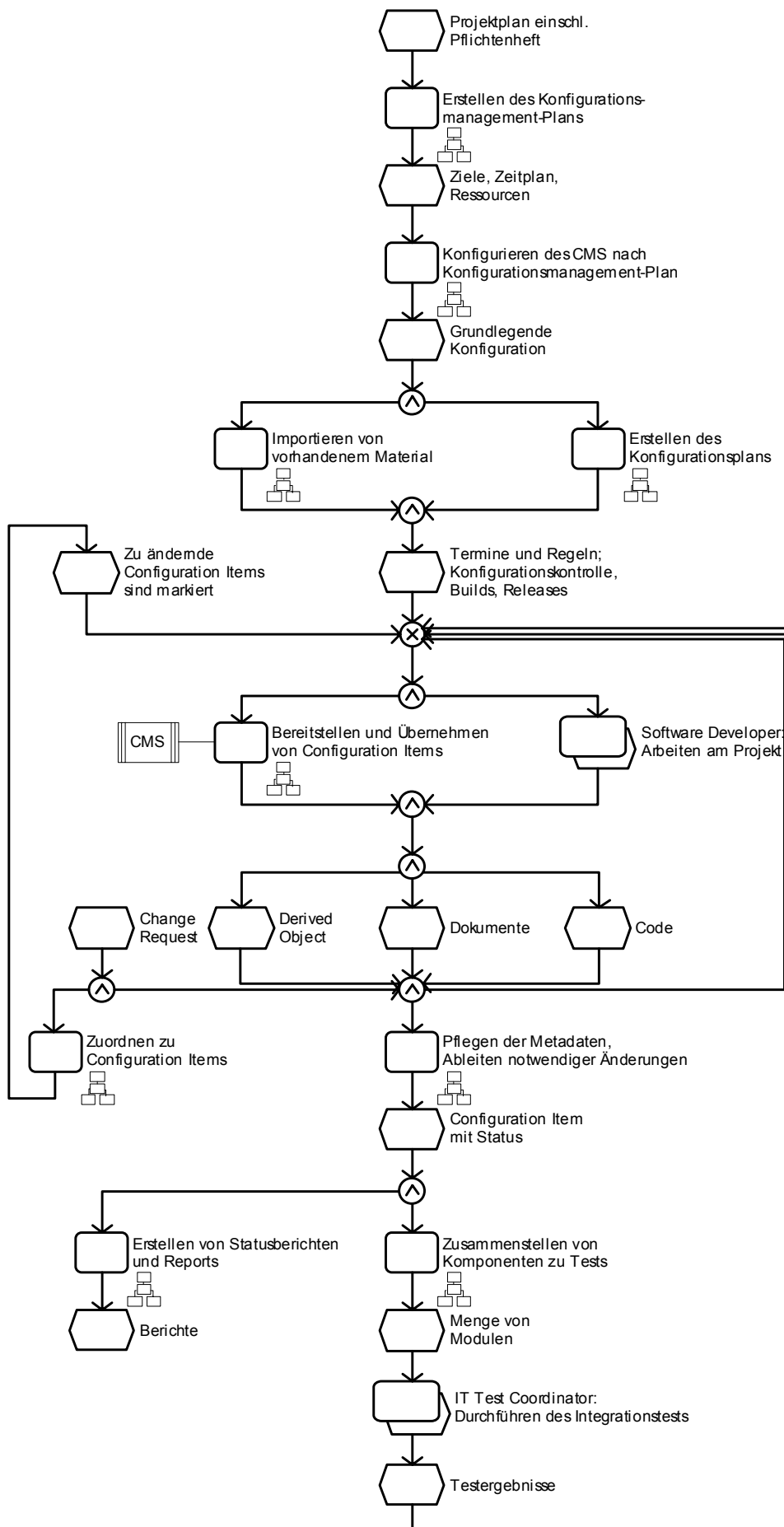


Abbildung G (vorherige Seite): Referenzprozess des IT Configuration Coordinator, Teil 1.

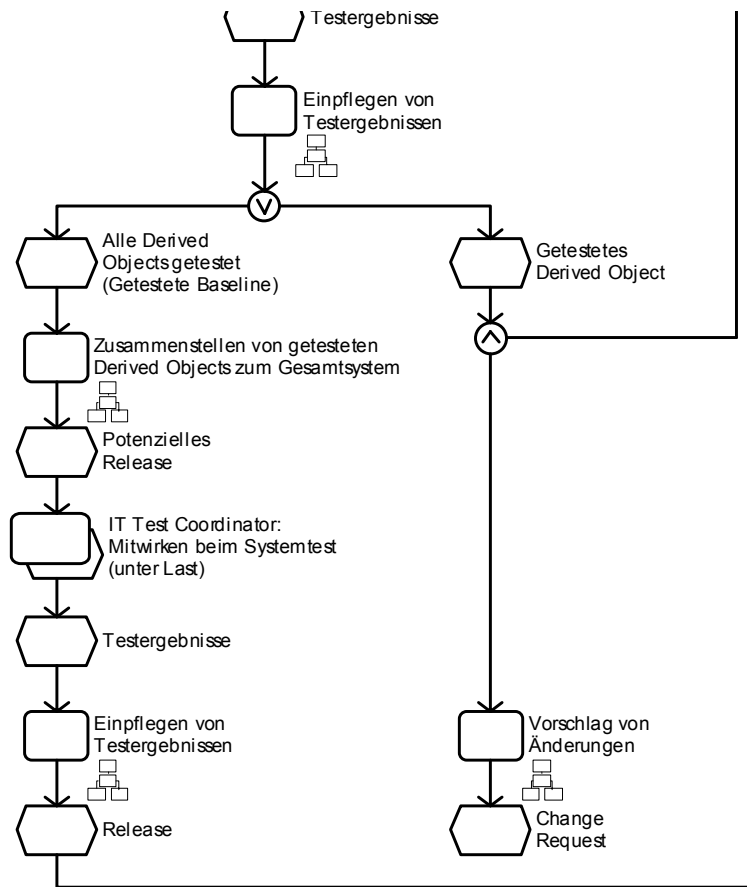


Abbildung H: Referenzprozess des IT Configuration Coordinator, Teil 2.

Dieser Referenzprozess ist typisch für das Configuration-Coordinator-Profil:

Der Referenzprozess ist im eigentlichen Sinne kein kontinuierlicher Prozess, sondern er besteht aus einzelnen Teilprozessen, die sich am „Life Cycle“ von Software-Produkten orientieren. Dabei sind die Tätigkeiten produktübergreifend angesiedelt. Der Referenzprozess beginnt mit der Erstellung eines Konfigurationsmanagement-Plans innerhalb der Projektplanung. Nach der Festlegung der Ziele und Ressourcen erfolgt eine Umsetzung des Konfigurationsmanagement-Plans in ein Configuration Management System (CMS) mit der Erstellung der Basiskonfiguration. Zu dieser Basiskonfiguration werden schon existierende Versionen von Configuration Items in die Basiskonfiguration übertragen. Damit endet das Tätigkeitsfeld des IT Configuration Coordinator in der Planungsphase. Den zweiten Schwerpunkt bildet die Verwaltung der Configuration Items mit den Teilprozessen „Bereitstellen und Übernehmen von Configuration Items“, „Pflegen von Metadaten“ und „Erstellen von Statusberichten und Reports“. Parallel zu diesen Verwaltungstätigkeiten sind die Teilprozesse zur Erstellung von Testumgebungen und die Verwaltung von Testergebnissen zu sehen. Die Tätigkeiten im Change Board sind das vierte Tätigkeitsfeld des IT Configuration Coordinator. Zentrales Dokument in diesem Tätigkeitsfeld ist der Change Request (CR). Die Teilprozesse in diesem Aufgabenbereich sind „Vorschlag von Änderungen“ und „Zuordnen des CR zu Configuration Items“.

Diese Tätigkeiten werden in kleinen wie in großen Projekten benötigt, in denen Software-Entwicklung eine Rolle spielt. Allerdings wird nicht in jedem Projekt jeder Teilprozess den gleichen Umfang und die gleiche Komplexität haben. So werden in großen Projekten sicherlich Tätigkeiten wie das Ein- und Auschecken von Configuration Items durch Software-



Entwickler erfolgen, und es wird insbesondere bei der Erstellung der Testumgebung eine enge Zusammenarbeit mit den Testkoordinatoren geben. In kleinen Projekten hingegen kann der IT Configuration Coordinator alle Aktivitäten seiner Tätigkeitsfelder allein durchführen.

### **3.1.2 Das Beispielprojekt**

Das „Course Registration System“-Projekt ist ein fiktives Projekt aus dem Schulungsbereich, in dem ein webbasiertes Registrierungssystem für Studenten und Professoren der Wiley-Hochschule entwickelt wird. Das Projekt soll das vorhandene System durch ein „state-of-the-art“-Online-System ersetzen, das Kursteilnehmern und Professoren den Zugang über PCs erlaubt. Dabei hat der IT Configuration Coordinator einen Software-Entwicklungsprozess zu begleiten, in dem mehrere Releases des neuen Systems vorgesehen sind. Weiterhin hat er verschiedene Arbeitsumgebungen und Testumgebungen innerhalb der einzelnen Entwicklungsphasen bereitzustellen. Vor dem Hintergrund dieses Beispielprojekts werden in den Beispielbeschreibungen die Tätigkeiten und Dokumente der einzelnen Teilprozesse aus Sicht des IT Configuration Coordinator betrachtet

### **3.1.3 Prozesskompass IT Configuration Coordinator**

1. Erstellen des Konfigurationsmanagement-Plans
2. Konfigurieren des CMS nach Konfigurationsmanagement-Plan
3. Importieren von vorhandenem Material
4. Erstellen des Konfigurationsplans
5. Bereitstellen und Übernehmen von Configuration Items
6. Zuordnen zu Configuration Items
7. Pflegen der Metadaten, Ableiten notwendiger Änderungen
8. Erstellen von Statusberichten, Reports
9. Zusammenstellen von Komponenten zu Tests
10. Einpflegen von Testergebnissen
11. Zusammenstellen von getesteten Derived Objects zu einem Gesamtsystem
12. Vorschlagen von Änderungen
13. Einpflegen von Testergebnissen

### **3.1.4 Teilprozesse des IT Configuration Coordinator**

Die Teilprozesse geben die Tätigkeiten in den verschiedenen Tätigkeitsfeldern innerhalb eines Entwicklungsprozesses und des Life Cycles von Software-Produkten ausführlich und detailliert wieder. Sie entsprechen so den Tätigkeiten eines IT Configuration Coordinator bei der Betreuung realer Software-Projekte, wovon eines als Grundlage zur Entwicklung des Referenz- und der Teilprozesse gedient hat und als Beispiel zur Veranschaulichung beschrieben wird.

Sicherlich werden nicht in jedem Software-Projekt alle Teilprozesse des IT Configuration Coordinator vorkommen. Insbesondere Tätigkeiten wie das Ein- und Auschecken von Configuration Items für Test- oder Entwicklungsumgebungen werden, wenn überhaupt, in enger Kooperation mit den Software-Entwicklern oder dem IT Test Coordinator durchgeführt. Ein IT Configuration Coordinator auf der Spezialistenebene sollte allerdings alle diese Tätigkeiten beherrschen. Die Betonung liegt dennoch auf den Prozessen der Verwaltung von Configuration Items und der Erstellung des Konfigurationsmanagement-Plans sowie dem Konfigurieren des CMS.

### 3.1.4.1 Erstellen des Konfigurationsmanagement-Plans

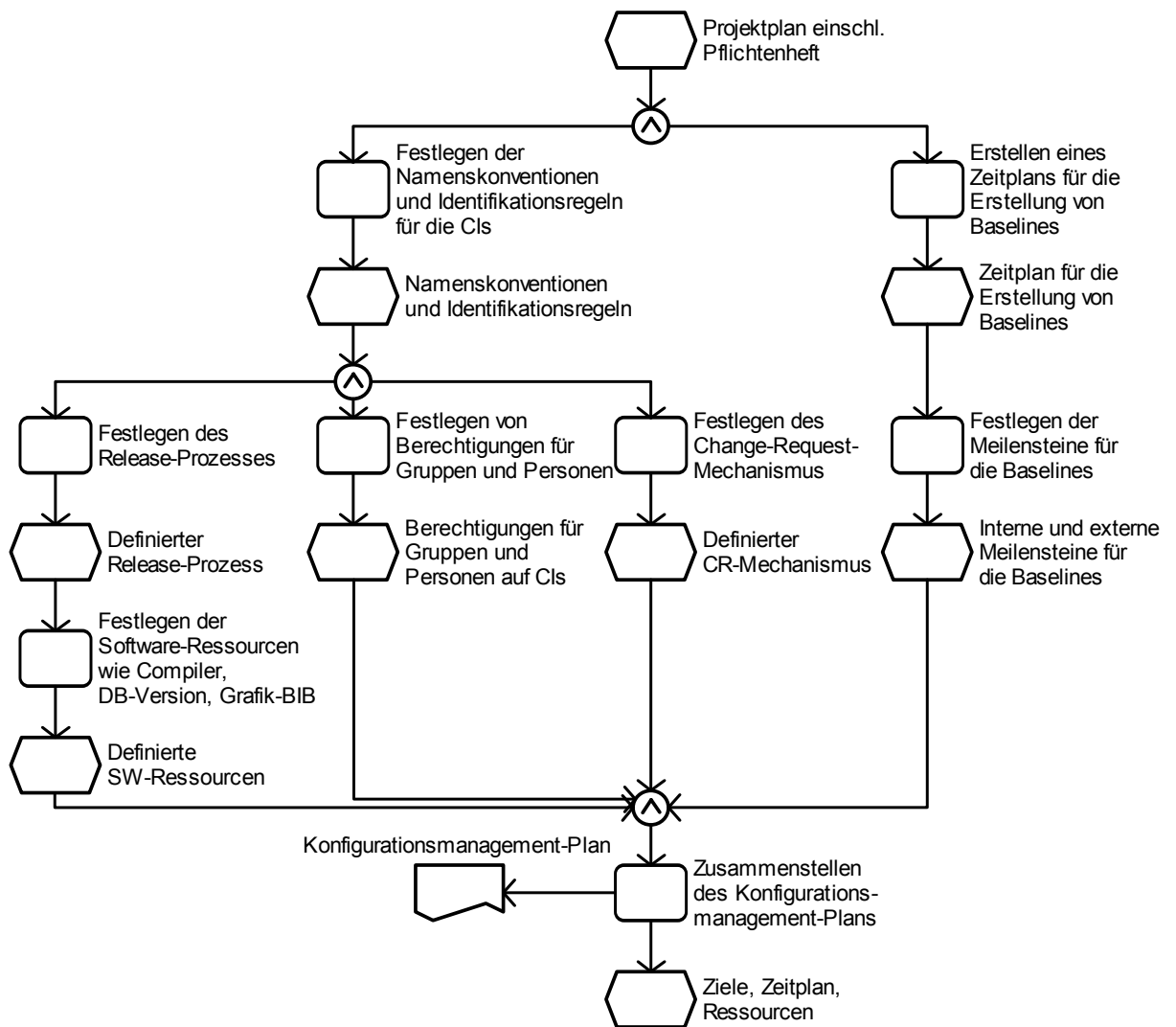


Abbildung 1: Erstellen des Konfigurationsmanagement-Plans.

Der Teilprozess ist Bestandteil der Planungsphase eines Software-Entwicklungsprozesses. In dieser Projektphase wird der Projektplan mit dem Testplan und dem Konfigurationsmanagement-Plan erstellt. In dem Konfigurationsmanagement-Plan werden die Namenskonventionen und die Identifikationsregeln für die Configuration Items festgelegt. Weiterhin werden die Zeitpunkte definiert, an denen Baselines von Configuration Items erstellt werden. Aus der Menge der erzeugten Baselines werden die internen und externen Meilensteine definiert. Die Definition des Release-Prozesses sowie des Change-Request-Mechanismus bilden weitere Kerntätigkeiten in diesem Teilprozess. Die Definition der Berechtigungen auf Configuration Items (CIs) und Arbeitsumgebungen sowie die Festlegung der Software-Ressourcen wie der zu verwendenden Compiler oder DB-Versionen sind weitere Tätigkeiten zur Erstellung des Konfigurationsmanagement-Plans.

### **3.1.4.1.1 Tätigkeiten: Erstellen des Konfigurationsmanagement-Plans**

- Festlegen der Namenskonventionen und der Identifikationsregeln für die CIs - eine zentrale Tätigkeit ist die Definition der Namenskonventionen und der Identifikationsregeln für die Configuration Items
- Erstellen eines Zeitplans für die Erstellung von Baselines - um einen möglichst aktuellen Stand der verschiedenen Arbeitsumgebungen der Software-Entwickler zu gewährleisten, sollten in der Entwicklungsphase zyklisch (z. B. einmal pro Woche) Baselines von den Configuration Items erstellt werden; die Festlegung dieser Regel und die Erstellung des darauf aufbauenden Zeitplans liegt im Aufgabenbereich des IT Configuration Coordinator innerhalb der Erstellung des Konfigurationsmanagement-Plans
- Festlegen der Meilensteine für die Baselines - mithilfe des Zeitplans für die Erstellung von Baselines legt der IT Configuration Coordinator fest, welche Versionen interne oder externe Meilensteine für die Configuration Items sind
- Festlegen des Release-Prozesses - die einzelnen Schritte eines Release-Prozesses müssen im Konfigurationsmanagement-Plan definiert werden; wenn z. B. innerhalb des Release-Prozesses eine Änderung des Datenbankschemas enthalten ist, muss im Release-Prozess festgehalten werden, dass eine Migration der Datenbestände beim Release-Wechsel zu erfolgen hat
- Festlegen der Berechtigungen für Gruppen und Personen - der IT Configuration Coordinator legt auf Modulebene der Configuration Items fest, welche Personen oder Gruppen diese Configuration Items bearbeiten dürfen
- Festlegen des Change-Request-Mechanismus - der Umgang mit Change Requests muss ebenfalls innerhalb des Konfigurationsmanagement-Plans definiert werden; dabei muss z. B. definiert werden, wer einen Change Request erstellen kann, welche Rolle den Change Request bewertet und wie der Change Request an das Entwicklungsteam weiterleitet wird
- Festlegen der Software-Ressourcen wie Compiler, DB-Version, Grafik-Bibliothek - aus dem Pflichtenheft können Vorgaben für die zu verwendenden Versionen von Fremdsoftware existieren; beispielsweise kann im Pflichtenheft die Version 7 einer Datenbank gefordert sein der IT Configuration Coordinator legt innerhalb dieser Tätigkeit die exakten Versionen (z. B. DB-Version 7.1.34) fest
- Zusammenstellen des Konfigurationsmanagement-Plans - in diesem Schritt werden die Ergebnisse aus dem Teilprozess zusammengestellt und im Konfigurationsmanagement-Plan festgehalten; der Arbeitsschritt liefert damit das Ergebnisdokument des Teilprozesses

### **3.1.4.1.2 Kompetenzfelder: Erstellen des Konfigurationsmanagement-Plans**

#### *Fähigkeiten/Fertigkeiten*

- Namenskonventionen festlegen können
- Identifikationsregeln für Configuration Items erstellen können
- Zeitpläne für Baselines erstellen können
- Meilensteine festlegen können
- Release-Prozesse definieren können
- Change-Request-Mechanismus definieren können
- Software-Ressourcen festlegen können
- Konfigurationsmanagement-Plan erstellen können
- Konfigurationsmanagement-Plan vorstellen können
- Berechtigungen auf CONFIGURATION ITEM vergeben können
- kommunizieren können

- zusammenarbeiten können
- sich durchsetzen können
- präsentieren können

#### *Wissen*

- Aufbau und Struktur des Release-Prozesses
- Aufbau und Struktur des Change-Request-Verfahrens
- Aufbau und Struktur von Berechtigungssystemen
- Aufbau und Struktur vom Konfigurationsmanagement-Plan

#### *Werkzeuge/Methoden*

- CCM-Werkzeuge
- Office-Werkzeuge
- Benutzung von Templates

### 3.1.4.1.3 Beispiel: Erstellen des Konfigurationsmanagement-Plans

Die Abbildung zeigt das Inhaltsverzeichnis des Konfigurationsmanagement-Plans für das Beispielprojekt.

<b>Configuration Management Plan</b>
<b>2 Introduction</b>
2.1 Purpose
2.2 Scope
2.3 Definitions, Acronyms, and Abbreviations
2.4 References
2.5 Overview
<b>3 Software Configuration Management</b>
3.1 Organization, Responsibilities, and Interfaces
3.2 Tools, Environment, and Infrastructure
<b>4 The Configuration Management Program</b>
4.1 Configuration Identification
4.1.1 Identification Methods
4.1.2 Project Baselines
4.2 Configuration and Change Control
4.2.1 Change Request Processing and Approval
4.2.2 Change Control Board (CCB)
4.3 Configuration Status Accounting
4.3.1 Project Media Storage and Release Process
4.3.2 Reports and Audits
<b>5 Milestones</b>
<b>6 Training and Resources</b>

Abbildung 2: Struktur eines Konfigurationsmanagement-Plans.

### 3.1.4.2 Konfigurieren des CMS nach Konfigurationsmanagement-Plan

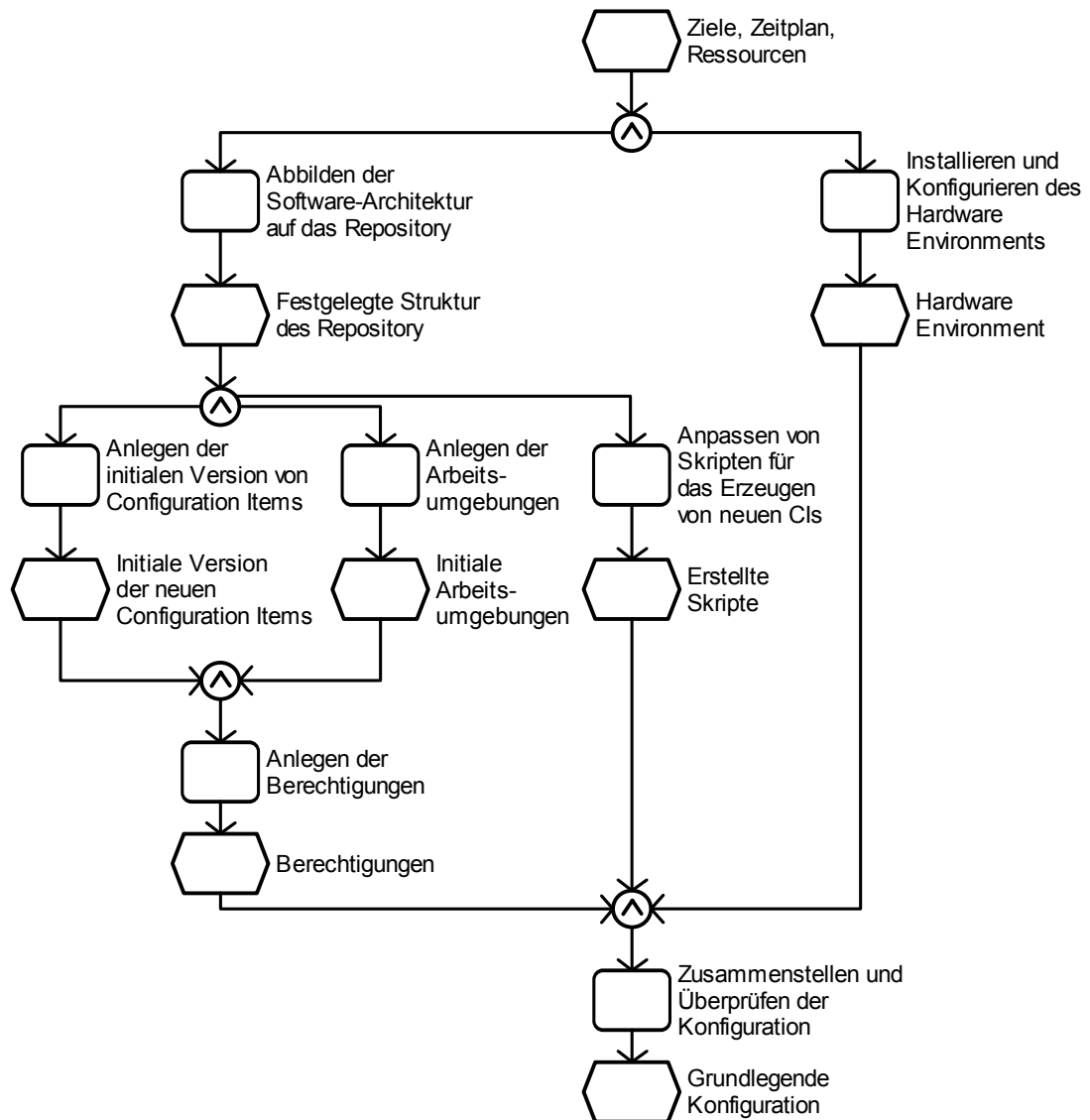


Abbildung 3: Konfigurieren des CMS nach Konfigurationsmanagement-Plan.

Im Teilprozess „Konfigurieren des CMS nach Konfigurationsmanagement-Plan“ werden die Entscheidungen aus dem Konfigurationsmanagement-Plan in das CMS umgesetzt. Dabei wird zuerst die Software-Architektur ins Repository umgesetzt. Anschließend werden die initialen Versionen für neue Configuration Items auf Modulebene angelegt. Parallel dazu kann die Referenzumgebung für die Arbeitsumgebungen angelegt werden. Daran anschließend vergibt der IT Configuration Coordinator die Berechtigungen zur Bearbeitung der verschiedenen Configuration Items. Damit der IT Configuration Coordinator nicht für das Anlegen neuer Configuration Items unterhalb der Modulebene zuständig ist, müssen in diesem Teilprozess noch Skripte für das Erzeugen neuer Configuration Items erstellt werden. Bevor die Konfiguration überprüft werden kann, muss das Hardware Environment installiert und konfiguriert werden.

In diesem Teilprozess werden die Definitionen und Regeln des Konfigurationsmanagement-Plans mithilfe des CMS umgesetzt. Diese Tätigkeiten sind stark vom eingesetzten CMS abhängig.

#### **3.1.4.2.1 Tätigkeiten: Konfigurieren des CMS nach Konfigurationsmanagement-Plan**

- Abbilden der Software-Architektur auf das Repository - die Software-Architektur auf der Ebene der einzelnen Module muss auf das Repository des CMS abgebildet werden
- Installieren und Konfigurieren des Hardware-Environments - hierbei geht es nur um das Installieren und Konfigurieren von projektabhängiger Hardware, wie z. B. eines Kartenlesers, und nicht um die Bereitstellung der Arbeitsplatzrechner oder entsprechender Server
- Anlegen der initialen Version von Configuration Items - für die zu diesem Zeitpunkt bekannten neuen Configuration Items wird in dem Repository entsprechend der Software-Architektur eine initiale Version angelegt
- Anlegen der Referenzumgebung der Arbeitsumgebungen - die Referenzumgebung für die Arbeitsumgebungen der Software-Entwickler wird erstellt
- Anpassen der Skripte für das Erzeugen von neuen CIs - die Skripte für das Anlegen neuer Configuration Items unterhalb der Modulebene werden für das neue Projekt erstellt
- Anlegen der Berechtigungen - der IT Configuration Coordinator vergibt die Rechte zur Bearbeitung der verschiedenen Configuration Items auf der Modulebene
- Zusammenstellen und Überprüfen der Konfiguration - nachdem der IT Configuration Coordinator die einzelnen Arbeitsschritte durchgeführt hat, überprüft er die Konfiguration auf Vollständigkeit, bevor er sie als grundlegende Konfiguration für weitere Arbeitsschritte frei gibt

#### **3.1.4.2.2 Kompetenzfelder: Konfigurieren des CMS nach Konfigurationsmanagement-Plan**

##### *Fähigkeiten/Fertigkeiten*

- Repositories im CMS anlegen können
- Referenzumgebung anlegen können
- Installieren und Konfigurieren des projektabhängigen Hardware-Environments können
- Version neuer Configuration Items anlegen können
- Skripten zum Anlegen von Configuration Items anpassen können
- Berechtigungen anlegen können
- zusammenarbeiten können
- selbstständig und zielorientiert arbeiten können
- Wesentliches von Unwesentlichem unterscheiden können
- logisch denken können
- abstrakt denken können
- Zusammenhänge erkennen können

##### *Wissen*

- Aufbau und Struktur von Arbeitsumgebungen
- Aufbau und Struktur von Berechtigungssystemen
- Aufbau und Struktur einer Referenzumgebung

- CMS-Konzepte
- Aufbau eines Hardware Environment

#### Werkzeuge/Methoden

- CCM-Werkzeuge
- Werkzeuge zum Skript erstellen

#### 3.1.4.2.3 Beispiel: Konfigurieren des CMS nach Konfigurationsmanagement-Plan

In dem Beispiel wird „Clear Case“ als CMS eingesetzt. Das Werkzeug erlaubt das Konfigurieren des CMS über das Einstellen verschiedener „Policies“. In dem Beispielprojekt wurde festgelegt, dass keine Konfiguration freigegeben werden darf, solange noch Configuration Items im Entwicklungsstatus sind. Durch die Einstellungen aus der Abbildung wurde in dem Beispiel der Freigabemodus für alle Configuration Items sehr restriktiv durch den IT Configuration Coordinator eingestellt.

Die Abbildung zeigt die verschiedenen „Policies“, die das CMS zum Konfigurieren für das Projekt bereitstellt.

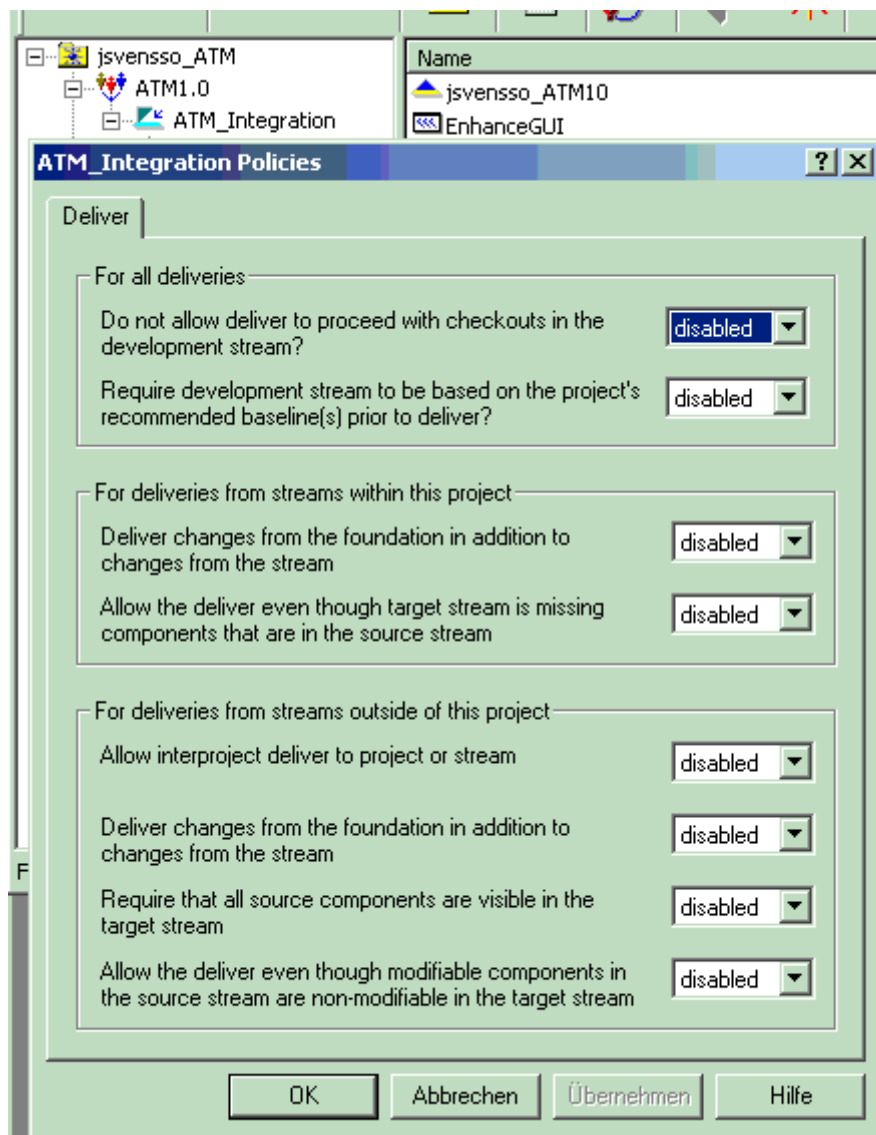


Abbildung 4: Konfigurieren des CMS durch Festlegen von „Policies“.



### 3.1.4.3 Importieren von vorhandenem Material

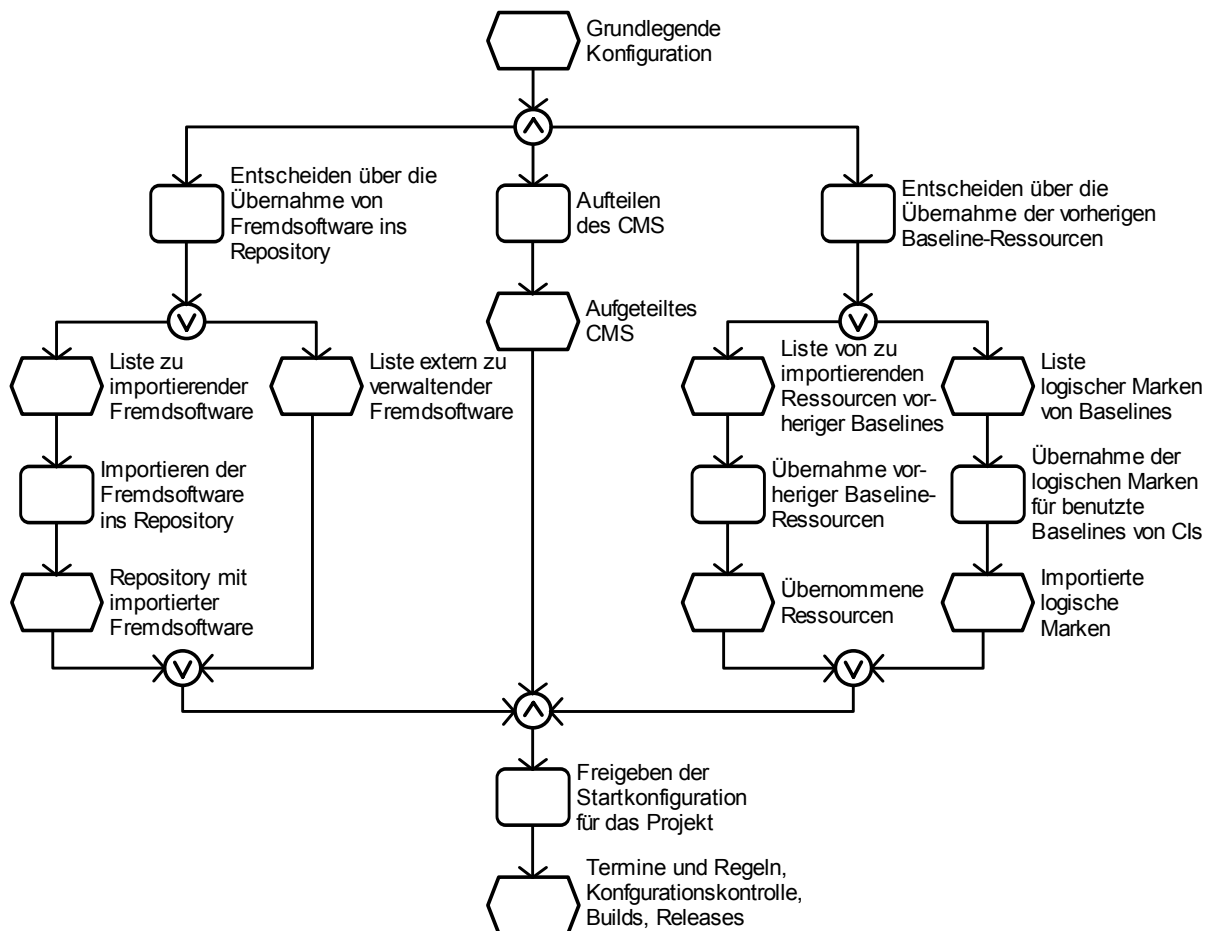


Abbildung 5: Importieren von vorhandenem Material.

Zu Beginn des Teilprozesses sind drei Entscheidungen vom IT Configuration Coordinator zu treffen:

1. Übernahme von Fremdsoftware ins Repository
2. Übernahme vorheriger Baseline-Ressourcen
3. Soll das CMS aufgeteilt werden, und wenn ja, wie?

Im Nachgang der Entscheidungen sind die entsprechenden Configuration Items in das Repository zu importieren, wobei nicht importierte vorherige Baseline-Ressourcen als logische Marken in das Repository zu übernehmen sind. Wenn diese Arbeitsschritte abgeschlossen sind, gibt der IT Configuration Coordinator die Startkonfiguration mit den Terminen und Regeln für das Projekt frei.

#### 3.1.4.3.1 Tätigkeiten: Importieren von vorhandenem Material

- Entscheiden über die Übernahme von Fremdsoftware ins Repository - der IT Configuration Coordinator muss in diesem Arbeitsschritt entscheiden, ob Fremdsoftware, die in dem Projekt benötigt wird, mit im Projekt verwaltet wird; Anhaltspunkt für eine solche Entscheidung ist, ob die Fremdsoftware noch in anderen

Projekten verwendet wird oder nicht; bei einer Verwendung in anderen Projekten sollte die Fremdsoftware projektunabhängig verwaltet werden

- Importieren der Fremdsoftware ins Repository - falls die Fremdsoftware nur in diesem Projekt verwendet wird, sollte der IT Configuration Coordinator die entsprechenden Configuration Items ins Repository importieren
- Aufteilen des CMS - um eine übersichtliche Struktur im Repository zu behalten, muss der IT Configuration Coordinator das Repository entsprechend aufteilen
- Entscheiden über die Übernahme vorheriger Baseline-Ressourcen - der IT Configuration Coordinator muss entscheiden, ob Ressourcen von vorherigen Baselines von Configuration Items mitverwaltet werden sollen; Anhaltspunkt für eine Übernahme von Configuration Items ist die geplante Weiterentwicklung von Basismodulen/Configuration Items in mehreren Projekten
- Übernahme vorheriger Baseline-Ressourcen - für die ausgewählten Baselines müssen die entsprechenden Configuration Items ins Repository übernommen werden.
- Übernahme der logischen Marken für die benutzten Baselines von CIs - für alle Configuration Items, die in dem Projekt nur verwendet werden sollen, sind logische Marken von Baselines in das Repository zu importieren

#### **3.1.4.3.2 Kompetenzfelder: Importieren von vorhandenem Material**

##### *Fähigkeiten/Fertigkeiten*

- CMS-Werkzeug einsetzen können
- Startkonfiguration erstellen können
- Startkonfiguration freigeben können
- Repository vom CMS strukturieren können
- Entscheidungen zur Baseline-Übernahme treffen können
- Fremdsoftware in die Konfiguration übernehmen können
- CMS in übersichtliche Strukturen aufteilen können
- vorherige Baselines ins Repository übernehmen können
- mit Fremdsoftware im Projekt umgehen können
- selbstständig und zielorientiert arbeiten können
- Wesentliches von Unwesentlichem unterscheiden können
- logisch denken können
- Zusammenhänge erkennen können
- Verknüpfungen erkennen können
- Genauigkeit/Sorgfalt

##### *Wissen*

- Aufbau von Software-Systemen
- Aufbau und Bestandteile von Baselines
- Aufbau von logischen Marken

##### *Werkzeuge/Methoden*

- CCM-Werkzeug
- Dateisystemmanagement-Werkzeuge (z. B. Explorer)
- technisches Englisch

#### **3.1.4.3.3 *Beispiel: Importieren von vorhandenem Material***

Im Projekt wurde mithilfe eines Wizards der letzte freigegebene Stand der Configuration Items aus dem Dateisystem importiert. Eine Übernahme aus anderen CMS wurde für das Beispiel nicht benötigt, da Configuration Items als einfache Textdateien vorlagen. Es wurde aus Zeitgründen auf den Import der bestehenden Daten eines vorher benutzten Repository verzichtet.

### 3.1.4.4 Erstellen des Konfigurationsplans

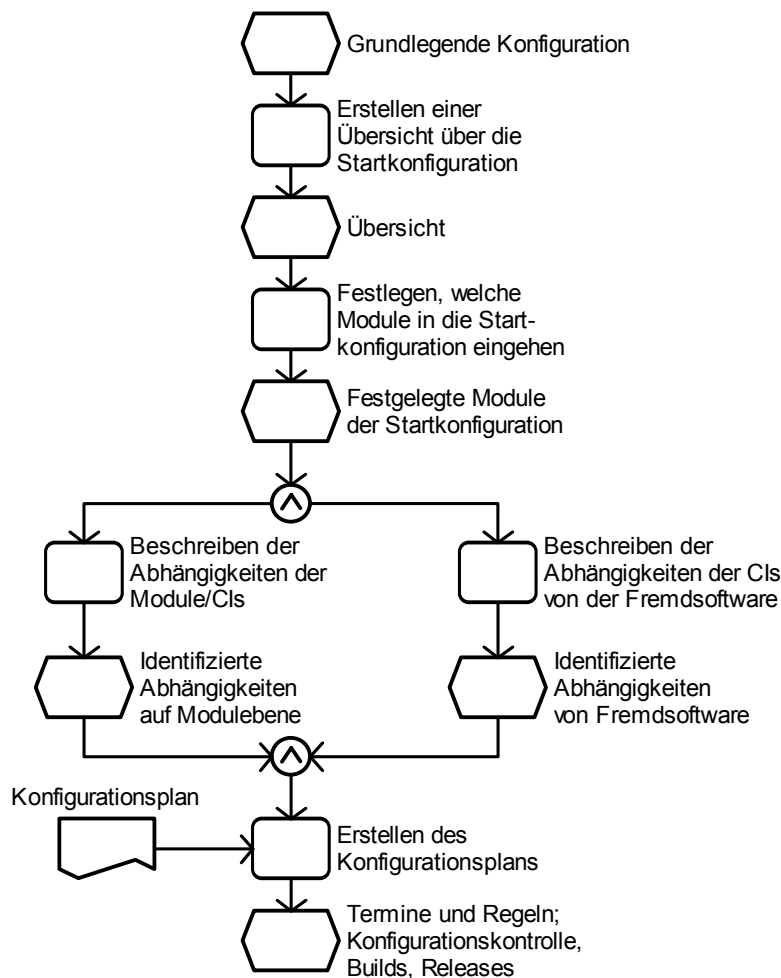


Abbildung 6: Erstellen des Konfigurationsplans.

Die Erstellung des Konfigurationsplans erfolgt in den Schritten „Erstellen einer Übersicht über die Startkonfiguration“, „Identifizieren der Abhängigkeiten zwischen den Modulen/CIs“ und „Identifizieren der Abhängigkeiten von Fremdsoftware“. Dabei arbeitet der IT Configuration Coordinator auf der Ebene von Modulen. Die Verfeinerung der Abhängigkeiten der Configuration Items auf Unit-Ebene ist durch den Software-Entwickler abzudecken.

#### 3.1.4.4.1 Tätigkeiten: Erstellen des Konfigurationsplans

- Erstellen einer Übersicht über die Startkonfiguration - entsprechend der grundlegenden Konfiguration wird eine Übersicht über die zur Konfiguration gehörenden Configuration Items auf Modulebene erstellt
- Beschreiben der Abhängigkeiten der Module/CIs - in diesem Arbeitsschritt werden die Abhängigkeiten der Configuration Items untereinander definiert
- Beschreiben der Abhängigkeiten von der Fremdsoftware - die Abhängigkeiten der Configuration Items von Fremdsoftware werden in dieser Tätigkeit identifiziert und dokumentiert; dabei sollten die Abhängigkeiten von Fremdsoftware über Basismodule gekapselt werden

- Erstellen des Konfigurationsplans - durch das Zusammenführen der drei Teilergebnisse entsteht der Konfigurationsplan für das Projekt

#### **3.1.4.4.2 Kompetenzfelder: Erstellen des Konfigurationsplans**

##### *Fähigkeiten/Fertigkeiten*

- Modulhierarchien erstellen können
- Module der Startkonfiguration festlegen können
- Übersicht über Startkonfiguration erstellen können
- Abhängigkeiten zwischen Modulen identifizieren können
- Abhängigkeiten zu Fremdsoftware beschreiben können
- Konfigurationsplan erstellen können
- selbstständig und zielorientiert arbeiten können
- Wesentliches von Unwesentlichem unterscheiden können
- logisch denken können
- Zusammenhänge erkennen können

##### *Wissen*

- Aufbau und Struktur des Konfigurationsplans
- Aufbau von Modulhierarchien
- Aufbau und Struktur einer Startkonfiguration
- Abhängigkeiten von Modulen
- Techniken der Software-Technologie
- Kapselung von Fremdsoftware

##### *Werkzeuge/Methoden*

- Modellierungswerkzeuge (z. B. Rational Rose, OEW, Together)
- Text Editor

#### **3.1.4.4.3 Beispiel Erstellen des Konfigurationsplans**

Die Abbildung zeigt die Struktur des Inhaltsverzeichnis eines Konfigurationsplans, der für das Beispiel erstellt wurde.

## **1 The Configuration Management Program**

---

### **1.1 Configuration Identification**

---

#### **1.1.1 Identification Methods**

See Wiley College CM Plan.

#### **1.1.2 Project Baselines**

Baselines are created at the end of each iteration, as described in the Software Development Plan.

### **1.2 Configuration and Change Control**

---

#### **1.2.1 Change Request Processing and Approval**

See Wiley College CM Plan.

The Configuration Control Manager and Project Manager roles of assigning work and reviewing/approving changes are performed by the lead of each team (see the organization chart in the Software Development Plan), who may delegate this responsibility as he/she deems appropriate.

#### **1.2.2 Change Control Board (CCB)**

There is no CCB, per Wiley College CM Plan.

### **1.3 Configuration Status Accounting**

---

#### **1.1.1 Project Media Storage and Release Process**

See Wiley College CM Plan.

#### **1.1.2 Reports and Audits**

See Wiley College Measurement Plan.

*Abbildung 7: Konfigurationsplan für das Beispielprojekt.*

### 3.1.4.5 Bereitstellen und Übernehmen von Configuration Items

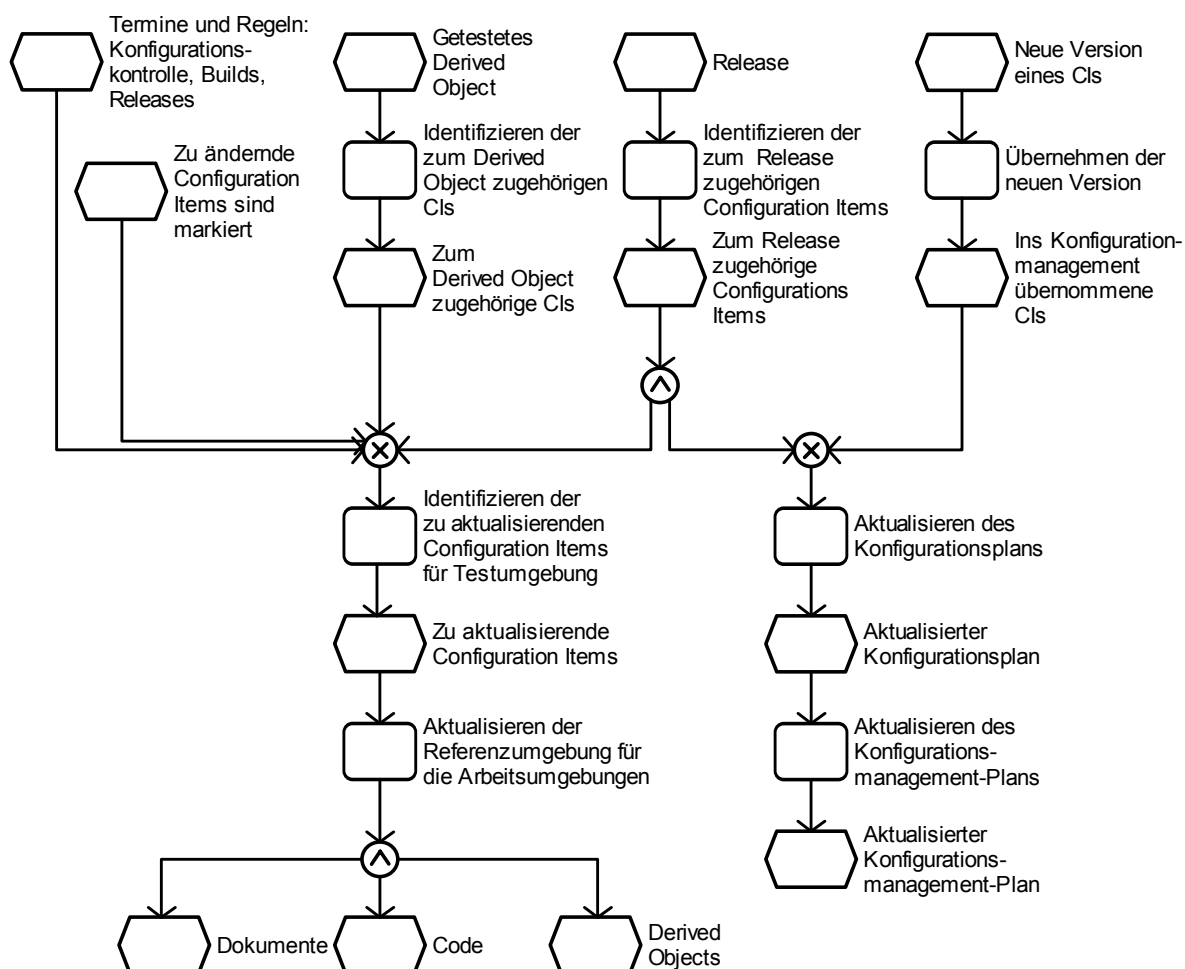


Abbildung 8: Bereitstellen und Übernehmen von Configuration Items.

Die Tätigkeiten in diesem Teilprozess lassen sich in die Bereiche „Identifizieren der bereit-zustellenden Configuration Items“, „Aktualisieren der Testumgebung“, „Übernehmen von Configuration Items“ und „Aktualisieren des Konfigurationsmanagement-Plans und Konfigurationsplans“ gliedern.

Für das Aktualisieren der Testumgebung müssen zuerst die Configuration Items zu den Eingangsdokumenten (Getestetes Derived Object, Release, Builds) identifiziert werden. Für diese markierten Configuration Items werden die abhängigen Configuration Items bestimmt und ebenfalls markiert. Die Configuration Items werden dann in einer Referenzumgebung für die verschiedenen Arbeitsumgebungen bereitgestellt.

Nach der Erzeugung von Releases und für die Übernahme neuer Versionen von Configuration Items aus den Arbeitsumgebungen müssen der Konfigurationsmanagement-Plan und der Konfigurationsplan aktualisiert werden.

Zum Übernehmen und zum Bereitstellen von Configuration Items für die Entwicklungsumgebungen sind Funktionalitäten (Check-In, Check-Out) entweder durch Einsatz eines CMS oder entsprechender Skripte bereitzustellen.

#### **3.1.4.5.1 Tätigkeiten: Bereitstellen und Übernehmen von Configuration Items**

- Identifizieren der zum Derived Object zugehörigen Configuration Items - zu den getesteten Derived Objects werden alle Configuration Items mithilfe des Konfigurationsplans ermittelt, da nach einem erfolgreichen Test diese Configuration Items für die Aktualisierung der Entwicklungsumgebungen vorgemerkt werden
- Identifizieren der zu aktualisierenden Configuration Items - in diesem Schritt findet eine Überprüfung der Versionen der Configuration Items in den Entwicklungsumgebungen und den zuletzt getesteten Versionen der Configuration Items statt; als Ergebnis werden Listen von zu aktualisierenden Configuration Items je Typ (Code, Dokumentation) geliefert
- Identifizieren der zum Release zugehörigen Configuration Items - zu dem getesteten Release werden alle Configuration Items mithilfe des Konfigurationsplans ermittelt, da nach einer Freigabe eines Releases die zugehörigen Configuration Items für eine Aktualisierung der Arbeitsumgebungen vorgemerkt werden
- Aktualisieren der Referenzumgebung für die Arbeitsumgebungen - nachdem die zu aktualisierenden Configuration Items identifiziert wurden, werden die aktuellen Versionen in die Referenzumgebung für die Arbeitsumgebungen importiert; damit stehen sie für ein Aktualisieren der Arbeitsumgebungen der Software-Entwickler zur Verfügung
- Aktualisieren des Konfigurationsplans - für jede Statusänderung und jede neue Version eines Configuration Item muss der Konfigurationsplan aktualisiert werden
- Aktualisieren des Konfigurationsmanagement-Plans - im Konfigurationsmanagement-Plan werden neue Versionen von Komponenten und neue Baselines von Produkten erfasst; deshalb ist der Konfigurationsmanagement-Plan nur bei der Freigabe von Releases und nach dem Test von Derived Objects für den Fall, dass es sich um Komponenten handelt, zu aktualisieren
- Übernehmen einer neuen Version eines Configuration Items - wenn neue Versionen von Configuration Items aus den Entwicklungsumgebungen ins CMS eingecheckt werden, wird diesen Versionen der Status ‚ungetestet‘ zugewiesen

#### **3.1.4.5.2 Kompetenzfelder: Bereitstellen und Übernehmen von Configuration Items**

##### *Fähigkeiten/Fertigkeiten*

- zu aktualisierende Configuration Items erkennen können
- Versionen mit der Konfigurationsmanagement-Software verwalten können
- Zusammenfassen und Aktualisieren der Configuration Items der Entwicklungsumgebung über Check-Out-/Check-In-Mechanismen
- Configuration Items verschiedener Typen für die Entwicklungsumgebung bereitstellen können
- Konfigurationsmanagement-Plan führen können
- die zum Release zugehörigen Configuration Items identifizieren können
- neue Version eines Configuration Items übernehmen können
- Referenzumgebung aktualisieren können
- Arbeitsumgebungen aktualisieren können
- Derived Object Configuration Items zuordnen können
- selbstständig und zielorientiert arbeiten können
- logisch denken können
- analysieren können
- abstrakt denken können



- Zusammenhänge erkennen können
- Genauigkeit/Sorgfalt

#### *Wissen*

- Struktur und Aufbau von Derived Objects
- Struktur und Aufbau eines Konfigurationsmanagement-Plans
- Struktur und Aufbau eines Konfigurationsplans
- Struktur und Aufbau von Arbeitsumgebungen
- Skriptsprachen

#### *Werkzeuge/Methoden*

- CMS-Werkzeug
- Werkzeug zur Skripterstellung

#### **3.1.4.5.3 Beispiel: Bereitstellen und Übernehmen von Configuration Items**

Siehe Teilprozess 3.1.4.3 „Importieren von vorhandenem Material“.

### 3.1.4.6 Zuordnen des Change Request zu Configuration Items

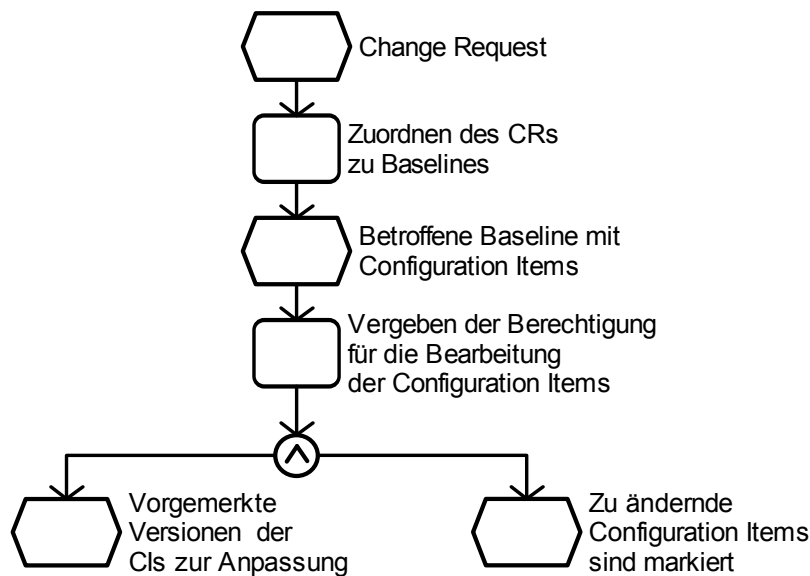


Abbildung 9: Zuordnen des Change Request zu Configuration Items.

In diesem Teilprozess werden die einzelnen Tätigkeiten bei der Zuordnung der Configuration Items zu einem Change Request (CR) beschrieben. Da im Change Request Funktionalitätsänderungen auf Komponentenebene beschrieben sind, werden im ersten Schritt die zu ändernden Komponenten identifiziert. Zu den entsprechenden Komponenten werden dann die Baselines identifiziert. Danach werden die betroffenen Versionen der Configuration Items markiert und neuere Versionen des Configuration Items für die notwendigen Anpassungen vorgemerkt. Zusätzlich werden die Berechtigungen zur Bearbeitung der Configuration Items vom IT Configuration Coordinator vergeben. Als Ergebnis dieses Teilprozesses sind die Configuration Items mit der entsprechenden Version dem Change Request zugeordnet und neuere Versionen der Configuration Items für die Anpassung im CMS vorgemerkt.

Die Basis für alle Tätigkeiten in diesem Teilprozess sind der Konfigurationsmanagement-Plan und der Konfigurationsplan. Falls das CMS die Funktionalität zum Vormerken nicht bereitstellt, muss der IT Configuration Coordinator dies mit anderen Hilfsmitteln übernehmen.

#### 3.1.4.6.1 Tätigkeiten: Zuordnen des Change Request zu Configuration Items

- Zuordnen des Change Request (CR) zu Baselines - in einem Change Request stehen Änderungen bezüglich der Funktionalität des Gesamtsystems oder einzelner Komponenten; in dieser Tätigkeit werden daher zuerst die zu ändernden Komponenten identifiziert; das Ergebnis ist eine Liste von Komponenten mit deren Versionsnummer
- Vergeben der Berechtigung für die Bearbeitung der Configuration Items - in diesem Schritt werden die Berechtigungen für die Bearbeitung der Configuration Items festgelegt

### **3.1.4.6.2 Kompetenzfelder: Zuordnen des Change Request zu Configuration Items**

#### *Fähigkeiten/Fertigkeiten*

- Configuration Items der betroffenen Komponente identifizieren können
- Change Requests zu den Baselines der Komponenten zuordnen können
- Configuration Items auswählen können
- Versionen von Configuration Items vormerken können
- Berechtigungen für Configuration Items vergeben können
- Configuration Items den verantwortlichen Entwicklern zuordnen können
- Änderungen an Configuration Items aus CR kommunizieren können
- Arbeitssystematik anwenden können
- selbstständig und zielorientiert arbeiten können
- Wesentliches von Unwesentlichem unterscheiden können
- logisch denken können
- zuordnen können
- Zusammenhänge erkennen können
- Sorgfalt/Genauigkeit

#### *Wissen*

- Struktur und Aufbau des Change Request
- Struktur und Aufbau des Konfigurationsplans

#### *Werkzeuge/Methoden*

- CMS Software
- CCMS Software

### **3.1.4.6.3 Beispiel: Zuordnen des Change Request zu Configuration Items**

In dem Beispielprojekt wurde ein Werkzeug eingesetzt, das die kombinierte Bearbeitung von Change Requests und Konfigurationen erlaubt. Die erste Abbildung zeigt die Zuordnung eines betroffenen Configuration Item zu einer Baseline und damit zu einem Change Request.

*Abbildung 10: Zuordnung eines Configuration Item zur Baseline.*

Im Beispielprojekt wurde eine funktionale Erweiterung des Course Registration System um eine Überprüfung der maximalen Anzahl von Kursteilnehmer pro Kursus über einen Change Request gestellt. Diesem Change Request werden nun mittels des eingesetzten Werkzeugs „Clear Case“ die zu ändernden Configuration Items zugeordnet.

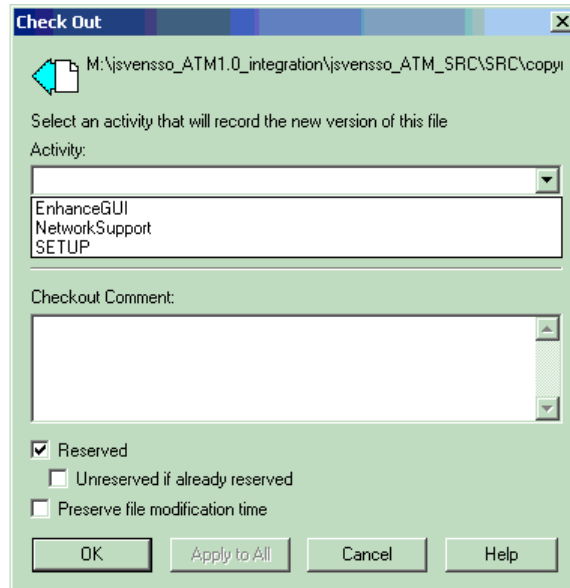


Abbildung 11: Zusammenstellen von ausgecheckten Configuration Items zu einer neuen Baseline.

Die zweite Abbildung zeigt, wie im zweiten Schritt die geänderten Configuration Items zu einer neuen Baseline zusammengestellt werden.

### 3.1.4.7 Pflegen der Metadaten, Ableiten notwendiger Änderungen

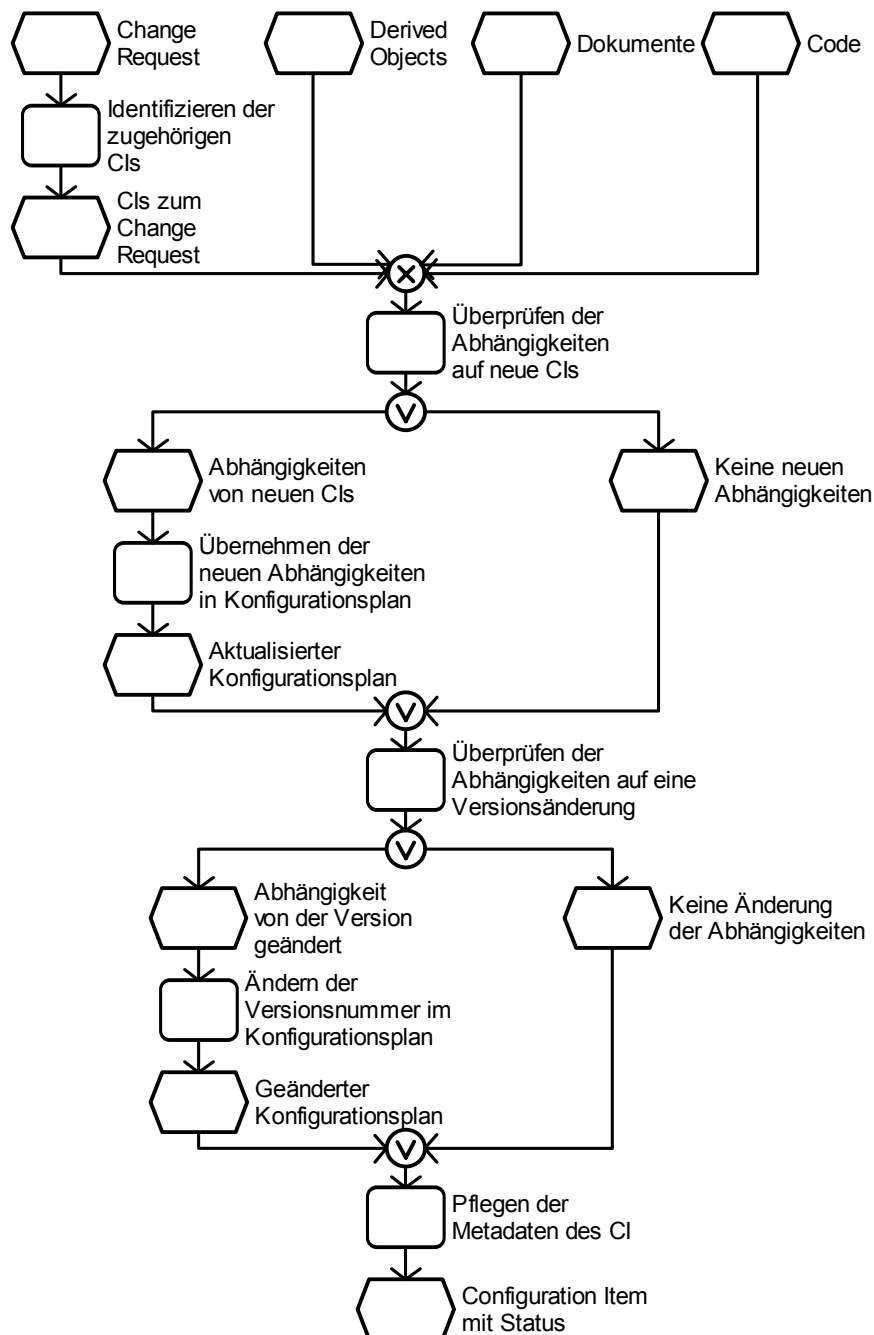


Abbildung 12: Pflegen der Metadaten, Ableiten notwendiger Änderungen.

Diese Aufgabe lässt sich in die Teilaufgaben „Ableiten der notwendigen Änderungen bei den Abhängigkeiten der Configuration Items“ und „Pflegen der Metadaten“ zerlegen. Die Ableitung der Abhängigkeiten lässt sich weiter in „Überprüfungen auf neue Abhängigkeiten der Configuration Items“ und „Überprüfung auf Änderungen in den Abhängigkeiten in der Version der Configuration Items“ aufspalten. Falls eine Überprüfung eine Änderung anzeigt, ist eine Anpassung des Konfigurationsplans und ggf. des Konfigurationsmanagement-Plans durchzuführen, und diese Änderung ist im CMS einzutragen. Wenn ein Change Request diesen Teilprozess anstößt, müssen vor den Überprüfungen der Configuration Items erst die

vom Change Request betroffenen Configuration Items identifiziert werden. Die andere Teilaufgabe ist die Pflege der Metadaten der Configuration Items. Welche Metadaten zu den Configuration Items gepflegt werden müssen, hängt stark vom eingesetzten CMS ab.

#### **3.1.4.7.1 Tätigkeiten: Pflegen der Metadaten, Ableiten notwendiger Änderungen**

- Identifizieren der zugehörigen Configuration Items - alle von einem Change Request betroffenen Configuration Items müssen mithilfe des Konfigurationsplans identifiziert werden, wenn die Zuordnung vom Change Request zu den Configuration Items nicht im CMS abgespeichert ist
- Überprüfen der Abhängigkeiten auf neue Configuration Items (CIs) - für alle Eingangsdokumente müssen die zugehörigen Configuration Items auf neue Abhängigkeiten überprüft werden
- Übernehmen der neuen Abhängigkeiten in den Konfigurationsplan - falls es neue Abhängigkeiten gibt, müssen diese Abhängigkeiten im Konfigurationsplan eingetragen werden und die entsprechenden Beziehungen im CMS angelegt werden.
- Überprüfen der Abhängigkeiten auf Versionsänderung - für alle Configuration Items der Eingangsdokumente muss überprüft werden, ob eine Anpassung auf eine neue Version der Configuration Items vorgenommen werden muss
- Übernehmen der neuen Versionsnummer in den Konfigurationsplan - falls es Versionsänderungen in den Abhängigkeiten gibt, müssen diese Abhängigkeiten im Konfigurationsplan angepasst und die entsprechenden Beziehungen im CMS gepflegt werden
- Ändern der Versionsnummer im Konfigurationsplan - falls es Versionsänderungen in den Abhängigkeiten gibt, müssen diese Abhängigkeiten im Konfigurationsplan angepasst und die entsprechenden Beziehungen im CMS gepflegt werden
- Pflegen der Metadaten des Configuration Items - falls ein CMS eingesetzt wird, erfolgt die Pflege der Metadaten meistens automatisch; andernfalls sind die Metadaten wie Autor , Erstellungs-, Änderungsdatum und Status über Listen zu pflegen

#### **3.1.4.7.2 Kompetenzfelder: Pflegen der Metadaten, Ableiten notwendiger Änderungen**

##### *Fähigkeiten/Fertigkeiten*

- Configuration Items zu Change Request zuordnen können
- Abhängigkeiten von Configuration Items in Bezug auf Versionen überprüfen können
- Konfigurationsplan führen können
- Versionsnummer in Konfigurationsplan übernehmen können
- Metadaten der Configuration Items ändern können
- Änderungen der Abhängigkeiten bei Versionsänderungen erkennen können
- neue Abhängigkeiten in Konfigurationsplan eintragen können
- Listen führen können
- Configuration Items Entwicklern zuordnen können
- Versionsnummern vergeben können
- notwendige Änderungen der Arbeits- oder Testumgebungen ableiten können

*Wissen*

- Aufbau und Struktur eines Change Request
- Aufbau und Struktur von Konfigurationsplänen
- Abhängigkeitsregeln für Configuration Items
- Versionierung von Configuration Items
- Aufbau und Struktur der Versionsnummer

*Werkzeuge/Methoden*

- CMS Software
- Text Editor

**3.1.4.7.3 Beispiel: Pflegen der Metadaten, Ableiten notwendiger Änderungen**

Das in dem Beispielprojekt eingesetzte Werkzeug pflegt die Metadaten wie Ersteller, Autor der letzten Änderung, Änderungsdatum und Versionsnummer bei jedem Check-In-Vorgang automatisch.

Wenn für die Pflege der Metadaten kein CMS-Werkzeug eingesetzt wird, muss der IT Configuration Coordinator die Metadaten gemäß dem Konfigurationsmanagement-Plan manuell aktualisieren.

### 3.1.4.8 Erstellen von Statusberichten und Reports

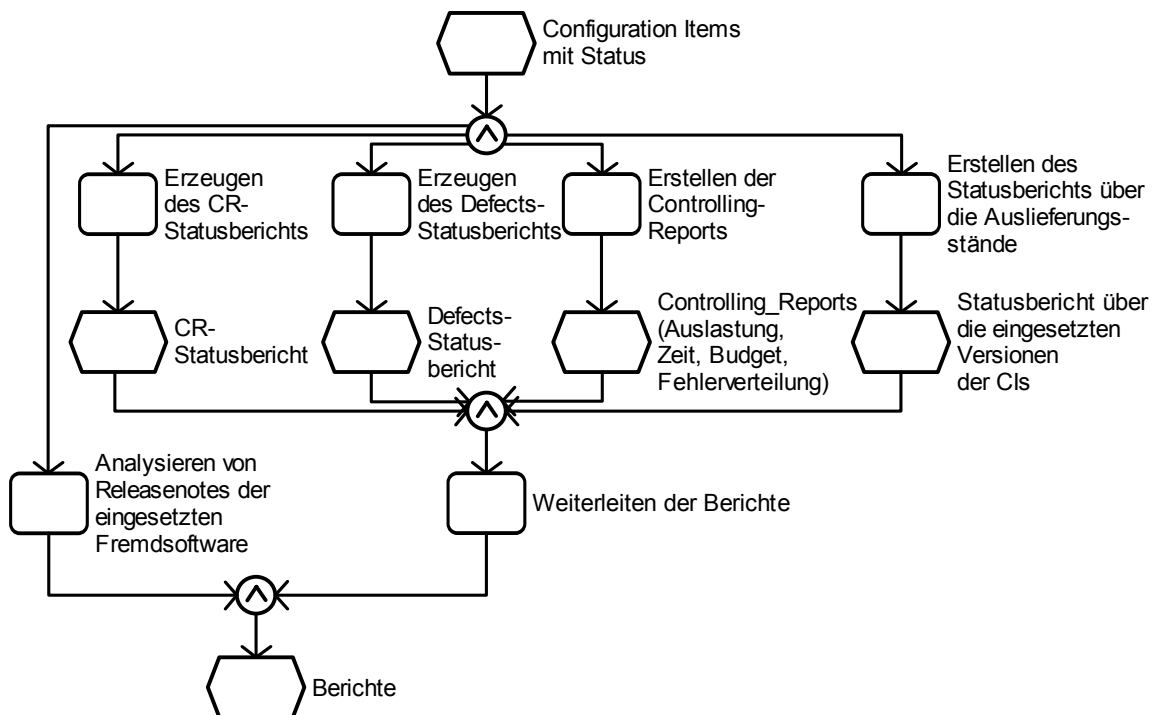


Abbildung 13: Erstellen von Statusberichten und Reports.

Der Teilprozess definiert die Berichte und Reports, die vom IT Configuration Coordinator erstellt werden und an die Projektleiter bzw. das Steuerungskomitee für die Software-Entwicklung weitergeleitet werden. Die zyklisch zu erstellenden Berichte sind der Change Request-Statusbericht und der Bericht mit der Übersicht über die Defect-Verteilung. Der Bericht mit der Übersicht über die Auslieferungsstände enthält die Zuordnung, welche Version/Baseline bei welchem Kunden im Einsatz ist. Die Controlling Reports werden je nach Unternehmenspolitik zyklisch und/oder auf Anfrage erstellt. In diesen Reports werden Aufwände, Entwicklungszeiten und Anzahl der Fehler pro Configuration Item dargestellt.

Das Aussehen und die inhaltliche Darstellung der Berichte sind stark firmenabhängig. Im Normalfall existieren Templates für die einzelnen Berichte und Reports (meist Bestandteil eines QM-Systems).

#### 3.1.4.8.1 Tätigkeiten: Erstellen von Statusberichten und Reports

- Erzeugen des CR-Statusberichts - der Change Request-Statusbericht enthält eine Übersicht über den Stand der Change Requests (CR) für ein Projekt/Produkt; in dem Bericht werden die Anzahl der Change Requests pro Status aufgelistet
- Erzeugen des Defect-Berichts - im Defect-Bericht wird die Verteilung der Defects über die Configuration Items dargestellt; ein zweiter Teil besteht in der Darstellung der Defects über die Entwicklungsphase des Projekts
- Erstellen der Controlling-Reports - die verschiedenen Controlling-Reports, die vom IT Configuration Coordinator erstellt werden, hängen stark von der jeweiligen Firmenpolitik ab; ein Beispiel für einen typischen Controlling-Report ist die



Aufschlüsselung von Aufwänden (Zeit, Ressourcen, Budget) auf die einzelnen Configuration Items, Baselines und Projekte

- Erstellen des Statusberichts über die Auslieferungszustände – in diesem Statusbericht werden die ausgelieferten Versionen/Baselines der Configuration Items für jeden Kunden festgehalten, d. h. dieser Statusbericht muss bei jedem Einspielen von Bug-Fixes oder neuen Releases aktualisiert werden
- Analysieren von Releasenotes der eingesetzten Fremdsoftware - da der IT Configuration Coordinator die zentrale Stelle ist, die weiß, welche Fremdsoftware in welchem Projekt eingesetzt wird, sollten die Releasenotes der Patches der Fremdsoftware über den IT Configuration Coordinator an die entsprechenden Entwickler weitergeleitet werden
- Weiterleiten der Berichte - die einzelnen Berichte werden an die entsprechenden Personen weitergeleitet

#### **3.1.4.8.2 Kompetenzfelder: Erstellen von Statusberichten und Reports**

##### *Fähigkeiten/Fertigkeiten*

- Reports definieren können
- Reports erstellen können
- Reports auswerten können
- Reports an die entsprechenden Personen weiterleiten können
- Reports den CRs zuordnen können
- Reports den Configuration Items zuordnen können
- Statusberichte über Auslieferungsstände erstellen können
- Ergebnisse der Reports kommunizieren können
- Releasenotes von Fremdsoftware analysieren können
- Releasenotes zu Baselines von eigenen Produkten zuordnen können

##### *Wissen*

- Aufbau und Inhalt von CR-Statusberichten
- Aufbau und Inhalt von Defect-Berichten
- Aufbau und Inhalt von Statusberichten über die Auslieferungszustände
- Aufbau und Inhalt von Controlling-Berichten

##### *Werkzeuge/Methoden*

- Text Editor
- Tabellen-Programm (Excel u. a.)
- Templates für die einzelnen Reports

#### **3.1.4.8.3 Beispiel: Erstellen von Statusberichten und Reports**

In dem Beispielprojekt wurden verschiedene Reports mithilfe des CMS definiert. Für das Beispielprojekt wurde mithilfe des Statusberichts „Pending Activities“ eine Übersicht über sich verzögernde Tätigkeiten bei der Entwicklung der Software-Komponenten erstellt. Dieser Bericht wurde an die Projektleitung weitergeleitet.

### 3.1.4.9 Zusammenstellen von Komponenten zu Tests

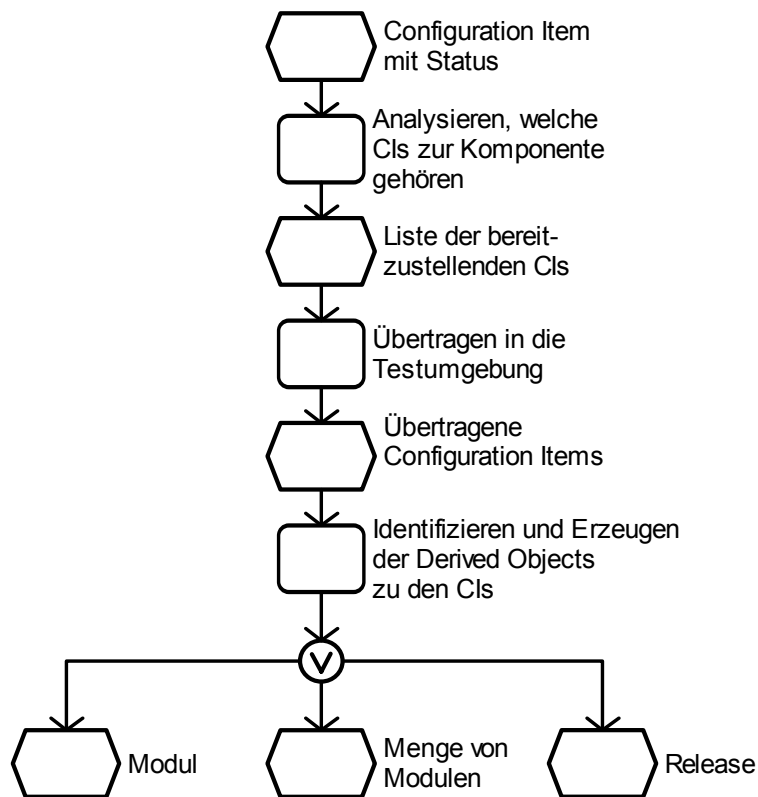


Abbildung 14: Zusammenstellen von Komponenten zu Tests.

In dem ersten Schritt muss in diesem Teilprozess vom IT Configuration Coordinator und IT Test Coordinator festgelegt werden, welche Komponenten für den Test bereitgestellt werden müssen. Anschließend werden die Configuration Items in die Testumgebung übertragen. Abschließend werden die Derived Objects von den Configuration Items in der Testumgebung vom IT Configuration Coordinator erzeugt. Dies geschieht wieder in Zusammenarbeit mit dem IT Test Coordinator.

#### 3.1.4.9.1 Tätigkeiten: Zusammenstellen von Komponenten zu Tests

- Analysieren, welche Configuration Items zur Komponente gehören - für die Zusammenstellung der Komponenten zu Tests müssen zuerst die Configuration Items mit den entsprechenden Versionen aus dem Konfigurationsplan oder mithilfe des CMS ermittelt werden; als Ergebnis liefert diese Tätigkeit die Liste der Configuration Items
- Übertragen in die Testumgebung - je nach Testart werden dann die entsprechenden Configuration Items mit den Derived Objects für die Testumgebung bereitgestellt und in Zusammenarbeit mit dem IT Test Coordinator in die Testumgebung übertragen
- Identifizieren der Derived Objects zu den Configuration Items - im zweiten Schritt müssen zu den Configuration Items noch die Derived Objects (Object-Files, Bibliotheken, falls vorhanden gespeicherte Testdaten) ermittelt werden

### 3.1.4.9.2 Kompetenzfelder: Zusammenstellen von Komponenten zu Tests

#### Fähigkeiten/Fertigkeiten

- Derived Objects zu den Configuration Items identifizieren können
- Configuration Items einer Komponente identifizieren können
- Configuration Items in eine Testumgebung übertragen können
- die notwendigen Derived Objects in der Testumgebung erzeugen können
- Liste der bereitzustellenden Configuration Items erstellen können
- Art der Derived Objects den Testarten zuordnen können
- Derived Objects für die entsprechende Testart bereitstellen können

#### Wissen

- Aufbau und Struktur von Testumgebungen
- Testarten und dazugehörige Testmethoden
- Aufbau, Struktur und Abhängigkeiten von Derived Objects
- Aufbau und Struktur von Baselines
- Zusammenhang zwischen Testart und zu erzeugenden Derived Objects

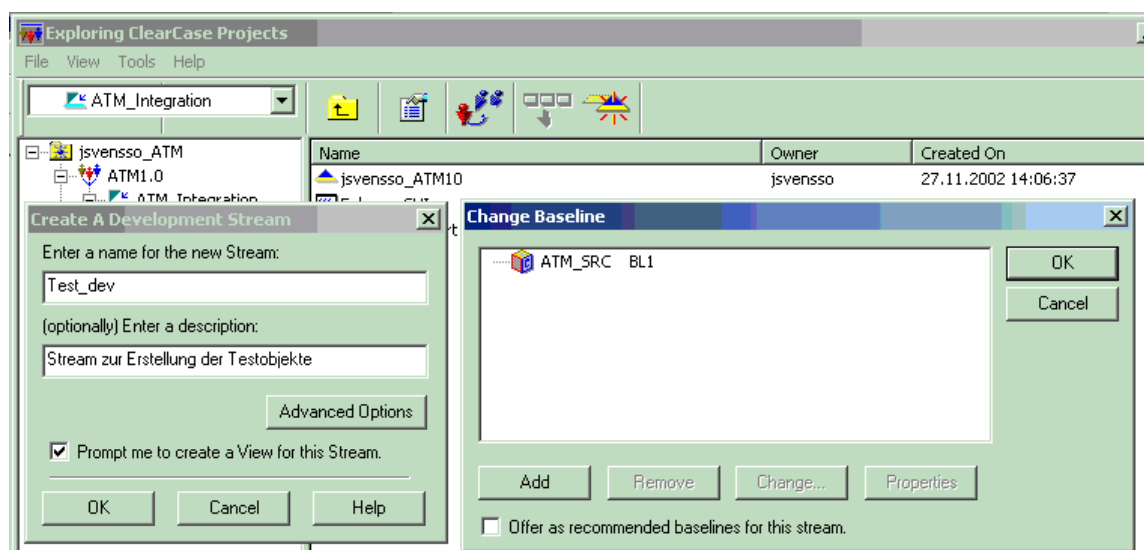
#### Werkzeuge/Methoden

- CMS Software
- Testumgebung
- Testwerkzeug

### 3.1.4.9.3 Beispiel: Zusammenstellen von Komponenten zu Tests

Für das Beispielprojekt wurde mithilfe des CMS eine spezielle Testkonfiguration angelegt. Anschließend wurden die Baselines der Configuration Items dieser Testkonfiguration zugeordnet. Dadurch waren die Configuration Items der Komponenten für den Test identifiziert und zusammengestellt. Hierfür wurden die im CMS abgelegten Zugehörigkeiten von Configuration Items zu Baselines genutzt.

Die Abbildung zeigt die Oberfläche des CMS beim Zusammenstellen der Configuration Items durch den Configuration Manager zu einem Integrationstest.



### 3.1.4.10 Einpflegen von Testergebnissen (Integrationstest)

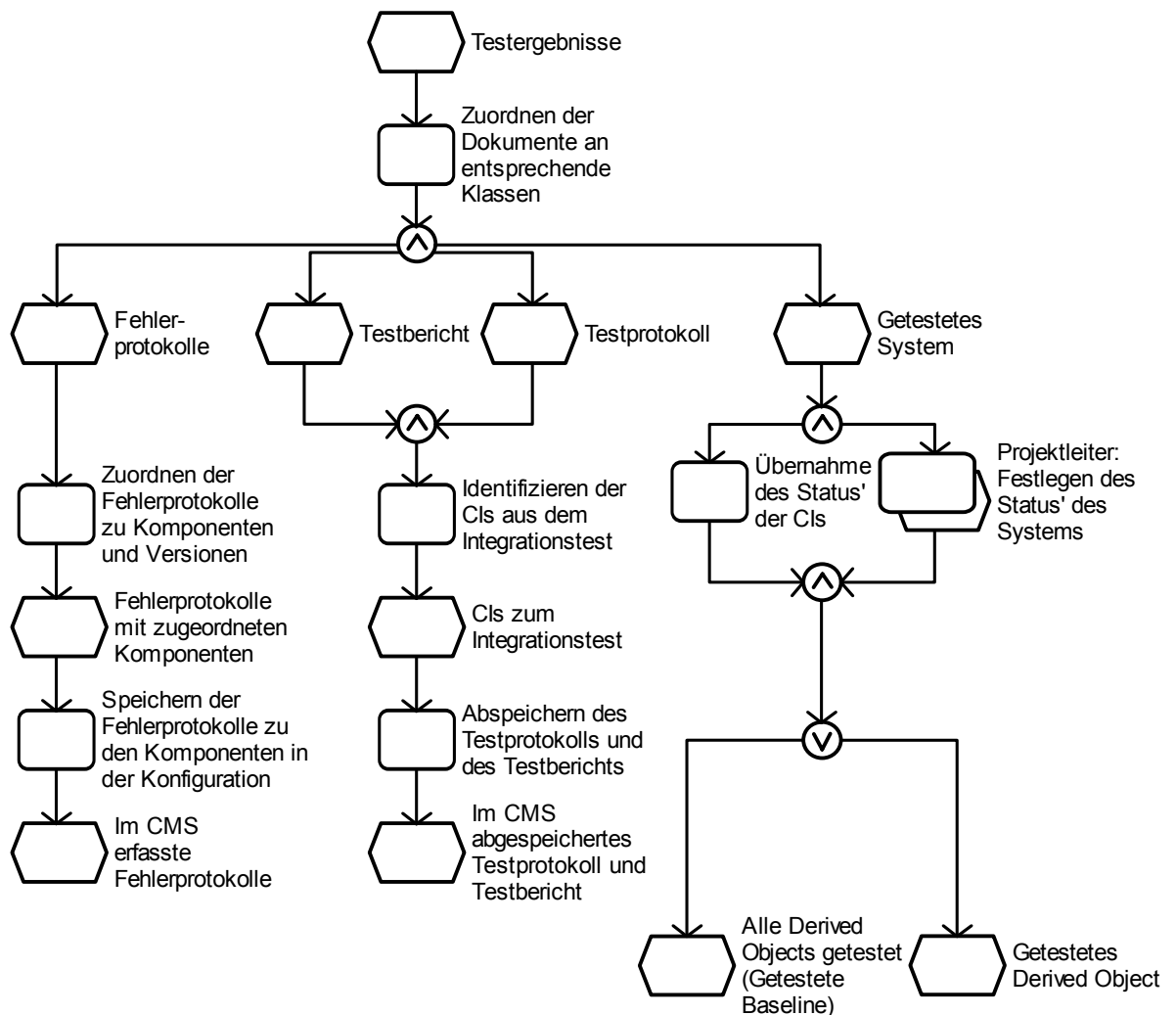


Abbildung 15: Einpflegen von Testergebnissen (Integrationstest).

In diesem Teilprozess sind die Tätigkeiten beschrieben, die der IT Configuration Coordinator beim Einpflegen der Testergebnisse aus dem Integrationstest durchführen muss. Als Eingangsdocuments aus dem Integrationstest, den der IT Test Coordinator durchführt, werden Fehlerprotokolle, das Testprotokoll und der Testbericht zur Configuration der Komponenten übernommen. Für die Fehlerreports wird eine Zuordnung auf Configuration-Item-Ebene erstellt. Zusätzlich wird der Status aus dem Testbericht jedem Configuration Item zugeordnet und im CMS abgespeichert.

Falls kein CMS zur Verfügung steht, müssen die Stati der Configuration Items im Konfigurationsmanagement-Plan verwaltet werden.

#### 3.1.4.10.1 Tätigkeiten: Einpflegen von Testergebnissen (Integrationstest)

- Zuordnen der Fehlerprotokolle zu Configuration Items - die beim Integrationstest erstellten Fehlerprotokolle werden den entsprechenden Versionen der Configuration

Items zugeordnet; die Configuration Items können dabei Versionen einzelner Builds sein.

- Speichern der Fehlerprotokolle zu den Configuration Items im CMS - zur Dokumentation des Integrationstests werden die Fehlerprotokolle inklusive der Zuordnung zu den Versionen der Configuration Items im CMS abgespeichert; damit können die Fehlerprotokolle zu den Versionen der Configuration Items zur Fehlerverfolgung nachträglich eingesehen werden
- Identifizieren der Configuration Items aus dem Integrationstest - der Testbericht und das Testprotokoll müssen den Versionen der Builds zugeordnet werden; dazu müssen die entsprechenden Configuration Items der Builds identifiziert werden
- Abspeichern des Testprotokolls und des Testberichts - nachdem die Configuration Items der Builds aus dem Integrationstest identifiziert wurden, müssen der Testbericht und das Testprotokoll im CMS abgespeichert werden; zusätzlich müssen die Beziehungen zwischen Testbericht und Testprotokoll zu den Versionen der Komponenten im CMS erfasst werden
- Übernahme des Status' der CIs - der Status der Builds aus dem Testbericht muss den Versionen der Configuration Items der Builds zugewiesen und im CMS gespeichert werden

#### **3.1.4.10.2 Kompetenzfelder: Einpflegen von Testergebnissen (Integrationstest)**

##### *Fähigkeiten/Fertigkeiten*

- Fehlerprotokolle den Configuration Items zuordnen können
- Configuration Items zu einem Testbericht identifizieren können
- Configuration Items zu einem Testprotokoll identifizieren können
- Configuration Items zu einem Fehlerprotokoll identifizieren können
- Testbericht im CMS abspeichern können
- Testprotokolle im CMS abspeichern können
- Status von Configuration Items ändern können
- Status vom getesteten System ändern können
- Status von Change Requests ändern können

##### *Wissen*

- Aufbau von Konfigurationen mit Configuration Items
- Aufbau und Inhalt eines Testberichts
- Aufbau und Inhalt eines Testprotokolls
- Aufbau und Inhalt eines Fehlerprotokolls
- Einpflegen der Stati von Configuration Items in den CR

##### *Werkzeuge/Methoden*

- CMS Software
- CCMS

#### **3.1.4.10.3 Beispiel: Einpflegen von Testergebnissen (Integrationstest)**

Siehe Teilprozess 3.1.4.5

### 3.1.4.11 Zusammenstellen von getesteten Derived Objects zum Gesamtsystem

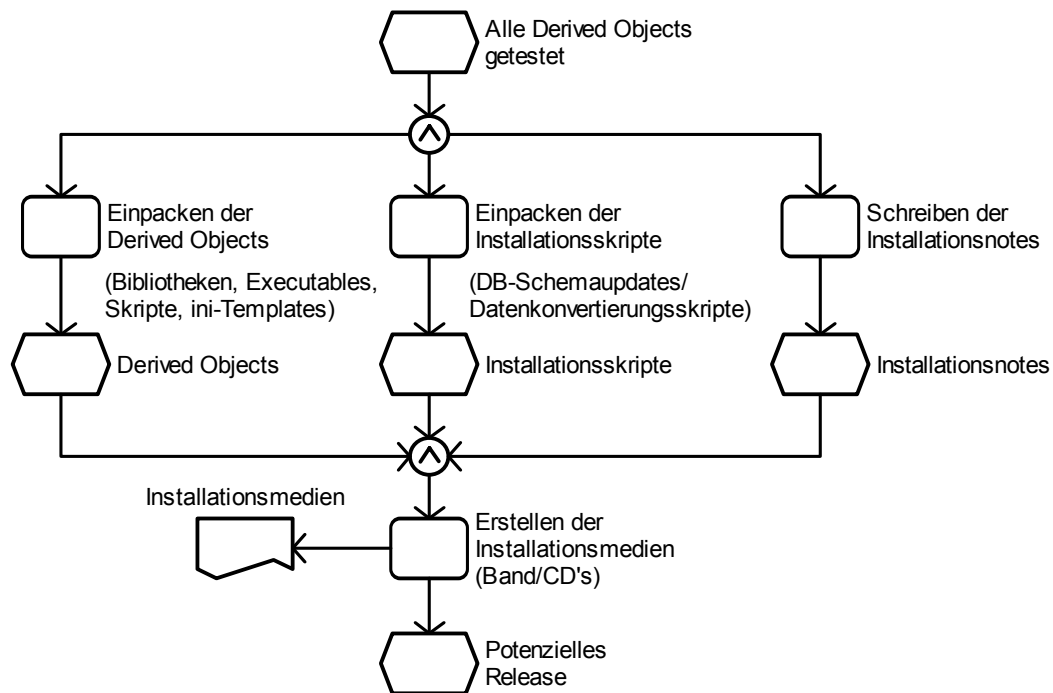


Abbildung 16: Zusammenstellen von getesteten Derived Objects zum Gesamtsystem.

Der Teilprozess enthält vier Tätigkeiten, die zum Erstellen eines Installationsmediums durchgeführt werden müssen. Im ersten Schritt werden alle Derived Objects (Bibliotheken, Exe-Dateien, Ini-Dateien, Skripte, Templates), die zum Release gehören, eingepackt. Im zweiten Schritt werden die Installationsskripte (Updates der Datenbankschema, Datenkonvertierungsskripte ...) zusammengestellt und eingepackt. Zum Gesamtsystem gehören des Weiteren die Installationsnotes, in denen der Prozess der Installation beschrieben ist.

#### 3.1.4.11.1 Tätigkeiten: Zusammenstellen von getesteten Derived Objects zum Gesamtsystem

- Einpacken der Derived Objects - alle Derived Objects zu dem Gesamtsystem werden durch den IT Configuration Coordinator zusammengestellt und dann eingepackt; Beispiele für Derived Objects sind Bibliotheken und Executables von Configuration Items oder zum Gesamtsystem gehörende Skripte, Ini-Files und Templates
- Einpacken der Installationsskripte - je nach Umfang und Art (Bug-Fix, Patches oder Release) ist eine Reihe von Installationsskripten zu erstellen; Beispiele hierfür sind Installationsskripte, die Änderungen am Datenbankschema durchführen, Anpassungsskripte der Daten entsprechend der Datenbankänderung
- Schreiben der Installationsnotes - in den Installationsnotes werden die einzelnen Schritte der Installation von Bug-Fixes, Patches oder neuen Releases beschrieben
- Erstellen der Installationsmedien - alle eingepackten Derived Objects, Installationsskripte und die Installationsnotes müssen auf das Installationsmedium übertragen werden

### **3.1.4.11.2 Kompetenzfelder: Zusammenstellen von getesteten Derived Objects zum Gesamtsystem**

#### *Fähigkeiten/Fertigkeiten*

- Installationsskripte erstellen können
- Installationsskripte einpacken können
- Dervied Objects zum Gesamtsystem zusammenstellen können
- Datenbankschema-Patches erstellen können
- Datenbank-Patches erstellen können
- Installationsnotes schreiben können
- Installationsbänder erstellen können
- Installations-CD erstellen können

#### *Wissen*

- Aufbau und Bestandteile eines Gesamtsystems
- Aufbau und Struktur von Installationsskripten
- Datenbankkenntnisse
- Datenbankschema-Kenntnisse
- Installationsmedien
- Aufbau und Inhalt von Installationsnotes

#### *Werkzeuge/Methoden*

- Text Editor
- Packprogramme (WinZip, WinRar, tar, ZIP)
- Setup-Programme (Wise)
- CD-/DVD-Brennprogramme

### **3.1.4.11.3 Beispiel: Zusammenstellen von getesteten Derived Objects zum Gesamtsystem**

Im Beispiel wurden alle Komponenten des neuen Course Registration System zu dem Gesamtsystem zusammengefügt. Dabei wurden nicht mehr Baselines genutzt, die auf Sourcen liegen, sondern die Baselines auf den Derived Objects. Weiterhin wurden zur Übernahme der Daten aus der alten Datenbank Migrationsskripte vom Datenbank-administrator geschrieben und zum Gesamtsystem hinzugefügt. Aus dem Gesamtsystem, den Installationsskripten, Migrationsskripten wurde dann eine Installations-CD für das Course Registration System erstellt. Dazu wurde als CD-Brennprogramm „Nero 5.5“ verwendet. Nachdem die Installations-CD erstellt war, wurde auf einem Testsystem die gesamte Installation durchgeführt und überprüft.

### 3.1.4.12 Vorschlägen von Änderungen

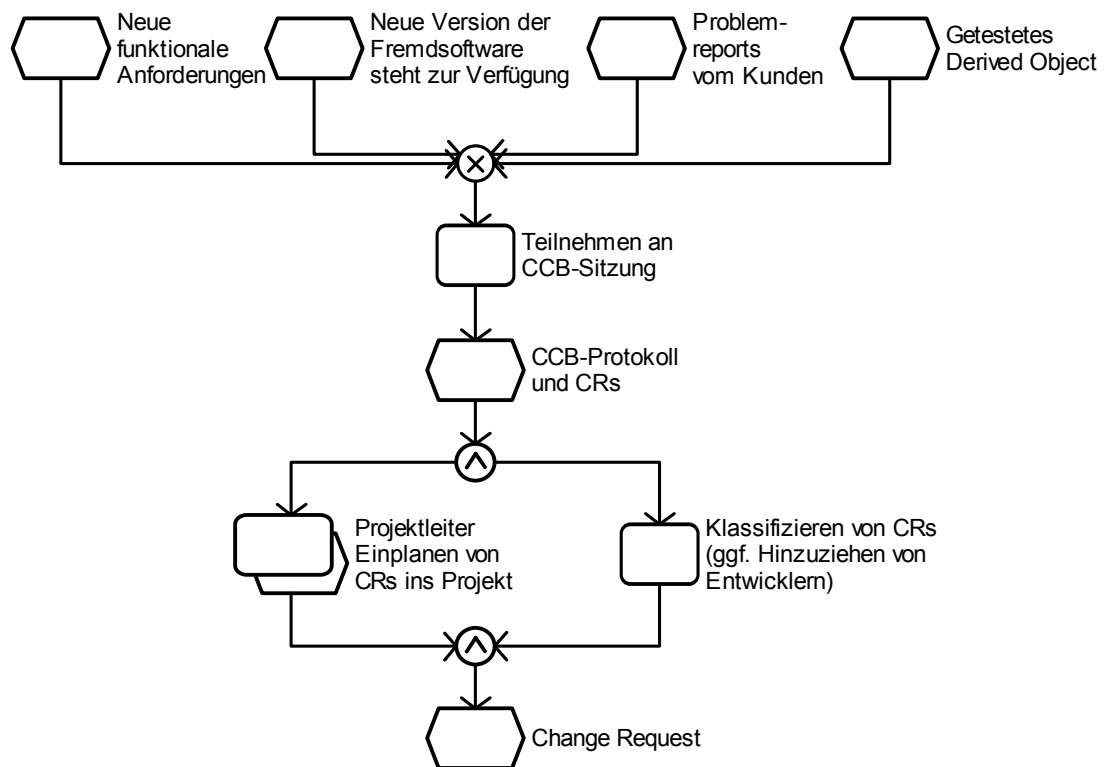


Abbildung 17: Vorschlägen von Änderungen.

Dieser Teilprozess enthält die Tätigkeiten des IT Configuration Coordinator im Zusammenhang mit Change Requests im Change Control Board (CCB). Im Change Control Board wird für neue (funktionale) Anforderungen, neue Versionen von Fremdsoftware, Problemreports oder freigegebene Baselines die Umsetzung in Change Requests diskutiert und beschlossen. Die neu angelegten Change Requests werden vom IT Configuration Coordinator mit den entsprechenden Entwicklern in Bezug auf die betroffenen Configuration Items klassifiziert. Parallel dazu plant der Projektleiter die Change Requests im Projekt ein.

#### 3.1.4.12.1 Tätigkeiten: Vorschlägen von Änderungen

- **Teilnehmen an Change Control Board-Sitzung** - der IT Configuration Coordinator nimmt an den Sitzungen des Change Control Boards teil; im Change Control Board werden neue Anforderungen und Problemreports diskutiert und über die Umsetzung mittels Change Request abgestimmt; des Weiteren wird über die Einführung neuer Versionen von Fremdsoftware oder neuer Baselines diskutiert und die Umsetzung mittels Change Request entschieden
- **Klassifizieren von CR** - die neuen Change Requests aus dem Change Control Board werden vom IT Configuration Coordinator unter Hinzuziehung der entsprechenden Entwickler bzgl. der Relevanz zu den Configuration Items klassifiziert; dadurch werden die entsprechenden Configuration Items, die zum Change Request gehören, identifiziert



### **3.1.4.12.2 Kompetenzfelder: Vorschlagen von Änderungen**

#### *Fähigkeiten/Fertigkeiten*

- Change Requests den Configuration Items zuordnen können
- neue Versionen von Fremdsoftware zu Projekten zuordnen können
- Change Requests den Projekten zuordnen können
- Anforderungen/Change Requests diskutieren können
- Versionsänderungen von Fremdsoftware kommunizieren können, CCB-Sitzungen vorbereiten können

#### *Wissen*

- Aufbau und Struktur von Change Requests
- Projektübersicht
- Projektstände
- genutzte Fremdsoftware in Projekten
- Aufbau und Funktionsweise des Change Control Board

#### *Werkzeuge/Methoden*

- CMS-Werkzeug
- CCMS-Werkzeug

### **3.1.4.12.3 Beispiel: Vorschlagen von Änderungen**

Die Abbildung zeigt ein Trouble Tickets System, in dem eine neue Anforderung an einen Web Shop als Änderungsanforderung angelegt wird. Neben der Liste der Trouble Tickets ist eine Detailansicht abgebildet, welche die einzelnen Attribute eines Änderungsvorschlags enthält.

Innerhalb der Change Controll Board-Sitzung wurde der Change Request mit der funktionalen Erweiterung des Course Registration System als einer von mehreren Change Requests vom Projektleiter vorgestellt. Der Change Request wurde dann von dem Change Control Board in Bezug auf seine Relevanz für das aktuelle und die geplanten Releases hin untersucht. Dabei wurden als Faktoren der Zeitplan des Releases und der Zeitbedarf für die Umsetzung der Anforderung, die benötigten Ressourcen gegenüber den zur Verfügung stehenden Ressourcen, die technischen Schwierigkeiten der Umsetzung und die Priorität der Anforderung betrachtet. In der Entscheidung des Change Control Boards wurde der Change Request als relevant angesehen, aus Zeitgründen aber in das nachfolgende Release verschoben.

The screenshot displays the Rational ClearQuest application window. The title bar reads "Rational ClearQuest - [CLSIC : Rational Demo (All Change Requests (All\_Requests))]". The menu bar includes File, Edit, View, Actions, Query, Window, and Help. The toolbar contains icons for various actions like Run Query, New Defect, and Print.

On the left, a tree view shows the workspace structure, including Queries, Charts, Reports, and various project-specific queries. The "All Change Requests" query is selected.

The main area shows a table of change requests with the following data:

id	record_type	State	
CLSIC00000036	EnhancementRequest	Submitted	Need to notify customer via email when order ships
CLSIC00000074	Defect	Submitted	Leak search complete
CLSIC00000092	Defect	Submitted	Wrong error for incorrect password
CLSIC00000073	Defect	Submitted	Starting main

Below the table, the "Result set" tab is active, showing a detailed view of the selected record (ID: CLSIC00000036). The view includes tabs for History, Requirements, ClearCase, Unified Change Management, Main, Analysis, Customer, Notes, Resolution, and Attachments. The "Main" tab is selected, displaying the following information:

- ID:** CLSIC00000036 **State:** Submitted
- Headline:** Need to notify customer via email when order ships
- Suite Project:** Webshop
- Customer Priority:** Medium
- Description:** It would be nice if ClassicsCD.com would send an e-mail notification when each order is shipped. The notice should include method of shipping and tracking number.

Buttons for Apply, Revert, Print Record, and Actions are visible on the right side of the detailed view.

In der Abbildung sind neben einer ID, die durch das Trouble Tickets System vergeben wird, ein Titel, das betroffene Projekt, eine Priorität und eine Beschreibung für den Änderungsvorschlag des IT Configuration Coordinator anzugeben.

### 3.1.4.13 Einpflegen von Testergebnissen (Systemtest)

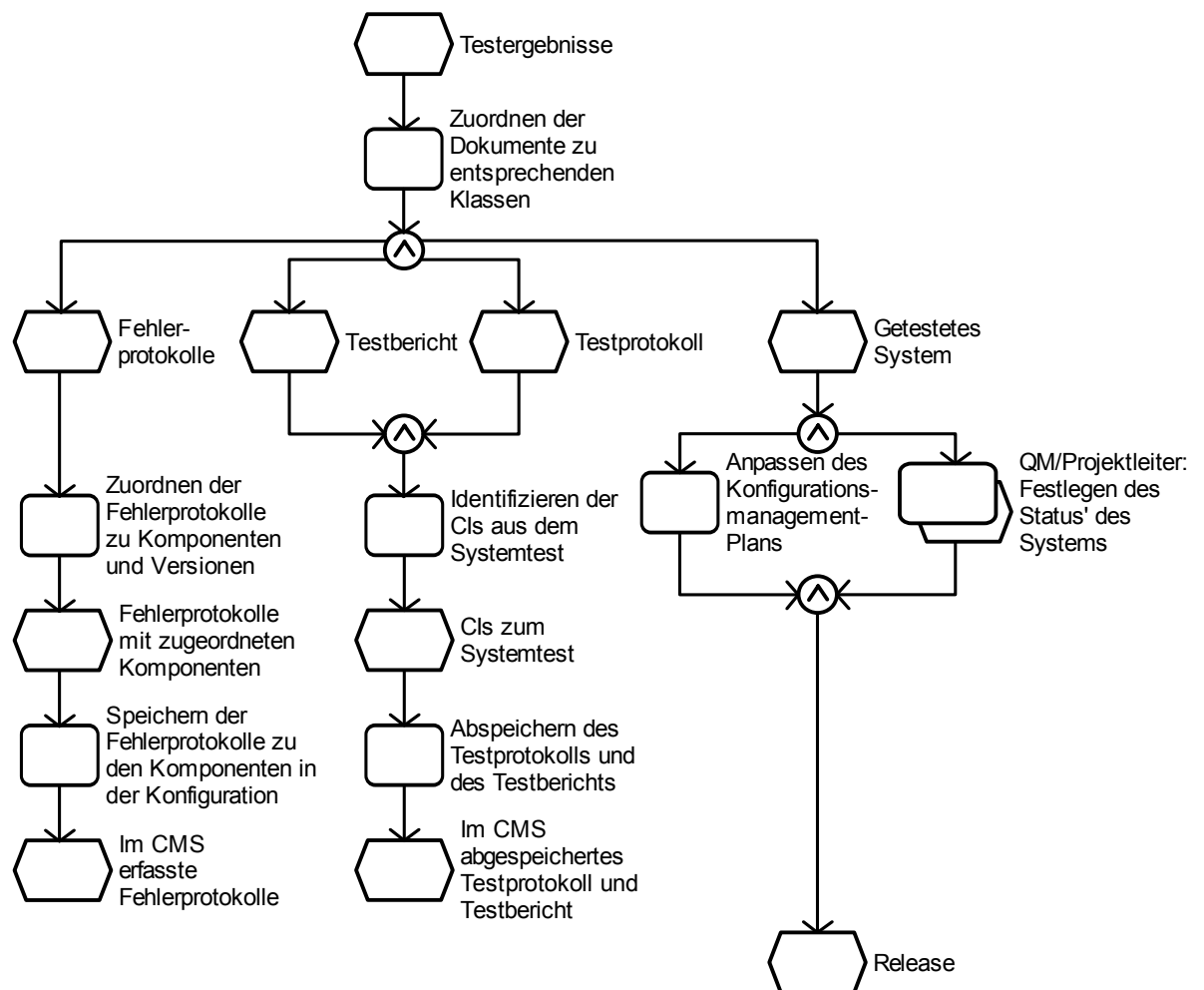


Abbildung 2: Einpflegen von Testergebnissen (Systemtest).

In diesem Teilprozess sind die Tätigkeiten beschrieben, die der IT Configuration Coordinator beim Einpflegen der Testergebnisse aus dem Systemtest durchführen muss. Als Eingangsdokumente aus dem Systemtest, den der IT Test Coordinator durchführt, werden Fehlerprotokolle, das Testprotokoll und der Testbericht zur Konfiguration der Komponenten übernommen. Für den Systemtest können die Testdokumente nur den Configuration Items auf Komponentenebene zugeordnet werden. Zusätzlich werden die Stati der Baselines und des Releases aus dem Testbericht für jedes Configuration Item im CMS abgespeichert.

#### 3.1.4.13.1 Tätigkeiten: Einpflegen von Testergebnissen (Systemtest)

- Zuordnen der Fehlerprotokolle zu Komponenten und Versionen - beim Systemtest werden die Use Cases aus der Anforderungsdefinition überprüft, dabei werden die Funktionalitäten auf Komponentenebene überprüft; deshalb werden die Fehlerprotokolle funktionalitäts-/komponentenbezogen erstellt; eine Zuordnung auf andere Configuration Items durch den IT Configuration Coordinator ist deshalb nicht möglich; für die Configuration Items der Komponenten muss dann noch die verwendete Version identifiziert

werden, bevor der Fehlerreport im CMS mit der Beziehung zu der Version der verursachenden Komponente eingepflegt wird.

- Speichern der Fehlerprotokolle in der Konfiguration - neben der Speicherung der Fehlerprotokolle zu den Komponenten des Systemtests müssen die Fehlerprotokolle zur Konfiguration des Gesamtsystems gespeichert werden
- Identifizieren der Configuration Items aus dem Systemtest - der Testbericht und das Testprotokoll müssen den Versionen der Komponenten zugeordnet werden; dazu müssen die entsprechenden Configuration Items der Komponenten identifiziert werden
- Abspeichern des Testprotokolls und des Testberichts - nachdem die Configuration Items der Komponenten identifiziert wurden, müssen der Testbericht und das Testprotokoll im CMS abgespeichert werden; zusätzlich müssen die Beziehungen zur Version des Gesamtsystems im CMS erfasst werden
- Anpassen des Konfigurationsmanagement-Plans - der Konfigurationsmanagement-Plan muss in Abstimmung mit dem Projektleiter aktualisiert werden; dabei legt der Projektleiter den Status für die Baselines der Komponenten und des Gesamtleases fest; diese Stati werden im Konfigurationsmanagement-Plans festgehalten

#### **3.1.4.13.2 Kompetenzfelder: Einpflegen von Testergebnissen (Systemtest)**

##### *Fähigkeiten/Fertigkeiten*

- Konfigurationsmanagement-Plan ändern können
- Fehlerprotokolle den Configuration Items zuordnen können
- Configuration Items zu einem Testbericht identifizieren können
- Configuration Items zu einem Testprotokoll identifizieren können
- Configuration Items zu einem Fehlerprotokoll identifizieren können
- Testbericht im CMS abspeichern können
- Testprotokolle im CMS abspeichern können
- Status von Configuration Items ändern können
- Status des getesteten Systems ändern können
- Status von Change Requests ändern können

##### *Wissen*

- Aufbau von Konfigurationen mit Configuration Items
- Aufbau und Inhalt eines Testberichts
- Aufbau und Inhalt eines Testprotokolls
- Aufbau und Inhalt eines Fehlerprotokolls
- Einpflegen der Stati von Configuration Items ins CR

##### *Werkzeuge/Methoden*

- CMS Software
- CCMS

#### **3.1.4.13.3 Beispiel: Einpflegen von Testergebnissen (Systemtest)**

Siehe Teilprozess 3.1.4.5