

Lehrstuhl für  
Informationstechnik im Maschinenwesen  
der Technischen Universität München

## **PDM-basierte Überwachung komplexer Entwicklungsprojekte**

Ralf Maximilian Jungkunz

Vollständiger Abdruck der von der Fakultät für Maschinenwesen der Technischen  
Universität München zur Erlangung des akademischen Grades eines

**Doktor-Ingenieurs (Dr.-Ing.)**

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing. Udo Lindemann

Prüfer der Dissertation: 1. Univ.-Prof. Dr.-Ing. Klaus Bender  
2. Univ.-Prof. Dr.-Ing. Andreas Karcher,  
Universität der Bundeswehr München

Die Dissertation wurde am 26.01.05 bei der Technischen Universität München  
eingereicht und durch die Fakultät für Maschinenwesen am 04.04.2005 angenommen.



# Vorwort

Die Dissertation entstand im Rahmen einer Forschungs Kooperation zwischen dem Lehrstuhl für Informationstechnik im Maschinenwesen (*itm*) der Technischen Universität München und der European Aerospace Defence and Space Company (EADS).

Bei meinem Doktorvater, Herrn Prof. Dr.-Ing. Klaus Bender, bedanke ich mich recht herzlich für seine zahlreichen Anregungen und für die Gewährung des nötigen Freiraumes. Darüber hinaus habe ich mich sehr für die Übernahme des Co-Referats durch Herrn Prof. Dr.-Ing. Andreas Karcher vom Institut für Angewandte Systemwissenschaften und Wirtschaftsinformatik an der Universität der Bundeswehr München gefreut. Außerdem danke ich Herrn Prof. Dr.-Ing. Udo Lindemann für die Übernahme des Prüfungsvorsitzes.

Diese Arbeit basiert auf Erkenntnissen, die ich während meiner vierjährigen Projektarbeit bei dem Luft- und Raumfahrtunternehmen EADS sammeln konnte. Besonders danke ich Herrn Dr.-Ing. Jörg Wirtz und Herrn Dipl.-Ing. Josef Vilsmeier für die zahlreichen Diskussionen, die wesentlich zum Reifen des Dissertationsthemas beitrugen.

Schließlich bedanke ich mich bei meiner Familie und Freunden, die mich auf unterschiedliche Art und Weise auf dem Weg zur Promotion unterstützten.

München, im Mai 2005

Ralf Maximilian Jungkunz



# Inhaltsverzeichnis

<b>Abbildungen .....</b>	<b>v</b>
<b>Tabellen.....</b>	<b>ix</b>
<b>Abkürzungen.....</b>	<b>xii</b>
<b>Glossar .....</b>	<b>xiv</b>
<b>1. Einleitung .....</b>	<b>1</b>
1.1 Hintergrund und Aufgabenstellung.....	1
1.2 Aufbau und Vorgehensweise .....	2
<b>2. Problemanalyse.....</b>	<b>5</b>
2.1 Analyse derzeitiger Entwicklungsprojekte.....	6
2.1.1 Begriffsklärung Entwicklungsprojekt.....	6
2.1.2 Stellenwert und Veränderungen von Entwicklungsprojekten .....	6
2.1.3 Fazit - die Koordination der Entwicklungsprojekte ist schwierig .....	8
2.2 Analyse derzeitiger Produkte .....	8
2.2.1 Begriffsklärung Produkt .....	8
2.2.2 Stellenwert und Veränderungen von Produkten .....	9
2.2.3 Fazit - Überfluss an Produktdaten .....	11
2.3 Analyse derzeitiger IT-Engineeringsysteme .....	12
2.3.1 Begriffsklärung IT-Engineeringsystem .....	12
2.3.2 Stellenwert und Veränderungen der IT-Engineeringsysteme .....	12
2.3.3 Fazit - die Werkzeuglandschaft ist kaum überschaubar .....	13
2.4 Das Überwachungsdefizit im Engineering.....	13
2.4.1 Begriffsklärung Engineeringüberwachung.....	14
2.4.2 Stellenwert und Aufgaben der Entwicklungsprojektüberwachung .....	15
2.4.3 Fazit - mangelnde Verlässlichkeit bisheriger Informationsressourcen.....	21

---

2.5	Resümee .....	22
2.5.1	Allgemeine Anforderungen an ein Überwachungskonzept .....	23
2.5.2	Allgemeiner Lösungsansatz - Transformationskonzept für IT-Informationen.....	25
<b>3.</b>	<b>Analyse bestehender Ansätze .....</b>	<b>27</b>
3.1	Analyse überwachungsrelevanter Standards .....	28
3.1.1	DIN 69904 - Elemente und Strukturen von Projektmanagementsystemen .....	28
3.1.2	ISO 10006 - Guidelines to quality in project management .....	29
3.1.3	DIN EN 13290 - Raumfahrt Projektmanagement.....	30
3.1.4	Fazit - keine Hinweise, was die traditionellen Überwachungsressourcen ersetzt .	30
3.2	Moderne Informationsressourcen im Engineering .....	31
3.2.1	Produktdatenmanagement Systeme (PDM).....	31
3.2.2	Team Data Management Systeme (TDM).....	32
3.2.3	Dokumenten-/Contentmanagement Systeme (DMS/CMS).....	33
3.2.4	Produktionsplanungs- und Steuerungssysteme (PPS) .....	34
3.2.5	Fazit - PDM enthält die geeignetsten Überwachungsressourcen.....	35
3.3	Ansätze zur überwachungsgerechten Aufbereitung von Informationen .....	35
3.3.1	SFB 336 - Teilprojekt I3 .....	36
3.3.2	TFB 29 - Teilprojekt T2.....	36
3.3.3	BMPF-Projekt Progress .....	38
3.3.4	Projektmanagementsysteme (PMS) .....	41
3.3.5	Entscheidungsunterstützende Systeme (EUS).....	42
3.3.6	Fazit - es fehlt ein PDM-übergreifender Informationsspeicher .....	44
3.4	Neutrale Datenmodelle mit PDM-Prinzip .....	45
3.4.1	PDM Enablers und die ISO 10303 .....	45
3.4.2	PDTnet, PDM Schema und das NCDM .....	46
3.4.3	Fazit - STEP bietet das geeignetste Referenzmodell .....	47
3.5	Resümee .....	47
3.5.1	Bewertungszusammenfassung .....	48
3.5.2	Konkrete Anforderungen an ein Überwachungskonzept.....	49
3.5.3	Konkreter Lösungsansatz - Transformationskonzept für PDM-Informationen.....	52

---

<b>4. Systematische Transformation von PDM-Informationen.....</b>	<b>55</b>
4.1 Metamodell der Transformationsunterstützung .....	56
4.2 Entwicklungsstrategie für wieder verwendbare Transformationsmuster.....	58
4.3 STEP-basierte Transformationsmuster .....	63
4.3.1 Transformationsmuster zur Sachfortschrittsüberwachung .....	63
4.3.2 Transformationsmuster zur Terminüberwachung.....	75
4.3.3 Transformationsmuster zur Qualitätsüberwachung.....	82
4.3.4 Transformationsmuster zur Belastungsüberwachung.....	89
4.3.5 Fazit - PDM-neutrale Konzeptvorlagen eines Überwachungswerkzeuges.....	91
4.4 Strategie zur Anpassung der Transformationsmuster .....	91
4.5 Systematischer Konfigurations- und Änderungsmanagementansatz.....	95
4.6 Intuitives Zugriffsprinzip .....	100
4.7 Leitlinien zur Umsetzung der Musterinstanzen .....	103
4.8 Resümee .....	104
<b>5. Die Transformationsbibliothek.....</b>	<b>105</b>
5.1 Beschreibung der Transformationsbibliothek .....	106
5.1.1 Systemarchitektur und Technologieauswahl .....	106
5.1.2 Datenbankentwurf.....	107
5.1.3 Benutzerschnittstelle (GUI).....	108
5.2 Anwendungsbeispiel Flugzeugentwicklung.....	109
5.3 Nutzung der Transformationsbibliothek .....	110
5.3.1 Beispiel: Meilensteintrendanalyse .....	110
5.3.2 Beispiel: Sachfortschrittsdiagramm.....	114
5.3.3 Beispiel: Gantt-Diagramm.....	115
5.3.4 Resultierender Konzeptentwurf des PDM-basierten Überwachungswerkzeuges	117
5.4 Fazit - systematische Unterstützung der PDM-Transformation.....	118
<b>6. Das PDM-basierte Überwachungswerkzeug .....</b>	<b>119</b>
6.1 Beschreibung des Überwachungswerkzeuges.....	120
6.1.1 Systemarchitektur und Technologieauswahl .....	120
6.1.2 Datenbankentwurf.....	121
6.1.3 Benutzerschnittstelle (GUI).....	122

---

6.2 Anwendungsbeispiel Flugzeugentwicklungsprojekt.....	123
6.3 Nutzung des Überwachungswerkzeuges.....	124
6.3.1 Beispiel: Gantt-Diagramm.....	124
6.3.2 Beispiel: Integriertes Meilenstein-Sachfortschrittsdiagramm .....	125
6.4 Fazit - verlässliche und aufwandsarme Engineeringüberwachung .....	126
<b>7. Resümee.....</b>	<b>129</b>
7.1 Zusammenfassung.....	129
7.2 Ausblick .....	131
<b>Anhang.....</b>	<b>133</b>
Anhang A: Transformationsmuster für Skalierungselemente .....	135
Anhang B: Transformationsmuster für Sachfortschrittsdiagramme .....	139
Anhang C: Transformationsmuster für Netzpläne .....	141
Anhang D: Transformationsmuster für Gantt-Diagramme .....	147
Anhang E: Transformationsmuster für Meilensteintrenddiagramme.....	153
Anhang F: Transformationsmuster für die Änderungsquote.....	155
Anhang G: Transformationsmuster für das Gewicht .....	157
Anhang H: Transformationsmuster für Belastungsdiagramme.....	159
Anhang I: Data Dictionary des Konfigurations- und Änderungsmanagementansatzes .....	161
<b>Literatur.....</b>	<b>163</b>



# Abbildungen

Abb. 1-1: Fokus der Arbeit .....	2
Abb. 1-2: Gesamtaufbau und Vorgehensweise.....	3
Abb. 2-1: Aufbau und Vorgehensweise Kapitel 2.....	5
Abb. 2-2: Abgrenzung von Forschungs- und Entwicklungsprojekten .....	6
Abb. 2-3: Anstieg der Variantenzahl .....	10
Abb. 2-4: Zunehmender Software- und Elektronikanteil .....	11
Abb. 2-5: Der Projektmanagement-Regelkreis.....	14
Abb. 2-6: Die reine Projektorganisation .....	16
Abb. 2-7: Die Einfluss-Projektorganisation.....	16
Abb. 2-8: Die Matrix-Projektorganisation.....	16
Abb. 2-9: Die Auftrags-Projektorganisation.....	16
Abb. 2-10: Entwicklungsprojektcontrolling Input und Output.....	17
Abb. 2-11: Sachfortschrittsdiagramm.....	18
Abb. 2-12: Netzplan.....	18
Abb. 2-13: Gantt-Diagramm.....	19
Abb. 2-14: Meilensteintrenddiagramm.....	19
Abb. 2-15: Belastungsdiagramm .....	20
Abb. 2-16: Kostendiagramm.....	20
Abb. 2-17: Mangelnde Verlässlichkeit traditioneller Controllingressourcen.....	22
Abb. 2-18: Transformationskonzept für Erzeuger- und Verwaltungssystemdaten .....	26
Abb. 3-1: Aufbau und Vorgehensweise Kapitel 3 .....	27
Abb. 3-2: Modell eines Projektmanagementsystems.....	29
Abb. 3-3: Produktstruktur eines Faserverbundbauteils in einem PDM-System.....	32
Abb. 3-4: Produktstruktur eines Faserverbundbauteils in einem TDM-System .....	33

---

Abb. 3-5: Aufbau des TFB 29-T2 Projektmonitors.....	37
Abb. 3-6: Projektcockpit.....	40
Abb. 3-7: Berechnung der geschätzten Forschungs- und Entwicklungsdauer .....	41
Abb. 3-8: Gantt-Diagramm in einem PMS.....	42
Abb. 3-9: Architektur eines EUS .....	43
Abb. 3-10: Produktstrukturaufbau nach dem STEP-AP214 Prinzip .....	46
Abb. 3-11: Die Lücken bei der Transformation von PDM-Informationen .....	49
Abb. 3-12: Systematische Transformation von PDM-Informationen .....	53
Abb. 4-1: Aufbau und Vorgehensweise Kapitel 4.....	55
Abb. 4-2: Aufbau der Transformationsunterstützung.....	58
Abb. 4-3: Modell des Transformationsprozesses .....	62
Abb. 4-4: Entwicklungsstrategie für Transformationsmuster .....	62
Abb. 4-5: Beispielhafte Kennlinienelemente für Sachfortschrittsdiagramme.....	64
Abb. 4-6: Datenbankmodell für das Sachfortschrittsbeispiel.....	69
Abb. 4-7: Beispielhafte Kennlinienelemente für Netzpläne.....	70
Abb. 4-8: Datenbankmodell für einen Netzplan.....	75
Abb. 4-9: Beispielhafte Kennlinienelemente für Gantt-Diagramme.....	76
Abb. 4-10: Datenbankmodell für ein Gantt-Diagramm.....	79
Abb. 4-11: Beispielhafte Kennlinienelemente für Meilensteintrenddiagramme.....	80
Abb. 4-12: Beispielhaftes Datenbankmodell für ein Meilensteintrenddiagramm.....	82
Abb. 4-13: Beispielhafte Kennlinienelemente für die Änderungsquote.....	83
Abb. 4-14: Datenbankmodell für eine Änderungsquote.....	85
Abb. 4-15: Beispielhafte Kennlinienelemente für das Gewicht .....	86
Abb. 4-16: Datenbankmodell für das Gewicht .....	88
Abb. 4-17: Beispielhafte Kennlinienelemente für Belastungsdiagramme .....	89
Abb. 4-18: Datenbankmodell für ein Belastungsdiagramm .....	91
Abb. 4-19: Strategie zur Anpassung der Transformationsmuster .....	94
Abb. 4-20: Ausschnitt einer Musterinstanz für ein Meilensteintrenddiagramm .....	95
Abb. 4-21: Das Modell des Verwaltungsprinzips.....	99
Abb. 4-22: Das Modell des Zugriffsprinzips.....	102

---

Abb. 5-1: Aufbau und Vorgehensweise Kapitel 5 .....	105
Abb. 5-2: Architektur und Technologieauswahl des Bibliotheksprototypen .....	106
Abb. 5-3: Transformation zwischen objektorientierten und relationalen Paradigma.	107
Abb. 5-4: Relationaler Datenbankentwurf in MS-Access .....	108
Abb. 5-5: Die Benutzerschnittstelle der Transformationsbibliothek.....	109
Abb. 5-6: Dokumentation des Änderungshintergrundes .....	111
Abb. 5-7: Spezifikation für eine Meilensteintrendanalyse .....	112
Abb. 5-8: Datenbankmodell des PDM-basierten Überwachungswerkzeuges.....	117
Abb. 6-1: Aufbau und Vorgehensweise Kapitel 6.....	119
Abb. 6-2: Architektur und Technologieauswahl des Überwachungswerkzeuges .....	121
Abb. 6-3: Relationaler Datenbankentwurf in MS-Access .....	122
Abb. 6-4: Die Benutzerschnittstelle des PDM-basierten Überwachungsprototyps....	123
Abb. 6-5: Balkendiagramm eines Flugzeugentwicklungsprojektes .....	125
Abb. 6-6: Integriertes Meilenstein-Sachfortschrittsdiagramm .....	126



# Tabellen

Tab. 4-1: Ausschnitt präsentierender Elemente von Sachfortschrittsdiagrammen.....	67
Tab. 4-2: Ausschnitt der Skalierungselemente und STEP .....	68
Tab. 4-3: Ausschnitt präsentierender Elemente von Netzplänen und STEP .....	73
Tab. 4-4: Ausschnitt präsentierender Elemente von Gantt-Diagrammen und STEP ...	77
Tab. 4-5: Ausschnitt präsentierender Elemente von Meilensteintrenddiagrammen.....	81
Tab. 4-6: Ausschnitt präsentierender Elemente der Änderungsquote und STEP .....	84
Tab. 4-7: Präsentierende Elemente des Gewichtes und STEP .....	87
Tab. 4-8: Ausschnitt präsentierender Elemente von Belastungsdiagrammen .....	90
Tab. 4-9: Data Dictionary Ausschnitt .....	100
Tab. 5-1: Musterinstanz einer Meilensteintrendanalyse .....	113
Tab. 5-2: Musterinstanz eines Sachfortschrittsdiagramms .....	115
Tab. 5-3: Musterinstanz eines Gantt-Diagramms .....	116
Tab. A-1: Transformationsmuster für Skalierungselemente (Teil 1) .....	135
Tab. A-2: Transformationsmuster für Skalierungselemente (Teil 2) .....	136
Tab. A-3: Transformationsmuster für Skalierungselemente (Teil 3) .....	137
Tab. A-4: Transformationsmuster für Sachfortschrittsdiagramme (Teil 1).....	139
Tab. A-5: Transformationsmuster für Sachfortschrittsdiagramme (Teil 2).....	140
Tab. A-6: Transformationsmuster für Netzpläne (Teil 1).....	141
Tab. A-7: Transformationsmuster für Netzpläne (Teil 2).....	142
Tab. A-8: Transformationsmuster für Netzpläne (Teil 3).....	143
Tab. A-9: Transformationsmuster für Netzpläne (Teil 4).....	144
Tab. A-10: Transformationsmuster für Netzpläne (Teil 5).....	145
Tab. A-11: Transformationsmuster für Gantt-Diagramme (Teil 1).....	147
Tab. A-12: Transformationsmuster für Gantt-Diagramme (Teil 2).....	148

---

Tab. A-13: Transformationsmuster für Gantt-Diagramme (Teil 3).....	149
Tab. A-14: Transformationsmuster für Gantt-Diagramme (Teil 4).....	150
Tab. A-15: Transformationsmuster für Gantt-Diagramme (Teil 5).....	151
Tab. A-16: Transformationsmuster für Meilensteintrenddiagramme.....	153
Tab. A-17: Transformationsmuster für die Änderungsquote .....	155
Tab. A-18: Transformationsmuster für das Gewicht.....	157
Tab. A-19: Transformationsmuster für Belastungsdiagramme (Teil 1) .....	159
Tab. A-20: Transformationsmuster für Belastungsdiagramme (Teil 2).....	160
Tab. A-21: Data Dictionary (Teil 1) .....	161
Tab. A-22: Data Dictionary (Teil 2) .....	162

# Abkürzungen

AA	<u>A</u> llgemeine <u>A</u> nforderung (alternativ: Lastenheftanforderung)
ADO	<u>A</u> ctiveX <u>D</u> ata <u>O</u> bject
AIM	<u>A</u> pplication <u>I</u> nterpreted <u>M</u> odel
ARM	<u>A</u> pplication <u>R</u> essource <u>M</u> odel
BI	<u>B</u> usiness <u>I</u> ntelligence
CAX	<u>C</u> omputer <u>A</u> ided <u>T</u> echnologien
EIS	<u>E</u> xecutive <u>I</u> nformation <u>S</u> ystem
ERP	<u>E</u> ngineering <u>R</u> essource <u>P</u> lanning
GUI	<u>G</u> raphical <u>U</u> ser <u>I</u> nterface
KA	<u>K</u> onkretisierte <u>A</u> nforderung (alternativ: Pflichtenheftanforderung)
MIS	<u>M</u> anagement <u>I</u> nformation <u>S</u> ystems
ODBC	<u>O</u> pen <u>D</u> atab <u>a</u> se <u>C</u> onnectivity
OLAP	<u>O</u> n <u>l</u> ine <u>A</u> nalytical <u>P</u> rocessing
PDM	<u>P</u> rodukt <u>d</u> aten <u>M</u> anagement
PMS	<u>P</u> rojekt <u>m</u> anagement <u>s</u> ystem
PPS	<u>P</u> roduktions <u>p</u> lanungs- und <u>S</u> teuerungssystem
STEP	<u>S</u> tandard for the <u>e</u> xchange of <u>p</u> roduct defined data
SQL	<u>S</u> tructured <u>Q</u> uery <u>L</u> anguage
TDM	<u>T</u> eam <u>D</u> ata <u>M</u> anagement





# Glossar

ARM	Das Application Resource Model bezeichnet ein anwendungsbezogenes Datenmodell in EXPRESS [Schi02].
Attribut	Attribute sind Eigenschaften von -> Entitäten. Sie sind durch einen Namen und einen Typ definiert.
Data Dictionary	Ein Verzeichnis, welches verschiedene Begrifflichkeiten aus verschiedenen Kontexten, aber gleicher Bedeutung für dasselbe Objekt gegenüberstellt, um so eine Abbildung zwischen den Begrifflichkeiten zu erhalten [Schi02].
Daten	Bloße Zeichenfolgen ohne Verwendungshinweise (z.B. Input oder Output aus Datenbanken oder Sensoren) [Bohn94], [Wehn99].
Data Mart	Sind kleinere spezielle -> Data Warehouses für bestimmte Themen bezogene Analysen [Lust99], [SchBanMer99].
Data Mining	Methode zur ungerichteten Analyse von Datensätzen (z.B. durch Clusteranalyse, Neuronale Netze) [Lust99], [Mert01].

---

Data Warehouse	Eine von operativen Datenbanken getrennte, endbenutzerorientierte Datenbank mit periodisch gefüllten Daten aus Produktivsystemen [Lust99], [GroGen00].
Dynamik	Merkmal für die Häufigkeit und (Un-) Regelmäßigkeit von Änderungen eines Systems [Schn94].
Engineeringüberwachung	→ Überwachung
Entität	Definiert ein Objekt, das aus Eigenschaften bzw. → Attributen und definierbaren Regeln besteht.
Entwicklungsprojekt	Dient der Erzeugung von marktfähigen Produkten [Litk93]. Die vorliegende Arbeit fokussiert sich auf die Domäne Maschinenbau.
Erzeugersystem	IT-System, das primär Daten erzeugt (z.B. CAx), wobei diese → Daten von anderen IT-Systemen verwaltet werden [VDI 2219].
Express	Bezeichnet eine textuelle und objektorientierte Datenmodelliersprache, die im Rahmen der internationalen Produktdatennorm → STEP verwendet wird [Phil00].
Express G	Bezeichnet die graphische Notationsform von EXPRESS.

---

Filterwert	Schränkt den Detaillierungsgrad von → Kennzahlen oder Kennlinien ein.
Gruppierungstechnik	Methode zum Zusammenfassen von Objekten zu Objektgruppen anhand definierter Ähnlichkeit. Es stehen drei verschiedene Verfahren zur Verfügung. Ordnungssysteme (z.B. Begriffssysteme, Klassifikationsschlüssel, Sachmerkmaleiste, Merkmalbasierte Gliederung und Beschreibung), multivariante Statistik (Cluster Analyse, Faktoren Analyse, Fuzzy Set) und schließlich die mathematische Analytik (z.B. Fourier-Desriptoren) [Zwic99].
Hilfsobjekte	Zu dieser Kategorie gehören Objekte, die den Einsatz von → Kernobjekten regeln.
Instanz	Mögliche Ausprägung einer → Entität.
Integriertes Produktmodell	Dient als logische Integrationsplattform für Anwendungssoftwaresysteme, die in den Phasen des Produktlebenszyklus eingesetzt werden [AndTri99].
Kennzahl	Messzahlen aus dem Rechnungswesen einer Unternehmung, welche die Leistungsfähigkeit unterschiedlicher Unternehmensbereiche oder verschiedene Unternehmen zueinander messbar machen sollten [SchBanMer99], [Lust99]. Kennzahlen werden zu Kennlinien, wenn man sie über die Zeit betrachtet.

---

Kernobjekte	Diese Kategorie fasst Objekte zusammen, die für das „Konzept zur systematischen Transformation von PDM-Informationen“ Transformationsunterstützung von zentraler Bedeutung sind.
Klasse	→ Entität
Komplexität	Beschreibt das Merkmal für die Überschaubarkeit/Handhabbarkeit eines Systems. Die Komplexität ist abhängig von Umfang und → Dynamik der (System-) Bausteine, Relationen und Zustände [GauEbbKal01], [Trip02].
Metadaten	→ Daten, die Teilstämme und deren Relationen innerhalb eines → PDM-Systems eindeutig beschreiben und identifizieren [Karc02].
Musterinstanz	Tabelle, welche über ein → Transformationsmuster hinaus die semantische Zuordnung zwischen STEP-Attributen und PDM-Attributen darstellt.
OLAP	Online Analytical Processing ist eine gerichtete Abfragemethode, die Endbenutzern einen mehrdimensionalen Zugriff auf Datenbestände (z.B. Data Warehouse, Data Mart) ermöglicht [Lust99], [SchBanMer99].
Paradigma	Eine bestimmte Art und Weise, die Problemwelt zu begreifen und Lösungskonzepte zu formulieren [JohSch99], [Glan02].

---

PDM	Produktdaten Management repräsentiert ein Konzept für die Verarbeitung von Produktdaten in den Phasen des Produktlebenszyklus und baut auf dem sog. → Integrierten Produktmodell auf [AndTri99], [Lang00].
Produktentwicklung	Der Begriff Produktentwicklung (bzw. Engineering) beschreibt einen Teilaspekt des Produktlebenszyklus, der die Bereiche Produktplanung (bzw. Anforderungsanalyse), Produktkonstruktion, -erprobung sowie Produktionsvorbereitung umfasst [Ehr195], [LinRei98].
Produktstruktur	Sie besteht einerseits aus Produkt definierenden Daten d.h. Stammdaten, wie z.B. Ersteller, Erstellungsdatum und Strukturdaten, wie z.B. Gültigkeiten und Verwendungshäufigkeit. Andererseits aus Produkt beschreibenden Daten, wie z.B. der Geometrie.
Projekt	Charakterisiert ein Vorhaben, das im wesentlichen durch seine Einmaligkeit der Bedingungen (z.B. Zielvorgabe, zeitliche, finanzielle und personelle Begrenzungen) gekennzeichnet ist [DIN 69901], [ISO 10006].
Überwachung	Kontrolle der Produktentwicklung in Entwicklungsprojekten des Maschinenbaus. Die Arbeit verwendet den Begriff alternativ zu „Engineeringüberwachung“.

---

Transformationsbibliothek	Die Arbeit versteht darunter ein Konzept, dass auf Grundlage von → Transformationsmustern und → Musterinstanzen den Paradigmenwechsel zwischen dem PDM- und dem Überwachungsansatz unterstützt.
Transformationsmuster	Tabelle, welche die semantische Zuordnung zwischen STEP-Attributen und den Elementen einer Überwachungskennlinie zeigt.
STEP	STEP (standard for the exchange of product model data) Synonym für die ISO 10303, beschreibt die rechnerinterpretierbare Darstellung und den Austausch von Produktdaten. Die Zielsetzung dieser Norm besteht darin, einen Mechanismus zur Verfügung zu stellen, der Produktdaten durch den gesamten Lebenszyklus eines Produktes hindurch, unabhängig von einem bestimmten Rechnersystem, beschreibt [ISO 10303-1], [AndTri99].
Steuerung	Die Steuerung sorgt für die Umsetzung von Korrekturmaßnahmen bei Projektstörungen. Grundlage der Steuerung bildet die → Überwachung des Projektgeschehens [Habe00].
Überwachung	Aufgabe einer Überwachung ist es den aktuellen Projektverlauf zu erfassen und ihn nach den Regeln des Projektmanagements darzustellen. Sie ist damit Grundlage für die → Steuerung [ISO 10006], [DIN 69901]. Die vorliegende Arbeit versteht dabei speziell die Überwachung der → Produktentwicklung in → Entwicklungsprojekten des Maschinenbaus.

# 1. Einleitung

## 1.1 Hintergrund und Aufgabenstellung

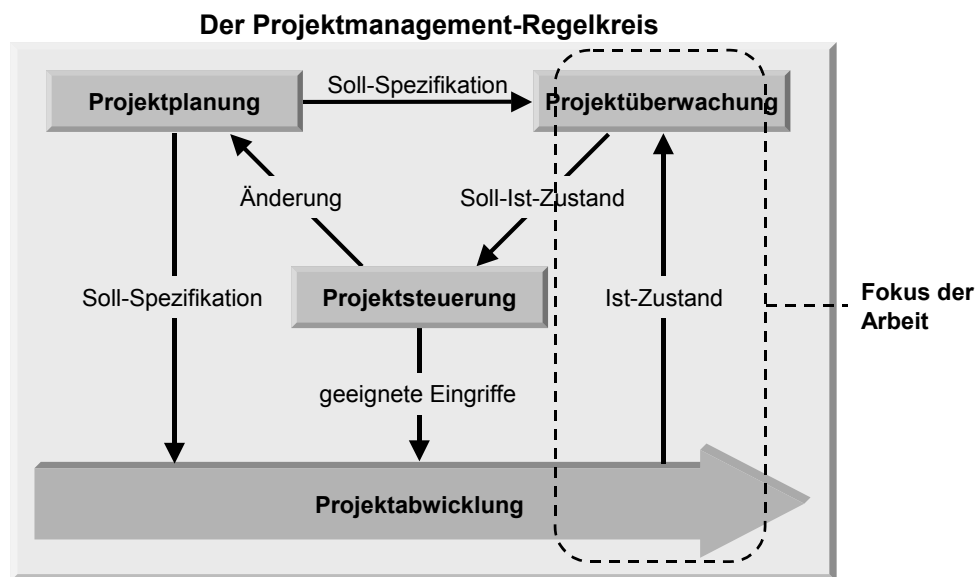
Eine verlässliche Überwachung des Engineerings ist Thema dieser Arbeit. Bis heute sind mit ihr enorme Kraftanstrengungen verbunden.

Die politischen Veränderungen der letzten Jahre haben zu einer weiteren Öffnung der internationalen Märkte geführt. Aufstrebende Volkswirtschaften, vor allem in Asien, Osteuropa und Südamerika, werden als Wachstumsmotor angesehen. Aber auch ein wirtschaftliches Engagement in diesen Regionen in Form von Tochterunternehmen oder Joint-ventures ist für die Sicherung der globalen Wettbewerbsfähigkeit von besonderer Bedeutung.

Internationale Aktivitäten bringen für Maschinenbauunternehmen eine Reihe von organisatorischen und strukturellen Veränderungen mit sich. So erfordern weltweite Entwicklungskooperationen eine klar definierte Arbeitsteilung [Schö99]. Ist dies nicht der Fall, muss mit Verzögerungen des Markteintritts gerechnet werden. Dies ist ein ernst zu nehmendes Problem, denn eine 6-monatige Verzögerung des Markteintritts (ca. 25%) führt zu Ertragseinbußen von durchschnittlich 33% [Wild03]. Der Erfolg bzw. der Misserfolg von Maschinenbauunternehmen hängt entsprechend stark vom Gelingen seiner Entwicklungsprojekte ab.

Ein besonderes Merkmal heutiger Produkte ist der starke funktionale Zusammenhang ihrer Mechanik-, Elektronik- und Softwarekomponenten. Infolgedessen entsteht eine Fülle an komplex vernetzten Produktinformationen, die über die unterschiedlichen Fachdisziplinen hinweg beherrscht werden müssen. Hilfe bieten Produktdatenmanagement-Systeme, die bereits viele Unternehmen einsetzen.

In diesem Entwicklungsumfeld trägt eine verlässliche Projektüberwachung zur Verwirklichung von geplanten Zielen wesentlich bei [Rind94], [Bend01]. Doch weist das derzeitige Engineeringcontrolling Defizite auf. Ursache dafür ist eine mangelnde Verlässlichkeit bisher üblicher Informationsressourcen. Die Dissertation greift diese Problemstellung auf und entwickelte ein Konzept, das den Wechsel zwischen dem PDM-Paradigma und dem Paradigma der Engineeringüberwachung vereinfacht. Den Fokus der Arbeit stellt Abbildung 1-1 dar.



**Abb. 1-1: Fokus der Arbeit**

## 1.2 Aufbau und Vorgehensweise

Die vorliegende Arbeit gliedert sich entsprechend der Abbildung 1-2 in insgesamt sieben Hauptkapitel. Sie werden im Folgenden kurz umrissen.

Kapitel 2 identifiziert ein Überwachungsdefizit im Engineering, dessen Ursache eine mangelnde Verlässlichkeit bisher üblicher Informationsressourcen ist. Aus dieser Problemstellung heraus entwickelt die Arbeit ein Pflichtenheft und einen allgemeinen Überwachungsansatz.

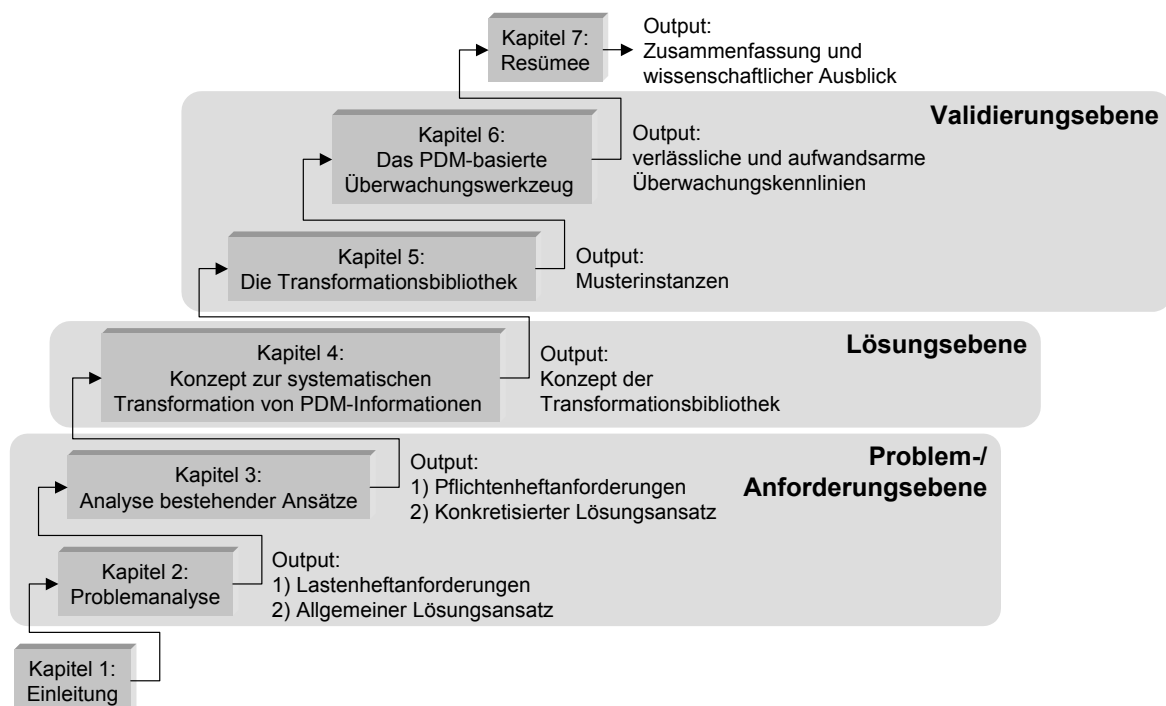


Kapitel 3 identifiziert überwachungsrelevante Ansätze, die im Rahmen eines Lösungskonzeptes genutzt werden können. Das Diskussionsergebnis fasst ein Pflichtenheft zusammen aus dem ein konkretisierter Überwachungsansatz entsteht.

Kapitel 4 greift die Ergebnisse aus Kapitel 3 auf und entwickelt ein „Konzept zur systematischen Transformation von PDM-Informationen“.

Kapitel 5 und Kapitel 6 verifizieren dieses Konzept. Kapitel 5 stellt eine Transformationsbibliothek vor, die zwischen dem PDM-Paradigma und dem Paradigma der Engineeringüberwachung vermittelt. Kapitel 6 zeigt ein PDM-basiertes Überwachungswerkzeug, das in der Entwicklung eines Luft- und Raumfahrtunternehmens schon produktiv eingesetzt wird.

Kapitel 7 fasst die Arbeitsergebnisse zusammen und gibt einen Ausblick auf zukünftige Forschungstätigkeiten.



**Abb. 1-2: Gesamtaufbau und Vorgehensweise**



## 2. Problemanalyse

Eine zuverlässige Überwachung von Entwicklungsprojekten trägt zur Verwirklichung von geplanten Zielen wesentlich bei. Jedoch weist das derzeitige Engineeringcontrolling Defizite auf. Ursache ist eine mangelnde Verlässlichkeit bisher üblicher Informationsressourcen. Kapitel 2 verfolgt drei Ziele. Zunächst wird dieses Überwachungsdefizit verifiziert. Darüber hinaus formuliert die Arbeit Lastenheftanforderungen, auf deren Grundlage ein entsprechend allgemeiner Lösungsansatz vorgestellt wird. Dazu stellt Kapitel 2.1 bis 2.3 das Umfeld der Engineeringüberwachung vor. Kapitel 2.4 diskutiert schließlich dessen Auswirkungen auf die Produktentwicklungsüberwachung. Den Aufbau gibt Abbildung 2-1 wieder.

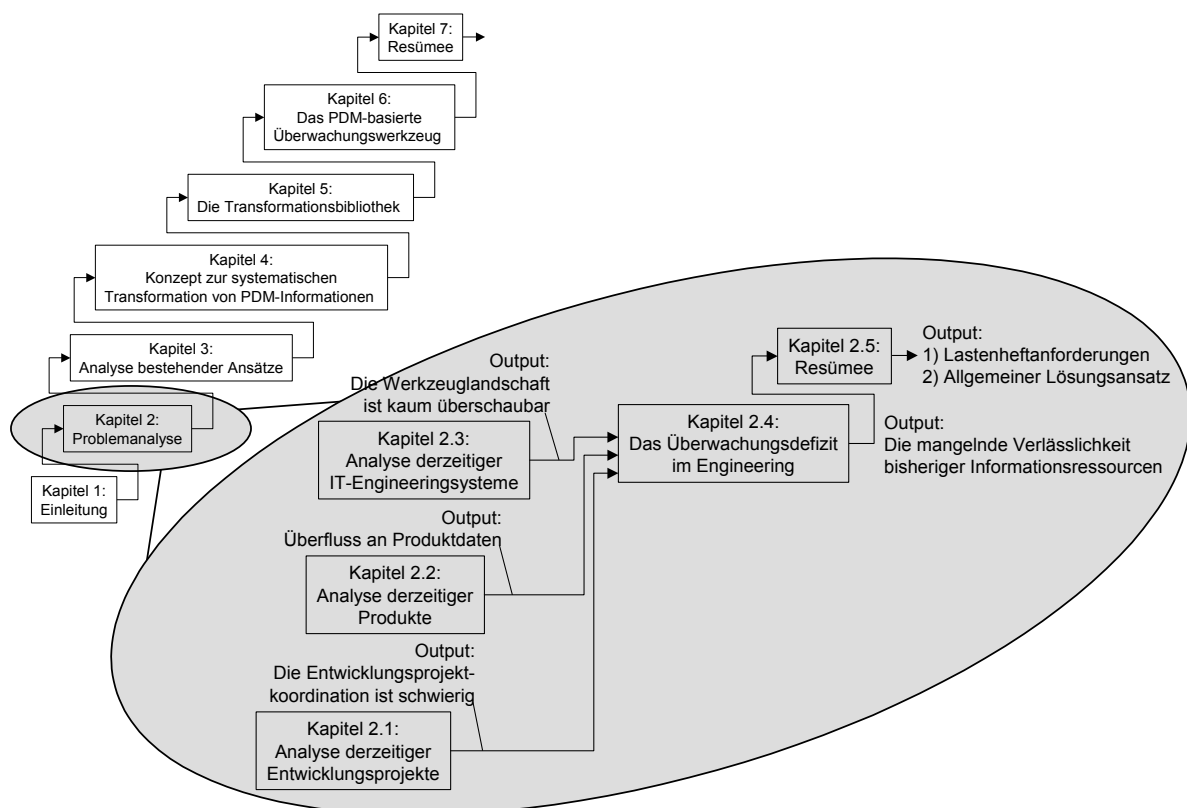


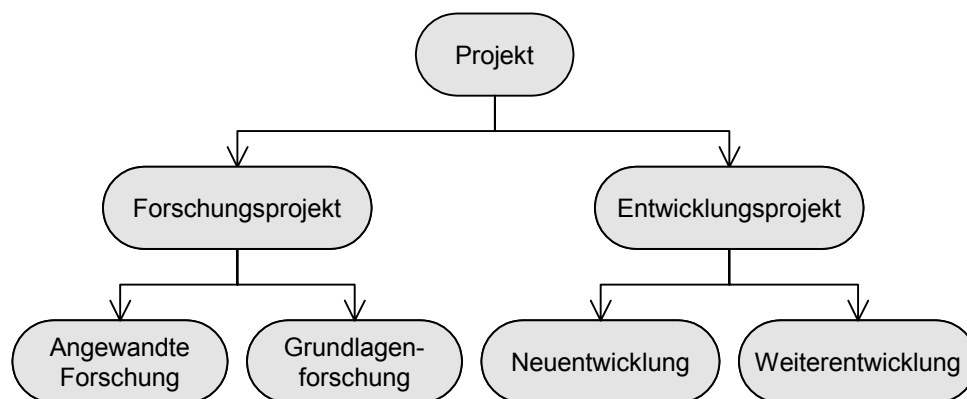
Abb. 2-1: Aufbau und Vorgehensweise Kapitel 2

## 2.1 Analyse derzeitiger Entwicklungsprojekte

Dieses Kapitel zeigt, dass sich die Koordination von Entwicklungsprojekten zunehmend schwieriger gestaltet. Dazu beschreibt die Arbeit den Begriff „Entwicklungsprojekte“ sowie deren Notwendigkeit und Veränderungen.

### 2.1.1 Begriffsklärung Entwicklungsprojekt

Ein *Projekt* ist ein Vorhaben, das im wesentlichen durch die Einmaligkeit seiner Bedingungen gekennzeichnet ist. Beispielsweise zeitliche, finanzielle und personelle Begrenzungen [DIN 69901], [DIN 19246]. Im Gegensatz zu *Forschungsprojekten* steht bei reinen *Entwicklungsprojekten* die Zweckorientierung im Vordergrund, d.h. die Generierung von marktfähigen Produkten (siehe Abb. 2-2) [Wick95]. Außerdem existieren bei ihnen schon zu Beginn konkrete Anforderungen bzgl. des Projektziels [Geig00], [Burg02]. Die vorliegende Arbeit schränkt den Entwicklungsprojektbegriff auf die Domäne Maschinenbau ein.



**Abb. 2-2: Abgrenzung von Forschungs- und Entwicklungsprojekten**

### 2.1.2 Stellenwert und Veränderungen von Entwicklungsprojekten

Da nach Schätzungen bis zu 80% des Produkterfolges im Entwicklungsbereich liegen, haben Entwicklungsprojekte enorme Bedeutung für den wirtschaftlichen Erfolg von Unternehmen [Sche92], [Bürg96]. Verzögert sich beispielsweise der Markteintritt um

6 Monate (ca. 25%), muss mit Ertragseinbußen von durchschnittlich 33% gerechnet werden [Lind02], [Wild03]. Der Erfolg eines Unternehmens hängt somit vom Ergebnis seiner Entwicklungsprojekte ab [PraSüd97], [Scho00], [MayChr01]. Die wesentlichen Veränderungen von Entwicklungsprojekten fasst die Arbeit wie folgt zusammen:

- a) Zunehmendes Concurrent Simultaneous Engineering
- b) Zunehmend virtuelle Unternehmen
- c) Zunehmende Multiprojektentwicklung

#### **a) Zunehmendes Concurrent Simultaneous Engineering (CSE)**

Der CSE-Ansatz ist in heutigen Entwicklungsprojekten allgegenwärtig [Lind02]. Die Grundidee besteht darin, dass örtlich verteilte Entwicklungsprozessschritte zeitlich parallel (simultaneous) in interdisziplinären Teams (concurrent) durchgeführt werden [Ehr195]. In der Literatur wird der Ansatz des CSE u.a. durch die Notwendigkeit sinkender Entwicklungszeiten und der zunehmenden Spezialisierung einzelner Entwicklungstätigkeiten begründet [EveEtAl98], [Schie02].

#### **b) Zunehmend virtuelle Unternehmen**

In den letzten Jahren hat eine zunehmende Dezentralisierung von Entwicklungsprojekten stattgefunden [SFB 374], [DobMus00]. Der Begriff, der sich in diesem Zusammenhang entwickelt hat, ist das sogenannte „virtuelle Unternehmen“. Der Begriff „virtuelles Unternehmen“ beschreibt einen losen Verbund von rechtlich unabhängigen Unternehmen, die für einen kurzen Zeitraum projektähnlich miteinander zusammenarbeiten. Die Entwicklungsprozesse in virtuellen Unternehmen sind entsprechend dynamisch und flexibel [LinRei98], [Jenn01]. Gründe für die Zunahme an virtuellen Unternehmen sind, dass Projektpartner individuelle Schlüsselfähigkeiten besitzen, die synergetisch kombiniert werden können [JohSch99], [GauEbbKall01], [Schi02], technologische und ökonomische Risiken aufteilbar sind [Ried00] und es ist möglich, Kostenvorteile in Ländern oder Regionen mit niedrigem Lohnkosten- bzw. Gehaltsniveau auszunutzen [Ried00].

### **c) Zunehmende Multiprojektentwicklung**

Unternehmen, die im Wettbewerb bestehen wollen, müssen ihre Produktpalette kontinuierlich verbessern und erneuern [LinRei98]. Die Folge ist, dass Entwicklungsprojekte nicht mehr autark sondern im integrierten Zusammenhang durchgeführt werden. Das bedeutet, dass Entwicklungsergebnisse projektübergreifend verwendet werden [Wild98], [Hill02].

#### **2.1.3 Fazit - die Koordination der Entwicklungsprojekte ist schwierig**

Bei heutigen Entwicklungsprojekten gestaltet sich die Koordination zunehmend mühsamer, weil immer mehr untereinander vernetzte Faktoren aufeinander abgestimmt werden müssen. Beispielsweise steigt die Anzahl der Kooperationspartner, Mitarbeiter, Prozessschritte, Aufgabenpakete und Entwicklungsstandorte mit Zunahme der virtuellen Unternehmensverbände. Durch die zuvor gezeigten Veränderungen ist aber auch davon auszugehen, dass sich dieser Trend in Zukunft verstärkt.

## **2.2 Analyse derzeitiger Produkte**

Dieses Kapitel stellt dar, dass heutige Produkte zu einem Überfluss an komplex vernetzten Produktdaten führen. Dazu beschreibt die Arbeit den Begriff „Produkte“ sowie deren Notwendigkeit und Veränderungen.

### **2.2.1 Begriffsklärung Produkt**

Bei einem Produkt handelt es sich um ein Erzeugnis, das als Ergebnis des Entwickelns und Konstruierens hergestellt oder angewendet wird. Das können materielle (z.B. Maschinen, Verfahren, Sachgüter), immaterielle Erzeugnisse (z.B. Programme, Dienstleistungen) oder auch Kombinationen sein [ThoEsc03], [VDI 2221]. Diese Arbeit versteht unter dem Begriff Produkt ausschließlich ein gegenständliches Erzeugnis des Maschinenbaus und verwandter Domänen, bei dem Software nicht das zentrale Element ist.

### **2.2.2 Stellenwert und Veränderungen von Produkten**

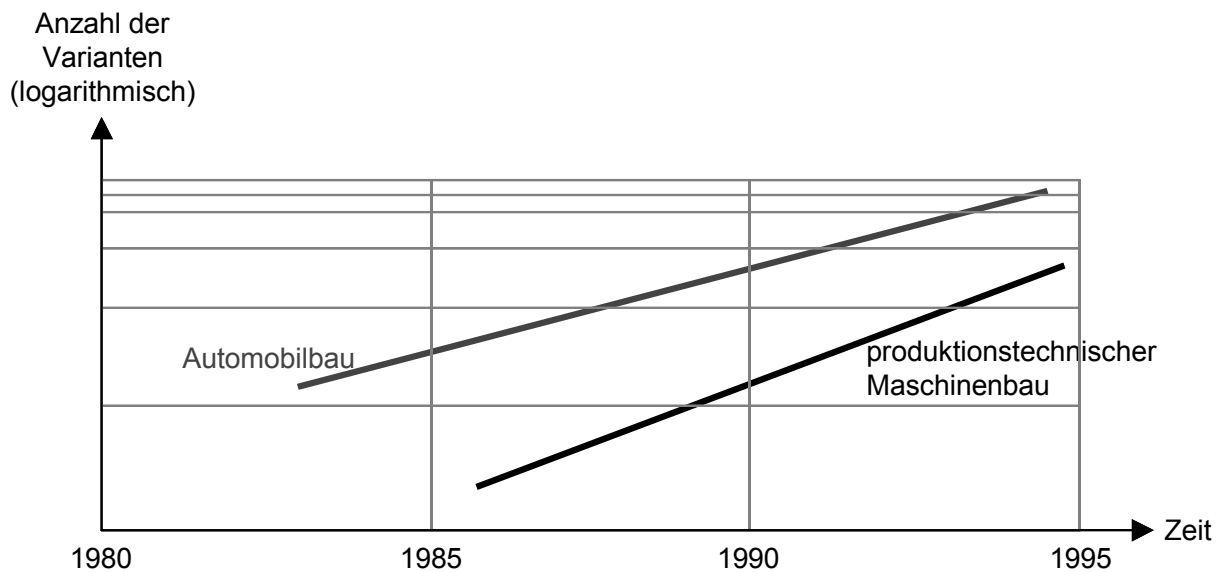
Durch innovative Produkte ist es möglich, Wettbewerbsvorteile und den wirtschaftlichen Erfolg eines Unternehmens zu sichern. Dieser Vorsprung ist aber nur temporär, weil davon auszugehen ist, dass andere Wettbewerber schon bald in der Lage sind, gleiche oder ähnliche Produkte verbilligt anzubieten [Stoc89], [JohSch99], [Geig00]. Die wesentlichen Veränderungen von Produkten fasst die Arbeit wie folgt zusammen:

- a) Zunehmende Vielfalt an Produktvarianten und Produktversionen
- b) Zunehmender Software- und Elektronikanteil
- c) Zunehmende Komponentenvernetzung

#### **a) Zunehmende Vielfalt an Produktvarianten und Produktversionen**

Produktvarianten sind zeitlich parallel existierende Ausprägungen ein und desselben Produktes, wie z.B. ein Fahrrad in den Varianten Sport, Komfort, Standard. Produktversionen sind zeitlich nacheinander entstehende Weiterentwicklungen eines Produktes, wie z.B. ein Sport-Fahrrad des Jahrgangs 2000 oder 2001.

Einerseits führt die fortschreitende Globalisierung dazu, dass individuelle Kundenwünsche immer kurzfristiger berücksichtigt werden [LinRei98], [FisZscTei03]. Auf der anderen Seite wächst die Vielfalt an Produktversionen sowie -varianten [GrabEtAl00], [Warn01] und es entsteht eine Fülle an Produktinformationen, die beherrscht werden müssen (siehe Abb. 2-3).

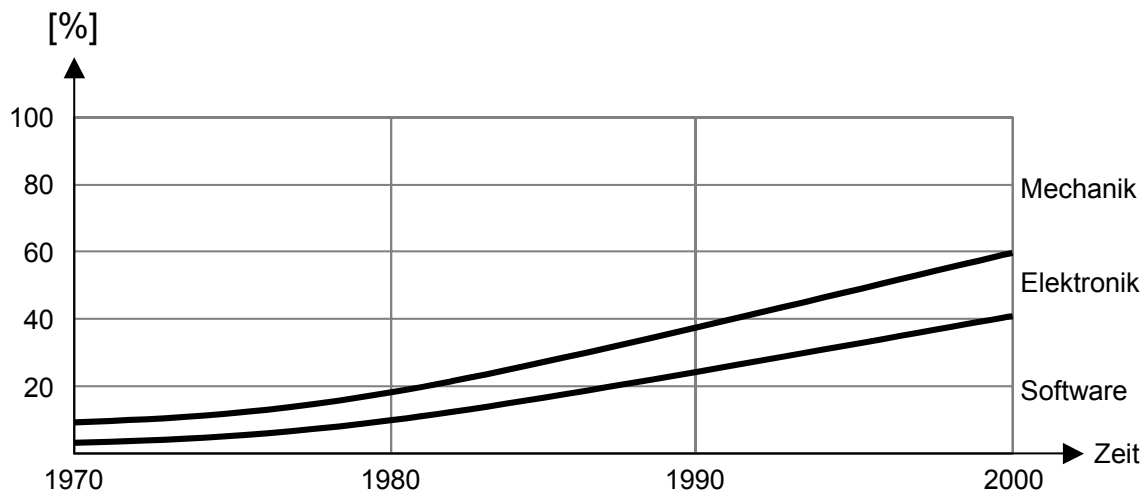


**Abb. 2-3: Anstieg der Variantenzahl [GerSch96]**

### **b) Zunehmender Software- und Elektronikanteil**

Früher wurden wesentliche Funktionen eines Produktes durch mechanische Komponenten realisiert. Heute drängt der Einsatz von Elektronik und Software diese immer mehr zurück [Glan02], [DomBot03], [Xul03]. Gründe hierfür sind u.a., dass Produktfunktionalitäten oft nur noch durch Software und Elektronik verwirklicht werden können (z.B. Sensoren). Software und Elektronik erlauben die flexiblere Gestaltung eines Produktes (z.B. Pumpen, Antriebe). Mit ihrer Hilfe lassen sich Produktfunktionen tendenziell preisgünstiger realisieren [Soli00], [BendEtAl01], [Wirt01], [Koca03].





**Abb. 2-4: Zunehmender Software- und Elektronikanteil**

### c) Zunehmende Komponentenvernetzung

Innovative Produkte zeichnen sich dadurch aus, dass zwischen den jeweiligen Produktkomponenten, starke funktionale Abhängigkeiten bestehen [GerSch96], [Iser99], [AndFröKlei00], [Wirt01]. Sie wird durch den zunehmenden Wunsch nach Produktfunktionalitäten getrieben, die oftmals nur noch im integrierten Zusammenspiel zwischen den jeweiligen Produktkomponenten wie Software, Elektronik, Hydraulik und Mechanik etc. verwirklicht werden können (z.B. variable Ventilsteuerung bei Verbrennungsmotoren).

### 2.2.3 Fazit - Überfluss an Produktdaten

Es zeigte sich, dass das Produktdatenvolumen mit zunehmendem Funktionsumfang und sinkender Lebensdauer der Produkte stark anwächst. Aus diesem Grund existiert heutzutage nicht ein Mangel, sondern ein Überfluss an Produktdaten.

## 2.3 Analyse derzeitiger IT-Engineeringsysteme

Dieses Kapitel veranschaulicht, dass die IT-Werkzeuglandschaft im Engineering kaum überschaubar ist. Dazu klärt die Arbeit den Begriff „IT-Engineeringsysteme“ sowie deren Notwendigkeit und Veränderungen.

### 2.3.1 Begriffsklärung IT-Engineeringsystem

IT-Systeme beinhalten Verfahren und Hilfsmittel, welche der automatisierten Verarbeitung von Daten unter Berücksichtigung ihrer Bedeutung (Semantik) dienen. Das *Engineering* (Produktentwicklung) verkörpert den planerischen, gestalterischen und organisatorischen Anteil der Produktentstehung [Ehr195]. CAx und FEM sind typische Vertreter von Erzeugersystemen im Engineering. PDM/EDM, DMS, WFS und TDM sind gängige Verwaltungssysteme im Engineering [VDI 2219].

### 2.3.2 Stellenwert und Veränderungen der IT-Engineeringsysteme

Der Einsatz von IT-Systemen im Engineering ist heute vielfach unumgänglich geworden. So ermöglichen IT-Systeme eine Verbesserung der Informationsverfügbarkeit und -qualität, so dass verteilt ablaufende Entwicklungsprojekte effizienter unterstützt werden können [Bend03], [Luka03]. Das bedeutet, mit Hilfe heutiger IT-Engineeringsysteme sinkt das Risiko von Fehlentwicklungen [BulBröWag99], [Jung99]. Die wesentlichen Veränderungen von IT-Engineeringsystemen fasst die Arbeit wie folgt zusammen:

- a) Zunehmender Kopplungsgrad zwischen IT-Engineeringsystemen
- b) Zunehmend individuelle Erzeuger- und Verwaltungssysteme

#### a) Zunehmender Kopplungsgrad zwischen IT-Engineeringsystemen

Durch die Verlagerung der Anwendungssoftware von zentralen Host Rechnern auf dezentrale Workstations existierten viele IT-Werkzeuginseln im Engineering. Entsprechend arbeiteten Konstrukteure und nachgelagerte Abteilungen oft mit

falschen, nicht mehr aktuellen Daten und Dokumenten [Ande01], [Ande03], [Karc02], [SFB 374]. Um die Datenqualität zu verbessern, gibt es seit einiger Zeit den Trend, diese IT-Werkzeuginseln zu vernetzen [BenKarFis01], beispielsweise CAD, FEM und DMU. So wird die Produktgeometrie in CAD modelliert, nach FEM übertragen, um sie dort entsprechend den Lasten-/Pflichtenheftanforderungen mechanisch und/oder thermisch zu belasten. Schließlich wird die Produktgeometrie nach DMU übertragen, wo entsprechende Kollisionsanalysen durchgeführt werden (Stichwort: virtueller Prototyp).

### **b) Zunehmend individuelle Erzeuger- und Verwaltungssysteme**

In Erzeugersystemen entstehen primär Daten, wobei diese von anderen IT-Systemen verwaltet werden (Verwaltungssysteme) [VDI 2219]. Beispiele für Erzeugersysteme sind Computer Aided Technologien (CAx), Finite Elemente Methode (FEM) und Digital Mock Up (DMU) [BulBröWag99], [Wirt01], [Karc02], [Vils04]. Zu den Verwaltungssystemen zählen Produktdatenmanagement/Engineering Daten Management (PDM/EDM), Team Data Management (TDM), Dokumentenmanagement/Content Management und Workflowmanagement. Seit vielen Jahren werden diese IT-Systeme in der Produktentwicklung eingesetzt. Um deren Unterstützungsgrad zu erhöhen, werden sie an unternehmensspezifische Aufgaben angepasst [Ande01], [Glan02], [Vils04].

### **2.3.3 Fazit - die Werkzeuglandschaft ist kaum überschaubar**

Heutzutage ist es zunehmend schwieriger geworden, einen aktuellen Überblick über die speziell für Entwicklungsaufgaben zugeschnittene IT-Werkzeuglandschaft zu behalten. So zeigte sich, dass der Grad der Verknüpfung zwischen individuellen IT-Lösungen ansteigt.

## **2.4 Das Überwachungsdefizit im Engineering**

Die vorangegangenen Kapitel hatten sich mit dem komplexen Engineeringumfeld beschäftigt. Dieses Kapitel wird zeigen, dass das derzeitige Engineeringcontrolling

Defizite aufweist, deren Ursache eine mangelnde Verlässlichkeit bisher üblicher Informationsressourcen ist. Dazu klärt die Arbeit den Begriff „Engineeringüberwachung“, deren Notwendigkeit und Aufgaben. Im weiteren Verlauf werden die Auswirkungen der komplexen Rahmenbedingungen auf die Engineeringüberwachung diskutiert.

### 2.4.1 Begriffsklärung Engineeringüberwachung

Das *Projektmanagement* beschäftigt sich mit Planung, Organisation, Überwachung (Controlling) und Steuerung aller Projektaspekte, um mit Hilfe eines kontinuierlichen Prozesses die Projektziele zu erreichen (siehe Abb. 2-5). Damit die *Projektsteuerung* Störungen frühzeitig korrigieren kann, ist es notwendig, den aktuellen Projektverlauf zu erfassen und ihn nach den Regeln des Projektmanagements aufzubereiten, beispielsweise die Arbeitsleistung, Termine, Ressourceneinsatz und Kosten [Litk93]. Dies fällt in den Aufgabenbereich der *Projektüberwachung* [ISO 10006], [DIN 69901]. Die vorliegende Arbeit versteht unter dem Begriff *Projektüberwachung* speziell die Kontrolle der Produktentwicklung in Entwicklungsprojekten des Maschinenbaus (Engineeringüberwachung).

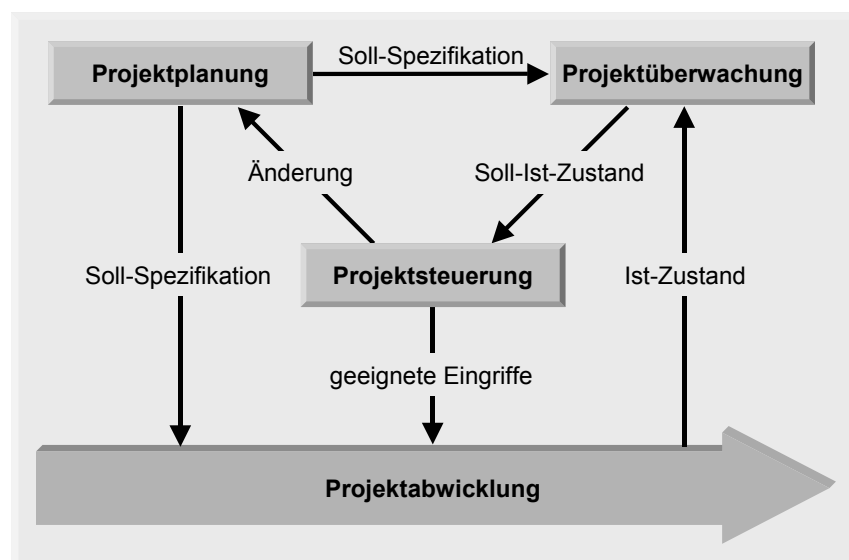


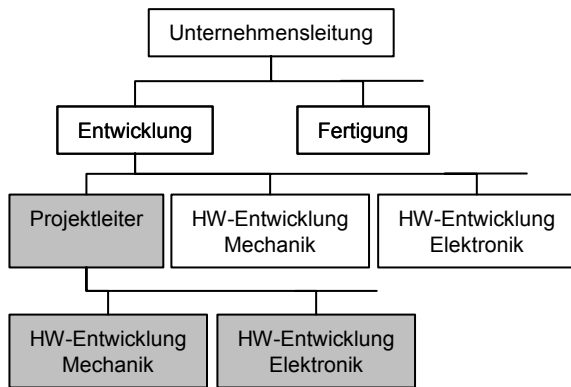
Abb. 2-5: Der Projektmanagement-Regelkreis [Burg02]

### 2.4.2 Stellenwert und Aufgaben der Entwicklungsprojektüberwachung

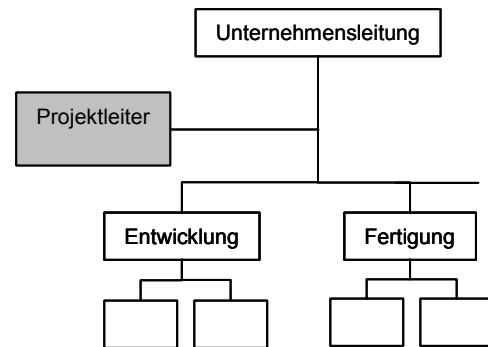
Allein die Entwicklung technisch perfekter Produkte reicht für den Markterfolg nicht aus. Für die Verwirklichung der Entwicklungsprojektziele ist es ebenso unerlässlich, ein Entwicklungsprojekt periodisch zu überwachen. Denn während der Durchführung ändern sich häufig die Projektziele oder es treten Störungen auf [Müll95], [Geig00], [Wild01]. Dabei gilt es, je früher Fehlentwicklungen erkannt werden, desto leichter, kürzer und kostengünstiger kann man gegensteuern [Burg02], [Rein02]. Bei großen Projekten nimmt allerdings der Arbeitsaufwand für die Kontrollarbeiten relativ stark zu, so dass er wirtschaftlich nur mit Hilfe einer IT-Unterstützung bewältigt werden kann [Rinz94]. Da, wie bereits zuvor erwähnt, die Engineeringüberwachung Teilaspekt des Projektmanagements ist, teilen sie sich auch die vier typischen Organisationsformen [Burg02], [Hill02]:

- 1) Reine-Projektorganisation (siehe Abb. 2-6)
- 2) Einfluss-Projektorganisation (siehe Abb. 2-7)
- 3) Matrix-Projektorganisation (siehe Abb. 2-8)
- 4) Auftrags-Projektorganisation (siehe Abb. 2-9)

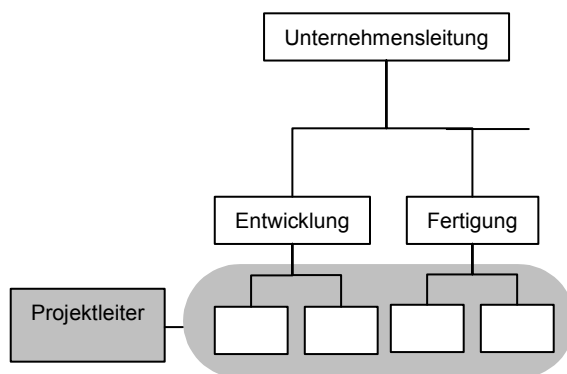
Bei der reinen Projektorganisation trägt der Projektleiter die Gesamtverantwortung und hat volle Weisungsbefugnis. Bei der Einfluss-Projektorganisation ist der Projektleiter nur Projektkoordinator. Bei der Matrix-Projektorganisation hat der Projektleiter die Projektverantwortung, aber nicht die volle Weisungsbefugnis. Bei der Auftragsprojektorganisation hat der Projektleiter die Projektverantwortung und fungiert als Auftraggeber [Burg02], [Hill02]. Die Wahl der jeweils geeigneten Organisationsform hängt von Faktoren wie Projektgröße ab [Burg02].



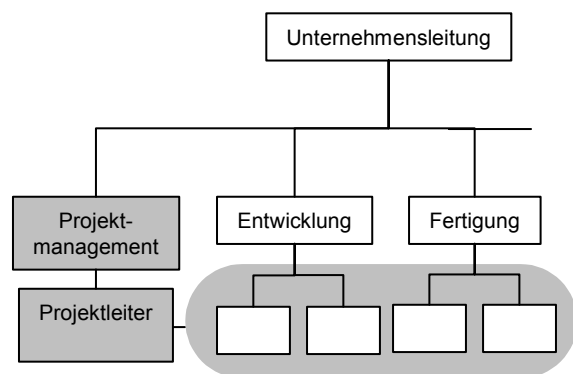
**Abb. 2-6: Die reine Projektorganisation**



**Abb. 2-7: Die Einfluss-  
Projektorganisation**



**Abb. 2-8: Die Matrix-  
Projektorganisation**



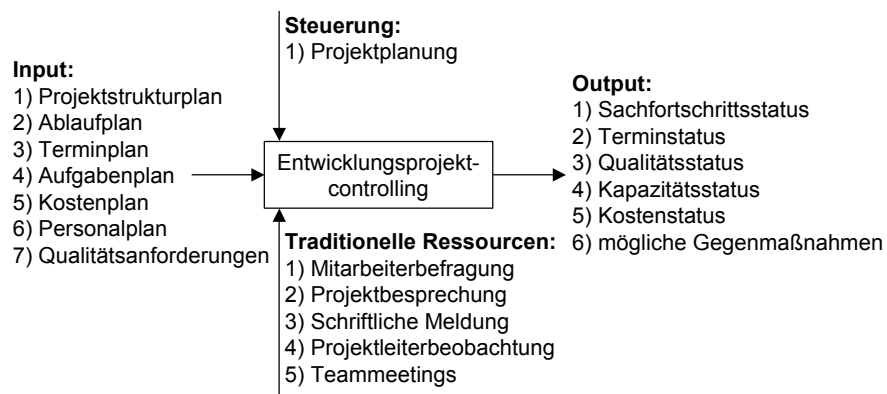
**Abb. 2-9: Die Auftrags-  
Projektorganisation**

Am meisten verbreitet ist das Entwicklungscontrolling in der Maschinenbauindustrie [Stoc89]. Sinn und Zweck des Entwicklungscontrollings (teilweise sind in der Literatur die Übergänge zwischen Projektcontrolling und Projektsteuerung fließend oder sie werden synonym verwendet) ist es, dass die Entwicklungsprojektziele (Soll-Zustände) entsprechend den Vorgaben aus der Entwicklungsprojektplanung erfüllt werden (siehe Abb. 2-10).

Zu überwachen sind die Vorgaben aus der Entwicklungsprojektplanung, wie z.B. Terminplan, Kostenplan, Qualitätsplan. Sie müssen von der Engineeringüberwachung

zu entsprechenden Leistungsparametern, wie z.B. Fertigstellungsgrad und Änderungsgrad, überführt werden. Da die Leistungsparameter in einer gegenseitigen Wechselwirkung stehen, müssen sie integriert betrachtet werden [Burg02], [Geig00], [Rein02], [JungWirt03]. Dabei wird die Datenbasis als Ressource zur Aggregation der Leistungsparameter traditionell durch Methoden wie Mitarbeiterbefragungen, Projektleiterbeobachtung, Projektbesprechungen, Teammeetings und schriftliche Meldungen erfasst, die in regelmäßigen Abständen durchgeführt werden müssen [Rinz94], [Schm96], [Bend01]. Entsprechend ergeben sich folgende sechs Aufgaben für die Projektüberwachung [Stoc89], [Gare91], [Litk93], [Rinz93], [Müll95], [Schm96], [Geig00], [Schn00], [Burg02], [Hill02], [Rein02]:

- a) Überwachung des Sachfortschritts
- b) Überwachung der Termine
- c) Überwachung der Qualität
- d) Überwachung der Kapazität
- e) Überwachung der Kosten
- f) Entwicklung von korrigierenden Maßnahmen bei Projektabweichungen



**Abb. 2-10: Entwicklungsprojektcontrolling Input und Output (Notation: IDEFO)**

### a) Überwachung des Sachfortschritts

Das Sachfortschrittscontrolling beschäftigt sich mit der Überwachung von Entwicklungsprojektleistung und wird in eine produktorientierte Sichtweise (z.B. Sachfortschritt bzgl. CAD-Zeichnungen), auch leistungsorientierte Sichtweise genannt, und in eine projektorientierte Sichtweise unterteilt (z.B. Sachfortschritt bzgl. Arbeitspakete) [Burg02], [Gare91]. Als Soll-Referenz für die Sachfortschrittsermittlung dienen Projektstruktur- und Aufgabenpläne sowie Qualitätsanforderungen aus der Entwicklungsprojektplanung. Zur Präsentation des projekt- oder produktbezogenen Sachfortschritts stehen verschiedene Kennzahlen (z.B. Fertigstellungsgrad, Erfüllungsgrad) und Kennlinien (z.B. Sachfortschritts-, Balkendiagramm, Netzplan) zur Verfügung (siehe Abb. 2-11 und Abb. 2-12).

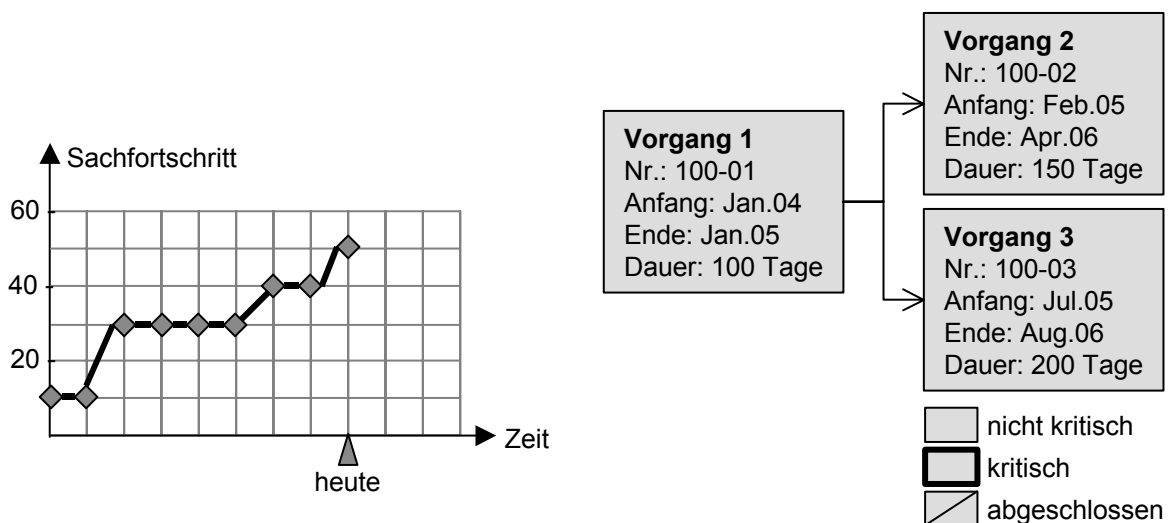


Abb. 2-11: Sachfortschrittsdiagramm

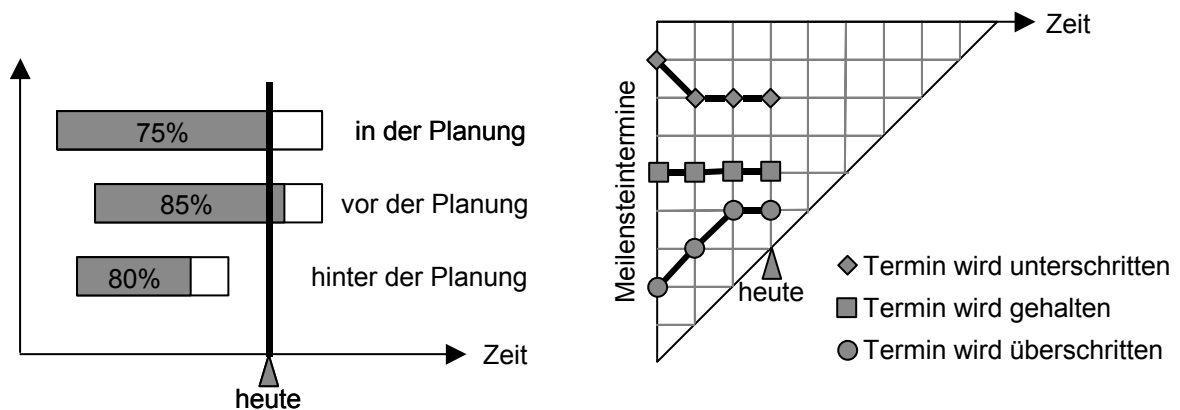
Abb. 2-12: Netzplan

### b) Überwachung der Termine

Das Termincontrolling beschäftigt sich mit der Überwachung von Entwicklungsprojektterminen. Als Soll-Referenz für die Terminermittlung dienen Personal-, Aufgaben- und Terminpläne aus der Entwicklungsprojektplanung. Zur Präsentation des Terminstatus stehen verschiedene Kennzahlen (z.B. Planabweichung,



Termintreue) und Kennlinien (z.B. Balken-/Gantt-Diagramm, Meilensteintrenddiagramm, Netzplan) zur Verfügung (siehe Abb. 2-13 und Abb. 2-14).



**Abb. 2-13: Gantt-Diagramm**

**Abb. 2-14: Meilensteintrenddiagramm**

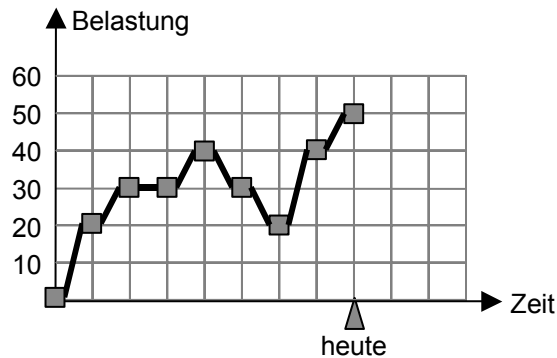
### c) Überwachung der Qualität

Das Qualitätscontrolling beschäftigt sich mit der Überwachung der Produkt- und Projektqualität. Zielsetzung ist die Einhaltung von Qualitätsstandards in Entwicklungsprojekten. Als Soll-Referenz für die Terminermittlung dienen Qualitätsanforderungen aus der Entwicklungsprojektplanung. Zur Präsentation des Produkt- und Projektqualitätsstatus werden unterschiedliche Kennzahlen benutzt (Änderungsgrad, Fehlerquote, Zuverlässigkeit, Gewicht, technische Wirksamkeit).

### d) Überwachung der Belastung

Das Belastungscontrolling beschäftigt sich mit der Überwachung der Projektressourcen. Zielsetzung ist es, eine gleichmäßige Auslastung der verfügbaren Ressourcen (Mitarbeiter, Werkzeuge, Räume) während eines gesamten Projektes sicherzustellen [Litk93]. Als Soll-Referenz für die Kapazitätsermittlung dienen u.a. Aufgaben- und Personalpläne aus der Entwicklungsprojektplanung. Der

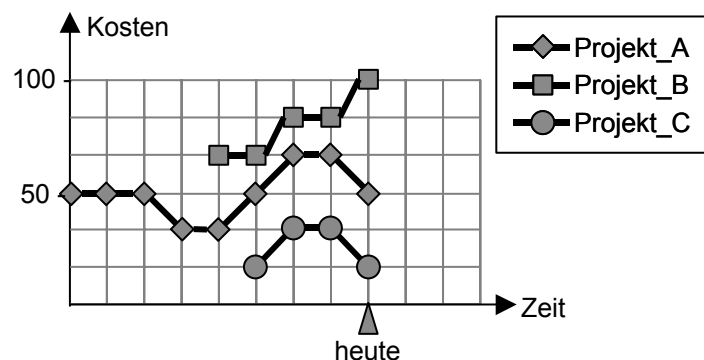
Kapazitätsstatus wird typischerweise mittels Kennzahlen wie Produktivität und Kennlinien wie Belastungsdiagramme präsentiert (siehe Abb. 2-15).



**Abb. 2-15: Belastungsdiagramm**

### e) Überwachung der Kosten

Das Kostencontrolling beschäftigt sich mit dem Controlling des Entwicklungsprojektbudgets. Dabei dient der Kostenplan aus der Entwicklungsprojektplanung als Soll-Referenz für die Kostenermittlung. Der Kostenstatus wird mittels gängiger Kennzahlen (Entwicklungskosten pro Stück) und Kennlinien (Kostendiagramm) präsentiert (siehe Abb. 2-16).



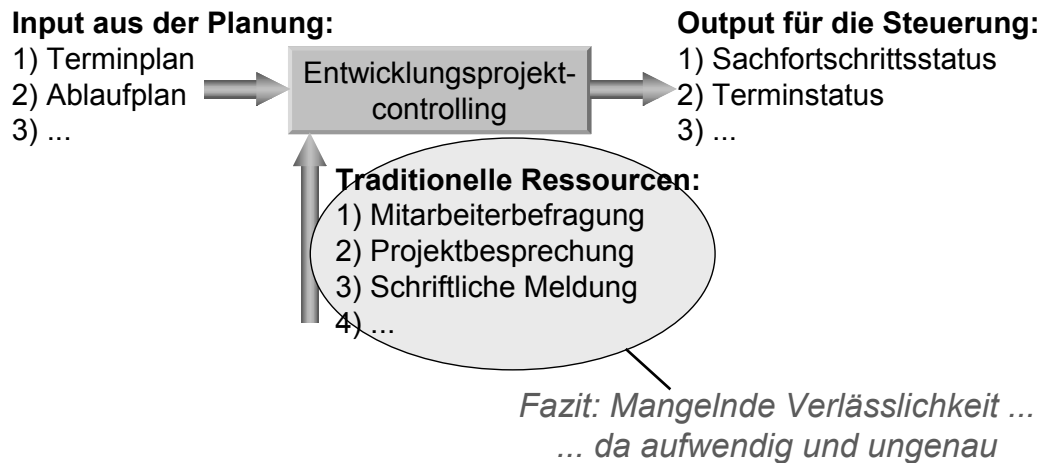
**Abb. 2-16: Kostendiagramm**

### **f) Entwicklung von korrigierenden Maßnahmen bei Projektabweichungen**

Korrigierende Maßnahmen haben das Ziel, die in der Entwicklungsprojektplanung spezifizierten Leistungsparameter, wie Termine, Leistung etc., im Einklang mit der aktuellen Entwicklungsprojektsituation zu halten. Dabei können Abweichungen von den Lastenheft- bzw. Pflichtenheftanforderungen durch unrealistische Entwicklungsprojektplanung (falsche Einschätzung des Zeitbedarfes), unvorhersehbare Änderungen (zusätzliche Kundenwünsche) und Fehler in der Arbeitsdurchführung (ungenügende Ressourcenauslastung) auftreten [Litk93]. Werden bei der Überwachung Abweichungen vom Plan festgestellt (Termin kann nicht gehalten werden), dann müssen die für das betroffene Arbeitspaket verantwortlichen Mitarbeiter Vorschläge ausarbeiten, wie diese Schwierigkeiten zu beseitigen sind [Rinz94]. Grundlage für die Entwicklung von möglichen Gegenmaßnahmen sind gegenübergestellte Analysen des aktuellen Projektzustandes durch das Controlling (z.B. Sachfortschritts- mit Meilensteintrenddiagramm). Sie können in drei Kategorien unterteilt werden: Veränderung der Leistungs-/Soll-Parameter im Lastenheft bzw. Pflichtenheft, Beibehaltung der Leistungsparameter durch entsprechende Optimierungsmaßnahmen und schließlich ein Entwicklungsprojektabbruch.

### **2.4.3 Fazit - mangelnde Verlässlichkeit bisheriger Informationsressourcen**

Kapitel 2 verifizierte, dass eine zuverlässige Überwachung von Entwicklungsprojekten anhand etablierter Kennlinien zur Verwirklichung von geplanten Zielen beiträgt. Jedoch weist das derzeitige Engineeringcontrolling Defizite auf. Ursache ist eine mangelnde Verlässlichkeit bisher üblicher Informationsressourcen wie Projektbesprechungen und formale Mitarbeiterbefragungen (siehe Abb. 2-17).



**Abb. 2-17: Mangelnde Verlässlichkeit traditioneller Controllingressourcen**

**[JunWirKar04]**

## 2.5 Resümee

Ziel der folgenden Arbeit ist es, unter Berücksichtigung der bisherigen Diskussionen allgemeine Anforderungen (Lastenheftanforderungen) zur Überwindung des Überwachungsdefizits im Engineering zu formulieren (siehe Kapitel 2.5.1). Darüber hinaus auf deren Basis einen entsprechend allgemeinen Lösungsansatz zu entwerfen (siehe Kapitel 2.5.2). Am Ende von Kapitel 3 werden sowohl Anforderungen als auch der Lösungsansatz konkretisiert (siehe Kapitel 3.5.2 bzw. 3.5.3).

### 2.5.1 Allgemeine Anforderungen an ein Überwachungskonzept

Die allgemeinen Anforderungen, die sich vor dem Hintergrund einer mangelnden Verlässlichkeit bisheriger Informationsressourcen ergeben, lassen sich wie folgt zusammenfassen:

AA 1: Allgemeine Anforderungen bzgl. der Überwachungsdatenbasis

AA 2: Allgemeine Anforderungen bzgl. Auswahl und Verdichtung

AA 3: Allgemeine Anforderungen bzgl. der Darstellung

AA 4: Allgemeine Anforderungen bzgl. eines Vorgehens

#### AA 1: Allgemeine Anforderungen bzgl. der Überwachungsdatenbasis

##### **AA 1.1: Produktive Erzeuger- und Verwaltungssysteme dienen als Informationsressource**

Begründung: Der bisherige Ansatz, eine separate Controllingdatenbasis auf Grundlage von Mitarbeiterbefragungen manuell zu pflegen, führt zu wenig verlässlichen Überwachungsergebnissen (vgl. Kapitel 2.4.3).

##### **AA 1.2: Prinzipien der Erzeuger- und Verwaltungssysteme dürfen weder verworfen noch Informationen in ihnen verändert werden**

Begründung: Die Unterstützung von Entwicklungsaufgaben durch speziell darauf zugeschnittene Erzeuger- und Verwaltungssysteme ist unumgänglich (vgl. Kapitel 2.3.3).

##### **AA 1.3: Der Umfang verwertbarer Informationen aus Erzeuger- und Verwaltungssystemen ist groß genug, um daraus die gängigsten Überwachungskennlinien entwickeln zu können**

Begründung: Um Projektabweichungen erkennen und geeignete Gegenmaßnahmen entwickeln zu können, muss das aktuelle Geschehen für die Engineeringüberwachung vollständig erfassbar sein (vgl. Kapitel 2.4.2).

**AA 1.4: Die Controllingdatenbasis lässt sich leicht an ein sich veränderndes Überwachungsumfeld anpassen**

Begründung: Typischerweise werden IT-Engineeringsysteme angepasst oder substituiert (vgl. Kapitel 2.3.3). Darüber hinaus stellte Kapitel 2.4.2 dar, dass sich Ziele im Laufe der Projektdurchführung verändern können (Stichwort: „moving target“).

**AA 2: Allgemeine Anforderungen bzgl. Auswahl und Verdichtung****AA 2.1 Die Transformation von Erzeuger- und Verwaltungssysteminformationen wird systematisch unterstützt**

Begründung: Es ist schwierig, die überwachungsrelevanten Informationen aus kaum überschaubaren IT-Werkzeuglandschaften (vgl. Kapitel 2.3.3) und aus einer Fülle potentiell interessanter Produktinformationen (vgl. Kapitel 2.2.3) zu identifizieren.

**AA 3: Allgemeine Anforderungen bzgl. der Darstellung****AA 3.1: Die Projektsituation wird anhand etablierter Überwachungskennlinien präsentiert**

Begründung: Das aktuelle Geschehen muss für die Engineeringüberwachung intuitiv erfassbar sein (vgl. Kapitel 2.4.3).

**AA 4: Allgemeine Anforderungen bzgl. eines Vorgehens****AA 4.1: Die Entwicklung eines Überwachungswerkzeuges wird systematisch unterstützt**

Begründung: IT-Werkzeuglandschaften werden kontinuierlich verändert (vgl. Kapitel 2.3.3) und Ziele ändern sich typischerweise im Laufe der Projektdurchführung (vgl. Kapitel 2.4.2).

### **2.5.2 Allgemeiner Lösungsansatz - Transformationskonzept für IT-Informationen**

Grundlage der hier vorgestellten Idee ist eine 4-Schichten-Architektur (siehe Abb. 2-18). Die unterste Schicht beinhaltet gemäß Anforderung AA 1.1 die für das Engineeringcontrolling potentiell interessanten Erzeuger- und Verwaltungssysteme der Produktentwicklung. Um die Lastenheftanforderung AA 1.2 zu befriedigen, werden in vordefinierten Zyklen diejenigen Informationen aus Erzeuger- und Verwaltungssystemen in eine zusätzliche Controllingdatenbasis gespiegelt, welche die Lastenheftanforderung AA 1.3 erfüllen. Diese Controllingdatenbasis ist entsprechend Anforderung AA 1.4 so aufgebaut, dass sie einerseits die gängigen Informationsbedürfnisse der Engineeringüberwachung abbilden kann und andererseits die typischen Prinzipien der Erzeuger- und Verwaltungssysteme im Engineering möglichst genau wiedergibt.

Die über der Controllingdatenbasis liegende „Auswahl- und Verdichtungsschicht“ ist aus vordefinierten Analysemechanismen zusammengesetzt, welche nach dem Baukastenprinzip individuell erstellt und auch während der Projektlaufzeit an sich verändernde Bedürfnisse angepasst werden können (vgl. AA 2.1). Die oberste Schicht (Überwachungscockpit) stellt die Ergebnisse in Form von etablierten Überwachungskennlinien und -kennzahlen integriert dar (vgl. AA 3.1).

Schließlich verfügt das zu verwirklichende Transformationskonzept über eine Entwicklungsstrategie, anhand derer auch weniger versierte Anwender, unter Berücksichtigung der zur Verfügung stehenden produktiven IT-Systeme, ein Überwachungswerkzeug für die Produktentwicklung konzipieren, umsetzen und weiterentwickeln können (vgl. AA 4.1).

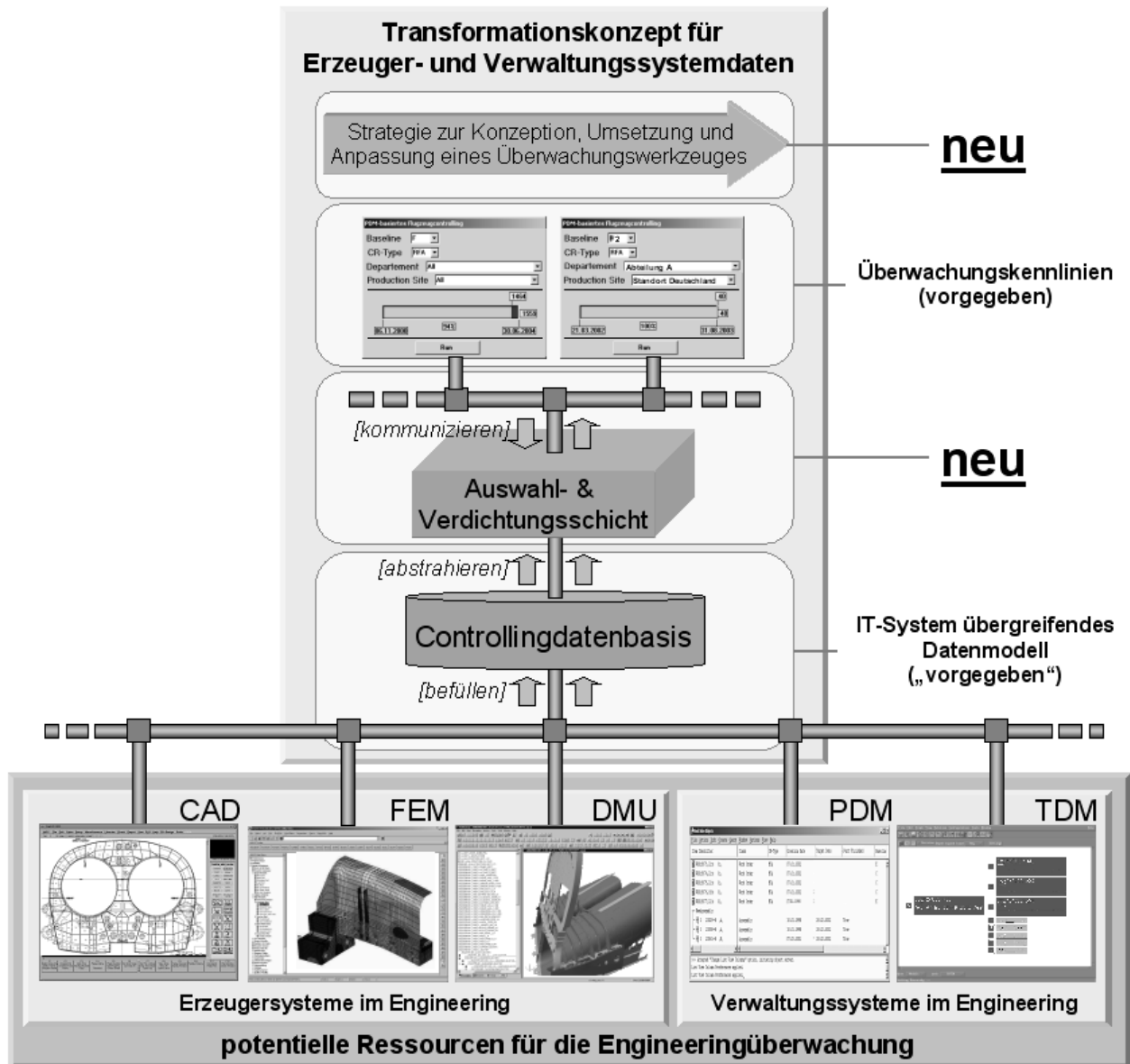


Abb. 2-18: Transformationskonzept für Erzeuger- und Verwaltungssystemdaten



### 3. Analyse bestehender Ansätze

Das vorangegangene Kapitel stellte fest, dass die zur Zeit genutzten Informationsquellen zu wenig verlässlichen Überwachungsergebnissen führen. Dieses Kapitel untersucht den Stand der Technik im Bezug auf die Überwachung komplexer Entwicklungsprojekte im Maschinenbau. Damit verfolgt dieses Kapitel zwei Ziele. Einerseits die Lastenheftanforderungen aus Kapitel 2.5.1 zu Pflichtenheftanforderungen weiterzuentwickeln, andererseits einen entsprechend konkretisierten Lösungsansatz auf Grundlage der Pflichtenheftanforderungen vorzustellen. Dazu stellt die Arbeit insgesamt siebzehn überwachungsrelevante Ansätze vor und diskutiert sie anhand der Vorgaben aus Kapitel 2.5.1. Den Aufbau dieses Kapitels gibt Abbildung 3-1 wieder.

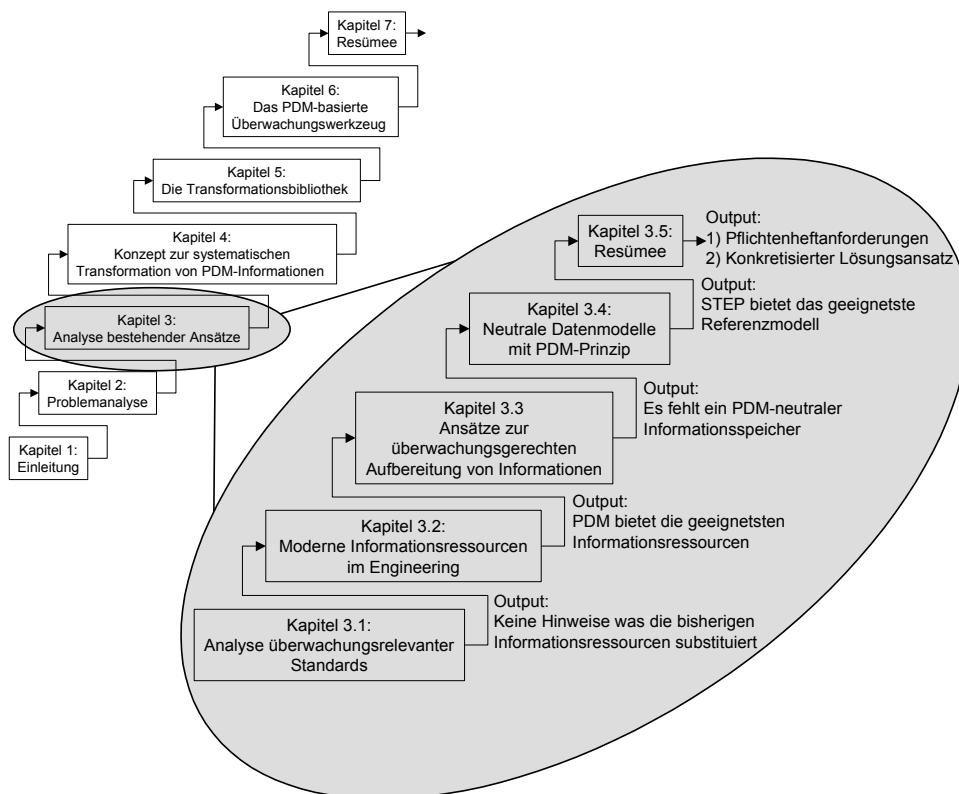


Abb. 3-1: Aufbau und Vorgehensweise Kapitel 3

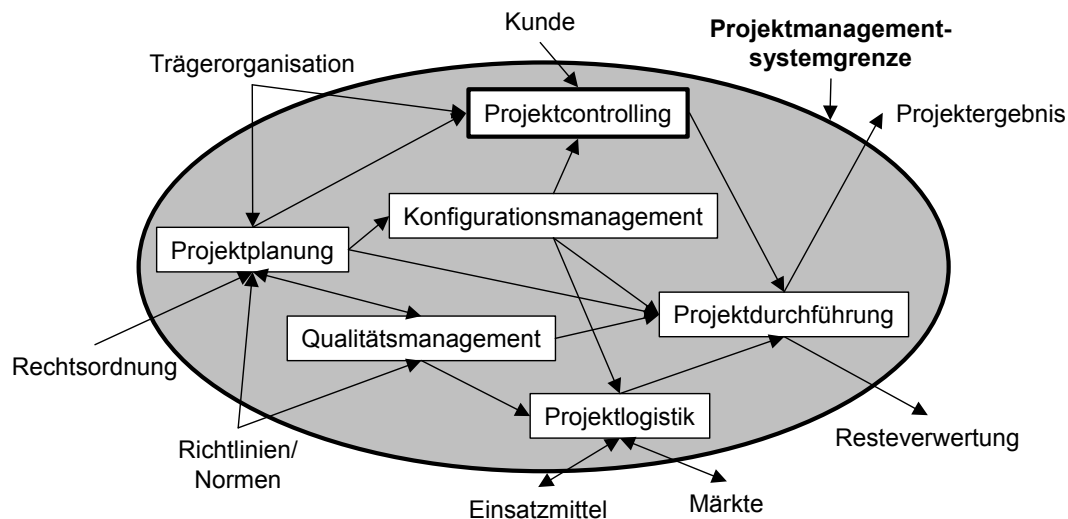
### **3.1 Analyse überwachungsrelevanter Standards**

Ziel von Kapitel 3.1 ist es, Standards zu identifizieren, die konkrete Anhaltspunkte für eine Substitution traditioneller Überwachungsressourcen bieten. Dazu stellt die Arbeit drei Richtlinien vor (DIN 69904, ISO 10006, DIN EN 14093) und diskutiert deren Stärken anhand der allgemeinen Anforderungen aus Kapitel 2.5.1.

#### **3.1.1 DIN 69904 - Elemente und Strukturen von Projektmanagementsystemen**

Mit Hilfe dieser Norm sollen Organisationen in der Lage sein, sowohl kleine und überschaubare als auch große und komplexe Projektmanagementsysteme einführen und aufrechterhalten zu können. Darüber hinaus, wie bestehende Werkzeuge zu verbessern sind [DIN 69904]. Dazu stellt die DIN 69904 sechs Anforderungen an Projektmanagementsystemen vor (Flexibilität, Universalität, Modularität, Kompatibilität, Transparenz, Prävention) [DIN 69904]. Flexibilität wird innerhalb dieses Standards als die Eigenschaft eines Projektmanagementsystems definiert, sich kurzfristig an neue bzw. veränderte Bedingungen anzupassen. Die Universalität des Werkzeuges soll ferner eine möglichst vielseitige Verwendung bzw. Nutzung garantieren. Zudem muss das Projektmanagementsystem modular aufgebaut sein, d.h. es ist aus mehreren Subsystemen zusammensetzen, die individuell entwickelt werden können. Damit das Projektmanagementsystem sowie dessen Subsysteme und Elemente mit angrenzenden Systemen verträglich ist, muss darauf geachtet werden, dass die Anforderung „Kompatibilität“ erfüllt wird. Durch Transparenz und Prävention soll schließlich garantiert werden, dass ein Projektmanagementwerkzeug einerseits Zusammenhänge und Abläufe sichtbar macht und andererseits ein frühzeitiger Eingriff in das Projektgeschehen möglich ist. [DIN 69904]. Neben diesen sechs Anforderungen stellt die DIN 69904 das allgemeingültige Modell eines Projektmanagementsystems und dessen Elemente vor, wie z.B. die Projektplanung, das Projektcontrolling etc. (siehe Abb. 3.2). Nach diesem Standard umfasst das Projektmanagementsystemelement „Projektcontrolling“ die Prozesse und Regeln, die innerhalb des Projektmanagementsystems zur Sicherung der Projektziele beitragen

durch die Erfassung von Ist-Daten, Soll-Ist-Vergleichen, Festlegung und Analyse der Abweichungen, Bewertung der Konsequenzen und Vorschlägen bzgl. Korrekturmaßnahmen, Mitwirken bei der Maßnahmenplanung und Überwachung ihrer Durchführung [DIN 69904].



**Abb. 3-2: Modell eines Projektmanagementsystems [DIN 69904]**

### 3.1.2 ISO 10006 - Guidelines to quality in project management

Ziel dieser Norm ist es, Anwender mit Projektmanagementenerfahrung insofern zu unterstützen, dass sie unabhängig von der Projektkomplexität, -dauer, -rahmenbedingungen, Projekt- und Produktart eine Verbesserung der Projektqualität erreichen können [ISO 10006]. Ein Maßnahmenkatalog, der an unternehmensspezifische Randbedingungen anzupassen ist, soll dies ermöglichen und ist nach unterschiedlichen Gesichtspunkten gruppiert (z.B. Controlling des Fortschritts, der Ressourcen, der Zeit, der Kosten, des Risikos und der Verträge). Beispielsweise sollen im Rahmen des Controllings diejenigen Teilschritte kontinuierlich identifiziert, evaluiert und dokumentiert werden, die zu Ineffizienzen

führen können. Sie sollen als Basis für Fortschritts- und Prozessevaluationen verwendet werden und um die verbleibenden Projektaktivitäten zu planen. Dabei ist gemäß der ISO 10006 sicherzustellen, dass das Sammeln und Analysieren der entsprechenden controllingrelevanten Daten effektiv und effizient durchgeführt wird.

### **3.1.3 DIN EN 13290 - Raumfahrt Projektmanagement**

Die DIN EN 13290 beinhaltet eine Reihe von Raumfahrtnormen, die sich mit dem Management, dem Engineering und der Produktsicherung in Raumfahrtprojekten und -anwendungen befassen [DIN EN 13290-1]. Die kosten- und zeitorientierte Planung, das Projektcontrolling und die Projektberichterstattung zu verbessern, ist Thema der DIN EN 13290-7. Sie stellt fest, dass die Inhalte einer manuell zu pflegenden Überwachungsdatenbasis nach drei Gesichtspunkten aufzubereiten sind: Arbeitsfortschritt, Projektkosten und Zeit. Die Überwachung des Arbeitsfortschritts und deren Dokumentation soll folgende Informationen berücksichtigen: abgeschlossene Arbeitspakete, laufende Arbeitspakete mit Angabe des Start- und Endtermins und abgeschlossene Arbeitspakete. Außerdem eine Erklärung, dass Reihenfolge, Abhängigkeiten und Vorgaben für geplante Arbeitspakete nach wie vor gültig sind. Die Projektkostenüberwachung und deren Berichterstattung soll in Abhängigkeit von Zulieferleistungen und sonstigen direkten Kosten, direkten Personalkosten etc. erfolgen.

### **3.1.4 Fazit - keine Hinweise, was die traditionellen Überwachungsressourcen ersetzt**

Bei Analyse der drei vorgestellten Richtlinien anhand der allgemeinen Anforderungen aus Kapitel 2.5.1 ist folgendes festzustellen. Alle drei Ansätze sind vergleichsweise allgemeingültig beschrieben. Auf die Lastenheftanforderungen AA 1.1, AA 1.2, AA 1.3, AA 1.4 und AA 4.1 und geht keine Richtlinie ein. Die ISO 10006 als auch die DIN EN 13290-7 haben den Bedarf an verlässlichen Informationsressourcen erkannt. Jedoch enthalten sie keine Hinweise, welche IT-Systeme genutzt werden sollten, um die Verlässlichkeit der Überwachungsergebnisse zu verbessern. Entsprechend

befriedigen sie auch nicht die Lastenheftanforderung AA 2.1 ein. Die DIN 69904 und die DIN EN 13290-7 verifizieren die Notwendigkeit einer integrierten Darstellung des Projektgeschehens (vgl. AA 3.1), doch nur die DIN EN 13290-7 geht hier noch einen Schritt weiter. Sie fordert die gleichzeitige Betrachtung von Kosten, Zeiten und Arbeitsfortschritt. Die Darstellungsform des Projektgeschehens bleibt bei den vorgestellten Standards offen. Es bleibt somit unklar, welche produktiven IT-Werkzeuge als Überwachungsgrundlage genutzt werden sollen. Kapitel 3.2 löst diese Fragestellung.

### **3.2 Moderne Informationsressourcen im Engineering**

Dieses Kapitel untersucht, inwieweit vier typische IT-Engineersysteme die bisherigen Quellen der Projektüberwachung ersetzen kann: Produktdatenmanagement Systeme (PDMS), Team Data Management Systeme (TDMS), Dokumenten-/Contentmanagement Systeme (DMS/CMS) und Produktionsplanungs- und Steuerungssysteme (PPS)

#### **3.2.1 Produktdatenmanagement Systeme (PDM)**

PDM, das synonym zu dem Begriff Engineering Data Management (EDM) verwendet wird, repräsentiert ein Konzept zur Integration und Verwaltung von Produktinformationen in allen Phasen des Produktlebenszyklus [Höfe99], [Gerh00], [Wirt01], [Glan02], [Ande03], [Karc02]. Der Ansatz des Produktdatenmanagements baut auf dem sogenannten Integrierten Produktmodell auf, das IT-Werkzeuge, wie z.B. CAD, FEM, DMU, auf Datenebene integriert [AndTri99], [AbrGer97] und Produktinformationen anhand einer frei definierbaren Produktstruktur darstellt (siehe Abb. 3-3). Zu diesen Produktinformationen zählen neben der reinen Produktgeometrie sogenannte Stamminformationen, wie z.B. Identifikationsnummer, Erstellungsdatum, Ersteller und Freigabestatus als auch Strukturinformationen, wie z.B. Gültigkeit, Kardinalität und Verwendung.

PDM-Systeme müssen auch während der Betriebsphase kontinuierlich angepasst werden, um firmenspezifische Integrations- und Verwaltungsaufgaben zu erfüllen [Wirt01]. Hierunter fallen z.B. Art und Umfang der Stamm- und Strukturinformationen, ebenso Typ und Funktionalität der Schnittstellen zu Erzeugersystemen, wie CAx und FEM sowie Verwaltungssystemen, wie TDM und DMS.

Beispiele für etablierte PDM-/TDM-Systeme sind TeamCenter (EDS), iMan (EDS) eMatrix (MatrixOne) und Windchill (Parametrix).

← Produktstruktur
Stamm- und Strukturdaten →

Item Identifier	Class	CR-Type	CR-Status Date	Life Cycle Sta	CR-Creator	Rev
W0000328,260083D1,12	Work Order	RFA	19990318	Des Closed	C0419	A
Has Resulting Part (hasRP)						
J52.....-.....,A,.....	Part/Assembly (ZPD)	---	---	Released	---	A
J52.....-.....,A,.....	Part/Assembly (ZPD)	---	---	Released	---	A
J52.....-.....,A,.....	Part/Assembly (ZPD)	---	---	Released	---	A
J52.....-.....,A,.....	Part/Assembly (ZPD)	---	---	Released	---	A
J52.....-.....,A,.....	Part/Assembly (ZPD)	---	---	Released	---	A
Uses Parts (uses)						
J52.....-.....,C,.....	Part/Assembly (ZPD)	---	---	Released	---	C
J52.....-.....,C,.....	Part/Assembly (ZPD)	---	---	Released	---	C
J52.....-.....,C,.....	Part/Assembly (ZPD)	---	---	Released	---	C

```

>>> Accepted "Has Affected Item" option, initiating object server.
>>> Accepted "Has Resulting Item" option, initiating object server.
5 objects found.

```

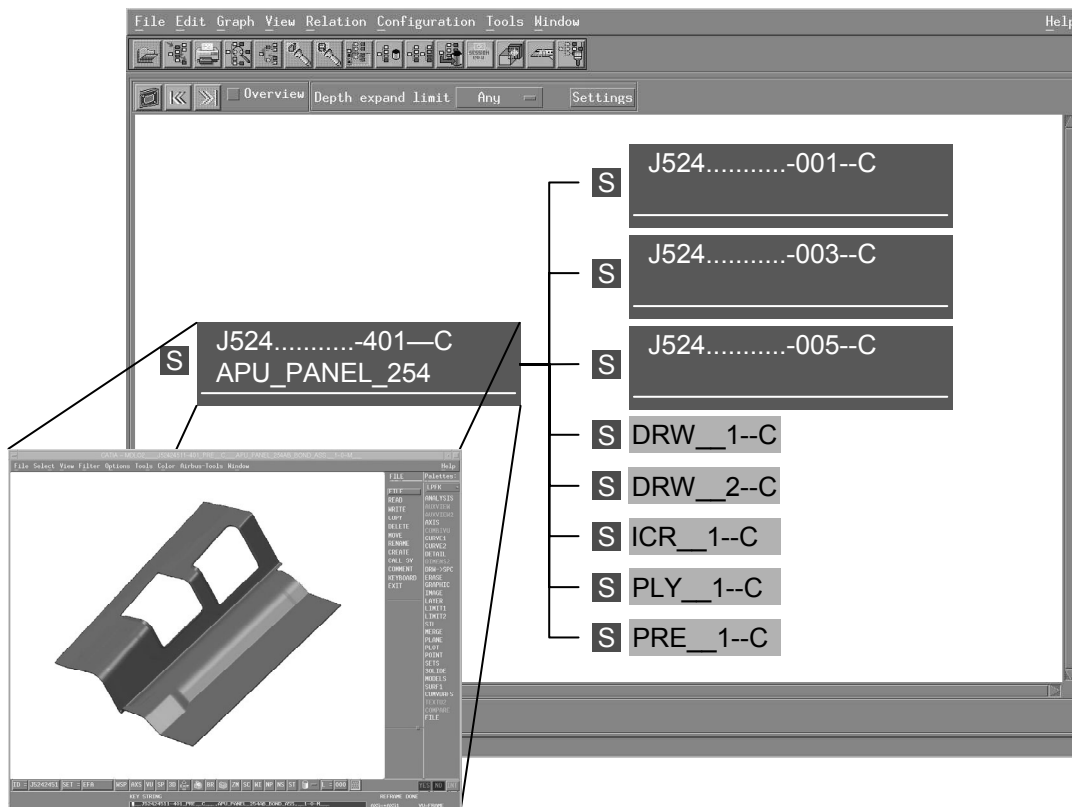
**Abb. 3-3: Produktstruktur eines Faserverbundbauteils in einem PDM-System (TeamCenter)**

### 3.2.2 Team Data Management Systeme (TDM)

TDM, das auch als „Dateimanager für CAD-Systeme“ bezeichnet wird, repräsentiert ein Konzept zur Verwaltung von CAD-Dateien, wie z.B. 2-/3D-CAD, DMU als auch deren Stamm- und Strukturinformationen anhand einer frei definierbaren Struktur (siehe Abb. 3-4). Meistens sind TDM-Systeme nur in der Lage, ein einziges

CAD-Dokumentenformat zu verwalten. Ursache ist, dass TDM- und CAD-Systeme von demselben Softwarehaus entwickelt werden.

Beispiele für etablierte TDM-Systeme sind VPM (Enovia), Pro-Intralink (Parametrics) und VirtualProductManager (IBM).



**Abb. 3-4: Produktstruktur eines Faserverbundbauteils in einem TDM-System (VPM)**

### 3.2.3 Dokumenten-/Contentmanagement Systeme (DMS/CMS)

DMS ist ein Konzept zur Verwaltung von Dokumenten, wie z.B. Lastenheft, Montageanleitung, Betriebsanleitung, Normen als auch deren Stamminformationen, wie z.B. ID, Erstellungsdatum, Ersteller, Reifegrad, und Strukturinformationen, wie z.B. Gültigkeiten und Kardinalitäten, anhand einer frei definierbaren Struktur [DIN EN 82045-1].

Der Unterschied zwischen DMS zu PDM liegt dabei darin, dass DMS PLC-Informationen auf Dateiebene integriert, PDM auf Datenebene. Beiden Ansätzen ist gemeinsam, dass sie typischerweise um Workflowfunktionen ergänzt sind.

Content Management (CM) repräsentiert eine Weiterentwicklung des DMS-Ansatzes. Hinter CM verbirgt sich ein Konzept für die Erzeugung, Verwaltung, Nutzung und Wartung von dynamischen Informationen in Webseiten, wie z.B. Wechselkurse, Preislisten, Teilelisten [KooJaeOff01].

Beispiele für DMS/CMS sind Documentum (Documentum), Domino.Doc (IBM) und Business Content Manager (Stellent).

### **3.2.4 Produktionsplanungs- und Steuerungssysteme (PPS)**

PPS, das synonym zu Engineering Resource Planning (ERP) verwendet wird, repräsentiert ein prozessorientiertes Konzept zur Planung, Steuerung und Überwachung der Produktion und des Vertriebs unter Mengen-, Termin- und Kapazitätsgesichtspunkten [Schm96], [Kurb98], [Lang00], [Wirt01], [Maut02], [Phil02], [VDI 2219]. Die vier Kernaufgaben von PPS sind Produktionsprogrammplanung, Produktionsbedarfsplanung, Fremdbezugsplanung und –steuerung und die Eigenbezugsplanung. Außerdem unterstützt der Ansatz die drei Querschnittsaufgaben Auftragskoordination, Lagerwesen und Controlling [DornEtAl90], [MucNic95], [Kurb98], [Marx00], [LucEve01].

Zu den wichtigsten PPS-Grunddaten zählen [Hack89], [Kurb98]:

- a) Teiledaten (z.B. Teilenummer, Bezeichnung, Werkstoff)
- b) Erzeugnisstrukturdaten (z.B. Menge, Variantenkennzeichen, Änderungsdatum)
- c) Arbeitsplandaten (z.B. Nummer des Arbeitsganges, Rüst- und Bearbeitungszeit)
- d) Betriebsmittel-, Arbeitsplätze-, Fertigungsstrukturdaten (z.B. Stundensatz, Leistungsgrad, Maschinen- und Personalangaben)



Beispiele für PPS sind R3 (SAP), SINET (Siemens Nixdorf), BaanSeries (IBM), Applications (Oracle) und People Soft (People Soft).

### **3.2.5 Fazit - PDM enthält die geeignetsten Überwachungsressourcen**

Da sich das TDM-Prinzip auf ein reines CAD-Modell Management beschränkt, sind innerhalb einer entsprechenden TDM-Datenbasis nur Informationen bzgl. des CAD-Modellfortschritts zu finden, wie z.B. in Form von neuen Teilstämmen und entsprechenden Stamminformationen. Zudem beinhaltet TDM keine Planungsdaten, wie z.B. Meilensteine. Bei DMS ist die Mehrzahl überwachungsrelevanter Informationen in Dateien implizit abgelegt, werden aber explizit in PDM verwaltet. PPS-Systeme beschränken sich auf zeit- und kostenbezogene Informationen für Produktion und Vertrieb (vgl. Lastenheftanforderung AA 1.3).

TDM, PDM und DMS präsentieren Informationen in Form von frei definierbaren Strukturen. PPS stellen Informationen anhand einer beliebig erzeugbaren Prozess-, Abteilungs- oder Aufgabenstruktur dar. Das bedeutet, die vorgestellten Informationssysteme erfüllen nicht die Lastenheftanforderung AA 2.1 und AA 3.1.

Das heißt, PDM bietet im Vergleich zu DMS, TDM oder PPS den größten Umfang und Detaillierungsgrad überwachungsrelevanten Daten für das Engineering, aber diese werden derzeit nicht anhand etablierter Überwachungskennlinien präsentiert. Kapitel 3.3 wird sich mit dieser Frage beschäftigen.

## **3.3 Ansätze zur überwachungsgerechten Aufbereitung von Informationen**

Entsprechend der Vorgabe aus Kapitel 3.2.5 untersucht dieses Kapitel Ansätze, die geeignet sein könnten, PDM-Informationen überwachungsgerecht zu analysieren. Sie sind in drei eher modernere (SFB 336-I3, TFB 29-T2, Progress) und zwei traditionelle Beispiele unterteilt (PMS, EUS).

### 3.3.1 SFB 336 - Teilprojekt I3

Schwerpunktmäßig wurde die Integration von Produktentwicklung und Montageplanung im abgeschlossenen Sonderforschungsbereich 336 behandelt (siehe [Schm96]). Ein Arbeitsgebiet des Teilprojektes I3 beschäftigte sich mit der Entwicklung eines Werkzeuges zur Unterstützung des standortverteilten Projektmanagements und -controllings.

Das Ergebnis zeigt sich als modulares Projektleitsystem „ProLeit“, das auf einem sogenannten „ProLeit-Regelwerk“, einer ProLeit-Datenbank und einer ProLeit-Benutzeroberfläche basiert. Die Module sind untereinander verknüpft. Mit Hilfe der graphischen Oberfläche des Prototypen lassen sich Petri-Netze, Gantt-Diagramme sowie Projektstrukturpläne erzeugen. Ihnen können folgende Metainformationen zugeordnet werden:

- a) Petri-Netze (Arbeitspaketname, Arbeitspaketverantwortlicher, Arbeitsbeginn, geplante Arbeitspaketkosten und geplante Arbeitspaketdauer)
- b) Gantt-Diagramme (Arbeitspaketname, Arbeitspaketbeginn und Arbeitspaketdauer)
- c) Projektstrukturpläne (Arbeitspaketname, Arbeitspaketverantwortlicher und Arbeitsbeginn)

Die Aktualisierung des Projektgeschehens in „ProLeit“ erfolgt manuell mit Hilfe von Dialogboxen.

### 3.3.2 TFB 29 - Teilprojekt T2

Ziel des Transferbereiches 29 war es, Ergebnisse des Sonderforschungsbereiches 336 bei einem Industriepartner umzusetzen (siehe [TFB29-T2]).

Das Ergebnis zeigt sich als Projektmonitor (siehe Abb. 3-5). Manuell zu pflegende Tabellen für Termine, Kosten, Gewichte und Funktionserfüllung bilden dessen Datenbasis. Die Tabelle „Termine“ enthält Datenfelder für die wichtigsten Projektphasen mit entsprechenden Meilensteinen. Die Tabelle „Kosten“ enthält Datenfelder für Werkzeug-, Material-, Entwicklungs- und Investitionskosten. Die Tabelle „Gewichtsstatus“ umfasst Datenfelder für Bauteile- und Gesamtgewichte. Mit Hilfe vordefinierter Verdichtungsalgorithmen werden die Datenfelder zu Überwachungskennlinien verdichtet.

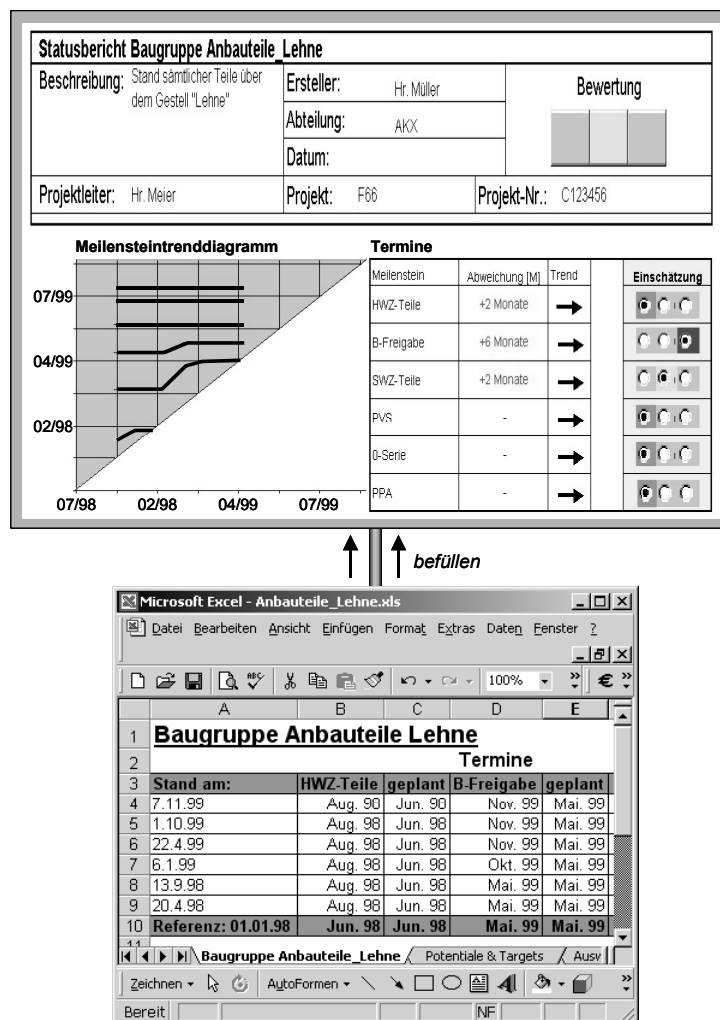


Abb. 3-5: Aufbau des TFB 29-T2 Projektmonitors

### 3.3.3 BMPF-Projekt Progress

Ziel des Projektes war es, einen Ansatz zu entwickeln, der den Zeitaufwand bis zur Markteinführung von Geräten und Systemen verkürzt. Der Schwerpunkt lag bei den Branchen Telekommunikation, Industrie- und Kfz-Elektronik (siehe [PROGRESS02]).

Ein Bereich beschäftigte sich mit der Entwicklung eines „Gesamtkonzeptes zur Effizienzsteigerung des Forschungs- und Entwicklungscontrollings“. Die wichtigsten Ergebnisse sind:

- a) Projektmonitoringkonzept
- b) Vorgehensmodell zur Prognose von Entwicklungszeiten

Das Projektmonitoringkonzept (siehe Abb. 3-6) gibt Hinweise zur Überwachung verschiedener Erfolgskriterien hinsichtlich ihrer Zielerreichung. Die Darstellung soll sowohl durch etablierte Überwachungskennlinien, wie z.B. Belastungs- und Sachfortschrittsdiagramme als auch mit Hilfe von Ampeln erfolgen. Das Konzept beinhaltet über dies hinaus folgende Erfolgskriterien zur Bewertung von Forschungs- und Entwicklungsprojekten:

- a) Finanzielle Sicht  
Entwicklungskosten, Herstellungskosten, Investitionskosten und Verkaufsprognose
- b) Zeitliche Sicht  
Lieferbeginn und Verfügbarkeit
- c) Technische Sicht  
Funktionalität, Parameter, Innovationsgrad und Qualität
- d) Marktorientierte Sicht  
Marktanalysestatus, Produkteinführungsstrategie und Unternehmensausrichtung am Markt

Diese Erfolgskriterien werden jeweils durch die Parameter Zielwert, Eskalationswert, Prognosewert und Prognosetoleranz beschrieben.

a) Zielwert

Dieser beschreibt den Sollwert eines Erfolgskriteriums (z.B. Herstellkosten = 2,00 €).

b) Eskalationswert

Er repräsentiert die Abweichung vom Sollwert (z.B. +/- 0,50 €).

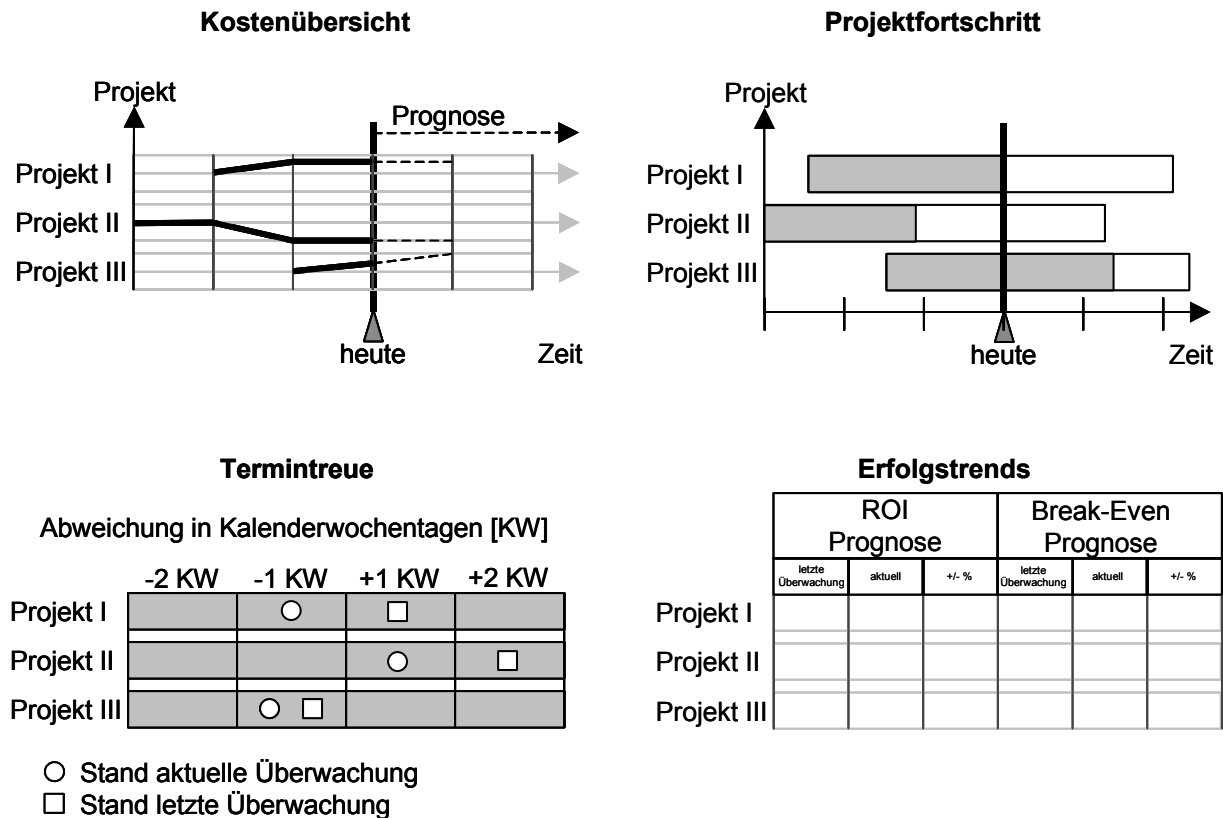
c) Prognosewert

Stellt aus Mitarbeitersicht den erwarteten künftigen Zielwert dar (z.B. Herstellkosten = 2,20 €).

d) Prognosetoleranz

Beschreibt die maximale Abweichung des künftigen Zielwertes (z.B. +/- 0,20 €).

Um den Unterstützungsgrad zu erhöhen, wird vorgeschlagen, existente Daten aus SAP-R3 und Microsoft-Projekt als Grundlage zur Berechnung der Erfolgskriterien zu nutzen.



**Abb. 3-6: Projektcockpit [PROGRESS02], [Wild03]**

Das „Vorgehensmodell zur Prognose von Entwicklungszeiten“ besteht aus drei Phasen:

a) Neutrales Vorhersagemodell

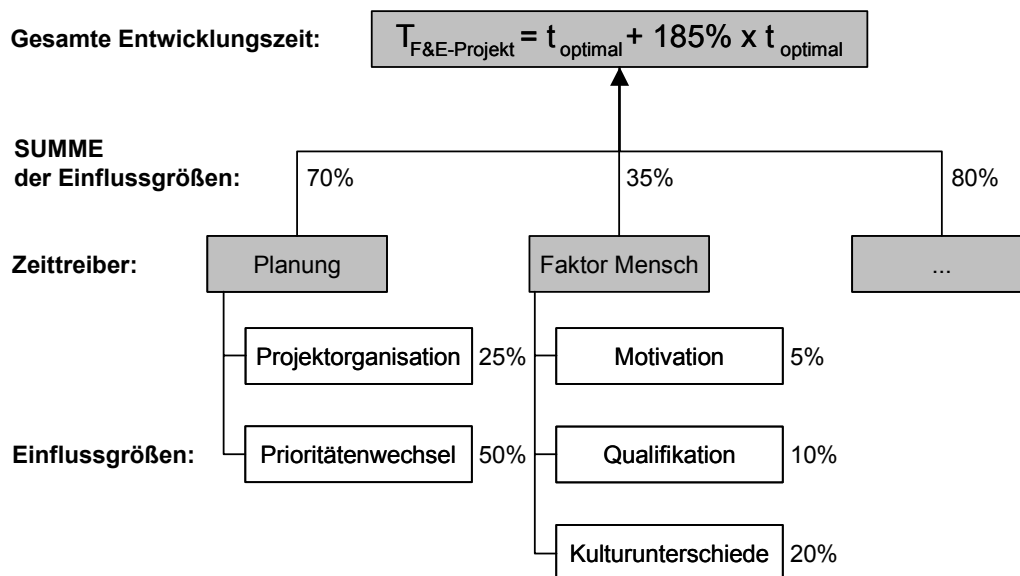
Hier findet eine Identifikation, Bewertung und Gruppierung von Kriterien statt, welche die Forschungs- und Entwicklungsdauer merklich beeinflussen (z.B. Grad der IT-Systemintegration, Prozessreife, Komplexität der Technologie, soziale Kompetenz, Projektorganisation).

b) Unternehmensspezifische Anpassung

Hier werden Wirkzusammenhänge zwischen den zuvor festgestellten Kriterien ermittelt.

## c) Operationalisierung

Hier erfolgt die Berechnung der geschätzten Forschungs- und Entwicklungsdauer (siehe Abbildung 3-7).

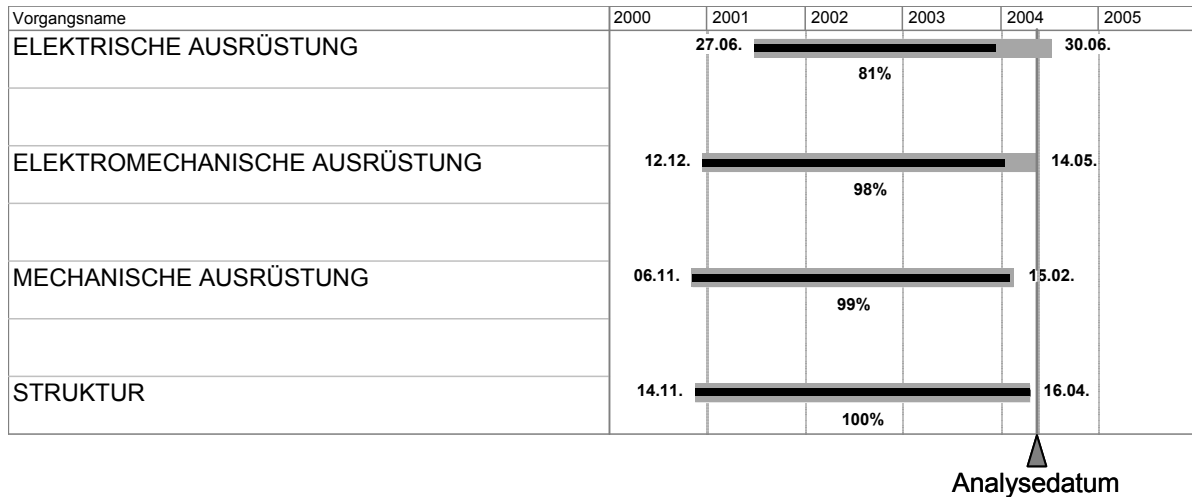


**Abb. 3-7: Berechnung der geschätzten Forschungs- und Entwicklungsdauer**

### 3.3.4 Projektmanagementsysteme (PMS)

PMS bieten die Möglichkeit zur Planung, Organisation und Steuerung aller Projektaspekte [ISO 10006], wie z.B. der Projektstruktur, des Projektablaufes, der Ressourcen, der Kapazitäten, der Termine, des Projektfortschritts und des Zielerfüllungsgrades [Litk93] (siehe Abb. 3-8).

Beispiele für PMS sind Microsoft-Projekt (Microsoft) und ARIS Projektmanager (IDS Scheer).

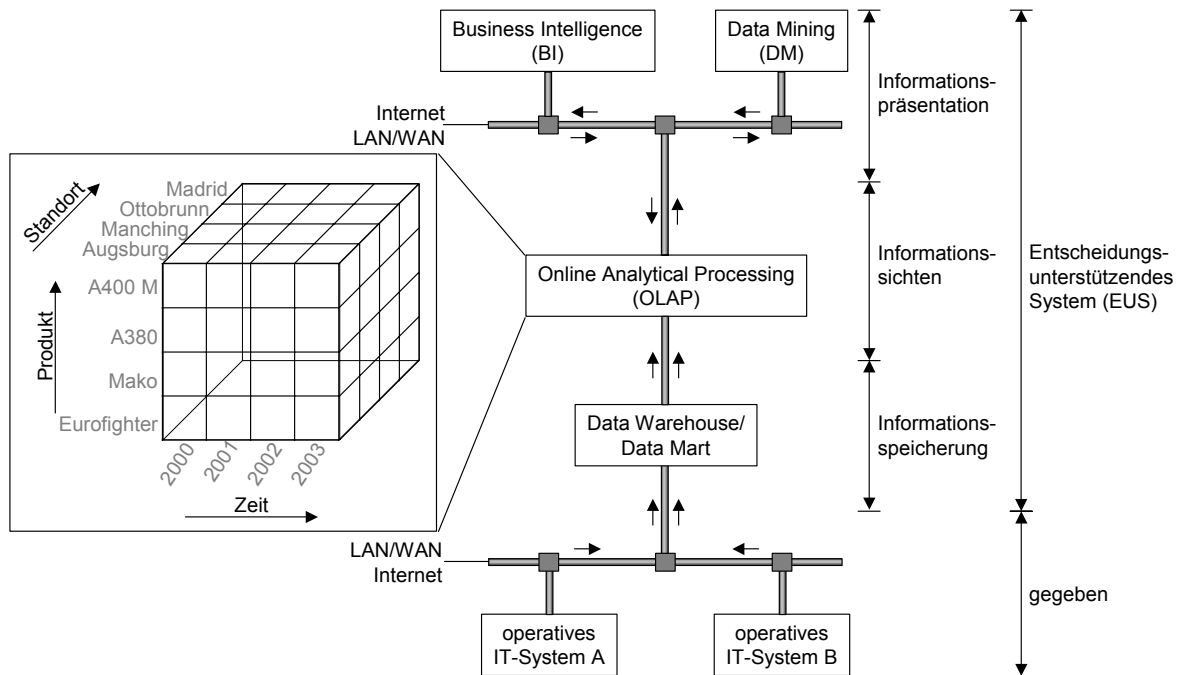


**Abb. 3-8: Gantt-Diagramm in einem PMS**

### 3.3.5 Entscheidungsunterstützende Systeme (EUS)

Die Aufgabe von EUS ist es, Daten aus verschiedenen Produktivsystemen zu verdichten und zielgruppenorientiert darzustellen [Lust99], [SchBanMer99]. Die prinzipielle Architektur eines EUS zeigt Abbildung 3-9. Alternative Begriffe zu EUS sind Executive Information Systems (EIS) und Decision Support Systems (DSS). Der Begriff des Managementinformationssystems (MIS) wird oft synonym zu EUS verwendet, obwohl MIS der Vorläufer von EUS ist. So besitzt MIS keine Möglichkeit zur unstrukturierten Suche nach Informationszusammenhängen (Stichwort: Data Mining) [SchBanMer99]. Expertensysteme, als Teilaspekt der künstlichen Intelligenz (KI), zählen nicht zu der Kategorie der EUS.





**Abb. 3-9: Architektur eines EUS [SchBanMer99]**

Den Bereich der Informationsspeicherung bildet ein so genanntes Data Warehouse, das eine themen- und zeitorientierte Datensammlung aus operativen IT-Systemen enthält [Lust99], [SchBanMer99], [GroGen00], [ChaSin04]. Dabei werden drei Data Warehouse Architekturansätze unterschieden [Lust99], [SchBanMer99], [GroGen00]:

- a) Zentrales Data Warehouse  
Es handelt sich um eine zusätzliche physikalische Datenbasis.
- b) Virtuelles Data Warehouse  
Die Daten der operativen Produktivsysteme werden direkt ausgewertet.
- c) Data Mart  
Es repräsentiert ein themen- bzw. abteilungsspezifisches Data Warehouse.

Den Bereich Informationssichten bildet das so genannte Online Analytical Processing (OLAP). OLAP, das auch als Datenwürfel bzw. Hypercube bezeichnet wird, beschreibt eine Datenmodellierungs- und Abfragemethode. Ziel ist es,

unterschiedlichen Endnutzern einen intuitiven Zugriff auf die Inhalte des Data Warehouses zu ermöglichen [Lust99], [SchBanMer99], [ChaSin04]. Dabei werden drei verschiedene Ausprägungen von OLAP unterschieden (ROLAP, MOLAP und HOLAP). ROLAP für relationale, MOLAP für multidimensionale und HOLAP für hybride Datenbanken [SchBanMer99].

Der Bereich Informationspräsentation bildet im EUS die Business Intelligence (BI) sowie das Data Mining (DM). Beiden dienen das OLAP als Datenquelle. BI analysiert die Inhalte vom OLAP entsprechend fester Algorithmen [SchBanMer99], [GroGen00], [SchBan04]. DM verdichtet die OLAP-Inhalte nach nicht vordefinierten Algorithmen, d.h. unstrukturiert. Zu ihnen zählen mathematische Algorithmen, wie z.B. die Clusteranalyse. Darüber hinaus Methoden der Künstlichen Intelligenz, wie z.B. die Entscheidungsbaummethode [Lust99], [SchBanMer99], [Lang00], [Mult00], [Mert01].

Beispiele für EUS sind Decision Support Suite (Pilot Software), Information Delivery System (SAS Institute), HOLOS (Segate) und Alea (Decisionware). MIS-Produkte sind TRIAGON (Siemens Nixdorf), SUIT (Hummingbird BI) und INFO (Segate).

### **3.3.6 Fazit - es fehlt ein PDM-übergreifender Informationsspeicher**

Trotz der gemeinsamen Vision „Nutzung der existenten digitalen Daten“ (vgl. AA 1.1) haben es die Forschungsprojekte SFB 336-I3, PROGRESS und TFB-29-T2 nicht geschafft, die operative Produktdatenbasis anzubinden (vgl. AA 2.1). Entsprechend erfolgt bei ihnen eine manuelle Erfassung von überwachungsrelevanten Daten innerhalb einer von den übrigen Produktivsystemen getrennten Datenbasis. So können diese Überwachungskonzepte ohne hohen Pflegeaufwand keine verlässlichen Überwachungskennlinien darstellen. Auch fehlt den Forschungsprojekten eine Strategie, welche die Konzeption, Umsetzung und Anpassung eines Überwachungswerkzeuges systematisch unterstützt (vgl. AA 4.1).

Die Stärken der Forschungsprojekte liegen in der Informationspräsentation (vgl. AA 3.1). Dabei adressiert der TFB-29-T2 die Engineeringüberwachung am umfassendsten. Auf der anderen Seite würde es sich bei dem TFB-29-T2, bei PMS und EUS anbieten, diese an produktive PDM-Systeme anzubinden. Bisher gibt es jedoch keine Ansätze, wie dies zu bewerkstelligen ist. Auch fehlt ihnen eine Informationsbasis, die typische PDM-Prinzipien befriedigen kann (vgl. AA 1.4). Aus diesem Grund wird sich Kapitel 3.4 mit PDM-relevanten Datenmodellen beschäftigen.

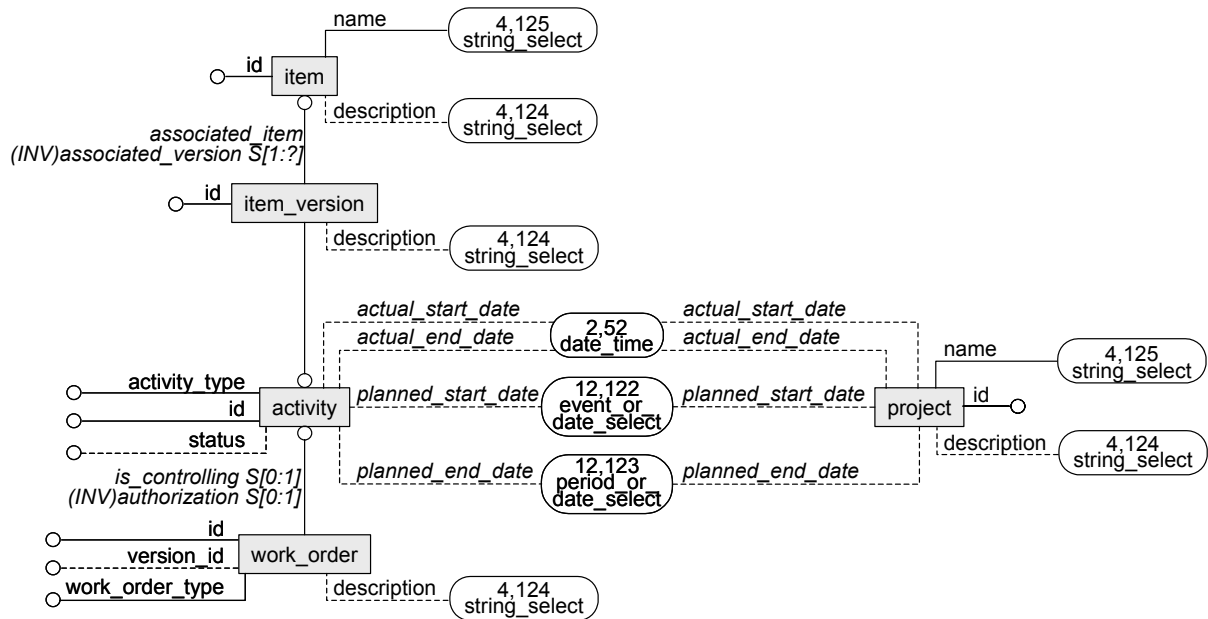
### **3.4 Neutrale Datenmodelle mit PDM-Prinzip**

Dieses Kapitel identifiziert das geeignetste neutrale Referenzmodell für den Informationsspeicher des Überwachungskonzeptes. Dazu diskutiert diese Arbeit PDM Enablers, ISO 10303, PDTnet, PDM Schema und NCDM anhand der Ergebnisse in Kapitel 2.5.1.

#### **3.4.1 PDM Enablers und die ISO 10303**

STEP (standard for the exchange of product model data) ist Synonym für eine Serie von ISO 10303-Standards, welche die rechnerinterpretierbare Darstellung und den Austausch von Produktdaten festlegen. STEP wurde aus etablierten Standards zum reinen Geometriedatenaustausch zwischen CAD Systemen entwickelt (SET, VDAFS, IGES).

STEP stellt ein Rahmenwerk zur Verfügung, das sowohl die Beschreibung als auch den Austausch von Produktdaten durch deren gesamten Lebenszyklus hindurch unabhängig von IT-System regelt. Über die bisherigen Datenaustauschstandards hinaus bietet STEP die Möglichkeit, Konfigurations-, Produktstruktur- und Gültigkeitsmanagementinformationen abzubilden (siehe Abb. 3-10). Ergänzende und weiterführende Informationen zum Thema STEP finden sich z.B. unter [AndTri99].



**Abb. 3-10: Produktstrukturaufbau nach dem STEP-AP214 Prinzip (Notation: Express G)**

Die OMG (Objekt Management Group) ist ein Zusammenschluss aus verschiedenen PDM-Systemvendoren. Im Rahmen der OMG wurde ein auf den CORBA-Services basierendes Referenzmodell entwickelt, das den Informationsaustausch zwischen verschiedenen PDM-Systemen spezifiziert (PDM Enablers). So ermöglicht der PDM Enablers den direkten Zugriff auf bestimmte Objekte und Funktionen innerhalb eines PDM-Systems [VDI 2219]. Zu der PDM Enablers Richtlinie gehört auch ein Datenmodell, welches typische PDM-Systemfunktionalitäten enthält, wie z.B. das Produktstrukturmanagement mit entsprechenden Stamm- und Strukturinformationen und das Konfigurationsmanagement mit den dazugehörigen Gültigkeiten und Kardinalitäten.

### 3.4.2 PDTnet, PDM Schema und das NCDM

Neben den vergleichsweise etablierten Datenmodellen STEP und PDM Enablers wurden in den vergangenen Jahren verschiedene Ansätze unternommen, um weitere

PDM-Referenzmodelle zu entwickeln. Zu den bekannteren zählt das PDTnet, das PDM Schema und das NATO Product Data Model (NCDM).

Das PDTnet repräsentiert einen auf XML basierenden Ausschnitt vom STEP AP214-ARM. Es wurde im Rahmen eines Verbundprojektes der deutschen Automobilindustrie entwickelt [Schi02]. Das PDM Schema repräsentiert eine Schnittmenge der ISO 10303 Applikationsprotokolle AP 214, AP 212, AP 203 und AP 232 auf AIM-Level [VilWir01]. So ermöglicht es das PDM Schema, Informationen bzgl. Identifikations-, Konfigurations-, Produktstruktur- und Gültigkeitsmanagement etc. abzubilden.

Das NCDM basiert auf diesem PDM Schema und ist in der objektorientierten Notation „Express“ beschrieben (siehe ISO 10303-11). So enthält das NCDM über das PDM Schema hinaus zusätzliche Entitäten und Attribute, um die besonderen Anforderungen des Produktsupports zu erfüllen, wie z.B. Informationen zur Logistik- und Fehleranalyse [NCO00], [JunKarVil02].

### **3.4.3 Fazit - STEP bietet das geeignetste Referenzmodell**

Es zeigte sich, dass überwachungsrelevante Informationen sowohl mit den vorgestellten etablierten als auch mit den eher moderneren Referenzmodellen explizit abgebildet werden können (vgl. AA 1.3). Vor allem bietet hier STEP die umfangreichsten Möglichkeiten. Jedoch werden überwachungsrelevante Informationen auch bei STEP auf eine Weise dargestellt, die von der Engineeringüberwachung kaum intuitiv nutzbar ist (vgl. AA 3.1).

## **3.5 Resümee**

Kapitel 3.1 bis 3.4 hat insgesamt siebzehn überwachungsrelevante Ansätze vorgestellt und deren Stärken und Schwächen anhand der allgemeinen Anforderungen aus Kapitel 2.5.1 diskutiert. Ziel ist es nun, auf Basis einer Bewertungszusammenfassung (Kapitel 3.5.1) Pflichtenheftanforderungen zur Überwindung des in Kapitel 2.4 identifizierten

Überwachungsdefizits zu formulieren (siehe Kapitel 3.5.2). Zusätzlich gilt es, auf deren Grundlage einen im Vergleich zu Kapitel 2.5.2. detaillierteren Lösungsansatz zu entwerfen (siehe Kapitel 3.5.3).

### **3.5.1 Bewertungszusammenfassung**

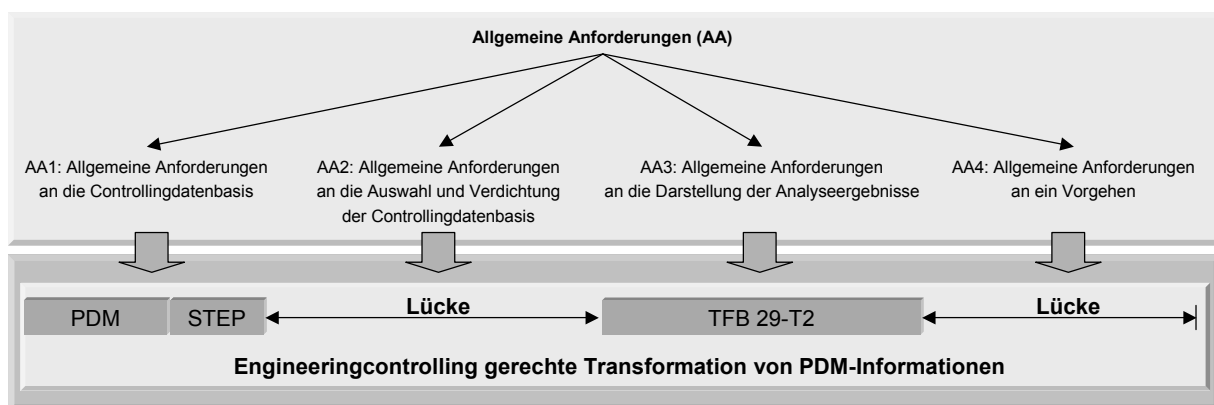
Kapitel 3.1 untersuchte drei überwachungsrelevante Standards (DIN 69904, ISO 10006, DIN EN 14093) anhand der Lastenheftanforderungen aus Kapitel 2.5.1. Wegen deren allgemeingültiger Formulierungen blieb jedoch die Frage nach realistischen Alternativen zu den bisher üblichen Informationsressourcen der Engineeringüberwachung offen.

Aus diesem Grund analysierte Kapitel 3.2 PDM, TDM, DMS/CMS sowie PPS, die vier typischen Vertreter von Verwaltungssystemen im Engineering. Hier zeigte sich, dass PDM im Vergleich die umfangreichste und verlässlichste Ressource effizient weiterzuverarbeitender Überwachungsdaten bietet. Es zeigte sich jedoch, dass das PDM- und das Überwachungsparadigma stark voneinander abweicht. Die überwachungsorientierte Aufbereitung von PDM-Informationen wird entsprechend erschwert.

Kapitel 3.3 untersuchte die fünf Ansätze zur Aufbereitung von Informationen SFB 336-I2, TFB 29-T2, Progress, PMS und EUS. Dabei wurde festgestellt, dass gerade die drei Forschungsprojekte viele konkrete Beispiele für Überwachungskennlinien bieten, wie z.B. Meilensteintrendanalysen oder Gantt-Diagramme. Die Forschungsprojekte haben die Vision von einer Nutzung existenter Informationsressourcen. Ihnen fehlen aber konkrete Hinweise, wie PDM-, PPS-, DMS- oder TDM- Informationen zu Überwachungskennlinien transformiert werden können (vgl. AA 2.1). Entsprechend ist es bei den drei Forschungsprojekten notwendig, eine zusätzliche Informationsbasis manuell zu pflegen, für deren Konzeption, Umsetzung und Anpassung keine Entwicklungsstrategie vorliegt (vgl. AA 4.1). Darüber hinaus, dass die diskutierten

Ansätze nicht auf einem Referenzmodell basieren, das typische Engineeringprinzipien befriedigen kann (vgl. AA 1.2 und AA 1.4).

Dementsprechend untersuchte Kapitel 3.4 neutrale Datenmodelle mit PDM-Prinzip: ISO 10303, PDM Enablers, PDTnet, PDM Schema und NCDM. Bei deren Diskussion stellte die Arbeit fest, dass STEP im Vergleich die meisten Engineeringcontrolling relevanten Informationen abbilden kann (vgl. AA 1.3). Abbildung 3-11 gibt die Bewertungszusammenfassung wieder.



**Abb. 3-11: Die Lücken bei der Transformation von PDM-Informationen**

### 3.5.2 Konkrete Anforderungen an ein Überwachungskonzept

Die konkreten Anforderungen, die sich durch die Diskussion von bestehenden Ansätzen ergeben, lassen sich wie folgt zusammenfassen:

KA 1: Konkretisierte Anforderungen bzgl. der Überwachungsdatenbasis

KA 2: Konkretisierte Anforderungen bzgl. Auswahl und Verdichtung

KA 3: Konkretisierte Anforderungen bzgl. der Darstellung

KA 4: Konkretisierte Anforderungen bzgl. eines Vorgehens

**KA 1: Konkrete Anforderungen bzgl. der Überwachungsdatenbasis****KA 1.1: Die bisherigen Informationsressourcen der Engineeringüberwachung werden durch PDM-Informationen ersetzt**

Begründung: PDM-Systeme bieten im Vergleich die verlässlichsten und umfangreichsten Informationsressourcen zur Engineeringüberwachung (siehe Kapitel 3.2.5).

**KA 1.2: Eine zusätzliche Controllingdatenbasis, die mit einer beliebigen PDM-Systemlandschaft koppelbar ist, basiert auf der ISO 10303**

Begründung:

- a) Die konsistente Integration und Verwaltung von Produktinformationen ist nur dann zu gewährleisten, wenn die PDM-Prinzipien weder verworfen noch die Informationen in PDM-Systemen durch das Überwachungskonzept verändert werden können (siehe Kapitel 3.2.1).
- b) Im Vergleich zu den anderen PDM-relevanten Datenmodellen ist STEP in der Lage, die größte Menge an überwachungsrelevanten Informationen abzubilden (siehe Kapitel 3.4.3).

**KA 2: Konkretisierte Anforderungen bzgl. der Auswahl und Verdichtung**

**KA 2.1: Wieder verwendbare Transformationsmuster zeigen die semantische Zuordnung zwischen STEP-Attributen und den Überwachungskennlinien des TFB 29-T2** Begründung: Im Vergleich ist STEP in der Lage, die umfangreichste Zahl von überwachungsrelevanten Informationen abzubilden. Die STEP-Semantik und Syntax ist jedoch aus Sicht des Engineeringcontrollings schwierig zu interpretieren (vgl. Kapitel 3.4.3).



**KA 2.2: Konsistente Verwaltung der Transformationsmuster**

Begründung: Ein Anwender, der ein PDM-basiertes Überwachungswerkzeug nutzt, kann nur dann die Überwachungskennlinien korrekt interpretieren, wenn ihm die sich verändernden Grundlagen der Transformation transparent sind (vgl. Kapitel 3.2.5).

**KA 3: Konkrete Anforderungen bzgl. der Darstellung****KA 3.1: Die Überwachungskennlinien werden entsprechend des TFB 29-T2 präsentiert**

Begründung: Im Vergleich adressiert der TFB 29 die Darstellungsbedürfnisse des Engineeringcontrollings am umfangreichsten (vgl. Kapitel 3.3.6).

**KA 3.2: Intuitiver Zugriff auf die formal beschriebenen Transformationsmuster** Begründung: Die Wieder- und Weiterverwendung von Transformationsmustern wird erleichtert.

**KA 4: Konkrete Anforderungen bzgl. eines Vorgehens****KA 4.1: Die Entwicklung, Anpassung und Umsetzung einer semantischen Zuordnung zwischen PDM, STEP und den Überwachungskennlinien wird systematisch unterstützt**

Begründung: Die Entwicklung, Anpassung und Umsetzung eines PDM-basierten Überwachungswerkzeuges muss trotz der sich dynamisch ändernden PDM-Systeme, Überwachungsanforderungen und trotz der Komplexität des STEP-Standards beherrschbar bleiben (vgl. Kapitel 3.2.5 und 3.4.3).

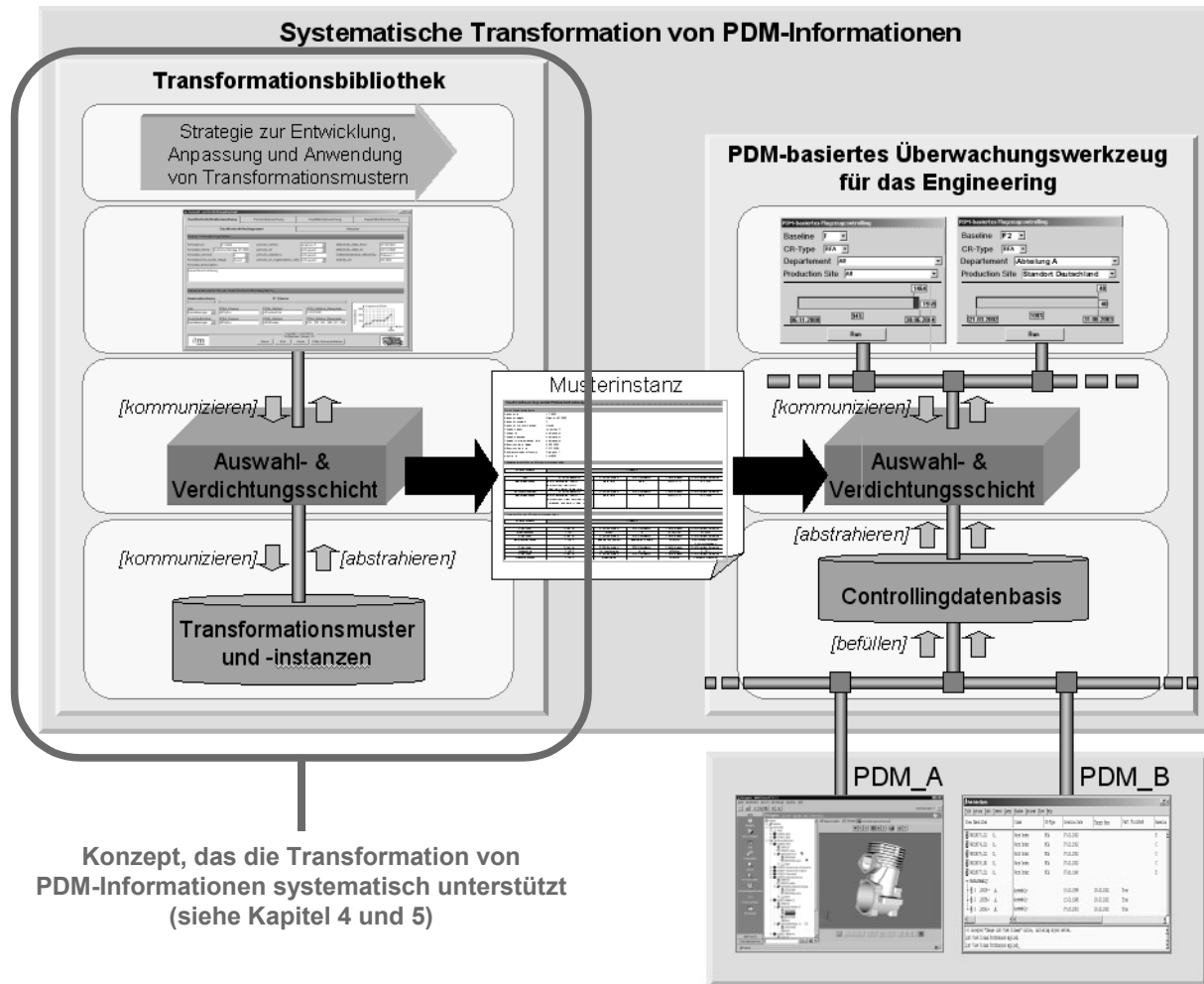
### **3.5.3 Konkreter Lösungsansatz - Transformationskonzept für PDM-Informationen**

Der zweiteilige Lösungsansatz, den Abbildung 3-12 zeigt, vereinigt die Pflichtenheftanforderungen KA 1 bis KA 4. Der linke Bildbereich gibt die Grundlage für das Metamodell in Kapitel 4.1 vor.

Im rechten Bildbereich ist das PDM-basierte Überwachungswerkzeug für das Engineering zu sehen. Die zentrale Herausforderung bei der Entwicklung eines solchen Werkzeuges ist der Semantikabgleich zwischen dem PDM- und dem Überwachungsparadigma. Diesen unterstützt eine Transformationsbibliothek, die der linke Bildbereich zeigt.

Die unterste Ebene des PDM-basierten Überwachungskonzeptes bilden PDM-Systeme (vgl. KA 1.1). Periodisch werden Informationen aus diesen operativen Systemen in eine eigenständige Controllingdatenbasis geladen, deren PDM-neutrales Datenbankmodell auf dem STEP-Standard basiert (vgl. KA 1.2). Mit Hilfe einer geschickten Auswahl und Verdichtung der nach dem STEP-Standard gespeicherten PDM-Informationen ist es möglich, Meilensteintrendanalysen, Gantt-Diagramme etc. zu erzeugen (vgl. KA 2.1 und KA 3.1).

Die unterste Schicht der Transformationsbibliothek besteht aus einer konfigurationsgesteuerten Sammlung wieder verwendbarer Transformationsmuster, die in Tabellenform eine semantische Zuordnung zwischen dem PDM- und dem Überwachungsparadigma zeigen (vgl. KA 2.1 und 2.2). Grundlage dieser Repositoryinhalte bildet eine Strategie, die es erleichtert, Transformationsmuster systematisch zu entwickeln, sie mit Hilfe eines intuitiven Bibliothekszugriffs an eine vorgegebene PDM-Landschaft systematisch anzupassen und zudem vorgibt, wie die Transformationsmuster bei der Entwicklung eines PDM-basierten Überwachungswerkzeuges anzuwenden sind (vgl. KA 3.3 und KA 4.1).





# 4. Systematische Transformation von PDM-Informationen

Aus der vergangenen Analyse hatte sich im wesentlichen herauskristallisiert, dass eine Transformation zwischen dem PDM- und dem Überwachungsparadigma die zentrale Herausforderung bei der Entwicklung und Anpassung eines PDM-basierten Überwachungssystems darstellt. Aus diesem Grund entwickelt das folgende Kapitel ein Konzept, das die Transformation von PDM-Informationen systematisch unterstützt. Dazu leitet die Arbeit ein Metamodell aus dem in Kapitel 3 erarbeiteten generellen Strukturkonzept ab. Anschließend gestalten die Kapitel 4.2 bis 4.7 die Kern- und Hilfsobjekte des Metamodells aus. Den Aufbau dieses Kapitels zeigt Abbildung 4-1.

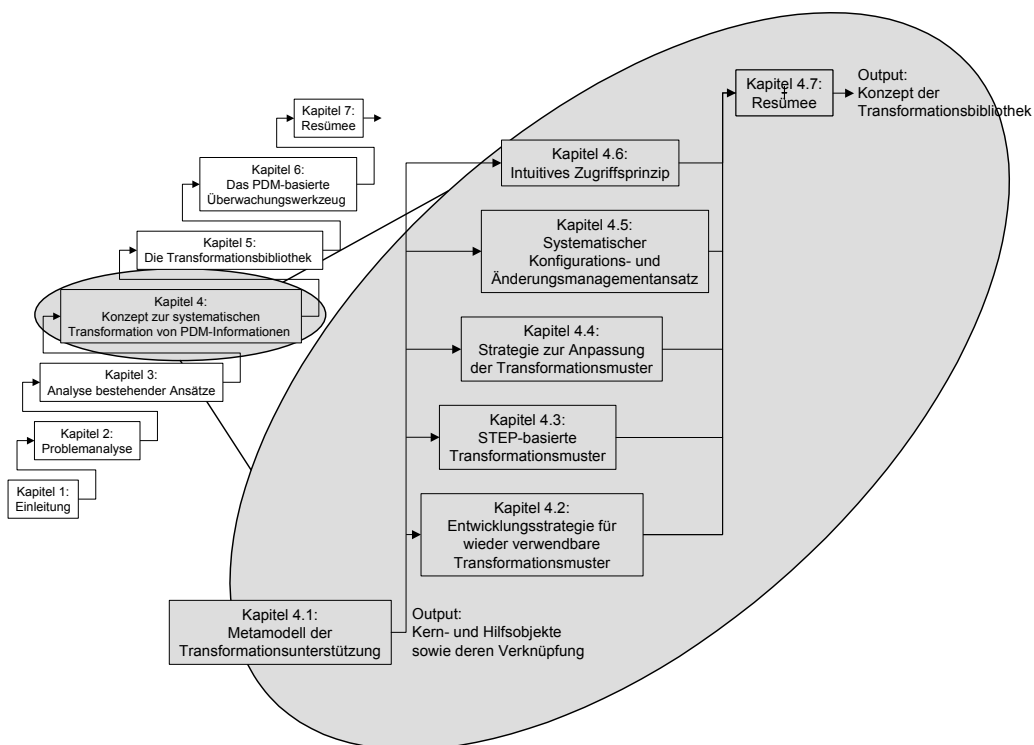


Abb. 4-1: Aufbau und Vorgehensweise Kapitel 4

#### 4.1 Metamodell der Transformationsunterstützung

Das in Abbildung 4.2 dargestellte Metamodell zeigt die Objekte der Transformationsunterstützung. Sie sind in zwei Kategorien unterteilt, die miteinander fest verbunden sind:

a) Kernobjekte

Diese Kategorie fasst Objekte zusammen, welche für die Transformationsunterstützung von zentraler Bedeutung sind: *Entwicklungsstrategie\_für\_wieder\_verwendbare\_Transformationsmuster*, *Wieder\_verwendbare\_Transformationsmuster*, *Strategie\_zur\_Anpassung\_der\_Transformationsmuster*, *Weiterverwendbare\_Musterinstanzen* und *Leitlinien\_zur\_Umsetzung\_der\_Musterinstanzen*.

b) Hilfsobjekte

Zu dieser Kategorie gehören Objekte, die den Einsatz der Kernobjekte regeln: *Transformationsbibliothek*, *Intuitives\_Zugriffsprinzip* sowie ein *Systematischer\_Konfigurations-\_und\_Änderungsmanagementansatz*.

Das Hilfsobjekt „Transformationsbibliothek“ ist das zentrale Bindeglied für die Kern- und Hilfsobjekte der Transformationsunterstützung. Darüber hinaus stellt es gewissermaßen den „Dreh- und Angelpunkt“ zur Entwicklung eines PDM-basierten Überwachungswerkzeuges dar.

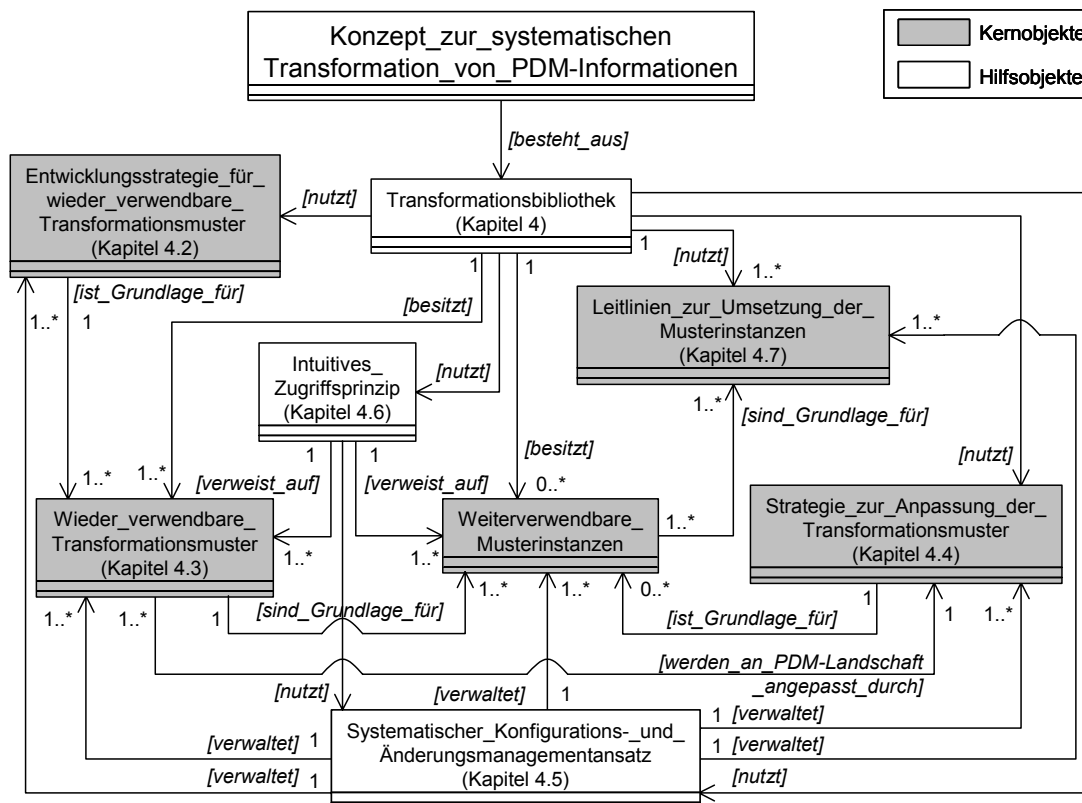
Von dem Hilfsobjekt „Transformationsbibliothek“ führt die gerichtete Assoziation „nutzt“ zu dem Hilfsobjekt „intuitives\_Zugriffsprinzip“, das die Kommunikation zwischen dem Nutzer auf der einen Seite und den beiden Kernobjekten „Wieder\_verwendbare\_Transformationsmuster“ und „Weiterverwendbare\_Musterinstanzen“ auf der anderen Seite, regelt (vgl. KA 3.2). Diese Aufgabe wird durch die zwei gerichteten Assoziationen „verweist\_auf“, ausgedrückt. Zudem zeigen die Kardinalitäten der beiden Assoziationsbeziehungen, dass es möglich ist, mehrere

„Wieder\_verwendbare\_Transformationsmuster“ und „Weiterverwendbare\_Musterinstanzen“ in der Transformationsbibliothek zu hinterlegen.

Das Kernobjekt „Wieder\_verwendbare\_Transformationsmuster“ ist Stellvertreter für eine Reihe von Mustern, welche die semantische Zuordnung zwischen den STEP-Attributen auf der einen Seite und einer Überwachungskennlinie auf der anderen Seite, formal und unabhängig von einem konkreten Anwendungsfall beschreiben (vgl. KA 2.1 und KA 3.2). Die gerichtete Assoziation „ist\_Grundlage\_für“ drückt aus, dass das Kernobjekt „Wieder\_verwendbare\_Transformationsmuster“ unter Zuhilfenahme des Kernobjektes „Entwicklungsstrategie\_für\_wieder\_verwendbare\_Transformationsmuster“ zu erzeugen ist (vgl. KA 4.1). Wegen der Anwendungsfall unabhängigen Auslegung des Kernobjektes „Wieder\_verwendbare\_Transformationsmuster“, muss es vor der Entwicklung eines PDM-basierten Überwachungswerkzeuges an eine konkrete PDM-Systemlandschaft angepasst werden. Das bedeutet, dass das Kernobjekt „Wieder\_verwendbare\_Transformationsmuster“ zu dem Kernobjekt „Weiterverwendbare\_Musterinstanzen“ zu überführen ist. Somit zeigt das Kernobjekt „Weiterverwendbare\_Musterinstanzen“ einerseits den fallspezifisch notwendigen STEP-Ausschnitt des Kernobjektes „Wieder\_verwendbare\_Transformationsmuster“ und andererseits die semantische Zuordnung zwischen dem fallspezifisch notwendigen STEP-Ausschnitt und den PDM-Attributen. Das Kernobjekt „Strategie\_zur\_Anpassung\_der\_Transformationsmuster“ unterstützt diesen Anpassungsvorgang (vgl. KA 4.1). Die gerichtete Assoziation „sind\_Grundlage\_für“ drückt aus, dass aus einem „Wieder\_verwendbaren\_Transformationsmuster“ eine Vielzahl von „Weiterverwendbaren\_Musterinstanzen“ erzeugt werden können.

Hinter dem Kernobjekt „Leitlinien\_zur\_Umsetzung\_der\_Musterinstanzen“ verbergen sich wichtige Grundsätze, die es zu beachten gilt, wenn „Weiterverwendbare\_Musterinstanzen“ als Grundlage zur Entwicklung eines PDM-basierten Überwachungswerkzeuges genutzt werden sollen (vgl. KA 4.1). Schließlich

übernimmt das Hilfsobjekt „Systematischer Konfigurations- und Änderungsmanagementansatz“ die Aufgabe einer konsistenten Verwaltung der beiden Kernobjekte „Wieder verwendbare Transformationsmuster“ und „Weiterverwendbare Musterinstanzen“ (vgl. KA 2.2).



**Abb. 4-2: Aufbau der Transformationsunterstützung  
(Notation: UML-Klassendiagramm)**

## 4.2 Entwicklungsstrategie für wieder verwendbare Transformationsmuster

Grundlage des hier zu erarbeitenden Kernobjektes „Entwicklungsstrategie für wieder verwendbare Transformationsmuster“ ist das Modell eines beispielhaften Transformationsprozesses (siehe Abbildung 4-3). Dieses Modell lässt sich aus dem in Abbildung 3-12 vorgestellten Lösungsansatz ableiten.



Die „Anwendungsebene“ beinhaltet vorhandene PDM-Systeme, deren Informationen die Grundlage für das Überwachungswerkzeug bilden. Aufgrund der ungleichen Philosophien der Informationsaufbereitung und -ablage muss jedoch die systematische Transformation von PDM-Informationen auf einer Ebene erfolgen (Transformationsebene), welche neben den STEP-Attributen und Kennlinienelementen auch Regeln (Transformationsregeln) beinhaltet:

a) Entsprechung (1:1)

Die Semantik eines STEP-Attributes ist mit der eines „präsentierenden Elementes“ bzw. „Skalierungselementes“ identisch. Dabei sind „präsentierende Elemente“ die elementaren Konstrukte, aus denen eine Überwachungskennlinie aufgebaut ist, wie z.B. Startdatum, Enddatum und Füllstand bei Gantt-Diagrammen. Die „Skalierungselemente“ regeln bei einer Überwachungskennlinie den Grad der Informationsdetaillierung.

b) Kombination bzw. Multiplikation (n:1 bzw. 1:n)

Bei einer „Kombination“ ergibt sich die Bedeutung eines „präsentierenden Elementes“ bzw. „Skalierungselementes“ aus einer Verknüpfung von mindestens zwei STEP-Attributen. Die „Multiplikation“ ist das Gegenteil einer „Kombination“. Das bedeutet, dass die Semantik eines STEP-Attributes mit mindestens einem „präsentierenden Element“ bzw. „Skalierungselement“ übereinstimmt.

c) Reduktion

Ausschnitte eines STEP-Attributes entsprechen der Semantik eines „präsentierenden Elementes“ bzw. „Skalierungselementes“.

d) Keine Entsprechung

Die Semantik eines „präsentierenden Elementes“ bzw. „Skalierungselementes“ kann nicht durch eine Kombination der bisherigen Transformationsregeln erfüllt werden.

Das in Abbildung 4-4 dargestellte UML-Aktivitätsdiagramm zeigt die „Entwicklungsstrategie für wieder verwendbare Transformationsmuster“, die aus dem Modell eines beispielhaften Transformationsprozesses abgeleitet wurde. Sie setzt sich aus den folgenden Aktivitäten zusammen: „Überwachungskennlinie festlegen“, „Kennlinienelemente ermitteln“, „STEP-Attribute und Kennlinienelemente semantisch vergleichen“, „Kennlinienelemente aufbrechen“ und „Semantische Verknüpfung formal dokumentieren“.

a) Überwachungskennlinie festlegen

Die Ermittlung der Lastenheftanforderungen an „wieder verwendbare Transformationsmuster“ ist Ziel dieser Aktivität. Dazu muss lediglich eine Überwachungskennlinie festgelegt werden.

c) Kennlinienelemente ermitteln

Ziel dieser Aktivität ist es, die Pflichtenheftanforderungen an „wieder verwendbare Transformationsmuster“ festzulegen. Dazu ist die Überwachungskennlinie in ihre „präsentierenden Elemente“ aufzuspalten, welche um sinnvolle „Skalierungselemente“ ergänzt werden. Beispielsweise besitzt ein Gantt-Diagramm die „präsentierenden Elemente“ Startdatum, Enddatum und Füllstand. Sinnvolle Skalierungselemente sind z.B. Abteilung, Projekt und Standorte.

d) STEP-Attribute und Kennlinienelemente semantisch vergleichen

Ziel dieser Aktivität ist es, unter Berücksichtigung der zuvor erwähnten Transformationsregeln die Bedeutungen der einzelnen „präsentierenden

Elemente“ und „Skalierungselemente“ mit den Definitionen der jeweiligen STEP-Attribute zu vergleichen, die im ISO 10303 Standard dokumentiert sind.

e) Kennlinienelemente aufbrechen

Bleibt ein semantischer Vergleich erfolglos, dann gibt es zwei Möglichkeiten. Entweder „präsentierende Elemente“ bzw. „Skalierungselemente“ sind tatsächlich nicht auf STEP projizierbar oder ein Kennlinienelement ist weiter aufzubrechen. Dies gilt es zu klären.

f) Semantische Verknüpfung formal dokumentieren

Ist ein semantischer Vergleich erfolgreich, so ist dieser in so genannten „Mapping-Tabellen“ zu dokumentieren. Sie repräsentieren Listen, die Datenfelder verschiedener Ausgangsdatenbanken zu Datenfeldern einer Zieldatenbank zuordnen [SchBanMer99]. Im Gegensatz zu diesem klassischen Verständnis wird hier die „Zieldatenbank“ durch die „präsentierenden Elemente“ und „Skalierungselemente“ der Überwachungskennlinien dargestellt und die „Ausgangsdatenbank“ durch die STEP-Attribute.

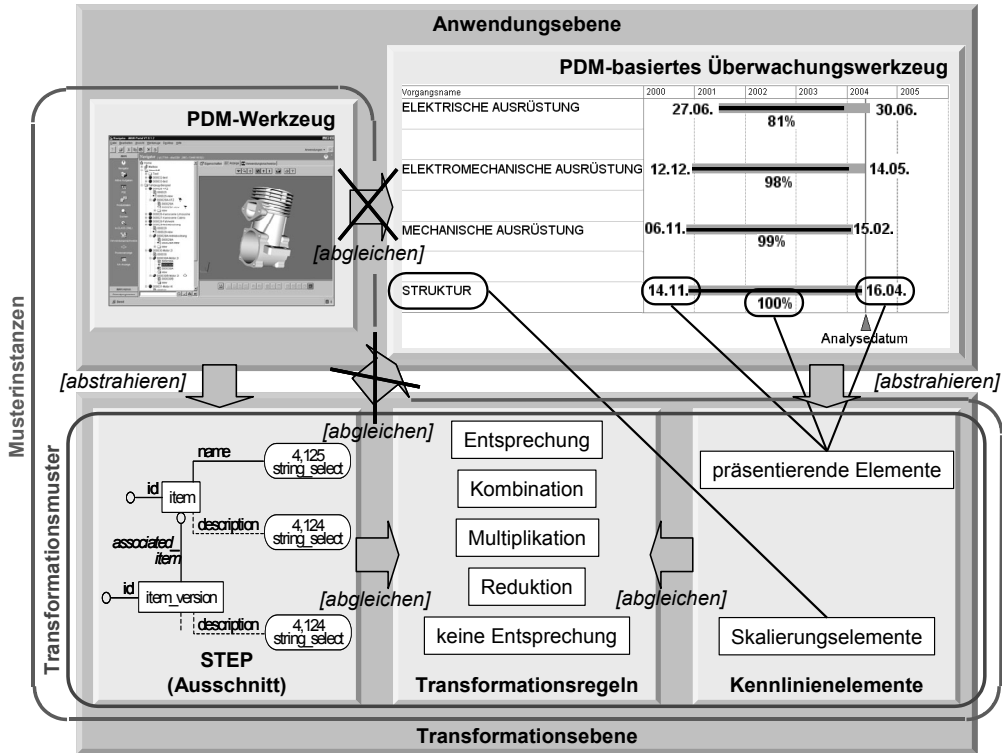


Abb. 4-3: Modell des Transformationsprozesses

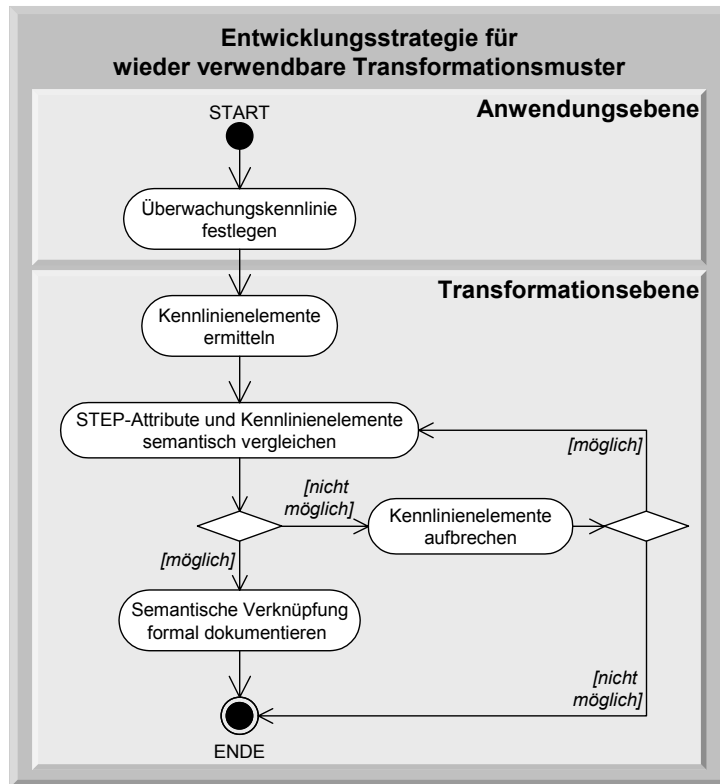


Abb. 4-4: Entwicklungsstrategie für Transformationsmuster

### 4.3 STEP-basierte Transformationsmuster

Ziel dieses Kapitel ist es „wieder verwendbare Transformationsmuster“ zu entwickeln, die auf STEP basieren und folgende Überwachungsfälle abdecken:

- 4.3.1 Sachfortschrittsüberwachung (d.h. Sachfortschrittsdiagramme und Netzpläne)
- 4.3.2 Terminüberwachung (d.h. Balkendiagramme und Belastungsdiagramm)
- 4.3.3 Qualitätsüberwachung (d.h. Änderungsquote und Gewicht)
- 4.3.4 Belastungsüberwachung (d.h. Belastungsdiagramme)

Konkrete Anwendungsbeispiele verdeutlichen die Nutzung der erarbeiteten Transformationsmuster.

#### 4.3.1 Transformationsmuster zur Sachfortschrittsüberwachung

Im folgenden werden STEP-basierte Transformationsmuster für die Sachfortschrittsüberwachung auf Basis der in Kapitel 4.2 vorgestellten Entwicklungsstrategie entworfen. Diese Transformationsmuster sind unterteilt in:

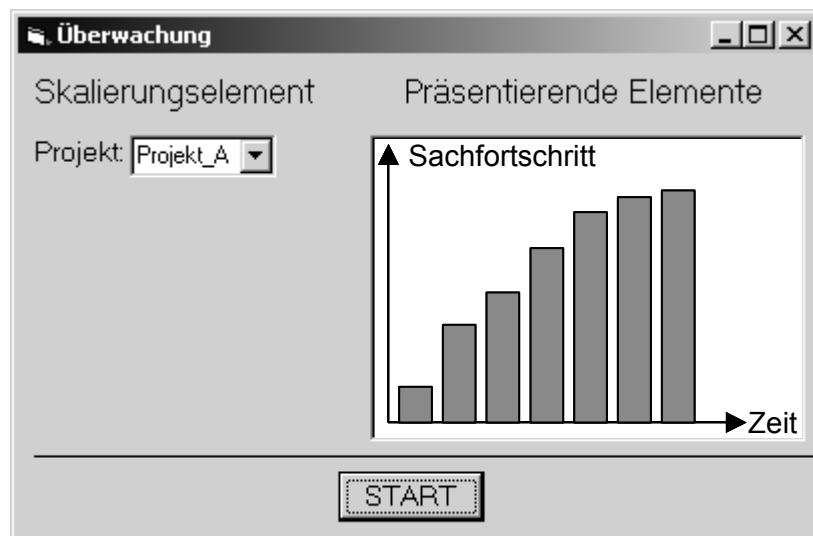
- a) Transformationsmuster für Sachfortschrittsdiagramme
- b) Transformationsmuster für Netzpläne

Die Transformationsmusterentwicklung für Sachfortschrittsdiagramme ist wie folgt unterteilt:

- a-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente
- a-2) Semantischer Vergleich und formales Dokumentieren
- a-3) Beispielhafte Nutzung des Transformationsmusters für Sachfortschrittsdiagramme

### a-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente

Der Screenshot (Abbildung 4-5) zeigt im rechten Bereich die „präsentierenden Elemente“ für ein Sachfortschrittsdiagramm, nämlich den Berichtszeitpunkt und Sachfortschritt. Der Sachfortschritt entspricht dem fertigen Arbeitsvolumen zu einem bestimmten Berichtszeitpunkt. Der linke Bereich zeigt exemplarisch das „Skalierungselement“ Projekt.



**Abb. 4-5: Beispielhafte Kennlinienelemente für Sachfortschrittsdiagramme**

### a-2) Semantischer Vergleich und formales Dokumentieren

Vergleicht man das präsentierende Element „Berichtszeitpunkt“ mit den Beschreibungen der STEP-Attribute, so lässt sich folgendes feststellen. Die Semantik von zwei STEP-Attributen („date“ und „time“) entspricht der des präsentierenden Elementes „Berichtszeitpunkt“. Die beiden STEP-Attribute und die Entität „date\_time“, die sie zusammenfasst, beschreibt STEP wie folgt:

- date\_time: Is the specification of a date and an optional time of day.
- date\_time.date: Specifies the calendar time, defined by year, month and day.
- date\_time.time: A moment of occurrence measured by hour, minute and second.

Vergleicht man die Bedeutung des Sachfortschrittes mit den Beschreibungen der STEP-Attribute, so lässt sich zunächst keine Übereinstimmung finden. Das bedeutet, es ist zu untersuchen, inwieweit das präsentierende Element Sachfortschritt in weitere Elemente aufgebrochen werden kann. Da Sachfortschritt soviel wie „fertiges Arbeitsvolumen“ bedeutet, liegt es nahe, die beiden Elemente „fertig“ und „Arbeitsvolumen“ als Vergleichsgrundlage heranzuziehen.

STEP bietet drei Möglichkeiten um Reifegradinformationen abzubilden. Zum einen enthält STEP eine eigene Entität für Reifegradinformationen, nämlich den „approval\_status“. Der notwendige Attributname heißt dabei „status\_name“. Zum anderen beinhalten die beiden Entitäten für Arbeitsanträge (work\_request) bzw. Vorgänge (activity) Reifegradinformationen. Der Attributname lautet in beiden Fällen „status“. STEP beschreibt diese Elemente wie folgt:

- approval\_status: The state of acceptance of some product data.
- approval\_status.status\_name: Specifies the terms characterizing the „approval\_status“.
- work\_request: Is the solicitation for some work to be done.
- work\_request.status: Specifies the status of the „work\_request“.
- activity: Is the fact of achieving or accomplishing an action.
- activity.status: Specifies the level of completion of the „activity“.

STEP erlaubt Statusinformationen von Arbeitsaufträgen (work\_order), Bauteilen/Baugruppen (item bzw. item\_version) und Projekten/Teilprojekten (project) sowohl über eine Verbindung mit dem Attribut „status“ der Entität „activity“ abzubilden als auch durch eine Verknüpfung mit dem Attribut „status\_name“ der Entität „approval\_status“. Zusätzlich ist eine Reifegradangabe bzgl. Arbeitsanträge (work\_request) bzw. Vorgänge (activity) durch deren Verknüpfung mit dem Attribut „status\_name“ der Entität „approval\_status“ möglich. Die Entitäten sind in STEP wie folgt definiert:

- item: Is either a single object or an unit in a group of objects. It collects the information that is common to all versions (“item\_version”) of the object.
- item\_version: Is a version of an “item” and serves as the collector of the data characterizing a physically realizable object in various application contexts.
- project: Is an identified program of work.
- work\_order: Is the authorization for one or more “activity”.

Tabelle 4-1 zeigt einen Ausschnitt des resultierenden Transformationsmusters für Sachfortschrittsdiagramme aus Anhang B. Die Spalten der gezeigten Tabelle enthalten dabei folgende Informationen:

- a) Nr.: Identifikationsnummer eines semantischen Abgleichs (z.B. F3, Z3)
- b) Element: Namen des präsentierenden Kennlinienelementes (z.B. Bauteile, Zeit)
- c) STEP-Entitäten: Textuelle Beschreibung des STEP-Datenmodellausschnittes, der für den Semantikabgleich benötigt wird (z.B. item <= item\_version => activity => actual\_start\_date => date\_time)
- d) STEP-Attribute: Die zentralen STEP-Attribute der Transformation, deren geeignete Kombination die Semantik des Kennlinienelementes wiedergibt (z.B. item.id, item\_version.id, date\_time.date, date\_time.time)



Name der gerichteten Assoziation zwischen den beiden STEP-Entitäten „item\_version“ und „item“

Gerichtete Assoziation von „item\_version“ nach „item“

Das STEP-Attribut „id“ ist Teil der STEP-Entität „item“

### "Präsentierende Elemente" von Sachfortschrittsdiagrammen und STEP

Sachfortschritt			
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:
F-6	bauteilebezogen	item <= associated_item <= item_version <= approval =>status => approval_status	item.id, item_version.id, approval_status.status_name
Zeit			
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:
Z-3	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	item.id, item_version.id, date_time.date, date_time.time

**Tab. 4-1: Ausschnitt präsentierender Elemente von Sachfortschrittsdiagrammen**

Während die präsentierenden Elemente eine Überwachungskennlinie darstellen, regeln Skalierungselemente den Grad der Informationsdetaillierung. Die Tabelle 4-2 zeigt eine Auswahl von Skalierungselementen, die im Kontext einer PDM-basierten Engineeringüberwachung relevant sind. Die vollständige Liste befindet sich im Anhang A. Analog zu Tabelle 4-1 befinden sich im linken Bildbereich die Namen der Skalierungselemente (z.B. Bauteile eines Projektes). Der rechte Bildbereich zeigt die textuelle Beschreibung des STEP-Datenmodellausschnittes, welcher für den Semantikabgleich zwischen den Skalierungselementen und STEP benötigt wird (z.B. item <= associated\_item <= item\_version => activity <= work\_programm <= project). Durch eine Überlagerung von Ausschnitten der präsentierenden und skalierenden Transformationsmuster ergibt sich das Datenbankmodell eines PDM-basierten Überwachungswerkzeuges. Abbildung 4-6 zeigt hierfür ein Beispiel.

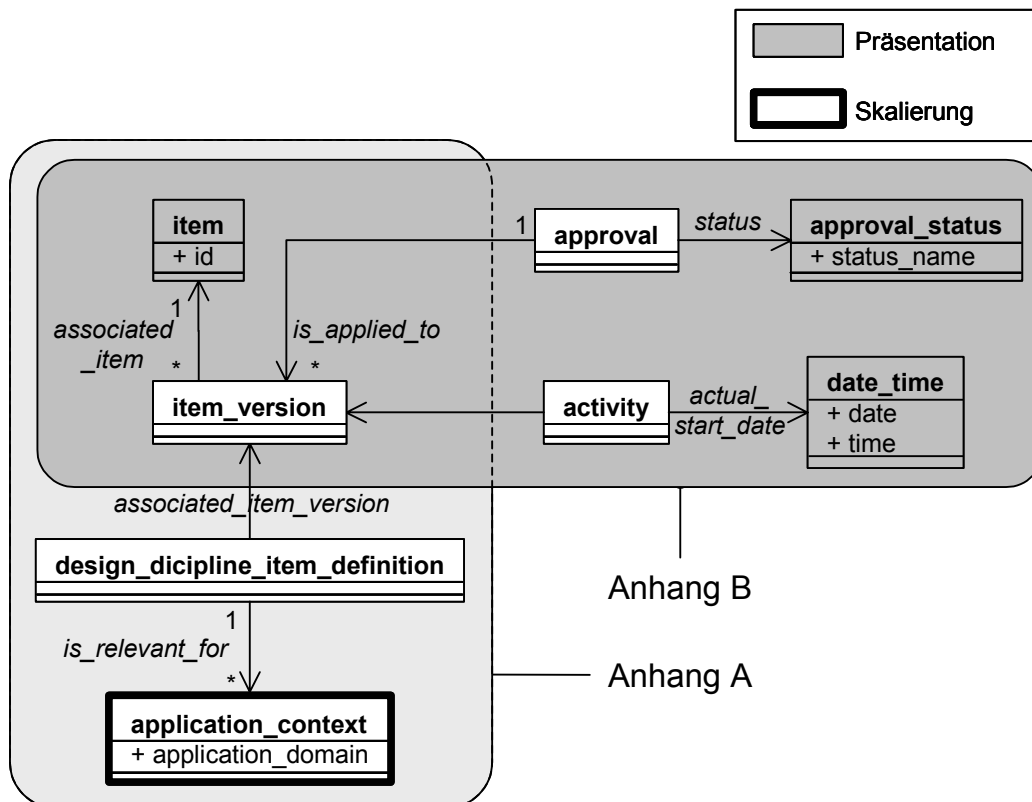
Zuordnung zwischen "Skalierungselementen" und STEP:			
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:
S-16	Bauteile eines Projektes	item <= associated_item <= item_version => activity <= work_programm <= project	item.id bzw. item_version.id, project.id
S-18	Bauteile eines Vorgangs	item <= associated_item <= item_version => activity	item.id bzw. item_version.id, activity.activity_type, activity.status, activity.id
S-27	Reifegrad von Arbeitsaufträgen	work_order => is_controlling => activity	work_order.id, work_order.version_id, work_order.work_order_type, activity.status

**Tab. 4-2: Ausschnitt der Skalierungselemente und STEP**

### a-3) Beispielhafte Nutzung des Transformationsmusters für Sachfortschrittsdiagramme

Durch Überlagerung von Transformationsmusterschnitten, die sich in Anhang A und B befinden, soll exemplarisch ein Datenbankmodell für das PDM-basierte Überwachungswerkzeug entwickelt werden. Dieses Datenmodell soll Informationen für ein bauteilebezogenes Sachfortschrittsdiagramm mit dem Skalierungselement „Anwendungsbereich“ abbilden, wie z.B. elektrische Ausrüstung oder Mechanik.

Das Transformationsmuster in Anhang B zeigt, dass der bauteilebezogene Sachfortschritt u.a. durch eine Kopplung zwischen den Entitäten „item“ und „approval\_status“ ausgedrückt werden kann (item <= associated\_item <= item\_version <= approval => status => approval\_status). Ferner, dass die Verbindung zwischen „item“ und „date\_time“ den bauteilebezogenen Berichtszeitpunkt wiedergibt (item <= item\_version => activity => actual\_start\_date => date\_time). Das Transformationsmuster in Anhang A zeigt, dass die Semantik des Anwendungsbereiches einer Verbindung zwischen den beiden Entitäten „item“ und „application\_context“ entspricht (application\_context <= ist\_relevant\_for <= design\_discipline\_item\_definition => associated\_item\_version => item\_version => associated\_item => item). Das resultierende Datenbankmodell für das PDM-basierte Überwachungswerkzeug stellt Abbildung 4-6 dar.



**Abb. 4-6: Datenbankmodell für das Sachfortschrittsbeispiel  
(Notation: UML-Klassendiagramm)**

## b) Transformationsmuster für Netzpläne

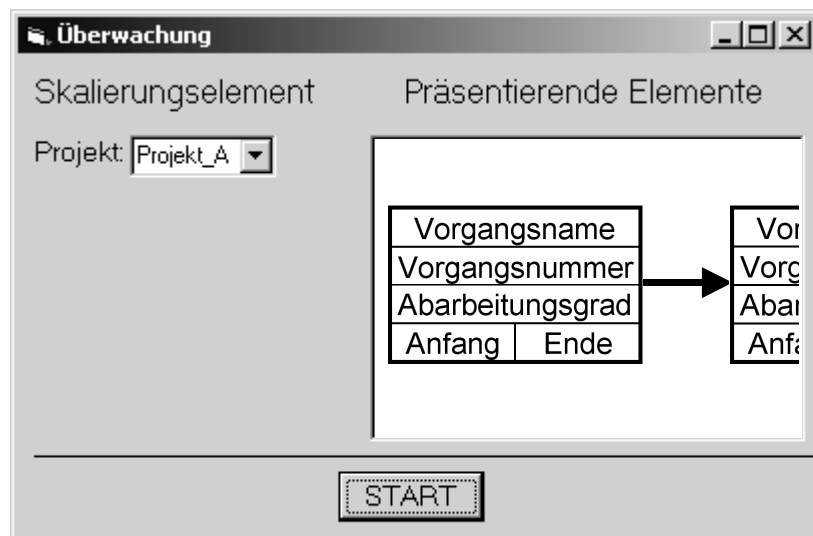
Die Transformationsmusterentwicklung für Netzpläne ist wie folgt unterteilt:

- b-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente
- b-2) Semantischer Vergleich und formales Dokumentieren
- b-3) Beispielhafte Nutzung des Transformationsmusters für Netzpläne

### **b-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente**

Analog zu den Sachfortschrittsdiagrammen soll ein Transformationsmuster für Netzpläne auf Basis von STEP entwickelt werden. Der rechte Bereich von Abbildung 4-7 zeigt die fünf „präsentierenden Elemente“ für einen Netzplan, genauer gesagt Vorgangsname, Vorgangsnummer, Abarbeitungsgrad, Vorgangsanfang und das Ende

des Vorgangs. Als Abarbeitungsgrad wird das Verhältnis zwischen fertigen Vorgängen zum gesamten Arbeitsvolumen bezeichnet. Er ist also dem Füllstand der Gantt-Diagramme gleichzusetzen. Im linken Teil der Abbildung ist das gleiche Skalierungselement zu erkennen wie im vorigen Beispiel.



**Abb. 4-7: Beispielhafte Kennlinienelemente für Netzpläne**

### b-2) Semantischer Vergleich und formales Dokumentieren

Der Vorgangsname entspricht in seiner Bedeutung dem STEP-Attribut „activity\_type“ und die Vorgangsnummer wird durch das STEP-Attribut „id“ ausgedrückt. Beide fasst die Entität „activity“ zusammen. Die Konstrukte beschreibt STEP wie folgt:

- activity: Is the fact of achieving or accomplishing an action.
- activity.id: Is the identifier of the „activity“.
- activity.activity\_type: Is the purpose of the „activity“.

Den Grad der Abarbeitung gibt das Verhältnis zwischen dem fertigen Arbeitsvolumen und dem gesamten Arbeitsvolumen wieder. Da dieser bei Netzplänen nur eine unterschiedliche Aufbereitung der gleichen STEP-Entitäten benötigt wie der

Sachfortschritt bei Sachfortschrittsdiagrammen, wird hier auf eine erneute Beschreibung der semantischen Transformation verzichtet.

Angaben zum Startdatum eines Vorganges werden durch die zwei STEP-Attribute „date“ und „time“ der STEP-Entität „date\_time“ ausgedrückt. Dabei ist jedoch zu beachten, dass STEP zwischen einem geplanten und einem tatsächlichen Startdatum unterscheidet. Die gerichtete Assoziation „planned\_start\_date“ zu der Entität „date\_time“ drückt ein geplantes Startdatum aus, d.h. ein Vorgang, ein Projekt, ein Arbeitspaket etc. wurde noch nicht begonnen. Die gerichtete Assoziation „actual\_start\_date“ zu der Entität „date\_time“ drückt hingegen ein tatsächliches Startdatum aus (siehe Tab. 4-3). STEP beschreibt die gerichteten Assoziationen wie folgt:

- planned\_start\_date: Specifies the date when the „activity“ is or was supposed to be started.
- actual\_start\_date: Specifies the date when the „activity“ actually started.

Ob die gerichtete Assoziation „planned\_start\_date“ oder „actual\_start\_date“ gültig ist, zeigt sich durch die Reifegradinformationen. Sie werden durch das Attribut „status\_name“ der Entität „approval\_status“ dargestellt. Ausnahmen bilden Arbeitsanträge (work\_request) und Vorgänge (activity). Sie besitzen ein eigenes Statusattribut (status).

Analog zum Startdatum eines Vorgangs werden Angaben zum Enddatum wahlweise durch die Attribute „date“ und/oder „time“ der STEP-Entität „date\_time“ abgebildet. Wie beim Startdatum unterscheidet STEP auch zwischen einem geplanten und einem tatsächlichen Enddatum. Die gerichtete Assoziation „planned\_end\_date“ mit der Entität „date\_time“ drückt ein geplantes Enddatum aus, d.h. ein Arbeitspaket, Projekt etc. wurde noch nicht beendet. Die gerichtete Assoziation „actual\_end\_date“ mit der

Entität „date\_time“ repräsentiert ein tatsächliches Enddatum. Die beiden gerichteten Assoziationen definiert STEP folgendermaßen:

- planned\_end\_date: Specifies the date when the „activity“ is or was supposed to be finished.
- actual\_end\_date: Specifies the date when the „activity“ actually finished.

Ob die gerichtete Assoziation „planned\_end\_date“ oder „actual\_end\_date“ gültig ist, beantwortet sich analog zum Vorgangstart durch entsprechende STEP-Attribute, die den Reifegrad darstellen.

**"Präsentierende Elemente" von Netzplänen und STEP**

<b>Vorgangname</b>			
<b>Nr.:</b>	<b>Element:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
NA-1	Name	activity	activity.activity_type,
<b>Vorgangsnummer</b>			
<b>Nr.:</b>	<b>Element:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
NU-1	Nummer	activity	activity.id
<b>Abarbeitungsgrad</b>			
<b>Nr.:</b>	<b>Element:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
A-4	arbeitsauftragsbezogen	work_order => is_controlling => activity	work_order.id, work_order.version_id, work_order.work_order_type, activity.status, activity.id, activity.activity_type
<b>Vorgangsanfang</b>			
<b>Nr.:</b>	<b>Element:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
VA-5	vorgangsbezogen	<b>Fall 1</b> die Abarbeitung des Vorgangs hat begonnen: activity => actual_start_date => date_time <b>Fall 2</b> die Abarbeitung des Vorgangs hat noch nicht begonnen: activity => planned_start_date => date_time	activity.id, activity.activity_type, activity.status, date_time.date, date_time.time
<b>Vorgangsende</b>			
<b>Nr.:</b>	<b>Element:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
VE-5	vorgangsbezogen	<b>Fall 1</b> die Abarbeitung des Vorgangs ist abgeschlossen: activity => actual_end_date => date_time <b>Fall 2</b> die Abarbeitung des Vorgangs ist noch nicht abgeschlossen: activity => planned_end_date => date_time	activity.id, activity.activity_type, activity.status, date_time.date, date_time.time

Tab. 4-3: Ausschnitt präsentierender Elemente von Netzplänen und STEP

**b-3) Beispielhafte Nutzung des Transformationsmusters für Netzpläne**

Ausgehend von den Transformationsmustern in Anhang A und C soll beispielhaft ein Datenbankmodell für das PDM-basierte Überwachungswerkzeug entwickelt werden,

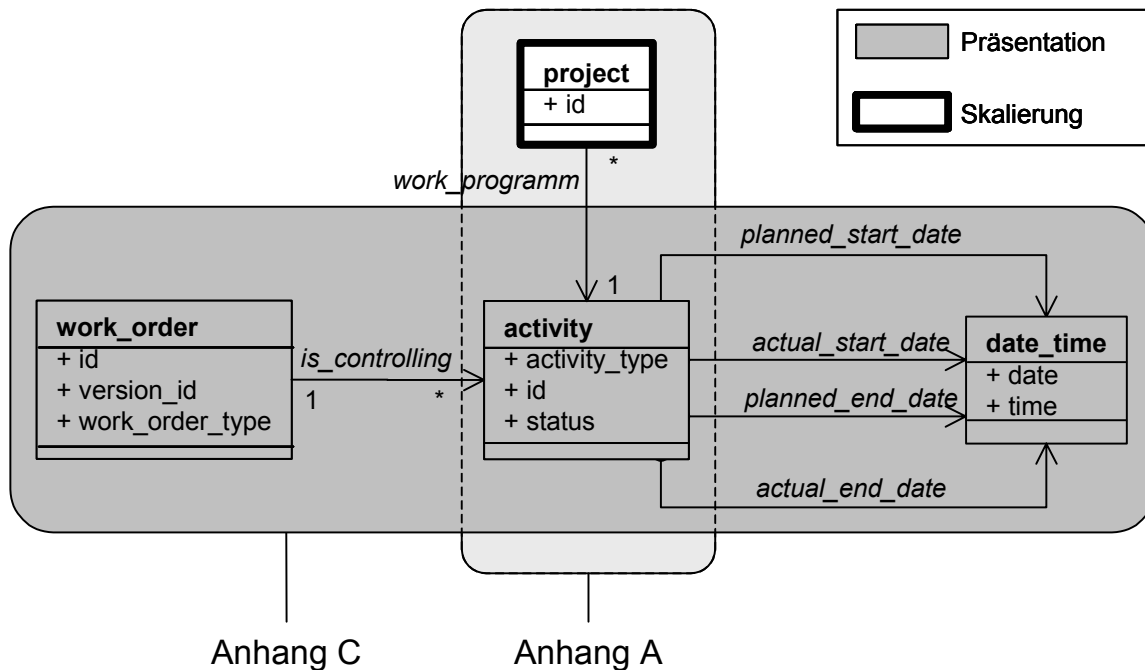
das Informationen für ein projektabhängiges Netzplandiagramm mit arbeitsauftragsbezogenem Füllstand bereitstellt.

Das Transformationsmuster für Netzpläne in Anhang C gibt vor, dass sowohl der Vorgangsname als auch die Vorgangsnummer durch die Entität „activity“ wiedergegeben wird. Die Semantik des Vorgangsstartes und des Vorgangsendes entspricht indessen einer Kopplung der beiden Entitäten „activity“ und „date\_time“.

Wurde ein Vorgang noch nicht begonnen (Fall 2), ist die Relation „planned\_start\_date“ zwischen den zwei Entitäten „activity“ und „date\_time“ zu verwenden (activity => planned\_start\_date => date\_time). Um Daten bzgl. des Endes eines bereits abgeschlossenen Vorgangs zu erhalten (Fall 1), bedarf es der Verbindung „actual\_end\_date“ zwischen der Entität „activity“ und der Entität „date\_time“ (activity => actual\_end\_date => date\_time).

Das Transformationsmuster für Skalierungselemente in Anhang A zeigt, dass die Bedeutung projektbezogener Vorgänge einer Verbindung zwischen den beiden Entitäten „activity“ und „project“ entspricht (activity <= work\_programm <= project). Das resultierende Datenbankmodell für das PDM-basierte Überwachungswerkzeug ist in Abbildung 4-8 dargestellt.





**Abb. 4-8: Datenbankmodell für einen Netzplan  
(Notation: UML-Klassendiagramm)**

### 4.3.2 Transformationsmuster zur Terminüberwachung

Im folgenden werden STEP-basierte Transformationsmuster für die Terminüberwachung auf Basis des in Kapitel 4.2 entwickelten Vorgehensmodells entworfen. Die Schritte sind wie folgt unterteilt:

- a) Transformationsmuster für Gantt-Diagramme
- b) Transformationsmuster für Meilensteintrenddiagramme

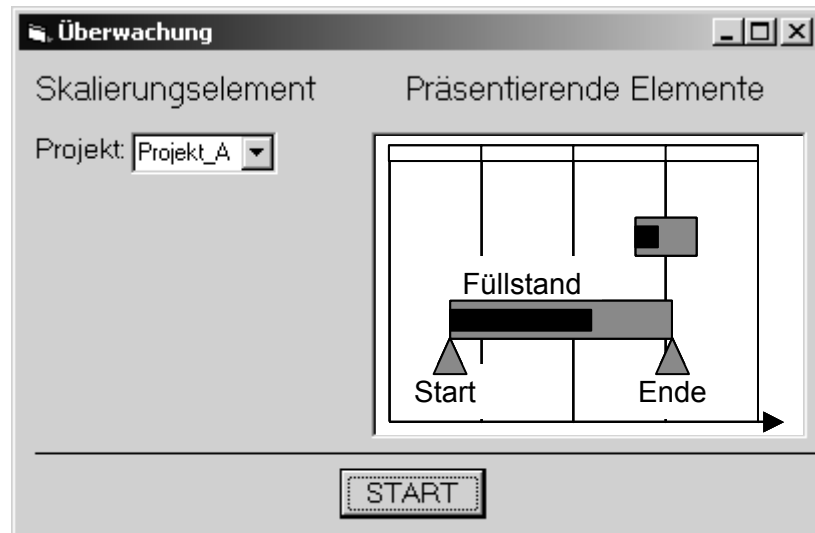
#### a) Transformationsmuster für Gantt-Diagramme

Die Transformationsmusterentwicklung für Gantt-Diagramme ist wie folgt gegliedert:

- a-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente
- a-2) Semantischer Vergleich und formales Dokumentieren
- a-3) Beispielhafte Nutzung des Transformationsmusters für Gantt-Diagramme

**a-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente**

Der rechte Bereich von Abbildung 4-9 zeigt die „präsentierenden Elemente“ von Balkendiagrammen, genauer gesagt Startdatum, Enddatum und Füllstand. Der Füllstand entspricht dem Verhältnis zwischen fertigem Arbeitsvolumen und gesamtem Arbeitsvolumen.



**Abb. 4-9: Beispielhafte Kennlinienelemente für Gantt-Diagramme**

**a-2) Semantischer Vergleich und formales Dokumentieren**

Die Semantik des Startdatums, Enddatums und Füllstandes entspricht der des Vorgangsstarts, Vorgangsendes bzw. Grades der Abarbeitung von Netz-Diagrammen. Aus diesem Grund verzichtet die Arbeit auf eine Wiederholung des semantischen Abgleichs mit STEP und verweist statt dessen auf das Transformationsmuster in Anhang D (siehe auch Tabelle 4-4).

**"Präsentierende Elemente" von Gantt-Diagrammen und STEP**

<b>Startdatum</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
SD-4	projekt-/teilprojektbezogen	<b>Fall 1</b> die Abarbeitung des Projektes oder Teilprojektes hat begonnen: activity <= work_programm <= project => actual_start_date => date_time <b>Fall 2</b> die Abarbeitung des Projektes oder Teilprojektes hat noch nicht begonnen: activity <= work_programm <= project => planned_start_date => date_time	project.id, activity.status,date_time.date, date_time.time
<b>Enddatum</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
ED-4	projekt-/teilprojektbezogen	<b>Fall 1</b> die Abarbeitung des Projektes oder Teilprojektes hat begonnen: activity <= work_programm <= project => actual_start_date => date_time <b>Fall 2</b> die Abarbeitung des Projektes oder Teilprojektes hat noch nicht begonnen: activity <= work_programm <= project => planned_start_date => date_time	project.id, activity.status,date_time.date, date_time.time
<b>Füllstand</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
FS-8	projekt-/teilprojektbezogen	project => work_programm => activity	project.id, activity.status, activity.id, activity.activity_type

**Tab. 4-4: Ausschnitt präsentierender Elemente von Gantt-Diagrammen und STEP**

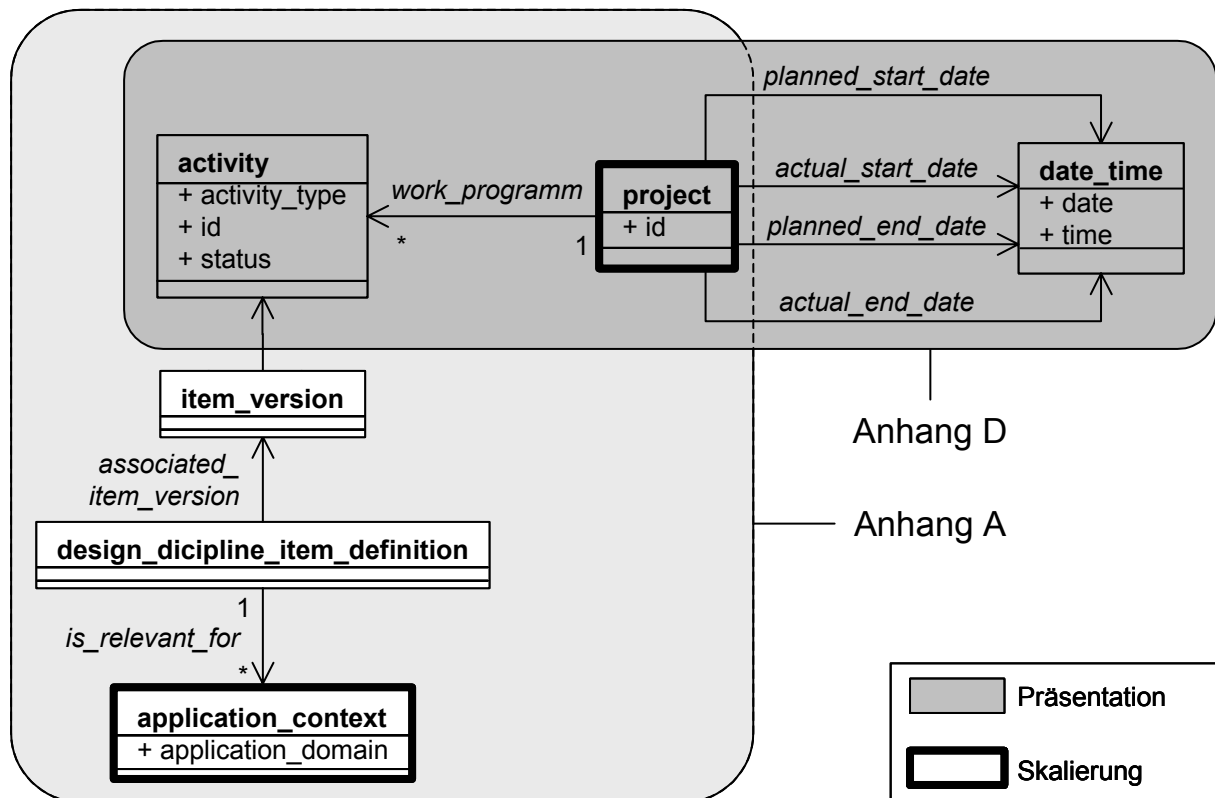
**a-3) Beispielhafte Nutzung des Transformationsmusters für Gantt-Diagramme**

Es soll ein Datenbankmodell entwickelt werden, auf dessen Grundlage sich ein projektbezogenes Gantt-Diagramm mit einer Möglichkeit zur Skalierung nach Projektverantwortlichkeiten erzeugen lässt. Dazu geben die Transformationsvorlage in Anhang A und D folgende Hinweise. Semantisch stimmt ein projektbezogener

Vorgangstart und Vorgangsende mit der Verbindung zwischen den beiden Entitäten „project“ und „date\_time“ überein. Dabei ist aber folgendes zu beachten. Um Informationen bzgl. des Projektstarts abzubilden, bedarf es der Verbindung „actual\_start\_date“ (Fall 1) zwischen den beiden Entitäten „project“ und „date\_time“ (project => actual\_start\_date => date\_time). Ist ein Projektbeginn noch in Planung (Fall 2), so gilt die gerichtete Assoziation „planned\_start\_date“ (project => planned\_start\_date => date\_time).

Damit Informationen bzgl. eines noch nicht abgeschlossenen Projektes abbildbar sind (Fall 2), bedarf es der gerichteten Assoziation „planned\_end\_date“ zwischen den zwei Entitäten „project“ und „date\_time“ (project => planned\_end\_date => date\_time). Im Gegensatz zur Entität „project“ enthalten die mit ihm assoziierten Vorgänge „activity“ ein Reifegradattribut. Das bedeutet, der projektbezogene Füllstand entspricht einer Verbindung zwischen diesen beiden Entitäten (project => work\_programm => activity).

Darüber hinaus enthält die Entität „project“ selbst kein Attribut über Projektverantwortlichkeiten. Die Transformationsvorlage im Anhang A zeigt aber, dass diese Information einer Verbindung zwischen den beiden Entitäten „application\_context“ und „project“ entspricht (application\_context <= is\_relevant\_for <= design\_discipline\_item\_definition => associated\_item\_version => item\_version => activity <= work\_programm <= project). Das resultierende Datenmodell gibt Abbildung 4-10 wieder.



**Abb. 4-10: Datenbankmodell für ein Gantt-Diagramm  
(Notation: UML-Klassendiagramm)**

### b) Transformationsmuster für Meilensteintrenddiagramme

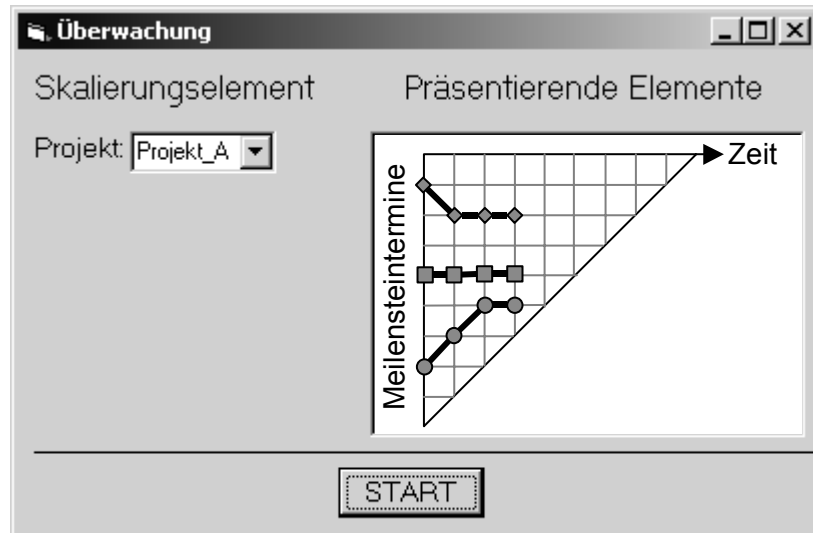
Die Transformationsmusterentwicklung für Meilensteintrenddiagramme ist wie folgt aufgebaut:

- b-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente
- b-2) Semantischer Vergleich und formales Dokumentieren
- b-3) Beispielhafte Nutzung des Transformationsmusters für Meilensteintrenddiagramme

#### **b-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente**

Die elementaren Bausteine von Meilensteintrenddiagrammen setzen sich aus den zwei Parametern Berichtszeitpunkt und Meilensteindatum zusammen

(siehe Abbildung 4-11). Das Meilensteindatum repräsentiert ein angenommenes Datum, bis zu welchem ein bestimmtes Arbeitsvolumen abgearbeitet werden soll.



**Abb. 4-11: Beispielhafte Kennlinienelemente für Meilensteintrenddiagramme**

### **b-2) Semantischer Vergleich und formales Dokumentieren**

Zum Abgleich der zwei Meilensteintrenddiagrammparameter Berichtszeitpunkt und Meilensteindatum lässt sich sagen, analog zum Sachfortschrittsdiagramm gibt den Berichtszeitpunkt die gerichtete Assoziation „actual\_start\_date“ zur Entität „date\_time“ wieder.

Angaben zum Meilensteindatum werden ebenfalls durch die Entität „date\_time“ abgebildet. Im Gegensatz zum Berichtszeitpunkt ist hier aber die gerichtete Assoziation „planned\_end\_date“ notwendig. Weil die Entität „date\_time“ für sich allein genommen keine Bedeutung hat, ist sie mit dem Arbeitsvolumen in Verbindung zu bringen. STEP erlaubt hier eine Vielzahl unterschiedlicher Kombinationsmöglichkeiten. Beispielsweise drückt die Verbindung „planned\_end\_date“ zwischen den Entitäten „project“ und „date\_time“ das Meilensteindatum von Projekten bzw. Teilprojekten aus (siehe Tabelle 4-5).

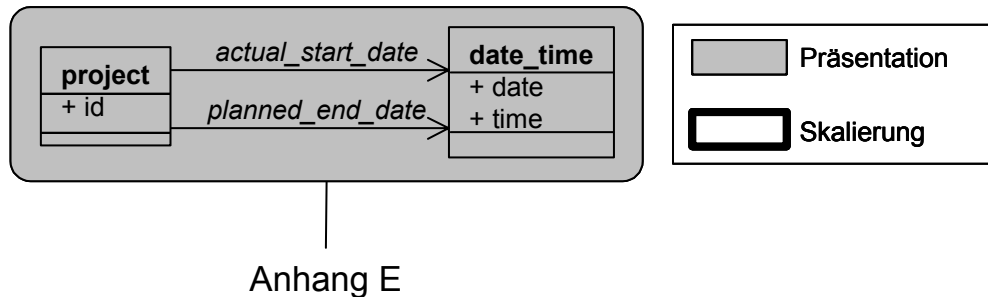
**"Präsentierende Elemente" von Meilensteintrenddiagrammen und STEP**

<b>Meilensteindatum</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
M-3	projekt-/teilprojektbezogen	project => planned_end_date => date_time	project.id, date_time.date, date_time.time
<b>Zeit</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
Z-3	projekt-/teilprojektbezogen	project => actual_start_date => date_time	project.id, date_time.date, date_time.time

**Tab. 4-5: Ausschnitt präsentierender Elemente von Meilensteintrenddiagrammen****b-3) Beispielhafte Nutzung des Transformationsmusters für****Meilensteintrenddiagramme**

Um den projekt-/teilprojektbezogenen Meilensteinverlauf darstellen zu können, entwickelt die Arbeit beispielhaft ein Datenbankmodell für ein PDM-basiertes Überwachungswerkzeug. Als Referenz dient das Transformationsmuster im Anhang E.

Auf der einen Seite zeigt das Transformationsmuster im Anhang E, dass ein projekt-/teilprojektbezogenes Meilensteindatum semantisch der gerichteten Assoziation „planned\_end\_date“ zwischen den beiden Entitäten „project“ und „date\_time“ entspricht (project => planned\_end\_date => date\_time). Andererseits, dass die Bedeutung eines projektbezogenen Berichtszeitpunktes mit der gerichteten Assoziation „actual\_start\_date“ zwischen den beiden Entitäten „project“ und „date\_time“ gleichkommt (project => actual\_start\_date => date\_time). Den Aufbau des resultierenden Datenmodells zeigt Abbildung 4-12.



**Abb. 4-12: Datenbankmodell für ein Meilensteintrenddiagramm  
(Notation: UML-Klassendiagramm)**

### 4.3.3 Transformationsmuster zur Qualitätsüberwachung

Im folgenden werden STEP-basierte Transformationsmuster für die Qualitätsüberwachung auf Basis des in Kapitel 4.2 entwickelten Vorgehensmodells partiell erarbeitet. Diese sind unterteilt in:

- a) Transformationsmuster für die Änderungsquote
- b) Transformationsmuster für das Gewicht

#### a) Transformationsmuster für die Änderungsquote

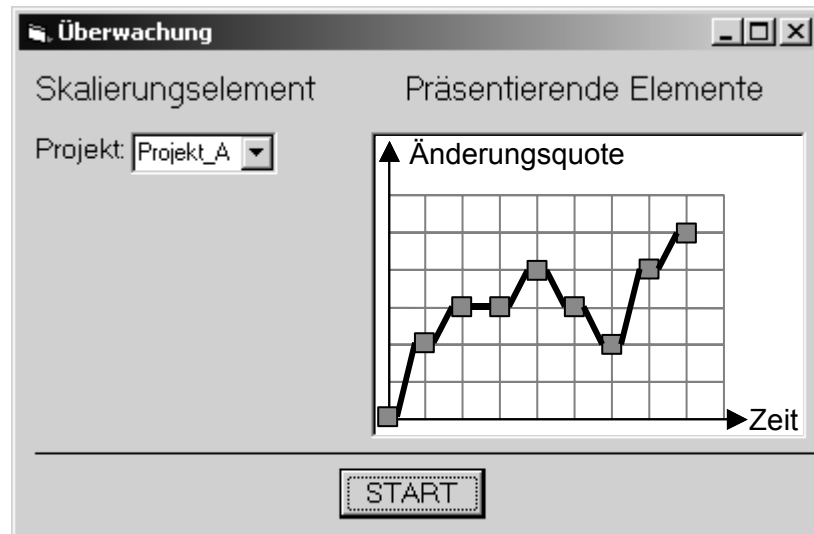
Die Transformationsmusterentwicklung für die Änderungsquote ist wie folgt gegliedert:

- a-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente
- a-2) Semantischer Vergleich und formales Dokumentieren
- a-3) Beispielhafte Nutzung des Transformationsmusters für die Änderungsquote

#### a-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente

Der rechte Bildbereich von Abbildung 4-13 zeigt die beiden präsentierenden Kennlinienelemente Änderungsquote und Berichtszeitpunkt. Die Änderungsquote entspricht dabei der Anzahl der Änderungen eines bestimmten Arbeitsvolumens.





**Abb. 4-13: Beispielhafte Kennlinienelemente für die Änderungsquote**

#### **a-2) Semantischer Vergleich und formales Dokumentieren**

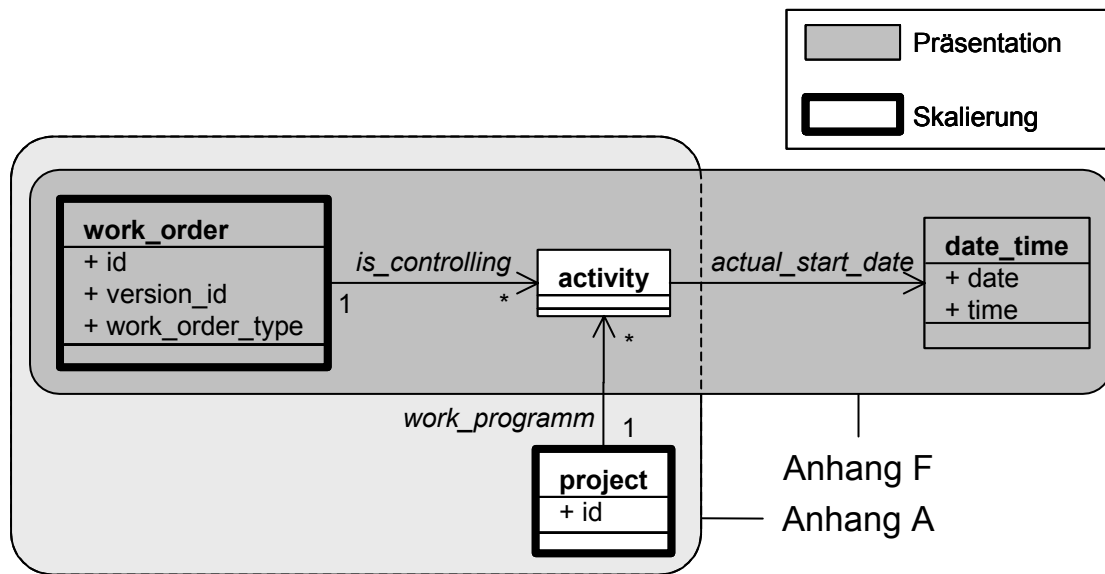
Die Semantik des Berichtszeitpunktes entspricht der gerichteten Assoziation „actual\_start\_date“ zur Entität „date\_time“, wie zuvor bei dem Sachfortschritts- und Meilesteintrenddiagramm. Wie bereits beschrieben, ist die Entität „date\_time“ für sich alleine genommen ohne Bedeutung. Das heißt, sie ist mit der Änderungsquote in Verbindung zu bringen. Angaben zur Änderungsquote sind nicht explizit im STEP-Datenmodell abgebildet und müssen daher aus bestehenden STEP-Attributen aggregiert werden, wie etwa die Änderungsquote von Arbeitsanträgen, Arbeitsaufträgen und Bauteilen/Baugruppen. Beispielsweise lässt sich aus der Verbindung zwischen der Entität „work\_order“ und der Entität „date\_time“ die Änderungsquote von Arbeitsaufträgen ermitteln (siehe Tabelle 4-6).

**"Präsentierende Elemente" der Änderungsquote und STEP**

Änderungsquote			
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:
A-2	arbeitsauftragsbezogen	work_order	work_order.id, work_order.version_id, work_order.work_order_type
Zeit			
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:
Z-2	arbeitsauftragsbezogen	work_order => is_controlling => activity => actual_start_date => date_time	work_order.id, work_order.version_id, work_order.work_order_type, date_time.date, date_time.time

**Tab. 4-6: Ausschnitt präsentierender Elemente der Änderungsquote und STEP****a-3) Beispielhafte Nutzung des Transformationsmusters für die Änderungsquote**

Im folgenden soll ein Datenmodell für die Änderungsquote von Arbeitspaketen in Abhängigkeit von Projekten ermittelt werden. Dazu geben die Transformationsvorlagen in Anhang A und F folgende Hinweise. Die Semantik der arbeitspaketebezogenen Änderungsquote lässt sich durch die beiden Attribute „id“ und „version\_id“ der Entität „work\_order“ ausdrücken. Den Berichtszeitpunkt gibt die gerichtete Assoziation „actual\_start\_date“ zwischen den Entitäten „work\_order“ und „date\_time“ wieder (work\_order => ist\_controlling => activity => actual\_start\_date => date\_time). Das resultierende Datenmodell zeigt Abbildung 4-14.



**Abb. 4-14: Datenbankmodell für eine Änderungsquote  
(Notation: UML-Klassendiagramm)**

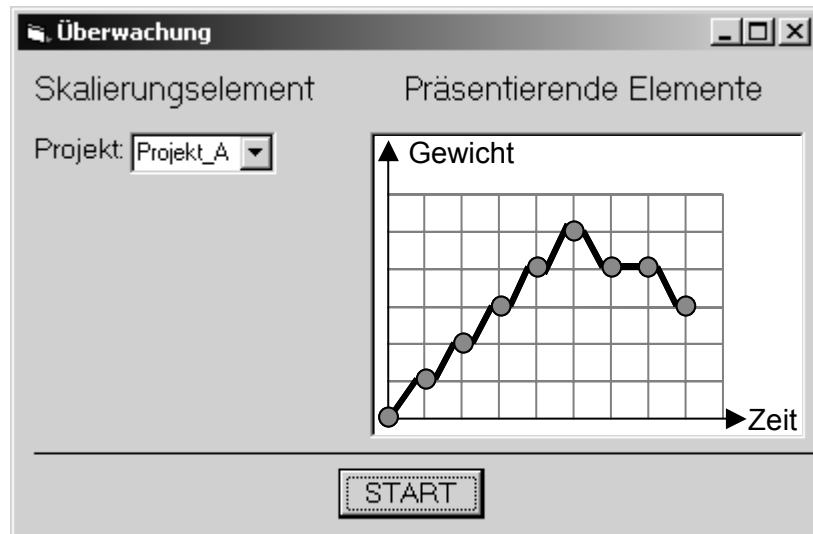
### b) Transformationsmuster für das Gewicht

Die Transformationsmusterentwicklung für das Gewicht ist wie folgt unterteilt:

- b-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente
- b-2) Semantischer Vergleich und formales Dokumentieren
- b-3) Beispielhafte Nutzung des Transformationsmusters für das Gewicht

#### **b-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente**

Die beiden präsentierenden Kennlinienelemente Gewicht und Berichtszeitpunkt zeigt der rechte Bereich von Abbildung 4-15.



**Abb. 4-15: Beispielhafte Kennlinienelemente für das Gewicht**

### **b-2) Semantischer Vergleich und formales Dokumentieren**

Die Bedeutung des Berichtszeitpunktes beim Gewicht entspricht wie zuvor der gerichteten Assoziation „actual\_start\_date“ zur Entität „date\_time“, sie ist aber ohne eine Verknüpfung zu weiteren Entitäten nicht aussagekräftig genug. Das bedeutet, es ist eine Verbindung zu den Bauteilen und deren Gewicht aufzubauen (siehe Tabelle 4-7).

Wie schon angedeutet, separiert STEP Gewichtsinformationen auf mehrere Entitäten, die untereinander zu verknüpfen sind. STEP stellt Bauteilinformationen durch die beiden Entitäten „item“ bzw. „item\_version“ dar, das Gewicht wird aber auf zwei Entitäten verteilt. Das Attribut „unit\_name“ der Entität „unit“ beschreibt die Einheit des Gewichtes (z.B. kg), das Attribut „value\_component“ der Entität „numerical\_value“ drückt hingegen die dazugehörige Menge aus (z.B. 10). STEP definiert diese Konstrukte wie folgt:

- unit: Is a quantity chosen as a standard in terms of which other quantities may be expressed.

- unit.unit\_name: Specifies the term representing the kind of „unit“ (e.g. gram, litre, volt).
- numerical\_value: Is a quantity expressed with a “unit”.
- numerical\_value.value\_component: Specifies the quantity of the “numerical\_value”.

### **"Präsentierende Elemente" des Gewichtes und STEP**

<b>Gewicht</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
G-1	bauteilebezogen	item_version => associated_item => item => material <= described_material <= material_property_ association => associated_property_ value => material_property_value => definition => material_property <= (ABS)property => allowed_unit => unit <= unit_component <= (ABS)value_with_unit => numerical_value	item.id, item_version.id, unit.unit_name, numerical_value.value_ component
<b>Zeit</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
Z-1	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	item.id, item_version.id, date_time.date, date_time.time

**Tab. 4-7: Präsentierende Elemente des Gewichtes und STEP**

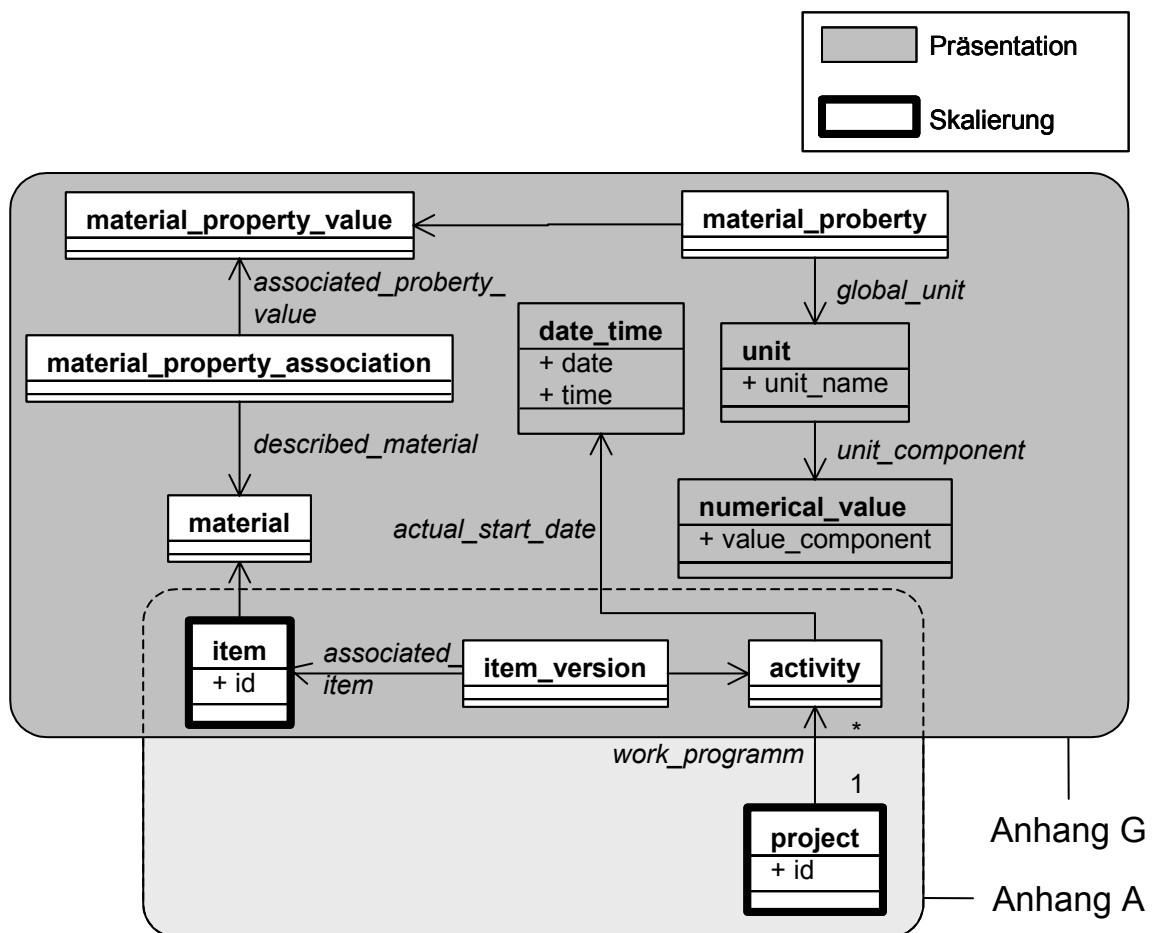
#### **b-3) Beispielhafte Nutzung des Transformationsmusters für das Gewicht**

In Abhängigkeit von Projekten soll das Bauteilegewicht ermittelt werden. Dazu bedarf es eines Datenmodells, das Abbildung 4-16 zeigt. Grundlage des Datenmodells sind die Transformationsmuster in Anhang A und G.

Das Transformationsmuster in Anhang G gibt vor, dass die Semantik des Bauteilegewichtes durch die Verknüpfung zwischen den drei Entitäten „item“, „unit“

und „nominal\_value“ erfüllt wird ( $\text{item\_version} \Rightarrow \text{associated\_item} \Rightarrow \text{item} \Rightarrow \text{material} \Leftarrow \text{described\_material} \Leftarrow \text{material\_property\_association} \Rightarrow \text{associated\_property\_value} \Rightarrow \text{material\_property\_value} \Rightarrow \text{definition} \Rightarrow \text{material\_property} \Leftarrow (\text{ABS})\text{property} \Rightarrow \text{allowed\_unit} \Rightarrow \text{unit} \Leftarrow \text{unit\_component} \Leftarrow (\text{ABS})\text{value\_with\_unit} \Rightarrow \text{numerical\_value}$ ).

Die Bedeutung des Berichtszeitpunkt entspricht darüber hinaus einer Verbindung zwischen den beiden Entitäten „item“ und „date\_time“ ( $\text{item} \Leftarrow \text{associated\_item} \Leftarrow \text{item\_version} \Rightarrow \text{activity} \Rightarrow \text{actual\_start\_date}$ ).



**Abb. 4-16: Datenbankmodell für das Gewicht  
(Notation: UML-Klassendiagramm)**

#### 4.3.4 Transformationsmuster zur Belastungsüberwachung

Nachfolgend wird ein STEP-basiertes Transformationsmuster für die Belastungsdiagramme auf Basis des in Kapitel 4.2 entwickelten Vorgehensmodells erarbeitet. Die Schritte sind wie folgt unterteilt:

- a-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente
- a-2) Semantischer Vergleich und formales Dokumentieren
- a-3) Beispielhafte Nutzung des Transformationsmusters für Belastungsdiagramme

##### a-1) Festlegen des Überwachungsauftrages und ermitteln der Kennlinienelemente

Abbildung 4-17 zeigt rechts die beiden „präsentierenden Elemente“ für ein Belastungsdiagramm Berichtszeitpunkt und Belastung (d.h. dem noch offenen Arbeitsvolumen).

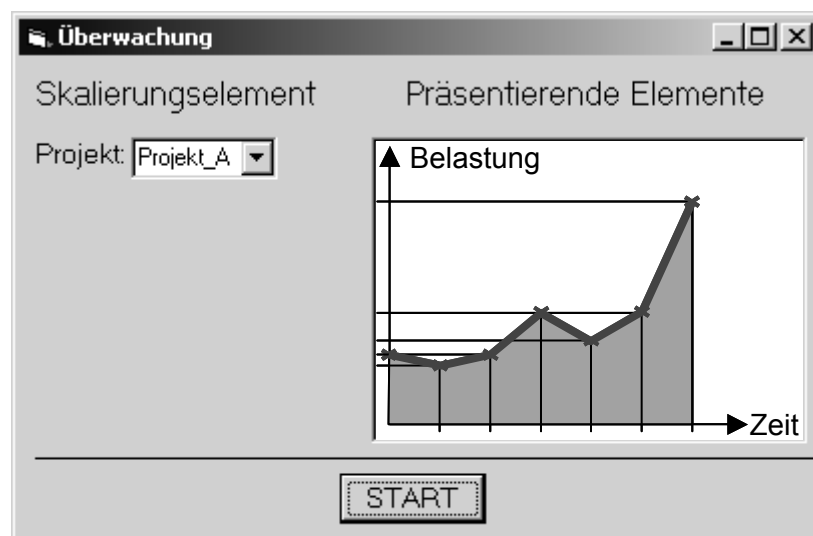


Abb. 4-17: Beispielhafte Kennlinienelemente für Belastungsdiagramme

##### a-2) Semantischer Vergleich und formales Dokumentieren

Die Transformation des Berichtszeitpunktes wurde bereits gezeigt. Außerdem beschreiben die gleichen STEP-Ausschnitte sowohl die Belastung als auch den

Sachfortschritt. Deshalb verzichtet die Arbeit auf eine Wiederholung der STEP-Transformation und verweist statt dessen auf das Transformationsmuster im Anhang H (siehe auch Tabelle 4-8).

### **"Präsentierende Elemente" von Belastungsdiagrammen und STEP**

<b>Belastung</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
B-6	Bauteile	item <= associated_item <= item_version <= approval =>status => approval_status	item.id, item_version.id
<b>Zeit</b>			
<b>Nr.:</b>	<b>Elemente</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
Z-3	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	item.id, item_version.id, date_time.date, date_time.time

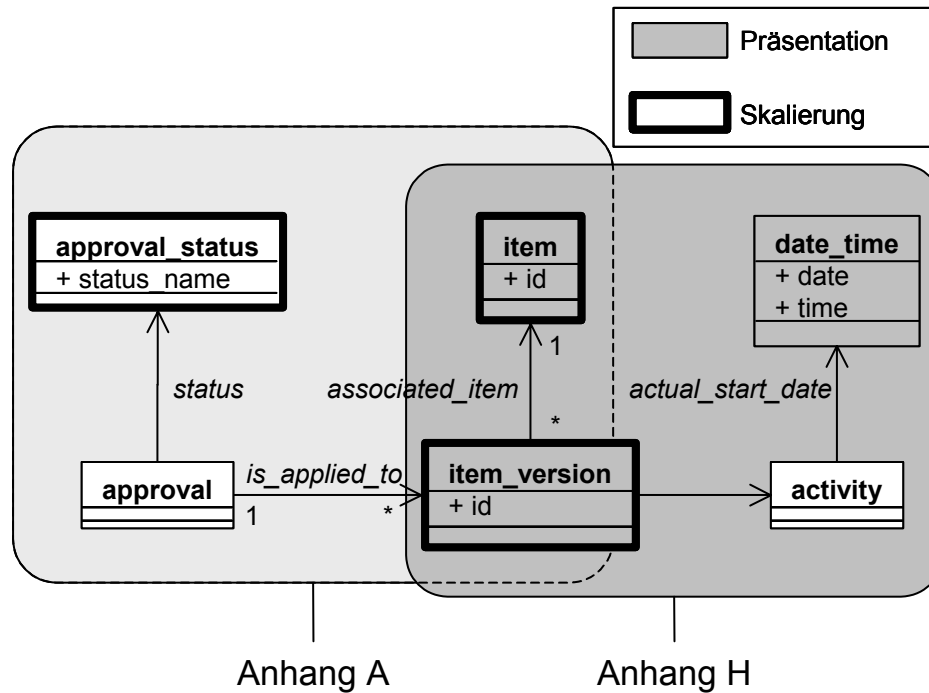
**Tab. 4-8: Ausschnitt präsentierender Elemente von Belastungsdiagrammen**

#### **a-3) Beispielhafte Nutzung des Transformationsmusters für Belastungsdiagramme**

Es soll beispielhaft ein Datenbankmodell erzeugt werden, auf dessen Grundlage die Belastung durch Bauteile dargestellt werden kann. Eine Auswahl möglicher Präsentationselemente für Belastungsdiagramme zeigt Anhang H. Eine Reihe denkbarer Skalierungselemente präsentiert Anhang A.

Anhang H veranschaulicht, dass die Zeitinformationen der Bauteile eine Verbindung zwischen den Entitäten „item“ und „date\_time“ erfordert (item <= associated\_item <= item\_version => activity => actual\_start\_date => date\_time). Wo hingegen Anhang A zeigt, dass Reifegradinformationen bzgl. Bauteile durch eine Verbindung zwischen den Entitäten „item“ und „date\_time“ ausgedrückt werden (item <= associated\_item <= item\_version <= is\_applied\_to <= approval => status => approval\_status). Das resultierende Datenbankmodell für das PDM-basierte Überwachungswerkzeug zeigt Abbildung 4-18.





**Abb. 4-18: Datenbankmodell für ein Belastungsdiagramm  
(Notation: UML-Klassendiagramm)**

### 4.3.5 Fazit - PDM-neutrale Konzeptvorlagen eines Überwachungswerkzeuges

Dieses Kapitel entwickelte Transformationsmuster für Netzpläne, Sachfortschritts-, Balken-, Meilensteintrend-, Belastungsdiagramme, für die Änderungsquote, das Gewicht und deren Skalierung. Die Transformationsmuster zeigen in tabellarischer Form die semantische Verknüpfung zwischen STEP-Attributen und den Präsentations- bzw. Skalierungselementen der Überwachungskennlinien. Durch eine Überlagerung von Transformationsmusterschnitten ergibt sich das Datenbankmodell eines PDM-basierten Überwachungswerkzeuges.

## 4.4 Strategie zur Anpassung der Transformationsmuster

Aufgrund der allgemeingültigen Auslegung des Kernobjektes „Wieder\_verwendbare\_Transformationsmuster“ muss es vor der Entwicklung eines

PDM-basierten Überwachungswerkzeuges an eine konkrete PDM-Systemlandschaft angepasst werden. Genauer gesagt, das Kernobjekt „Wieder\_verwendbare\_Transformationsmuster“ ist zu dem Kernobjekt „Weiterverwendbare\_Musterinstanzen“ zu überführen.

Das in Abbildung 4-19 dargestellte UML-Aktivitätsdiagramm zeigt hierfür eine „Entwicklungsstrategie zur Anpassung der Transformationsmuster“, die sich aus den Aktivitäten „Überwachungsauftrag festlegen“, „Transformationsausschnitte ermitteln und überlagern“, „STEP- und PDM-Attribute semantisch vergleichen“, „PDM-Attribute aufbrechen“ und „Semantische Verknüpfung formal dokumentieren“ zusammensetzt.

a) Überwachungsauftrag festlegen

Die Ermittlung der Transformationsanforderungen (Lastenheftanforderungen) an „PDM-basiertes Überwachungswerkzeug“ ist Ziel dieser Aktivität. Dazu ist neben den gewünschten Kennlinien des künftigen PDM-basierten Überwachungswerkzeuges auch deren jeweilige Skalierung festzulegen, wie z.B. ein teilprojektbezogenes Meilensteintrenddiagramm.

b) Transformationsausschnitte ermitteln und überlagern

Durch Vergleichen der Transformationsanforderungen mit den Transformationsmustern in Anhang A bis H sind die relevanten Tabellenausschnitte zu ermitteln. Durch Überlagern bzw. Verknüpfen der Tabellenausschnitte ergibt sich das Datenbankmodell des PDM-basierten Überwachungswerkzeuges. Finden sich zu den gewünschten Kennlinienelementen dennoch keine Transformationsmuster, so sind diese anhand der Entwicklungsstrategie in Kapitel 4.2 zu erweitern.

## c) STEP- und PDM-Attribute semantisch vergleichen

Ziel dieser Aktivität ist es, unter Berücksichtigung der in Kapitel 4.2 dokumentierten Transformationsregeln, die STEP-Attribute der Transformationsausschnitte mit den PDM-Attributen semantisch zu vergleichen. Um dies zu ermöglichen, ist sowohl ein aktuelles Datenbankmodell des PDM-Systems als auch ein Data Dictionary erforderlich, welches die Bedeutung und den Inhalt der Datenbanktabellen offen legt, genauer gesagt Name, Typ, Länge, Pflichtattribut ja/nein, Attributbeispiele etc. (z.B. A-01, date\_effectivity\_from, DATE, 10, ja, 23.11.1972, beschreibt den Beginn der Datumsgültigkeit).

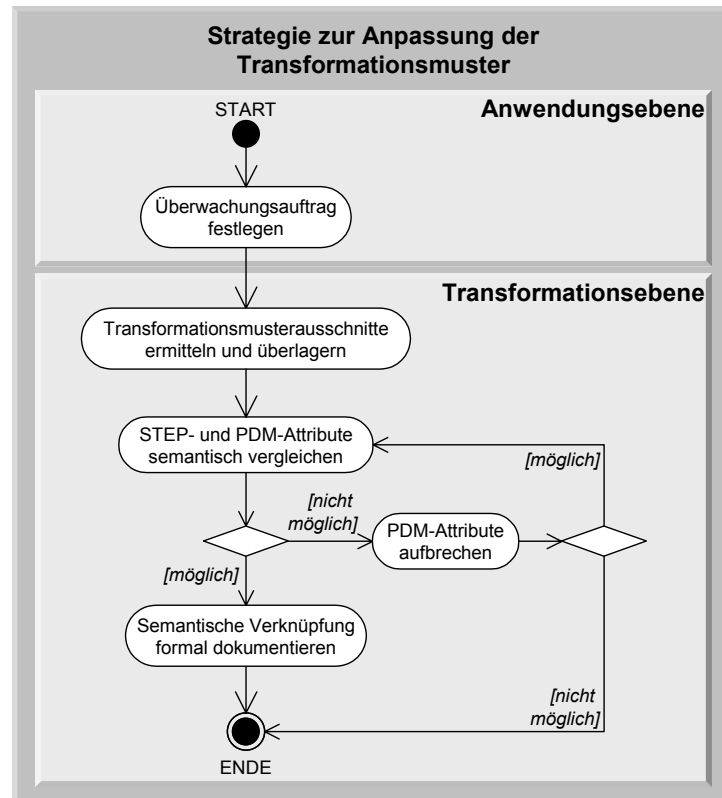
Da über einen Zeitraum von mehreren Jahren nicht automatisch garantiert werden kann, dass ein aktuelles PDM-Datenbankmodell zur Verfügung steht, bietet sich an, ein sogenanntes „Datenbank-Reverse-Engineering“ durchzuführen, d.h. eine Methode für das nachträgliche Dokumentieren von Datenbankstrukturen. Am komfortabelsten ist dabei die Nutzung von Standardsoftwareprogrammen, wie z.B. Visio von Microsoft. Darüber hinaus besteht aber auch die Chance, auf Basis entsprechender SQL-Abfragen an die PDM-Datenbank, deren Klassen, Attribute, Attributtypen, Beziehungen zwischen Klassen etc., nachträglich ein Datenbankmodell zu dokumentieren.

## d) PDM-Attribute aufbrechen

Ist eine semantische Kopplung zwischen STEP- und PDM-Attributen nicht möglich, dann gilt es zu untersuchen, ob PDM-Attribute in weitere Elemente aufgebrochen werden können.

## e) Semantische Verknüpfung formal dokumentieren

Ist ein semantischer Vergleich erfolgreich, so sind den STEP-Ausschnitten die entsprechenden PDM-Datenbankausschnitte zuzuordnen.



**Abb. 4-19: Strategie zur Anpassung der Transformationsmuster  
(Notation: UML-Aktivitätsdiagramm)**

Abbildung 4-20 zeigt beispielhaft u.a. den tabellarischen Ausschnitt einer „Weiterverwendbaren\_Musterinstanz“, deren Spalten folgende Informationen enthalten:

- e) Nr.: Identifikationsnummer eines semantischen Abgleichs zwischen Kennlinienelementen, STEP und PDM (z.B. M3, Z3)
- f) Element: Namen des Kennlinienelementes
- g) STEP-Entitäten: Textuelle Beschreibung des STEP-Datenmodellausschnittes, der für den Semantikabgleich mit den Kennlinienelementen benötigt wird (z.B. project => planned\_end\_date => date\_time)
- h) STEP-Attribute: Ihre Kombination entspricht der Semantik des Kennlinienelementes (z.B. date\_time.date, project.id)

- i) PDM-Entitäten: Textuelle Beschreibung des PDM-Datenmodellausschnittes, der für den Semantikabgleich mit STEP benötigt wird (z.B. s0WkrOdr => has\_WOPPr => WOPPr => has\_s7PhyAss => s7PhyAss)
- j) PDM-Attribute: Ihre Kombination entspricht der Semantik der STEP-Attribute (z.B. WOPPr.d0DesignTargetDate, s7PhyAss.ProjectName)

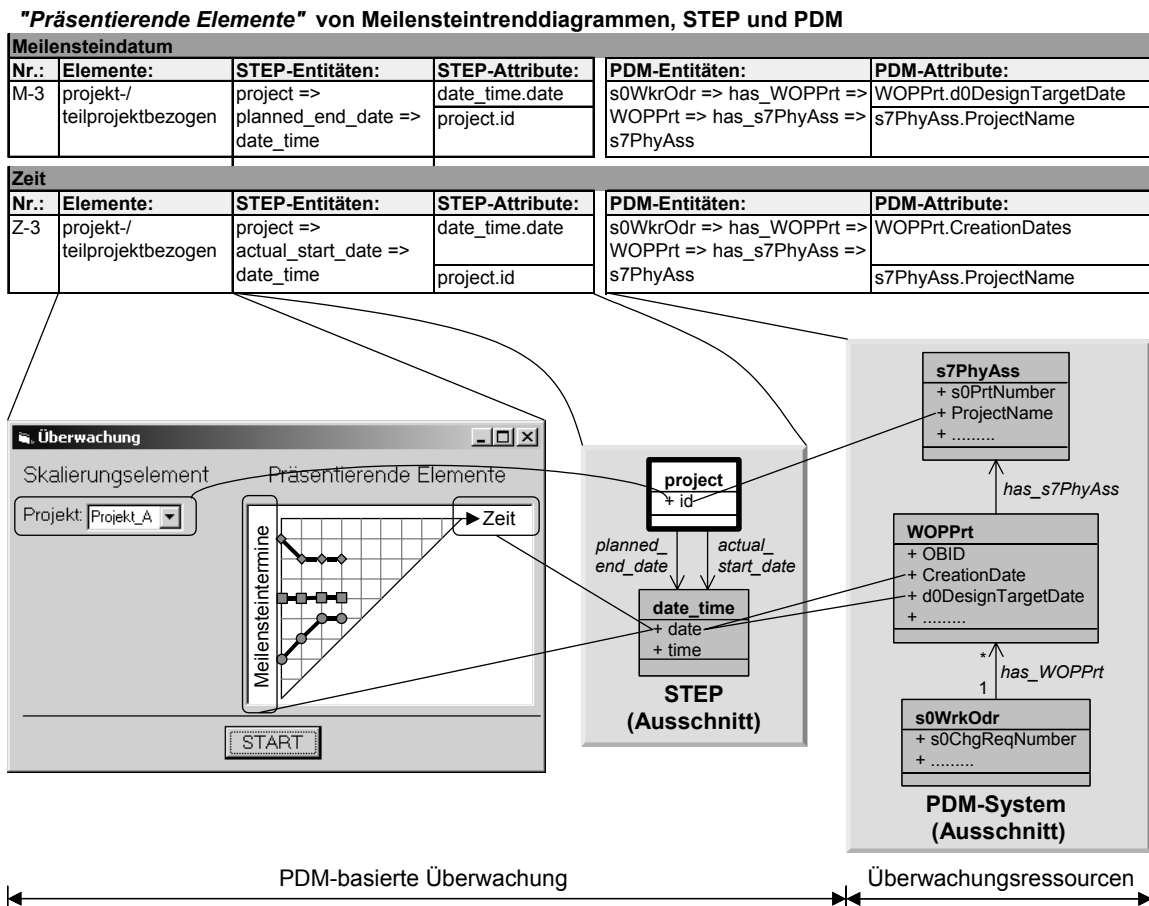


Abb. 4-20: Ausschnitt einer Musterinstanz für ein Meilensteintrenddiagramm

#### 4.5 Systematischer Konfigurations- und Änderungsmanagementansatz

Unter Berücksichtigung bestehender Ansätze entwirft die Arbeit ein systematisches Konfigurations- und Änderungsmanagement für das Metamodell zur Transformationsunterstützung.

Verwaltungsprinzipien finden sich u.a. in Richtlinien wie PDM Enablers, STEP, DIN EN 82045-2 und im Submodell Konfigurationsmanagement des V-Modells 97. PDM Enablers und STEP wurden bereits in Kapitel 3.4 vorgestellt. Die DIN EN 82045-2 beinhaltet ein auf der objektorientierten Notation „Express“ basierendes Datenmodell zur rechnerinterpretierbaren Beschreibung und den Austausch von Dokumentenmetadaten [DIN EN 82045-2]. Das Submodell Konfigurationsmanagement des V-Modells 97 beinhaltet Leitlinien, die sicherstellen sollen, dass Objekte eindeutig identifizierbar sind, Zusammenhänge und Unterschiede von verschiedenen Versionen einer Konfiguration erkennbar bleiben und Änderungen kontrolliert durchgeführt werden [Vers00]. Deren Vergleich ist wie folgt aufgebaut:

- a) Konfigurationsmanagementmöglichkeiten
- b) Änderungsmanagementmöglichkeiten
- c) Resümee

#### **a) Konfigurationsmanagementmöglichkeiten**

Sowohl PDM-Enablers als auch STEP, die DIN EN 82045-2 und das V-Model 97 bieten Konfigurationsmanagementmöglichkeiten.

PDM-Enablers erlaubt, los- (Klasse: LotEffectivity), datums- (Klasse: DatedEffectivity) und serialnummernbezogene (Klasse: SerialNumberEffectivity) Gültigkeitsinformationen mit Teilstämmen (Klasse: PartMaster) und Dokumenten (Klasse: DocumentMaster) zu verknüpfen. STEP besitzt ebenfalls die Möglichkeit, zeitraum- (Entität: duration) und datumsbezogene (Entität: date\_time) Gültigkeitsinformationen u.a. den Teilstämmen (Entität: item) und Dokumenten (Entität: document) zuzuordnen. Die DIN EN 82045-2 ist in der Lage, Dokumente (Entität: document) mit datums- (Entität: dated\_effectivity) bzw. zeitintervallbezogenen (Entität: time\_interval\_based\_effectivity) Gültigkeitsinformationen zu versehen. Das Submodell Konfigurationsmanagement regelt Konfigurationen über sogenannte „Konfigurations-Identifikationsdokumente“.

### **b) Änderungsmanagementmöglichkeiten**

Nicht nur STEP sondern auch PDM-Enablers und das Submodell Konfigurationsmanagement des V-Models 97 bieten die Möglichkeit, änderungsrelevante Informationen abzubilden.

PDM-Enablers und STEP unterscheiden dabei zwischen einem Änderungsantrag (PDM-Enablers-Klasse: EngChgRequest, STEP-Entität: change\_request) und einem Änderungsauftrag (PDM-Enablers-Klasse: EngChgOrder, STEP-Entität: work\_order). Das Submodell Konfigurationsmanagement des V-Models 97 gibt ein so genanntes „Konfigurations-Identifikationsdokument“ vor, in das ein Anwender Informationen bzgl. Änderungsnummer, geänderter Bereich, Grund der Änderung, Änderungsdatum, Änderungsversion und Änderungsverantwortlicher ablegen kann.

Mit PDM-Enablers, der DIN EN 82045-2 und STEP können Reifegradinformationen abgebildet werden. PDM-Enablers beinhaltet Reifegrade von Dokumenten und Teilstämmen (PDM-Enablers-Attribut: Statable), STEP darüber hinaus Reifegradinformationen bzgl. Änderungsanträgen (work\_request.status) und die mit ihnen verbundenen auszuführenden Aktivitäten (Attribut: activity.status). Die DIN EN 82045-2 beschränkt sich auf dokumentenbezogene Reifegradinformationen (Attribut: RASAD\_status.status\_value).

### **c) Resümee**

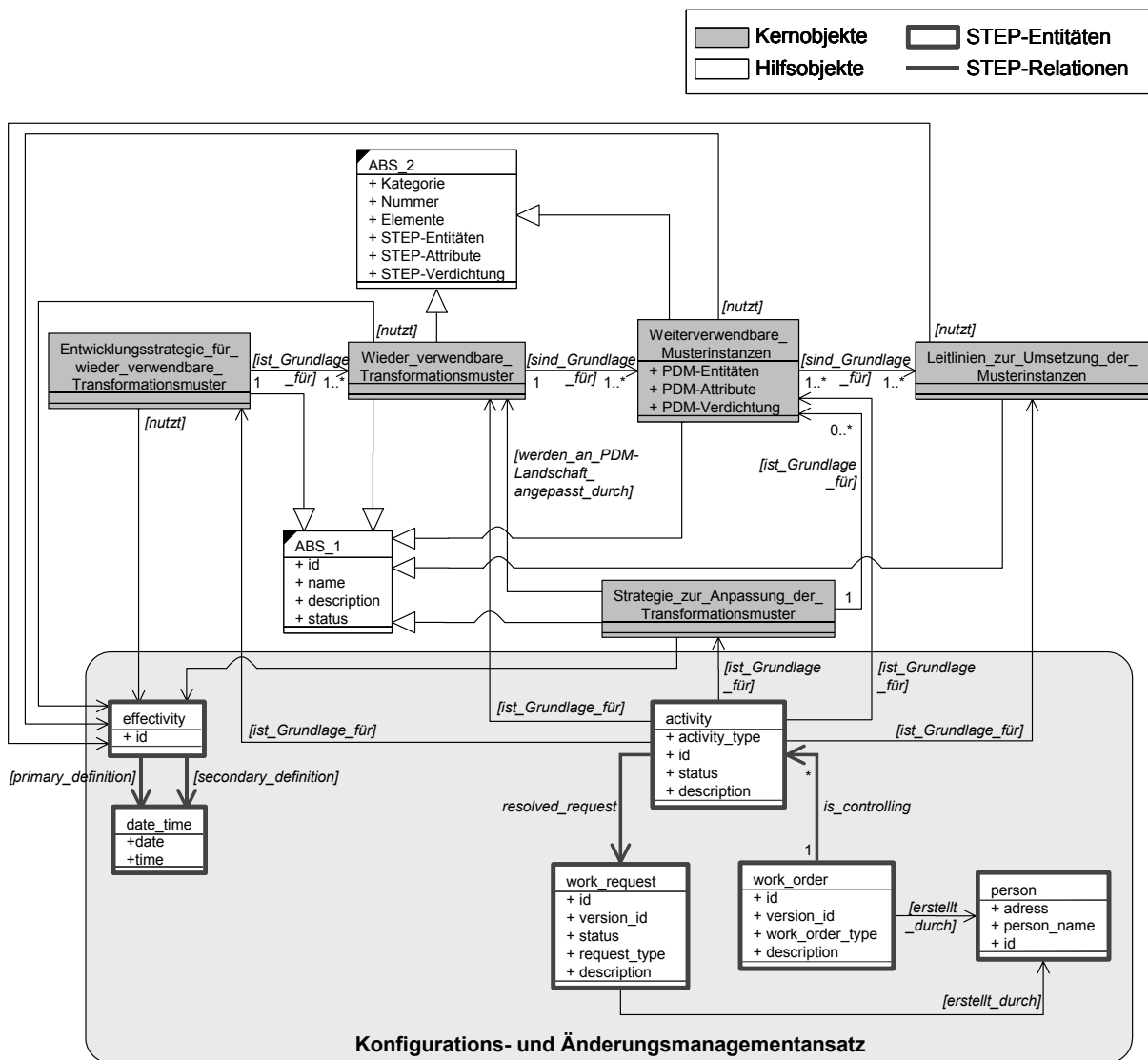
Trotz unterschiedlicher Philosophien, bieten alle vorgestellten Ansätze ähnlich umfangreiche Konfigurationsmanagementmöglichkeiten. Bei den Änderungsmanagementmöglichkeiten unterscheidet sich STEP von PDM-Enablers dadurch, dass es Reifegradinformationen umfassender abdeckt. Aus diesem Grund stellt die Arbeit im folgenden einen STEP-basierten Konfigurations- und Änderungsmanagementansatz für das Metamodell zur Transformationsunterstützung vor. Unter Berücksichtigung des Metamodells zur Transformationsunterstützung

(siehe Kapitel 4.1), der tabellarischen Transformationsvorlagen in Anhang A bis H und der zuvor erwähnten STEP-Entitäten, lässt sich ein UML-basiertes Klassendiagramm ableiten, das Abbildung 4-21 zeigt. Die grau hinterlegten Objekte (Kernobjekte), wie z.B. „Wieder\_verwendbare\_Transformationsmuster“ als auch deren Verknüpfungen untereinander, entsprechen denen aus dem Metamodell zur Transformationsunterstützung. Sie wurden allerdings mit zusätzlichen Hilfsobjekten und STEP-Entitäten verbunden.

Die Hilfsobjekte, wie z.B. „ABS\_1“, fassen Attribute zusammen, die an die Kernobjekte vererbt werden. Sie selbst sind abstrakt, d.h. sie werden nicht instanziiert. So vererbt die abstrakte Klasse „ABS\_1“ Identifikations- (z.B. „id“) und Reifegradattribute („status“) an alle Kernobjekte des Metamodells zur Transformationsunterstützung. Die abstrakte Klasse „ABS\_2“ vererbt obendrein Attribute, um die Transformationsmusterinformationen im Anhang A bis H abbilden zu können. Die Kategorie der STEP-Entitäten fasst die Objekte für das Konfigurations- und Änderungsmanagement der Kernobjekte zusammen, d.h. mit ihnen ist es möglich, Rückschlüsse über die Aktualität der Kernobjekte zu ziehen. Das datumsbezogene Konfigurationsmanagement der Kernobjekte erfolgt über die beiden STEP-Entitäten „effectivity“ und „date\_time“, welche über zwei gerichtete Assoziationen miteinander verbunden sind. Die gerichtete Assoziation „primary\_definition“ beschreibt den Beginn eines datumsbezogenen Gültigkeitsbereiches, die gerichtete Assoziation „secondary\_definition“ dagegen das Ende. Das Änderungsmanagement der Kernobjekte erfolgt mittels der vier STEP-Entitäten „activity“, „work\_request“, „work\_order“ und „person“. Das bedeutet, es unterstützt durch eine systematische Dokumentation die Änderungshintergründe zu verstehen. Über einen Arbeitsantrag (work\_request) sind Änderungswünsche zu dokumentieren (z.B. das Entwickeln neuer Transformationsmuster), deren Autor mit Hilfe der Klasse „person“ identifiziert wird. Wird der Arbeitsantrag akzeptiert d.h. `work_request.status = accepted`, so ist ein Arbeitsauftrag (work\_order) und entsprechend auszuführende Aktivitäten (activity) zu dokumentieren. Die



Kardinalitäten der gerichteten Assoziation „is\_controlling“ drücken aus, dass einem Arbeitsantrag mehrere auszuführende Aktivitäten zugeordnet werden können. Die gerichtete Assoziation „resolved\_request“ drückt aus, dass eine Aktivität genau mit einem Änderungswunsch verknüpfbar ist.

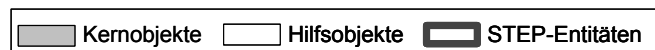


**Abb. 4-21: Das Modell des Verwaltungsprinzips  
(Notation: UML-Klassendiagramm)**

Beispielhaft zeigt Tabelle 4-9 einen Ausschnitt eines Data Dictionary's für den Konfigurations- und Änderungsmanagementansatz. Die vollständige Liste befindet sich im Anhang I. Analog zum UML-Klassendiagramm repräsentieren die grau

hinterlegten Felder die Kernobjekte des Metamodells zur Transformationsunterstützung. Die dick umrahmten Felder geben die aus STEP übernommenen Entitäten und deren Attribute wieder. Die weiß hinterlegten Felder beinhalten die Hilfsobjekte. Die Spalten des Data Dictionary's enthalten folgende Informationen:

- a) Nr.: Eindeutige Identifikationsnummer der Entität bzw. der dazugehörigen Attribute
- b) Entität: Namen der Entität (z.B. effectivity)
- c) Attribut: Namen des zur Entität gehörenden Attributes (z.B. id)
- d) Beschreibung: Erklärung der Entität bzw. des Attributs (z.B. Identifikationsnummer der Gültigkeit)
- e) Beispiel: Beispielhafte Belegung des Attributes (z.B. E-52098)



**"Data Dictionary" des Konfigurations- und Änderungsmanagementansatzes**

Nr.:	Entität:	Attribut:	Beschreibung:	Beispiel:
5	effectivity		Gültigkeit	
5-1		id	Identifikationsnummer der Gültigkeit	E-52098
10	Weiterverwendbare_Musterinstanzen		An eine konkrete PDM-Systemlandschaft angepasste Transformationsmuster	
10-1		PDM-Entitäten	Textuelle Beschreibung des PDM-Datenmodellausschnittes, der für den Semantikabgleich mit STEP benötigt wird	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss
10-2		PDM-Attribute	Ihre Kombination entspricht der Semantik der STEP-Attribute	WOPPr.d0DesignTargetDate, s7PhyAss.ProjectName
10-3		PDM-Verdichtung	Textuelle Beschreibung der Attributeverdichtung	Das höchste Datenfeld "d0DesignTargetDate" der Klasse "WOPPr" unter Berücksichtigung des Datenfeldes "ProjectName" der Klasse "s7PhyAss"

**Tab. 4-9: Data Dictionary Ausschnitt**

## 4.6 Intuitives Zugriffsprinzip

Unter Berücksichtigung bestehender Ansätze und typischer Zugriffsfälle stellt dieses Kapitel ein Zugriffsprinzip für das Metamodell zur Transformationsunterstützung vor.

Weil das Kernobjekt „Wieder\_verwendbare\_Transformationsmuster“ vor der Entwicklung eines PDM-basierten Überwachungswerkzeuges zu einer „Weiterverwendbaren\_Musterinstanz“ überführt werden muss und wegen der allgemeingültigen Auslegung der anderen Kernobjekte, unterscheidet die Arbeit folgende zwei Zugriffsfälle (siehe Abb. 4-22):

- a) Kernobjekt suchen
- b) Kernobjekt ändern und konfigurieren

#### **a) Anwendungsfall 1: Kernobjekte suchen**

Bestehende Verfahren zur Objektsuche werden in nichthierarchische Ansätze, wie z.B. Case Based Reasoning oder Metadatensuche, teilhierarchische Ansätze, wie z.B. Thesauri sowie hierarchische Ansätze, wie z.B. die Klassifikationsbaummethode, unterteilt.

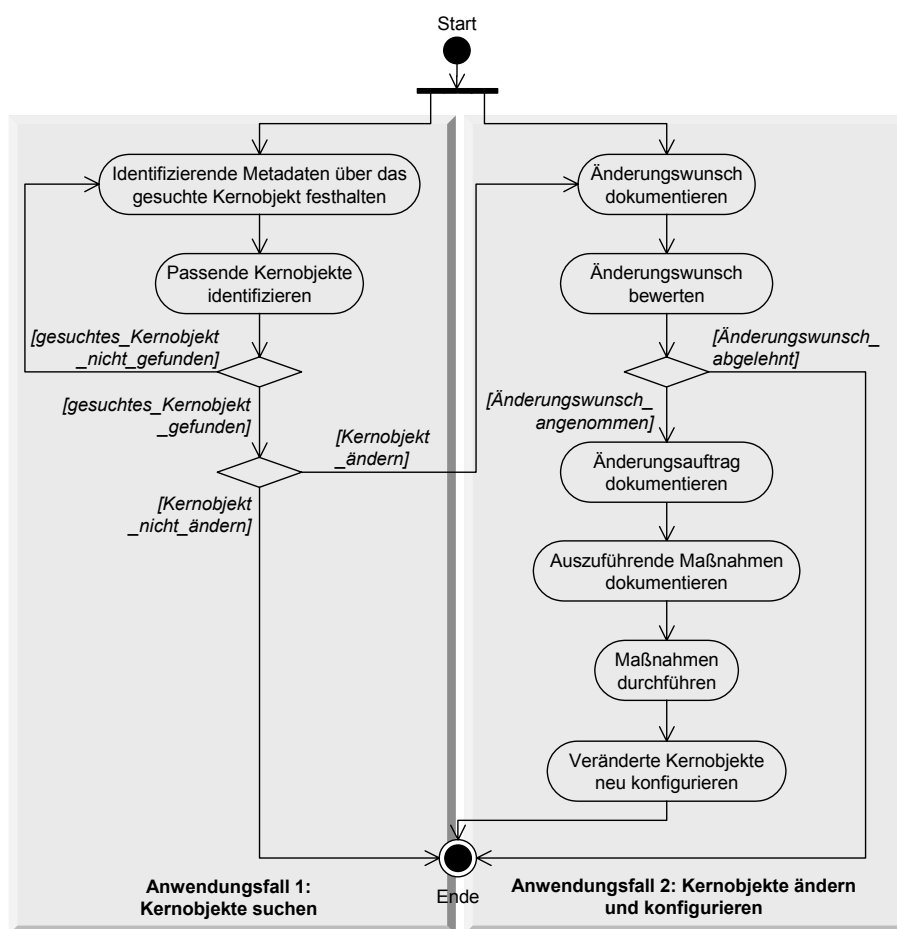
Da die Kernobjekte bisher nichthierarchisch aufgebaut sind, erfolgt deren Suche über identifizierende Metadaten. Diese Metadaten fasst die abstrakte Klasse „ABS\_1“ zusammen, genauer gesagt „id“, „name“, „description“ und „status“. Auf Grundlage einer Festlegung dieser identifizierenden Metadaten gilt es, die passenden Kernobjekte zu identifizieren.

#### **b) Anwendungsfall 2: Kernobjekte ändern und konfigurieren**

Das Ändern und Konfigurieren der Kernobjekte des Metamodells zur Transformationsunterstützung wird durch ein Verwaltungsprinzip vorgegeben, das in Kapitel 4.5 entwickelt wurde.

Das bedeutet, dass ausgehend von einem akzeptierten Änderungswunsch (Entitäten: work\_request und person) dazugehörige Änderungsaufträge (Entitäten: work\_order und person) und auszuführende Maßnahmen (Entität: activity) zu beschreiben sind.

Handelt es sich bei dem Änderungswunsch um eine Anpassung oder Neuerstellung von Transformationsmustern, so ist weiter nach Kapitel 4.2 vorzugehen. Handelt es sich bei dem Änderungswunsch um eine Anpassung oder Neuerstellung von Transformationsmustern, so ist nach Kapitel 4.4 zu verfahren. Nach Beendigung der Änderungsmaßnahmen sind die betroffenen Kernobjekte neu zu konfigurieren, d.h. zeitraumbezogene Gültigkeitsinformationen sind mit Hilfe der beiden STEP-Entitäten „effectivity“ und „date\_time“ zu vergeben.



**Abb. 4-22: Das Modell des Zugriffsprinzips**

## 4.7 Leitlinien zur Umsetzung der Musterinstanzen

Die Arbeit zeigt im folgenden die wichtigsten Grundsätze, die es zu beachten gilt, wenn Musterinstanzen als Grundlage zur Entwicklung eines PDM-basierten Überwachungswerkzeuges genutzt werden sollen. Sie sind wie folgt zusammengefasst:

- a) Qualitätsprüfung der PDM-Informationen
- b) Festlegen der Datenharmonisierung und der Datenladung
- c) Anforderungen an das Überwachungswerkzeug systematisch vervollständigen

### a) Qualitätsprüfung der PDM-Informationen

Die Verlässlichkeit eines PDM-basierten Überwachungswerkzeuges hängt stark von der Qualität der zugrundeliegenden PDM-Informationen ab. Das bedeutet, es ist erst dann sinnvoll, Musterinstanzen umzusetzen, wenn sichergestellt ist, dass die zur Überwachung benötigten PDM-Informationen vollständig und richtig gepflegt werden. Dies gilt es, durch entsprechende Datenbankabfragen systematisch zu überprüfen. Im Besonderen betrifft es die optionalen Datenfelder, die keine „Default-Werte“ besitzen.

### b) Festlegen der Datenharmonisierung und der Datenladung

Typischerweise unterscheiden sich die Datenbankstrukturen von produktiven PDM-Systemen. Deshalb sind die Themen Datenharmonisierung und Datenladung besonders wichtig, wenn ein Überwachungswerkzeug mit mehreren PDM-Systemen gekoppelt werden sollen. Grundlage für harmonische PDM-Informationen in der STEP-basierten Controllingdatenbasis sind einheitliche Formate, Kodierungen und Detaillierungsgrade. Sie gilt es festzulegen.

Einheitliche Ladezeitpunkte gehören zur erstmaligen Datenladung und zur fortlaufenden Aktualisierung der Controllingdatenbasis. Die Wahl des Ladeprozesses, der vollständig oder inkrementell durchgeführt werden kann, sollte von der Größe des zu transferierenden Datenvolumens und der Frequenz der Datenaktualisierung abhängig gemacht werden.

Zum Ladeprozess gehört auch, einen Kopplungstyp zwischen dem Überwachungswerkzeug und den PDM-Systemen auszuwählen. Das im Rahmen des STEP-Standards spezifizierte Austauschformat „P21“ bietet im Vergleich zu anderen Schnittstellenvarianten den Vorteil, dass es von den gängigen PDM-Systemen unterstützt wird. Da jedoch nicht auszuschließen ist, dass sich der Umfang der ausgeleiteten PDM-Informationen von den Vorgaben der Musterinstanzen abweicht, muss dies überprüft werden.

### **c) Anforderungen an das Überwachungswerkzeug systematisch vervollständigen**

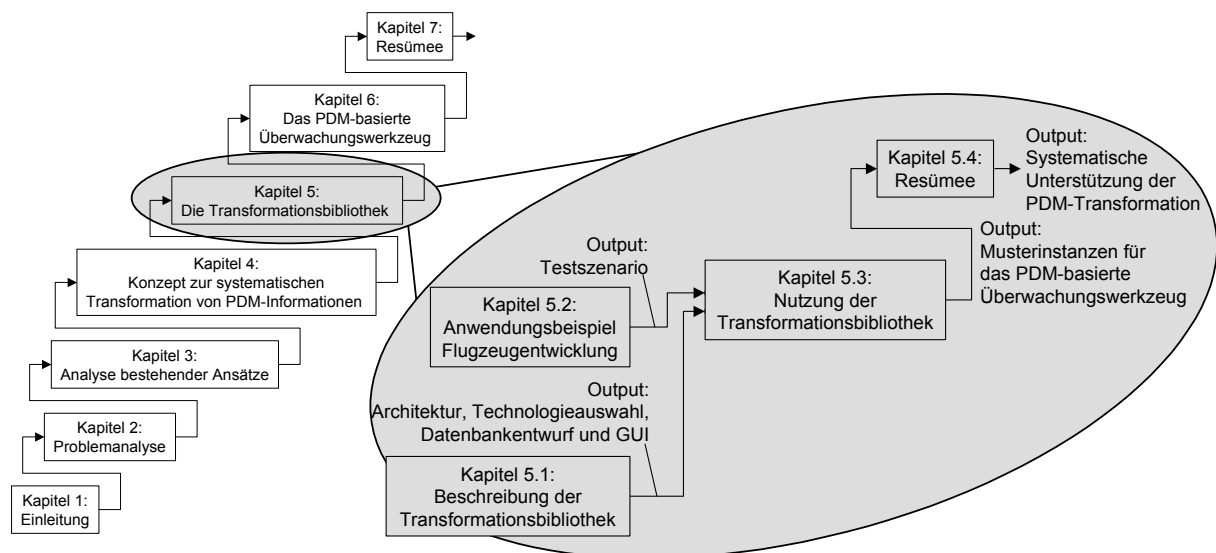
Neben den bisherigen Vorgaben sind weitere Lasten- und Pflichtenheftanforderungen notwendig, um eine systematische Realisierung und einen planmäßigen Test des PDM-basierten Überwachungswerkzeuges zu gewährleisten. Zu den Pflichtenheftanforderungen zählen u.a. Werkzeugzugriff, Benutzerführung und Detaillierungsgrad einer Überwachungskennlinie. Zu den Lastenheftanforderungen gehören z.B. eine Festlegung der Programmiersprache und der Datenbank. Beide Anforderungsklassen gilt es systematisch zu ermitteln, konsequent zu analysieren und formal zu beschreiben. Vorgehensmodelle zur Softwareentwicklung, wie z.B. Wasserfallmodell, Spiralmodell oder das V-Modell 97, geben dafür zahlreiche Hilfestellungen.

## **4.8 Resümee**

Ein PDM-basiertes Überwachungssystem bietet dem Engineeringcontrolling verlässliche Informationen, die Entwicklung ist aber schwierig. Die zentrale Herausforderung ist die Transformation zwischen dem PDM-Paradigma und dem Paradigma der Engineeringüberwachung. Hier greift die Arbeit an, in dem sie in Kapitel 4 die Kern- und Hilfsobjekte eines „*Konzeptes zur systematischen Transformation von PDM-Informationen*“ entwickelte, dessen objektorientiertes Metamodell Kapitel 4.1 vorstellte.

## 5. Die Transformationsbibliothek

Dieses Kapitel stellt eine Transformationsbibliothek vor, die auf dem in Kapitel 4 entwickelten Transformationskonzept basiert. Kapitel 5.1 zeigt die Architektur und technische Umsetzung des Bibliotheksprototyps. Anhand eines Anwendungsbeispiels aus der Luftfahrtindustrie (Kapitel 5.2) erstellt Kapitel 5.3 die Musterinstanzen für eine Meilensteintrendanalyse, Sachfortschritts- und Balkendiagramm. Die drei Musterinstanzen sind Vorgaben für das PDM-basierte Überwachungswerkzeug, das Kapitel 6 beschreibt. Den Aufbau dieses Kapitels stellt Abbildung 5.1 dar.



**Abb. 5-1: Aufbau und Vorgehensweise Kapitel 5**

## 5.1 Beschreibung der Transformationsbibliothek

### 5.1.1 Systemarchitektur und Technologieauswahl

Grundlage des hier vorgestellten Bibliotheksprototyps ist eine 2-Schichten Architektur, die innerhalb des Betriebssystems Windows 2000 realisiert wurde. Die Benutzerschnittstelle des Bibliotheksprototyps ist in der Programmiersprache Visual Basic 6.0 implementiert. Um den Nutzern dynamische Abfragen an die Datenbasis zu erlauben, wurden zusätzlich ActiveX Komponenten eingebettet (z.B. für Combo-, Textboxen, TabStrip). Außerdem werden ActiveX Data Objects (ADO's) für die Kommunikation zwischen der Benutzerschnittstelle und dem Datenbanktreiber (OLEDB 4.0) der MS-Access Datenbank verwendet (siehe Abb. 5-2).

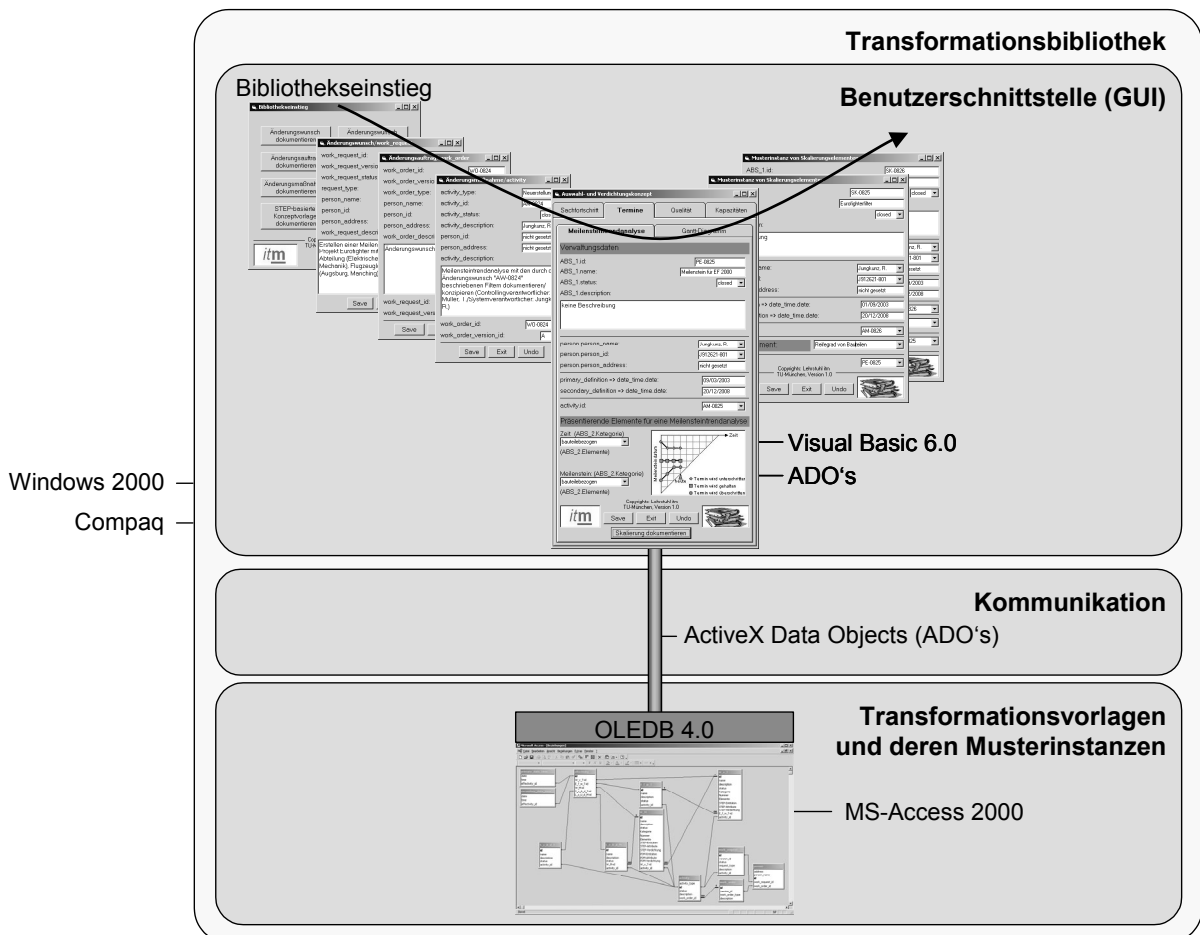
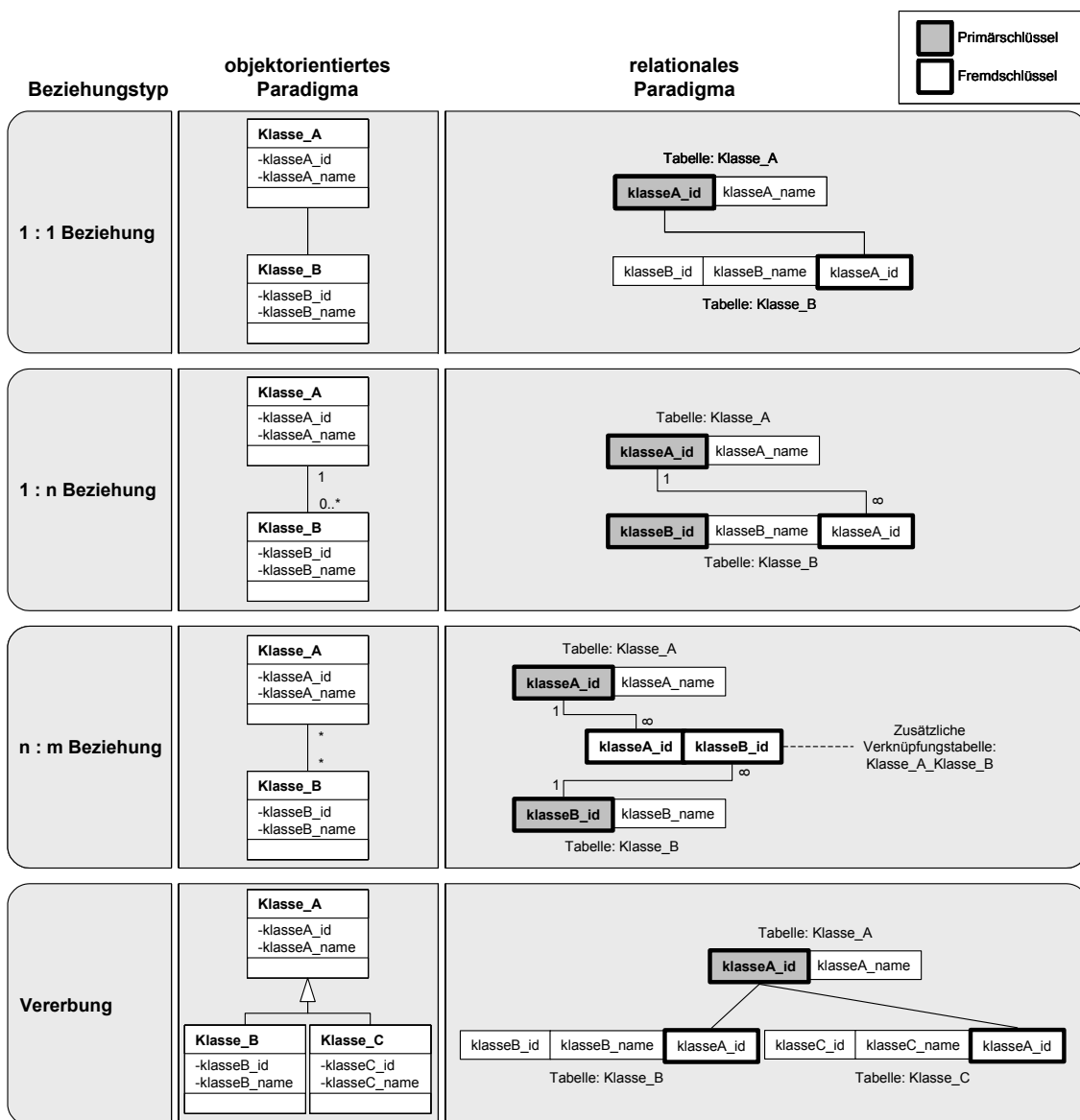


Abb. 5-2: Architektur und Technologieauswahl des Bibliotheksprototyps



### 5.1.2 Datenbankentwurf

Das UML-Klassendiagramm aus Kapitel 4.5 dient als Grundlage für die Datenbankimplementierung in MS-Access. Um dieses objektorientierte Modell in einer relationalen Datenbank umsetzen zu können, muss es jedoch entsprechend umgewandelt werden. Um diesen Vorgang zu verdeutlichen, fasst Abbildung 5-3 die vier wichtigsten Beziehungstypen zusammen. Das Ergebnis der Datenmodellüberführung zeigt Abbildung 5-4.



**Abb. 5-3: Transformation zwischen objektorientierten und relationalen Paradigma**

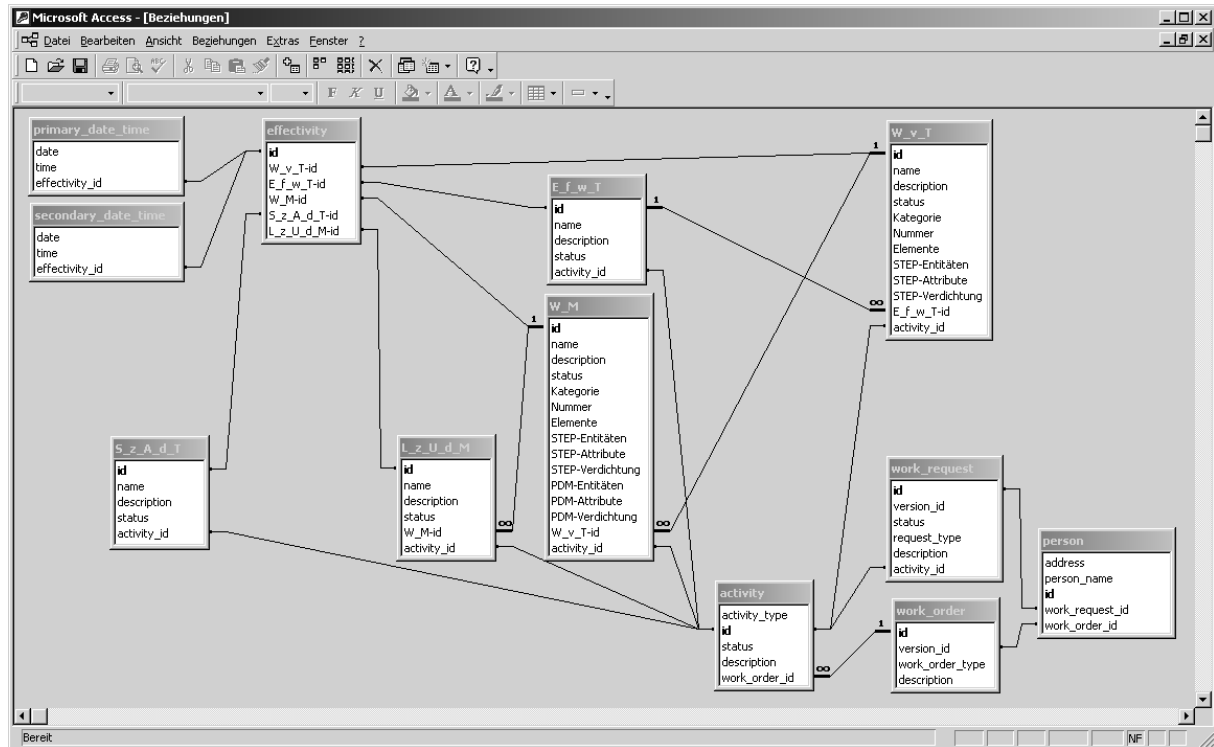


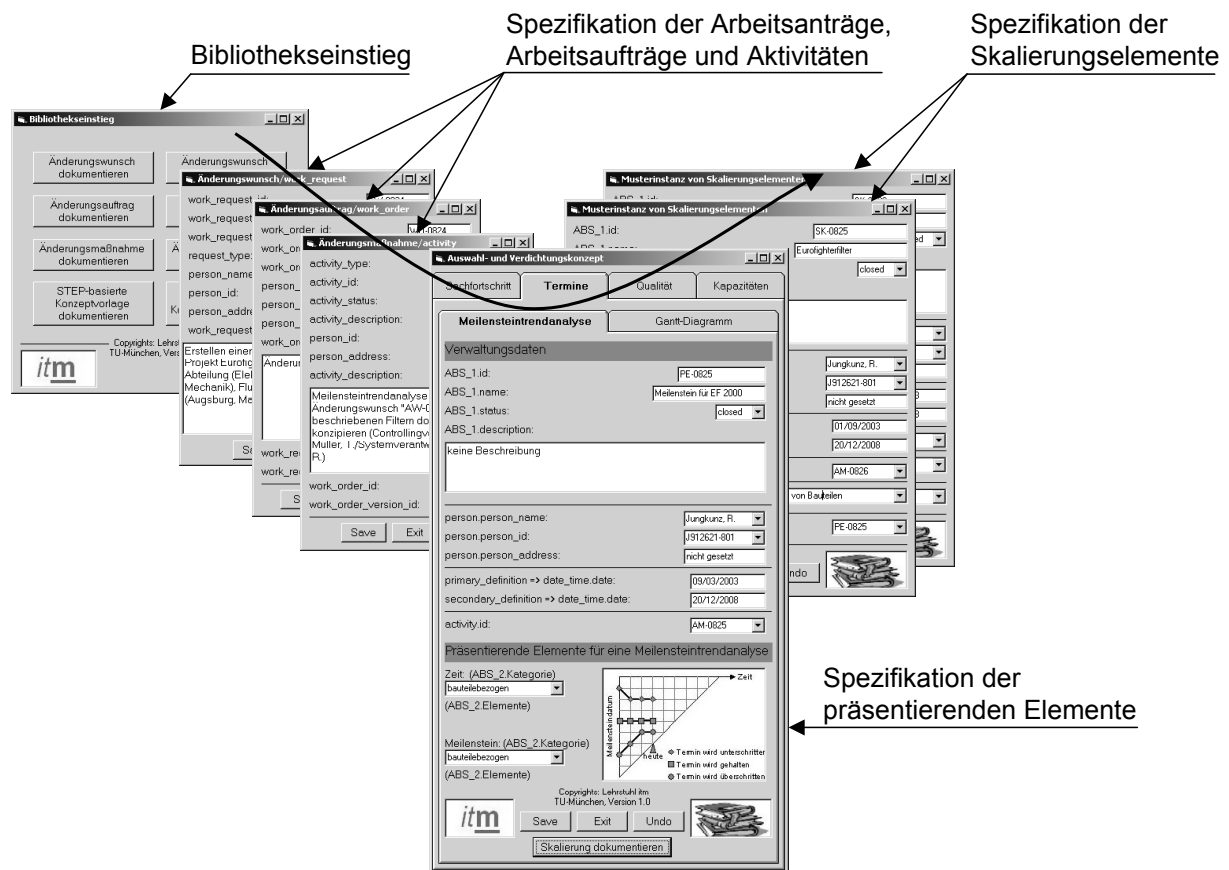
Abb. 5-4: Relationaler Datenbankentwurf in MS-Access

### 5.1.3 Benutzerschnittstelle (GUI)

Das in Kapitel 4.6 entwickelte Zugriffsprinzip dient als Grundlage für die Funktionalitäten und die entstandene Benutzeroberfläche der Transformationsbibliothek. Sie werden anschließend anhand von Abbildung 5-5 beschrieben.

Über einen Bibliothekseinstieg gelangt der Nutzer in einen Bereich, der zur Spezifikation von Änderungswünschen dient. Über die sich anschließenden Bildschirmfenster sind der dazugehörige Änderungsauftrag und die durchzuführenden Aktivitäten festzulegen. Unter Zuhilfenahme einer Menüleiste (Abb. 5-5, Mitte) navigiert der Nutzer zur gewünschten Transformationsvorlage (z.B. Meilensteintrenddiagramm). Dort selektiert er zunächst mit Hilfe von Auswahlboxen die gewünschten Elemente (z.B. abteilungsbezogenes Start-/ bzw. Enddatum, Arbeitspakete bezogener Füllstand). Schließlich sind die für eine Kennlinie relevanten

Filter anhand von Auswahlboxen zu spezifizieren (z.B. projektbezogener und standortbezogener Filter).



**Abb. 5-5: Die Benutzerschnittstelle der Transformationsbibliothek**

## 5.2 Anwendungsbeispiel Flugzeugentwicklung

Die EADS, zweitgrößtes Luft- und Raumfahrtunternehmen der Welt, entwickelt und betreut im Geschäftsbereich Militärflugzeuge Hochleistungsflugzeuge, wie Tornado und Eurofighter im typischerweise multinationalen Kontext. Anhand von Praxisbeispielen aus einem Forschungsprojekt zwischen der EADS-M und dem Lehrstuhl für Informationstechnik (*itm*) wird der Bibliotheksprototyp genutzt und auf seine Leistungsmerkmale hin untersucht. Diese sind durch die konkretisierten Anforderungen (Lastenheftanforderungen) vorgegeben. Das bei der EADS-M

eingesetzte und angepasste PDM-System TeamCenter von EDS (ehemals Metaphase von SDRC) dient als Referenz zur Erstellung von Musterinstanzen.

### 5.3 Nutzung der Transformationsbibliothek

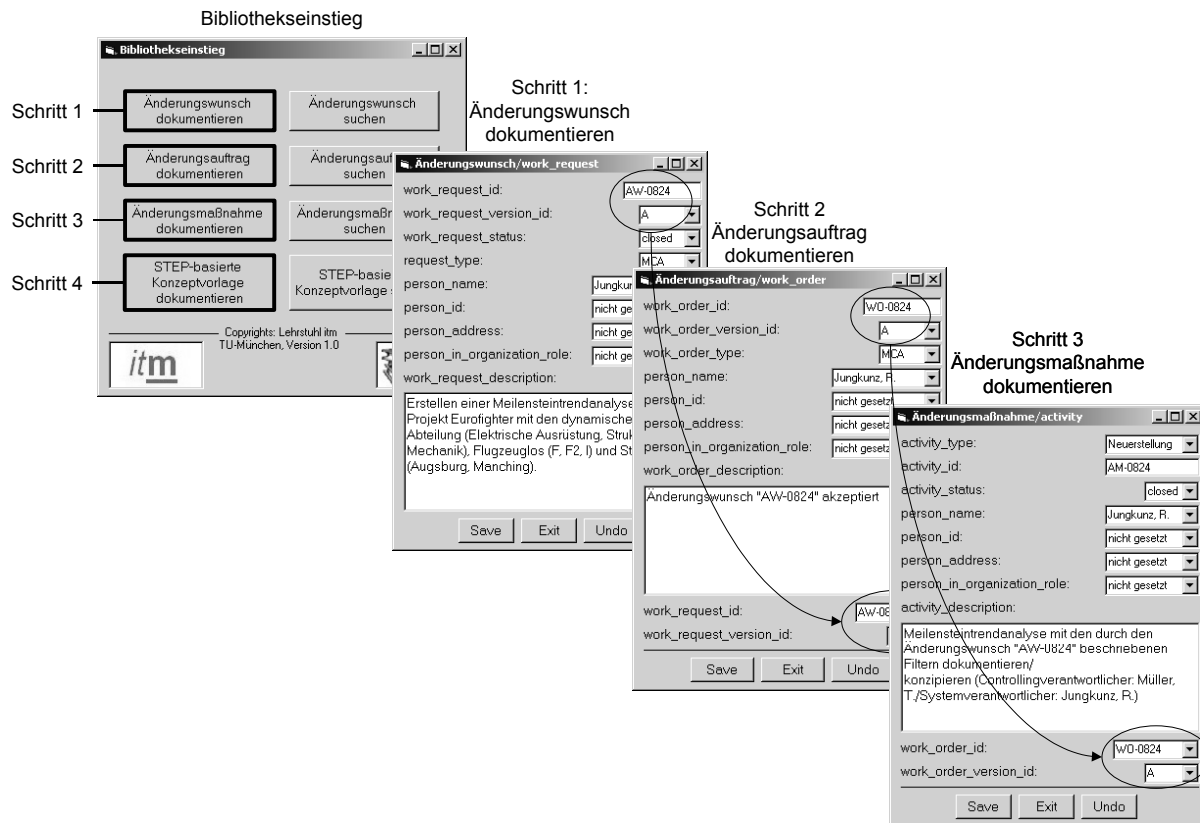
#### 5.3.1 Beispiel: Meilensteintrendanalyse

Um die konzeptionelle Grundlage zur Erstellung einer PDM-basierten Überwachungssoftware zu beschreiben, gibt die Arbeit eine Methode nach Kapitel 4.6 vor. Der erste Schritt ist somit das Dokumentieren eines Änderungswunsches (Entitäten: work\_request und person) mit Hilfe der Transformationsbibliothek (siehe Abb. 5-6). Beispielfhaft wurden hier innerhalb des Bildschirmfensters „Änderungswunsch“ folgende Anforderungen für eine PDM-basierte Meilensteintrendanalyse dokumentiert (STEP-Attribut: work\_request.description):

- a) Skalierungselement: Projekt (Eurofighter)
- b) Skalierungselement: Flugzeuglos (F, F2, I)
- c) Skalierungselement: Standort (Augsburg, Manching)
- d) Präsentierendes Element: Abteilungsbezogene Meilensteine (Elektrische Ausrüstung, Struktur, Mechanik)

Diese Änderungswunschangaben werden mit identifizierenden Metadaten versehen, die durch das Konfigurations- und Änderungsmanagementmodell in Kapitel 4.5 vorgegeben sind, genauer gesagt work\_request.id, work\_request.version\_id, work\_request.status, person.address, person.person\_name und person.id. Ausgehend von einem akzeptierten Änderungswunsch, ist im zweiten Schritt ein entsprechender „Änderungsauftrag“ zu dokumentieren (STEP-Attribut: work\_order.description) und analog zum Änderungswunsch mit den identifizierenden Metadaten work\_order.id, work\_order.version\_id, work\_order.work\_order\_type, person.address, person.person\_name und person.id zu versehen.

Im letzten Schritt wurden die zu dem Änderungsauftrag gehörenden Änderungsmaßnahmen spezifiziert (STEP-Attribut: activity.description) und um die identifizierenden Metadaten activity.activity\_type, activity.id und activity.status ergänzt.

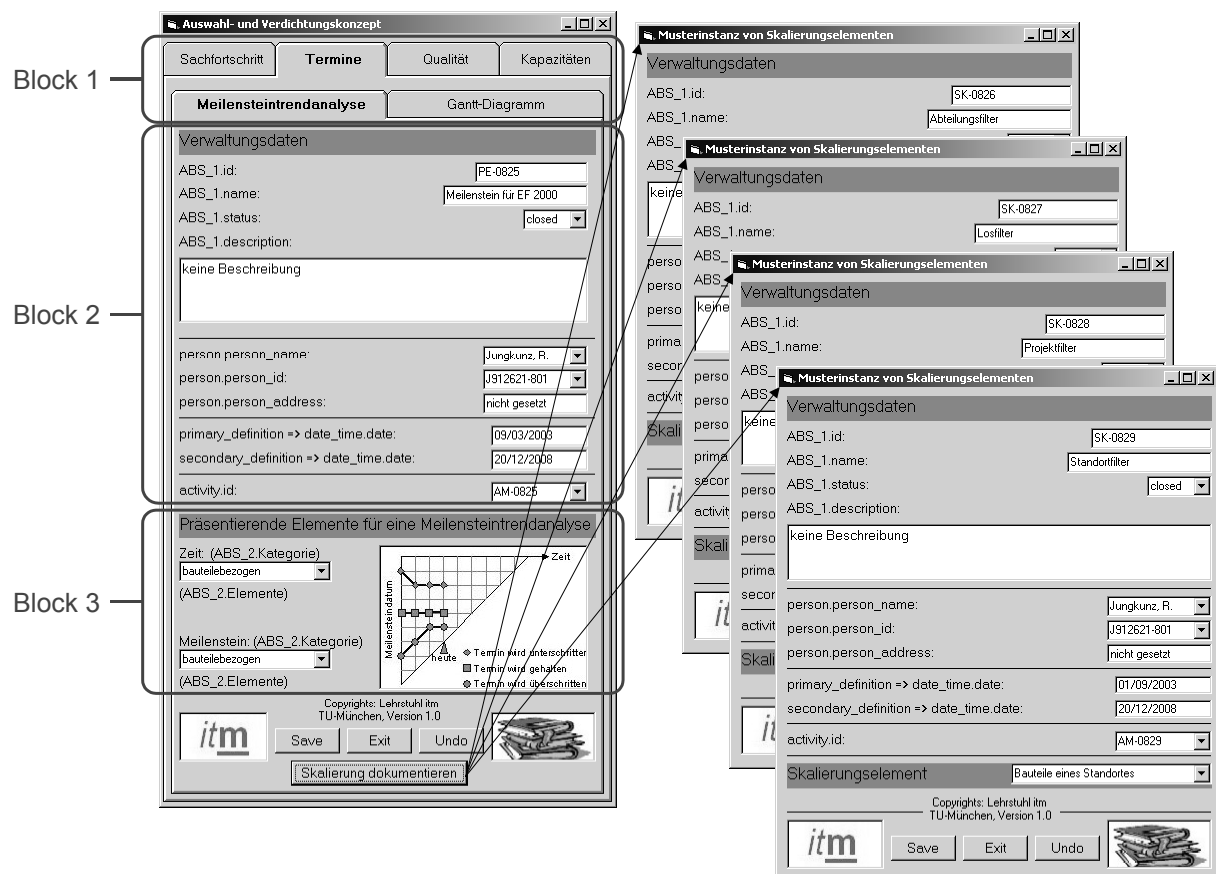


**Abb. 5-6: Dokumentation des Änderungshintergrundes**

Im Anschluss an die Dokumentation des Änderungswunsches, des Änderungsauftrages und der auszuführenden Aktivitäten, sind auf Grundlage der vorgegebenen Skalierungselemente und präsentierenden Elemente der Meilensteintrendanalyse die Musterinstanzen zu erstellen. Abbildung 5-7 stellt die wichtigsten Funktionalitäten der Transformation dar, die in drei Blöcke unterteilt sind:

- Der erste Block bietet dem Nutzer eine Navigation, die ihn ausgehend von der Analyseintension zu einer konkreten Überwachungskennlinie führt, wie beispielsweise eine Meilensteintrendanalyse.

- b) Der zweite Block beinhaltet verwaltungsrelevante Informationen der Musterinstanz. Sie sind in den Attributen der STEP-Entitäten „ABS\_1“, „person“, „date\_time“ und „activity“ hinterlegt.
- c) Im dritten Block sind die gewünschten präsentierenden bzw. skalierenden Kennlinienelemente zu wählen, wie z.B. ein bauteilebezogenes Meilensteindatum und bauteilebezogener Berichtszeitpunkt. Anhand dieser Informationen ist die Transformationsbibliothek in der Lage, in den Transformationsvorlagen (Anhang A bis H) die dazugehörigen STEP-Datenmodellausschnitte zu erkennen und diese auszugeben. Die Komplexität der Transformationsvorlagen bleibt somit dem Bibliotheksanwender weitgehend verborgen.



**Abb. 5-7: Spezifikation für eine Meilensteintrendanalyse**

Die Tabelle 5-1 zeigt das Resultat der Bemühungen, die Musterinstanz für die TeamCenter-basierte Meilensteintrendanalyse.

Im linken Bereich (Überwachungssicht) befinden sich die präsentierenden bzw. skalierenden Kennlinienelemente für die Meilensteintrendanalyse. Im mittleren Bereich (STEP-Sicht) sind die dazugehörigen STEP-Datenmodellausschnitte sowie die transformationsrelevanten STEP-Attribute zu finden. Die STEP-Sicht ist das systemneutrale Bindeglied zwischen den Überwachungskennlinien und den transformationsrelevanten PDM-Datenmodellausschnitten und PDM-Attributen.

Überwachungssicht		STEP-Sicht		PDM-Sicht	
<b>"Präsentierende Elemente" von Meilensteintrenddiagrammen, STEP und PDM</b>					
<b>Meilensteindatum</b>					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
M-2	bauteilebezogen	item <= associated_item <= item_version => activity => planned_end_date => date_time	date_time.date item.id	WOPPrt => has_s7PhyAss => s7PhyAss	WOPPrt.d0DesignTargetDate s7PhyAss.s0PartNumber
<b>Zeit</b>					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
Z-2	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	date_time.date item.id	WOPPrt => has_s7PhyAss => s7PhyAss	WOPPrt.CreationDate s7PhyAss.s0PartNumber
<b>"Skalierungselemente" des Meilensteintrenddiagramms, STEP und PDM</b>					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
S-4	Anwendungsbereich von Bauteilen	application_context <= is_relevant_for <= design_discipline_item_definition => associated_item_version => item_version => associated_item => item	application_context. application_domain item.id	s7PhyAss	s7PhyAss.s0SDR s7PhyAss.s0PartNumber
S-15	Bauteile eines Loses	lot_configuration <= item <= associated_item <= item_version	lot_configuration.lot_id item.id	s0WkrOdr => has_WOPPrt => WOPPrt => has_s7PhyAss => s7PhyAss	s0WkrOdr.s0BaselineClass s7PhyAss.s0PartNumber
S-16	Bauteile eines Projektes	item <= associated_item <= item_version => activity <= work_programm <= project	item.id project.id	s0WkrOdr => has_WOPPrt => WOPPrt => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber s0WkrOdr.ProjectName
S-17	Bauteile eines Standortes	item <= associated_item <= item_version => activity => organization	item.id organization.id	s0WkrOdr => has_WOPPrt => WOPPrt => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber s0WkrOdr.ChgReqCreator

**Tab. 5-1: Musterinstanz einer Meilensteintrendanalyse**

### 5.3.2 Beispiel: Sachfortschrittsdiagramm

Zusätzlich wurde zum Konzeptentwurf für eine PDM-basierte Meilensteintrendanalyse gefordert, dass eine Musterinstanz für ein TeamCenter-basiertes Sachfortschrittsdiagramm zu erstellen ist. Die Randbedingungen wurden wie folgt geschildert. Die präsentierenden Elemente des Sachfortschrittsdiagramms „Berichtszeitpunkt“ und „Sachfortschrittshöhe“ sollten bauteilebezogen ermittelt werden. Dem Anwender des Überwachungswerkzeuges muss es ferner möglich sein, das Sachfortschrittsdiagramm mit Hilfe der zwei dynamischen Skalierungselementen Flugzeuglos (F, F2, I) und Standort (Augsburg, Manching) auf seine persönlichen Informationsbedürfnisse anzupassen. Diese galt es um zwei statische Skalierungselemente, genauer gesagt Projekt (Eurofighter) und Abteilungen (Elektrische Ausrüstung, Mechanik, Struktur, Elektrik), zu ergänzen. Die Arbeit verzichtet hier auf eine entsprechend detaillierte Beschreibung der Änderungshintergründe und verweist statt dessen auf das Beispiel in Kapitel 5.3.1. Die resultierende Musterinstanz eines TeamCenter-basierten Sachfortschrittsdiagramms zeigt Tabelle 5-2.



**"Präsentierende Elemente" des Sachfortschrittsdiagramms, STEP und PDM**

Sachfortschritt					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
F-6	bauteilebezogen	item <= associated_item <= item_version <= approval =>status => approval_status	item.id  approval_status.status_ name	s7PhyAss	s7PhyAss.s0PartNumber  s7PhyAss.s0MaturityStage
Zeit					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
Z-3	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	item.id   date_time.date	WOPPr => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber   WOPPr.CreationDate
"Skalierungselemente" des Sachfortschrittsdiagramms, STEP und PDM					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
S-4	Anwendungsbereich von Bauteilen	application_context <= is_relevant_for <= design_dicipline_item _definition => associated_item_ version => item_version => associated_item => item	application_context. application_domain  item.id	s7PhyAss	s7PhyAss.s0SDR  s7PhyAss.s0PartNumber
S-15	Bauteile eines Loses	lot_configuration <= item <= associated_item <= item_version	lot_configuration.lot_id  item.id	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss	s0WkrOdr.s0BaselineClass  s7PhyAss.s0PartNumber
S-16	Bauteile eines Projektes	item <= associated_item <= item_version => activity <=	item.id project.id	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber s0WkrOdr.ProjectName
S-17	Bauteile eines Standortes	item <= associated_item <= item_version => activity => organization	item.id organization.id	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber s0WkrOdr.ChgReqCreator

**Tab. 5-2: Musterinstanz eines Sachfortschrittsdiagramms****5.3.3 Beispiel: Gantt-Diagramm**

Neben dem PDM-basierten Meilensteintrenddiagramm und Sachfortschrittsdiagramm sollte das TeamCenter-basierte Überwachungswerkzeug eine Möglichkeit zur Darstellung der aktuellen Terminalsituation in Form eines Gantt-Diagramms aufweisen. Die weiteren Vorgaben, die es hier zu berücksichtigen galt, wurden wie folgt geschildert. Die drei präsentierenden Gantt-Diagrammelemente „Startdatum“, „Enddatum“ und „Füllstand“ sind ausschließlich bauteilebezogen zu ermitteln. Die drei dynamischen Skalierungselemente Abteilung (Elektrische Ausrüstung, Struktur, Mechanik), Flugzeuglos (F, F2, I) und Standort (Augsburg, Manching) sollten es den Überwachungswerkzeugnutzern erleichtern, genau den Detaillierungsgrad an Informationen zu erhalten, die ihr Tagesgeschäft verlangt. Die drei dynamischen Skalierungselemente galt es um das statische Skalierungselement Projekt (Eurofighter) zu ergänzen (siehe Tab. 5-3).

**"Präsentierende Elemente" des Gantt-Diagramms, STEP und PDM**

Startdatum					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
SD-3	bauteilebezogen	<b>Fall 1</b> die Abarbeitung der Bauteile hat begonnen: item <= associated_item <= item_version => activity => actual_start_date => date_time <b>Fall 2</b> die Abarbeitung der Bauteile hat noch nicht begonnen: item <= associated_item <= item_version => activity => planned_start_date => date_time	item.id          date_time.date	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber          WOPPr.CreationDate

Enddatum					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
ED-3	bauteilebezogen	<b>Fall 1</b> die Abarbeitung der Bauteile ist abgeschlossen: item <= associated_item <= item_version => activity => actual_end_date => date_time <b>Fall 2</b> die Abarbeitung der Bauteile ist noch nicht abgeschlossen: item <= associated_item <= item_version => activity => planned_end_date => date_time	item.id          date_time.date	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber          WOPPr.d0DesignTargetDate

Füllstand					
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
FS-7	bauteilebezogen	item <= associated_item <= item_version => activity	item.id approval_status.status_name	s7PhyAss	s7PhyAss.s0PartNumber s7PhyAss.s0MaturityStage

**"Skalierungselemente" des Gantt-Diagramms, STEP und PDM**

Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	PDM-Entitäten:	PDM-Attribute:
S-4	Anwendungsbereich von Bauteilen	application_context <= is_relevant_for <= design_discipline_item_definition => associated_item_version => item_version => associated_item => item	application_context. application_domain    item.id	s7PhyAss	s7PhyAss.s0SDR    s7PhyAss.s0PartNumber
S-15	Bauteile eines Loses	lot_configuration <= item <= associated_item <= item_version	lot_configuration.lot_id  item.id	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss	s0WkrOdr.s0BaselineClass s7PhyAss.s0PartNumber
S-16	Bauteile eines Projektes	item <= associated_item <= item_version => activity <=	item.id project.id	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber s0WkrOdr.ProjectName
S-17	Bauteile eines Standortes	item <= associated_item <= item_version => activity => organization	item.id organization.id	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss	s7PhyAss.s0PartNumber s0WkrOdr.ChgReqCreator

**Tab. 5-3: Musterinstanz eines Gantt-Diagramms**

### 5.3.4 Resultierender Konzeptentwurf des PDM-basierten Überwachungswerkzeuges

Kapitel 5.3.1 bis 5.3.3 erstellten jeweils eine Musterinstanz für das TeamCenter-orientierte Überwachungswerkzeug der EADS Flugzeugentwicklung. Um zu verdeutlichen, wie die Präsentations- und Skalierungselemente im integrierten Zusammenhang stehen, wurden diese Konzeptentwürfe zu einem objektorientierten Datenmodell überführt. Das Ergebnis zeigt sich als UML-Klassendiagramm (siehe Abb. 5-8) und dient dem Überwachungswerkzeug in Kapitel 6 als konzeptionelle Grundlage. Durch Ergänzungen um weitere Skalierungselemente oder durch die Wahl unterschiedlicher präsentierender Kennlinienelemente kann der Umfang und damit die Datenbankkomplexität des TeamCenter-basierten Überwachungswerkzeuges beliebig gesteigert werden.

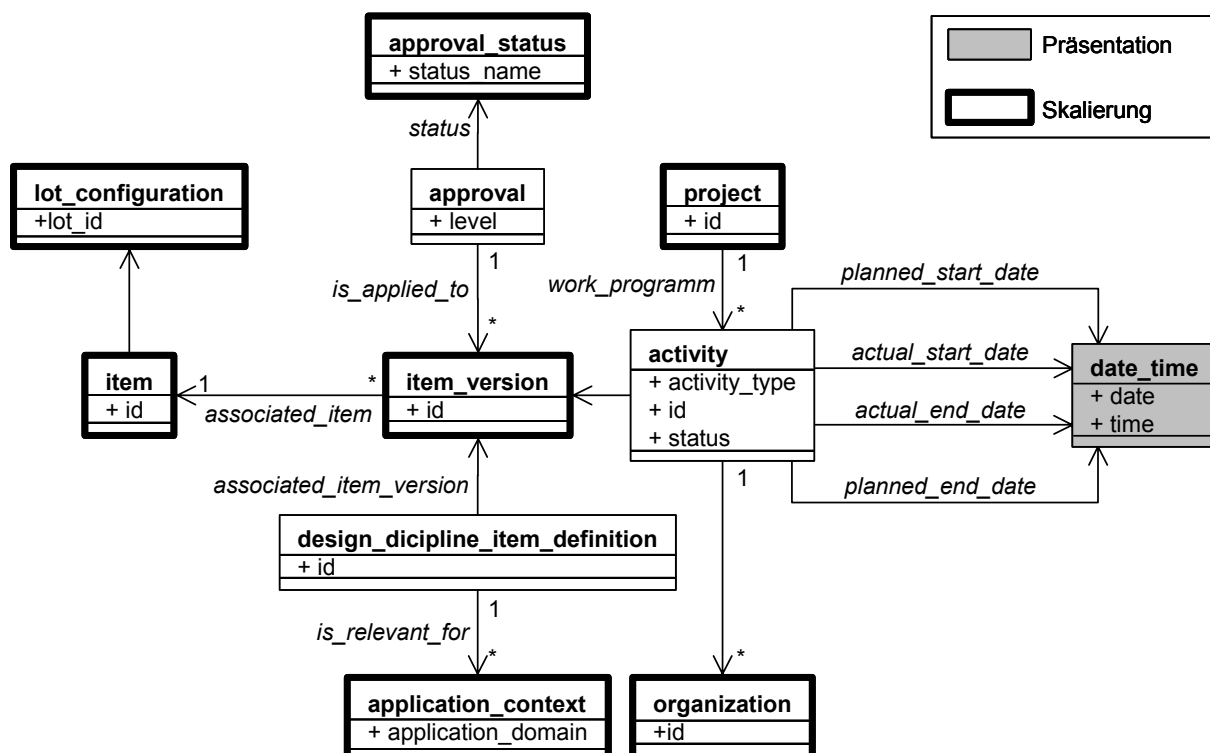


Abb. 5-8: Datenbankmodell des PDM-basierten Überwachungswerkzeuges  
(Notation: UML-Klassendiagramm)

#### **5.4 Fazit - systematische Unterstützung der PDM-Transformation**

Auf Grundlage von Kapitel 4 stellte dieses Kapitel einen Bibliotheksprototypen vor, der die Transformation zwischen dem PDM-Paradigma und dem Paradigma der Projektüberwachung systematisch unterstützt. Unter Berücksichtigung eines konkreten Praxisbeispiels aus der Flugzeugindustrie erstellte die Arbeit drei Musterinstanzen für das PDM-basierte Überwachungswerkzeug in Kapitel 6 Meilensteintrendanalyse, Sachfortschritts- und Balkendiagramm. Da die Pflichtenheftanforderungen, die Kapitel 3.5.2 an das zugrunde liegende Bibliothekskonzept stellte, durch den hier vorgestellten Bibliotheksprototypen erfüllt werden konnten, genauer gesagt KA 1.1, KA 1.2, KA 2.1, KA 2.2, KA 3.2 und KA 4.1, gilt die Tragfähigkeit des Bibliothekskonzeptes als nachgewiesen.

## 6. Das PDM-basierte Überwachungswerkzeug

Auf Grundlage der in Kapitel 5 erarbeiteten Musterinstanzen stellt dieses Kapitel ein PDM-basiertes Überwachungswerkzeug vor. Kapitel 6.1 zeigt die Architektur und technische Umsetzung des Überwachungsprototyps. Auf Basis eines Anwendungsbeispiels aus der Flugzeugentwicklung (Kapitel 6.2) präsentiert Kapitel 6.3 ein Balkendiagramm und integriertes Meilenstein-Sachfortschrittsdiagramm. Den Aufbau dieses Kapitels zeigt Abbildung 6.1.

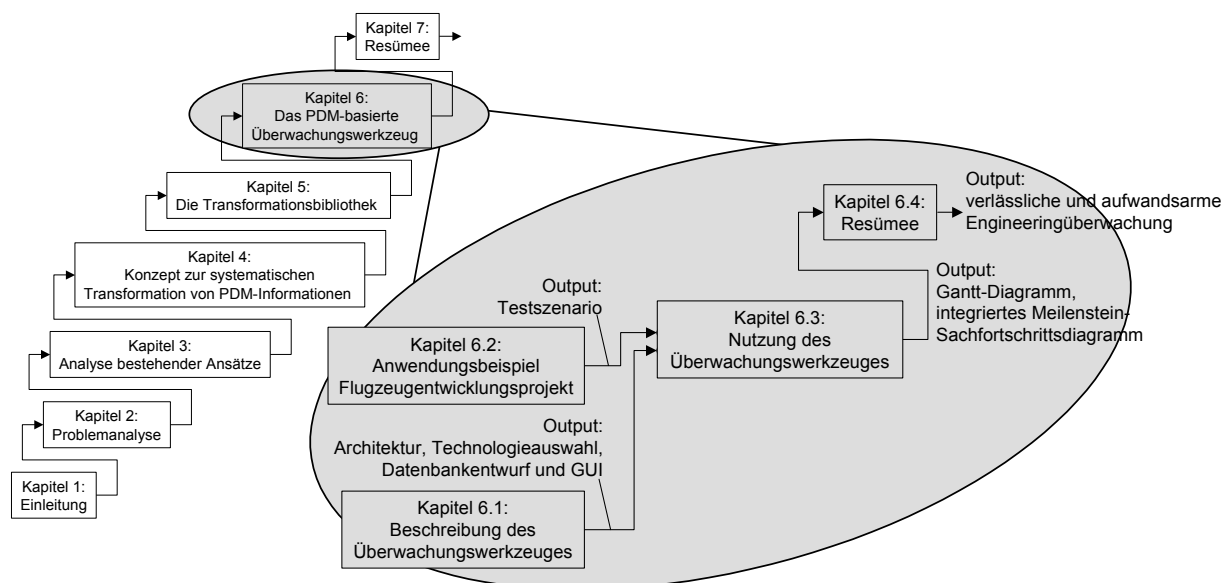


Abb. 6-1: Aufbau und Vorgehensweise Kapitel 6

## 6.1 Beschreibung des Überwachungswerkzeuges

### 6.1.1 Systemarchitektur und Technologieauswahl

Das PDM-System TeamCenter der Firma EDS dient u.a. der Entwicklungsabteilung bei EADS-M als zentrale Integrations- und Arbeitsplattform. Die Produktinformationen von TeamCenter werden innerhalb einer relationalen Datenbank von Oracle gehalten, die auf dem Betriebssystem Unix implementiert wurde. Die TeamCenter Clients laufen hingegen auf dem Betriebssystem Windows 2000. Über ein 100Mbit/s Ethernet (Netzwerkprotokoll: TCP/IP) wird einmal pro Woche maximal 2 GB an ausgewählten Produktinformationen in Form von csv-Dateien (comma seperated values) aus der Oracle-Datenbank exportiert und mit Hilfe eines Ladescripts in die relationale Datenbank Access von Microsoft importiert (siehe Abb. 6-2).

Grundlage des hier vorgestellten Überwachungsprototyps, der für das Betriebssystem Windows 2000 realisiert wurde, ist eine 2-Schichten Architektur. Dabei war die Benutzerschnittstelle (Überwachungscockpit) wie zuvor der Bibliotheksprototyp mit Hilfe der Programmiersprache Visual Basic 6.0 realisiert worden. Sie ist damit besonders auf Windows-Plattformen zugeschnitten. Um den Benutzern dynamische Abfragen zu ermöglichen, sind in Visual Basic 6.0 ActiveX-Komponenten eingebettet (z.B. für Combo-, Textboxen, TabStrip). Darüber hinaus verwendet der Prototyp ActiveX Data Objects (ADO's) für die Kommunikation zwischen der Benutzerschnittstelle und dem Datenbanktreiber (OLEDB 4.0) der MS-Access Datenbank. Als Grundlage der Datenbankimplementierung dienten die drei Musterinstanzen aus Kapitel 5.

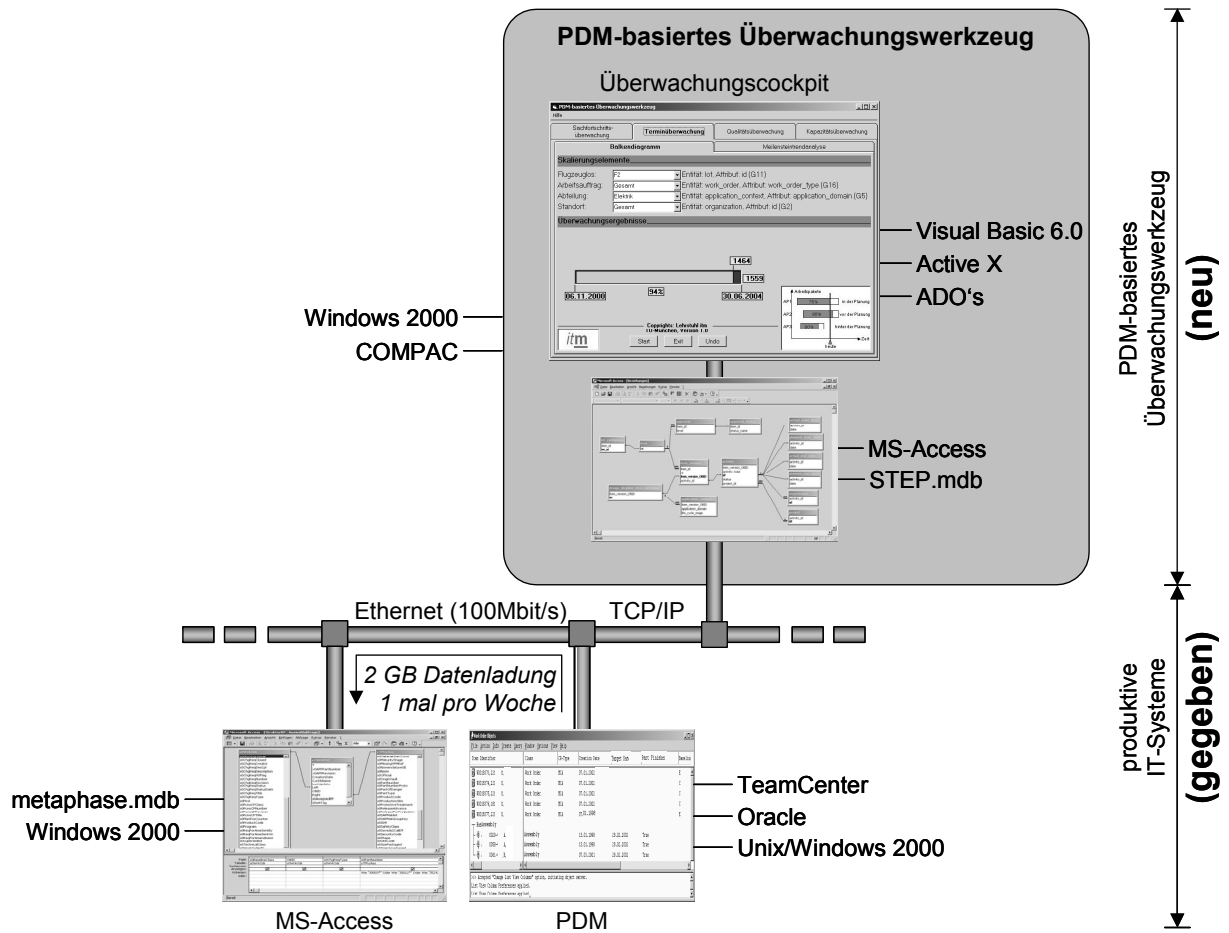
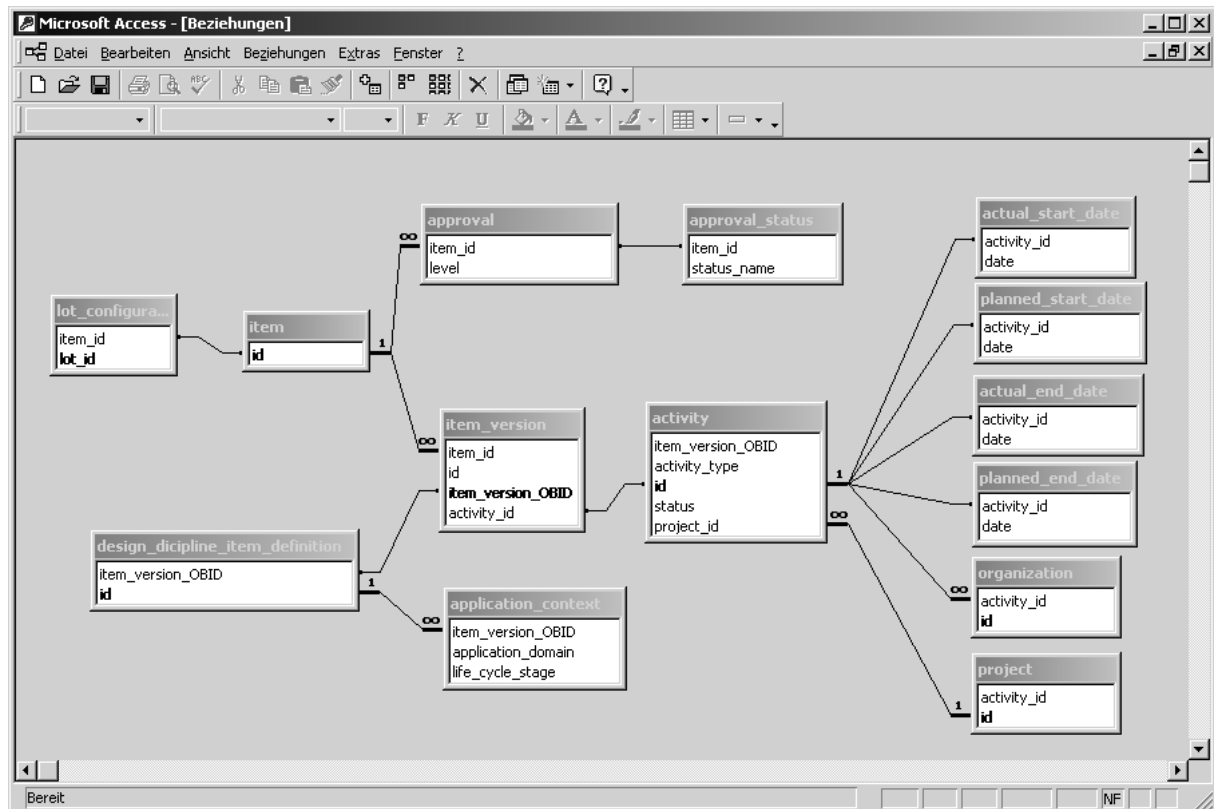


Abb. 6-2: Architektur und Technologieauswahl des Überwachungswerkzeuges

### 6.1.2 Datenbankentwurf

Das in Kapitel 5 entstandene UML-Klassendiagramm dient hier als Grundlage zur Entwicklung der STEP-basierten Datenbasis des PDM-basierten Überwachungswerkzeuges (vgl. Abb. 5-8). Dabei unterstützte wiederum die in Kapitel 5.1.2 vorgestellte Abbildung 5-3, das objektorientierte Konzept in einen relationalen Entwurf zu transformieren. Daher waren analog zu Kapitel 5 die MS-Access Datenbanktabellen um zusätzliche Felder (Fremdschlüssel) zu ergänzen, anhand derer die Objektbeziehungen aufgebaut werden konnten. Das Ergebnis wird durch Abbildung 6-3 repräsentiert.



**Abb. 6-3: Relationaler Datenbankentwurf in MS-Access**

### 6.1.3 Benutzerschnittstelle (GUI)

Abbildung 6-4 zeigt die entstandene Benutzeroberfläche (GUI) und Funktionalitäten des PDM-basierten Überwachungswerkzeuges, die in drei Blöcke unterteilt wurden:

- d) Der erste Block bietet dem Nutzer eine Navigation, die ihn ausgehend von der Analyseintension zu einer konkreten Überwachungskennlinie führt.
- e) Der zweite Block gibt dem Nutzer die Möglichkeit, die jeweilige Überwachungskennlinie an persönliche Informationsbedürfnisse anzupassen. Dies geschieht mit Hilfe derjenigen Skalierungselementen, die in den Musterinstanzen spezifiziert wurden (vgl. Kapitel 5).
- f) Im dritten Block visualisiert das Überwachungswerkzeug die Analyseergebnisse und ermöglicht damit dem Nutzer einen intuitiven und



verlässlichen Blick auf das derzeitige Geschehen im Produktentwicklungsbereich.

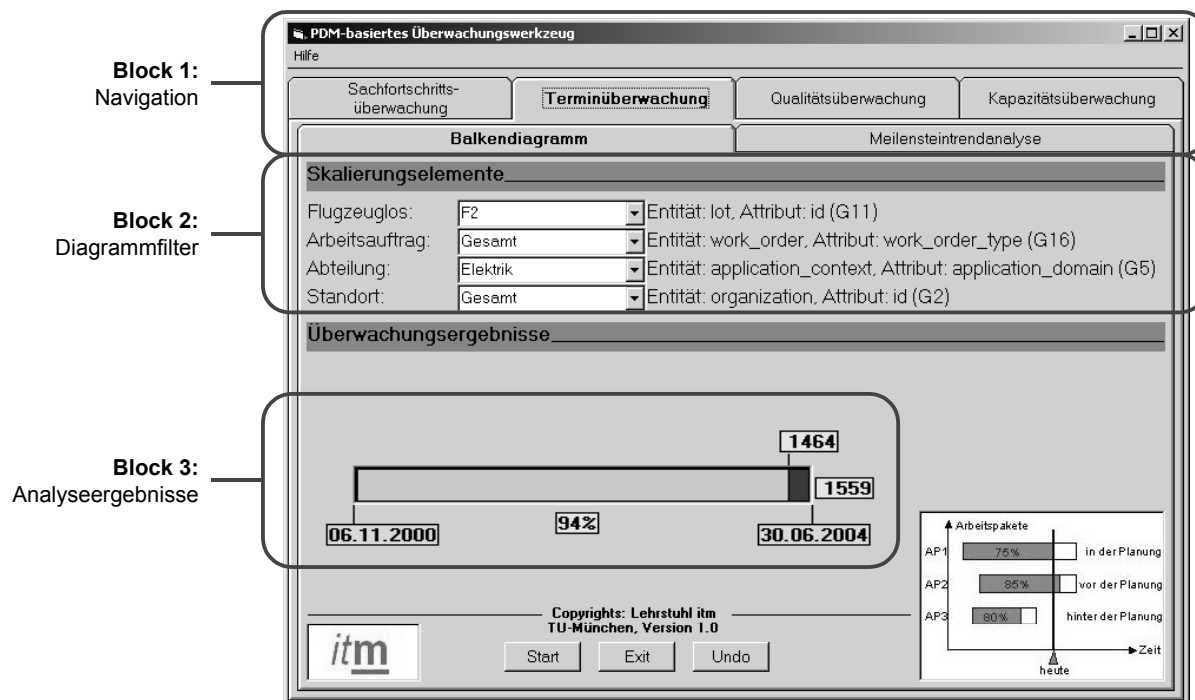


Abb. 6-4: Die Benutzerschnittstelle des PDM-basierten Überwachungsprototyps

## 6.2 Anwendungsbeispiel Flugzeugentwicklungsprojekt

Das durch den Geschäftsbereich Militärflugzeuge der EADS entwickelte und zu betreuende Hochleistungsflugzeug Eurofighter (EF-Typhoon) ist in vier multinationale Entwicklungs- und Bauanteile aufgeteilt. Die Entwicklung eines Flugzeugsystems (z.B. Flugsteuerung), welche typischerweise Bauanteile übergreifend ist, übernimmt ein einziger Projektpartner. Die physikalische Ausprägung der Flugzeugsysteme, wie z.B. die Fertigung von Kabelbäumen, unterliegen jedoch dem jeweiligen Bauanteileisigner. Die Endmontage und die Abnahme der Flugzeuge erfolgt länderspezifisch. Im Rahmen des bereits in Kapitel 5 erwähnten Kooperationsvorhabens entwickelte der Lehrstuhl für Informationstechnik (*itm*) der Technischen Universität München für die Abteilung Flugzeugentwicklung der EADS Militärflugzeuge ein PDM-basiertes Überwachungswerkzeug, das die Stamm- und

Strukturinformationen des dort eingesetzten und angepassten PDM-Systems TeamCenter von EDS als Informationsgrundlage nutzt. Im folgenden zeigt die Arbeit eine integrierte Darstellung des Entwicklungsprojektgeschehens speziell für die Flugzeugentwicklung.

## **6.3 Nutzung des Überwachungswerkzeuges**

### **6.3.1 Beispiel: Gantt-Diagramm**

Die Abbildung 6-5, bei der die Abteilungsnamen anonymisiert wurden, zeigt übersichtlich und entsprechend intuitiv die statische Sicht auf die Zielerreichung eines Teilprojekts. Durch den Fortschrittsbalken, der darstellt, wie viele Liefereinheiten (z.B. Zeichnungen) prozentual zur Gesamtlieferleistung bereits erbracht wurden, können rechtzeitig und zielgerichtet Ressourcen zur weiteren Zielerreichung eingesetzt werden. Bei dieser Form der graphischen Darstellung besteht aber die Gefahr, dass der Fortschrittsbalken den Konstruktionsverantwortlichen suggeriert, dass die Liefereinheiten (z.B. Zeichnungen) von Projektbeginn an in kontinuierlicher und linearer Form erbracht werden, was natürlich in der Praxis so oft nicht vorliegt. Das bedeutet, es ist eher damit zu rechnen, dass sich der Füllstand der Liefereinheiten (Zeichnungen) exponential über die Projektzeit hinweg entwickelt. Hinzu kommt, dass in das PDM-basierte Überwachungswerkzeug zusätzliche Features eingearbeitet wurden, die über die klassische Darstellungsform eines Balkendiagramms hinausreichen.

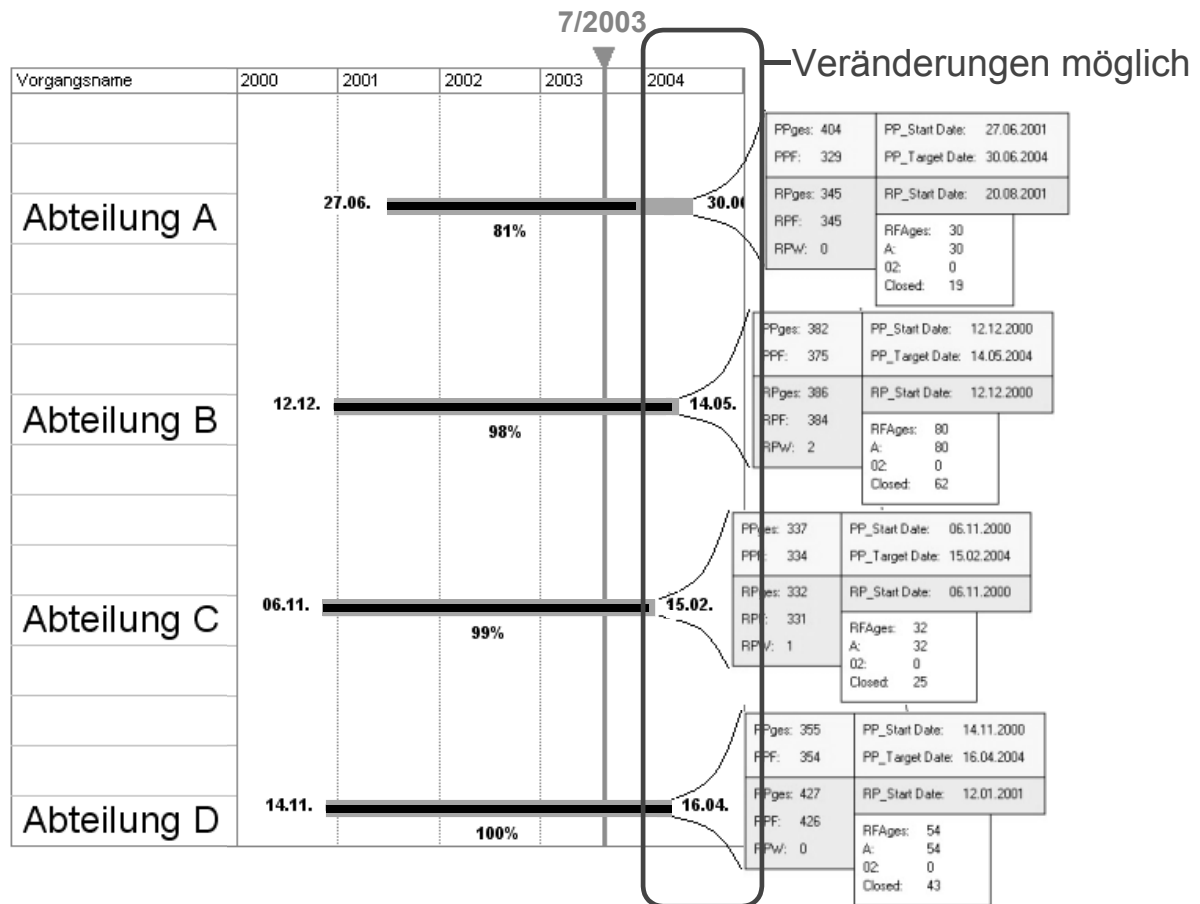


Abb. 6-5: Balkendiagramm eines Flugzeugentwicklungsprojektes

### 6.3.2 Beispiel: Integriertes Meilenstein-Sachfortschrittsdiagramm

In der Realität eines komplexen Entwicklungsprojektes kann man in der Regel nicht davon ausgehen, dass ein einmal spezifizierter Planungsrahmen über eine Entwicklungszeit von 2-3 Jahren konstant bleibt. Problematisch bei einer isolierten Sicht auf das Balkendiagramm ist, dass Schwankungen der Zielerreichung (Füllstand) aufgrund eines sich verändernden Planungshorizonts nicht direkt nachvollziehbar sind.

Aus diesem Grund ist es notwendig, ein weiteres Diagramm zu entwickeln, das sowohl die dynamische Entwicklung des Planungshorizontes widerspiegelt als auch die Entwicklung von Prognosen unterstützt (siehe Abb. 6-6). Das Diagramm, bei dem die Abteilungsnamen ebenso anonymisiert wurden wie die der Arbeitspakettypen, zeigt in

der oberen Kennlinie die Fortschreibung des geplanten Teilprojektendtermins aufgrund eines gesteigerten Änderungsumfanges, welche die unteren Balken darstellen und den entsprechend davon betroffenen Teilstämmen (Baugruppen, Bauteile), welche die unterste Kennlinie anzeigt.

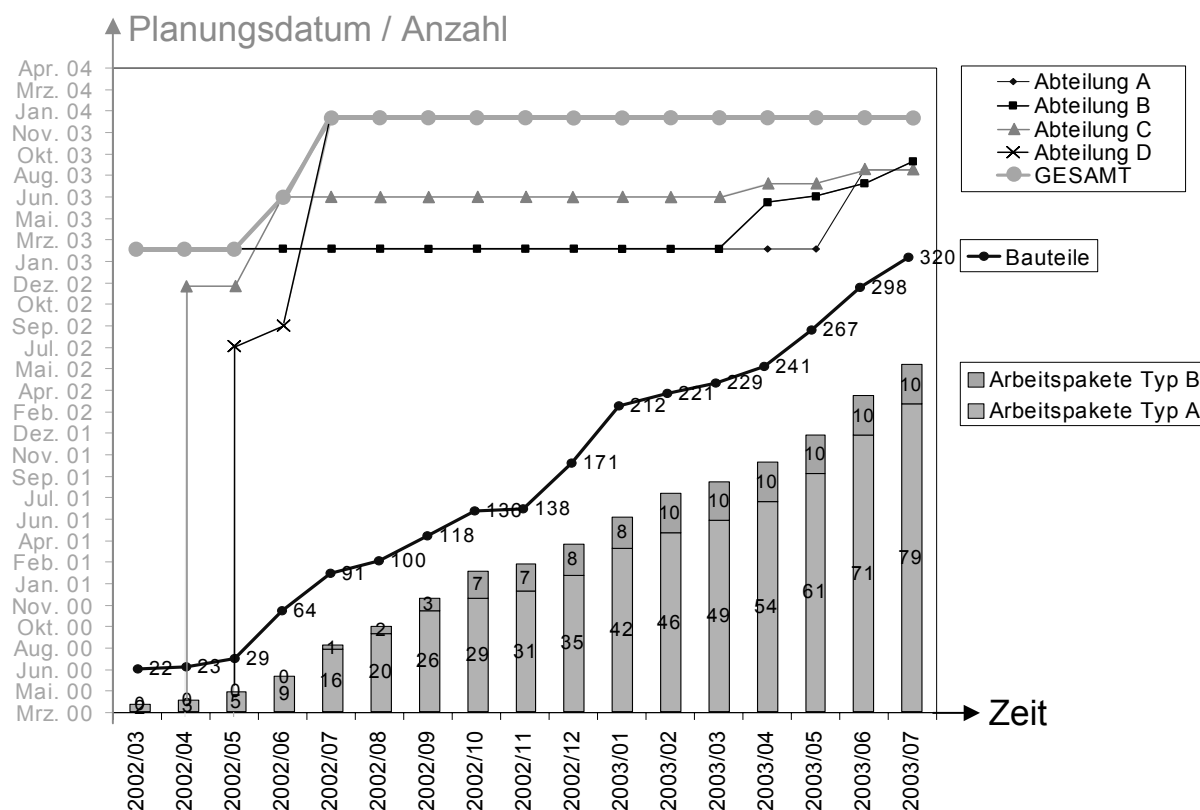


Abb. 6-6: Integriertes Meilenstein-Sachfortschrittsdiagramm

#### 6.4 Fazit - verlässliche und aufwandsarme Engineeringüberwachung

Anhand der drei Musterinstanzen, die Kapitel 5 entwickelte, stellte Kapitel 6 die Architektur und technologische Umsetzung eines PDM-basierten Überwachungswerkzeuges vor, das bereits in der Produktentwicklung eines Luft- und Raumfahrtunternehmens eingesetzt wird. Anhand einer Kopplung mit dem dort vorhandenen PDM-System TeamCenter (EDS) und zwei Analysebeispielen (Balkendiagramm, integriertes Meilenstein-Sachfortschrittsdiagramm) wurde gezeigt,

dass auf Basis des durch die Arbeit entwickelten „Konzeptes zur systematischen Transformation von PDM-Informationen“ eine verlässliche und aufwandsarme Übersicht auf das aktuelle Produktentwicklungsgeschehen erzeugbar ist. Sie lässt Störungen frühzeitig erkennen und hilft Gegenmaßnahmen zielgenau zu entwickeln.

Daneben unterstützt das PDM-basierte Überwachungswerkzeug eine optimale Ressourcenplanung und Ressourcenprognose in einem Entwicklungsprojekt. Bisherige Analyseergebnisse können darüber hinaus als Referenz für eine Ressourcenplanung ähnlicher neuer Flugzeugprojekte wertvolle Dienste leisten.

Die Pflichtenheftanforderungen, die Kapitel 3.5.2 an das PDM-basierte Überwachungswerkzeug stellte, konnten durch den hier vorgestellten Prototypen erfüllt werden, genauer gesagt KA 1.1, KA 1.2 und KA 3.1.



# 7. Resümee

## 7.1 Zusammenfassung

Eine zuverlässige Überwachung von Entwicklungsprojekten trägt zur Verwirklichung von geplanten Zielen wesentlich bei. Doch weist das derzeitige Engineeringcontrolling Defizite auf. Ursache dafür ist eine mangelnde Verlässlichkeit bisheriger Informationsressourcen.

Auf der Suche nach Substitutionsmöglichkeiten wurden Ansätze entdeckt, welche die Vision von der Nutzung existenter digitaler Informationen propagieren. Letztendlich musste aber eine ernüchternde Bilanz gezogen werden. Die Ansätze hatten es nicht geschafft, operative IT-Systeme als Informationsressourcen zu nutzen.

Es fiel aber auf, dass Produktdatenmanagement-Systeme im Vergleich die geeignetste Informationsressource für das Engineeringcontrolling sind. Die Entwicklung eines PDM-basierten Überwachungssystems ist aber schwierig. Dafür ist maßgeblich die Heterogenität zwischen dem PDM-Paradigma und dem Paradigma der Projektüberwachung verantwortlich. Es war somit die zentrale Herausforderung, die Paradigmentransformation zu unterstützen.

Aus dieser Erkenntnis heraus entwickelte die Dissertation in Kapitel 4 ein „Konzept zur systematischen Transformation von PDM-Informationen“, das sich aus den folgenden Elementen zusammensetzt:

### a) Metamodell der Transformationsunterstützung

Das UML-Klassendiagramm stellt die Kern- und Hilfsobjekte des Transformationskonzeptes sowie deren Beziehungen dar. Damit ist das

Metamodell der „Dreh- und Angelpunkt“ für eine Anpassung und Erweiterung des Transformationskonzeptes.

b) Entwicklungsstrategie für wieder verwendbare Transformationsmuster

Das UML-Aktivitätsdiagramm zeigt, wie unabhängig von einem konkreten Anwendungsfall eine semantische Zuordnung zwischen STEP-Attributen und Überwachungskennlinien zu entwickeln und formal zu beschreiben ist.

c) STEP-basierte Transformationsmuster

Die in tabellarischer Form dokumentierten Transformationsmuster stellen die semantische Zuordnung zwischen STEP-Attributen und den folgenden Überwachungskennlinien dar: Meilensteintrendanalyse, Gantt-, Belastungs- und Sachfortschrittsdiagramm, Netzplan, Änderungsquote und Gewicht.

d) Strategie zur Anpassung der Transformationsmuster

Das UML-Aktivitätsdiagramm zeigt, wie die Transformationsmuster an eine konkrete PDM-Systemlandschaft angepasst werden müssen. Das bedeutet, aus Transformationsmustern werden Musterinstanzen.

e) Systematischer Konfigurations- und Änderungsmanagementansatz

Das UML-Klassendiagramm regelt die konsistente Verwaltung der Transformationsmuster und Musterinstanzen.

f) Intuitives Zugriffsprinzip

Das UML-Aktivitätsdiagramm koordiniert die Interaktion zwischen dem Anwender auf der einen Seite und den konfigurationsgesteuerten Transformationsmuster und deren Musterinstanzen auf der anderen Seite.



g) Leitlinien zur Umsetzung der Musterinstanzen

Sie zeigen die wichtigen Grundsätze, die es zu beachten gilt, wenn Musterinstanzen als Grundlage zur Entwicklung eines PDM-basierten Überwachungswerkzeuges verwendet werden.

Schließlich verifizieren zwei Prototypen das „Konzept zur systematischen Transformation von PDM-Informationen“:

a) Prototyp einer Transformationsbibliothek

Er beinhaltet Transformationsmuster für die gängigsten Überwachungskennlinien, die im Rahmen der Arbeit entwickelt wurden (siehe Anhang A bis H). Die Nutzung der Transformationsbibliothek führte zu Musterinstanzen, welche dem PDM-basierten Überwachungswerkzeug als konzeptionelle Grundlage dienen.

b) Prototyp eines PDM-basierten Überwachungswerkzeuges

Das in der Entwicklung eines Luft- und Raumfahrtunternehmens schon produktiv eingesetzte Werkzeug ist durch eine geschickte Kopplung in der Lage, Informationen des PDM-Systems TeamCenter (EDS) in Form von Meilensteintrendanalysen, Gantt-, Belastungs- und Sachfortschrittsdiagrammen darzustellen.

## 7.2 Ausblick

Das in dieser Arbeit entwickelte „Konzept zur systematischen Transformation von PDM-Informationen“ unterstützt die Entwicklung eines PDM-basierten Überwachungswerkzeuges. Weitere Themen, die im Rahmen dieser Arbeit nicht verfolgt werden konnten, sind wie folgt zusammengefasst:

a) Das Transformationskonzept wurde an den Bedürfnissen der Produktentwicklung ausgerichtet. PDM-Systeme sind aber auch in anderen Lebenszyklusphasen im

Einsatz. Es liegt entsprechend nahe, das Transformationskonzept auch dort zu verwenden. Diese Möglichkeit gilt es zu überprüfen.

b) Eine verlässliche Kostenübersicht ist für die Engineeringüberwachung von Bedeutung. Mit STEP können diese Kosteninformationen abgebildet werden, sie befinden sich bisher aber in Produktplanungs- und Steuerungssystemen und nicht in Produktdatenmanagement-Systemen. Aus diesem Grund ist es sinnvoll, in das Transformationskonzept zusätzliche Transformationsmuster aufzunehmen. Deren Funktion muss es sein, dem Anwender die semantische Zuordnung zwischen den kostenrelevanten Ausschnitten des PPS- und STEP-Paradigmas zu zeigen.

c) Fehlende oder fehlerhafte Datensätze sind in produktiven PDM-Systemen nicht auszuschließen. Sie können aber zu falschen Überwachungsergebnissen führen. Deswegen ist es zweckmäßig, das Transformationskonzept durch geeignete Komponenten zu ergänzen, welche eine systematische Identifizierung und Bereinigung von falschen PDM-Datensätzen unterstützen (Stichwort: Data Cleansing).

# Anhang

- Anhang A: Transformationsmuster für Skalierungselemente
- Anhang B: Transformationsmuster für Sachfortschrittsdiagramme
- Anhang C: Transformationsmuster für Netzpläne
- Anhang D: Transformationsmuster für Gantt-Diagramme
- Anhang E: Transformationsmuster für Meilensteintrenddiagramme
- Anhang F: Transformationsmuster für die Änderungsquote
- Anhang G: Transformationsmuster für das Gewicht
- Anhang H: Transformationsmuster für die Belastungsdiagramme
- Anhang I: Data Dictionary des Konfigurations- und  
Änderungsmanagementansatzes



## Anhang A: Transformationsmuster für Skalierungselemente

<b>"Skalierungselemente" und STEP (AP 214-ARM):</b>			
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>
S-1	Anwendungsbereich	application_context	application_context.application_domain
S-2	Anwendungsbereich von Arbeitsanträgen	application_context <= is_relevant_for <= design_dicipline_item_definition => associated_item_version => item_version => activity => resolved_request => work_request	application_context.application_domain, work_request.id, work_request.version_id, work_request.status, work_request.request_type
S-3	Anwendungsbereich von Arbeitsaufträgen	application_context <= is_relevant_for <= design_dicipline_item_definition => associated_item_version => item_version => activity <= is_controlling <= work_order	application_context.application_domain, work_order.id, work_order.version_id, work_order.work_order_type
S-4	Anwendungsbereich von Bauteilen	application_context <= is_relevant_for <= design_dicipline_item_definition => associated_item_version => item_version => associated_item => item	application_context.application_domain, item.id bzw. item_version.id
S-5	Anwendungsbereich von Projekten/Teilprojekten	application_context <= is_relevant_for <= design_dicipline_item_definition => associated_item_version => item_version => activity <= work_programm <= project	application_context.application_domain, project.id
S-6	Anwendungsbereich von Vorgängen	application_context <= is_relevant_for <= design_dicipline_item_definition => associated_item_version => item_version => activity	application_context.application_domain, activity.activity_type, activity.status, activity.id
S-7	Arbeitsantrag	work_request	work_request.id, work_request.version_id, work_request.status, work_request.request_type
S-8	Arbeitsanträge eines Projektes	work_request <= resolved_request <= activity <= work_programm <= project	work_request.id, work_request.version_id, work_request.status, work_request.request_type, project.id
S-9	Arbeitsaufträge	work_order	work_order.id, work_order.version_id, work_order.work_order_type
S-10	Arbeitsaufträge eines Projektes	work_order => is_controlling => activity <= work_programm <= project	work_order.id, work_order.version_id, work_order.work_order_type, project.id
S-11	Baugruppentyp	assembly_definition	assembly_definition.assembly_type
S-12	Bauteile	item bzw. item_version	item.id bzw. item_version.id

**Tab. A-1: Transformationsmuster für Skalierungselemente (Teil 1)**

Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:
S-13	Bauteile von Arbeitsanträgen	item <= associated_item <= item_version => activity => resolved_request => work_request	item.id bzw. item_version.id, work_request.id, work_request.version_id, work_request.status, work_request.request_type
S-14	Bauteile von Arbeitsaufträgen	item <= associated_item <= item_version => activity <= is_controlling <= work_order	item.id bzw. item_version.id, work_order.id, work_order.version_id, work_order.work_order_type
S-15	Bauteile eines Loses	lot_configuration <= item <= associated_item <= item_version	lot_configuration.lot_id, item.id bzw. item_version.id
S-16	Bauteile eines Projektes	item <= associated_item <= item_version => activity <= work_programm <= project	item.id bzw. item_version.id, project.id
S-17	Bauteile eines Standortes	item <= associated_item <= item_version => activity => organization	item.id bzw. item_version.id, organization.id
S-18	Bauteile eines Vorgangs	item <= associated_item <= item_version => activity	item.id bzw. item_version.id, activity.activity_type, activity.status, activity.id
S-19	Bauteileklasse	specific_item_classification	specific_item_classification. classification_name, specific_item_classification. classification_type
S-20	Lebenszyklusphase	application_context	application_context.life_cycle_ stage
S-21	Los	lot_configuration	lot_configuration.lot_id, lot_configuration.lot_size
S-22	Organisation	organization	organization.delivery_address, organization.id, organization.organization_type, organization.postal_address, organization.visitor_address
S-23	Person	person	person.address, person.person_name, person. id
S-24	Produktklasse	product_class	product_class.id, product_class.level_type, product_class.version_id
S-25	Projekt/Teilprojekt	project	project.id
S-26	Reifegrad von Arbeitsanträgen	work_request	work_request.status
S-27	Reifegrad von Arbeitsaufträgen	work_order => is_controlling => activity	work_order.id, work_order.version_id, work_order.work_order_type, activity.status
S-28	Reifegrad von Bauteilen	item <= associated_item <= item_version <= is_applied_to <= approval => status => approval_status	item.id bzw. item_version.id, approval_status.status_name
S-29	Reifegrad von Projekten/Teilprojekten	project => work_programm => activity	project.id, activity.status

**Tab. A-2: Transformationsmuster für Skalierungselemente (Teil 2)**

Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:
S-30	Reifegrad von Vorgängen	activity	activity.status
S-31	Vorgang	activity	activity.activity_type, activity.status, activity.id
S-32	Vorgänge arbeitsantragsbezogen	activity => resolved_request => work_request	activity.activity_type, activity.status, activity.id, work_request.id, work_request.version_id, work_request.status, work_request.request_type
S-33	Vorgänge arbeitsauftragsbezogen	activity <= is_controlling <= work_order	activity.activity_type, activity.status, activity.id, work_order.id, work_order.version_id, work_order.work_order_type
S-34	Vorgänge bauteilebezogen	activity <= item_version => associated_item => item	activity.activity_type, activity.status, activity.id, item.id bzw. item_version.id
S-35	Vorgänge projekt- /teilprojektbezogen	activity <= work_programm <= project	activity.activity_type, activity.status, activity.id, project.id
S-36	Zeit	date_time	date_time.date, date_time.time

**Tab. A-3: Transformationsmuster für Skalierungselemente (Teil 3)**





## Anhang B: Transformationsmuster für Sachfortschrittsdiagramme

### "Präsentierende Elemente" von Sachfortschrittsdiagrammen und STEP (AP 214-ARM)

Sachfortschritt				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
F-1	arbeitsantragsbezogen	work_request	work_request.id, work_request.version_id, work_request.status, work_request.request_type	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "work_request", die Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_request"
F-2	arbeitsantragsbezogen	work_request <= resolved_request <= activity	work_request.id, work_request.version_id, work_request.request_type, activity.status	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_request"
F-3	arbeitsantragsbezogen	work_request <= resolved_request <= activity <= item_version <= is_applied_to <= approval => status => approval_status	work_request.id, work_request.version_id, work_request.request_type, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status", die Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_request"
F-4	arbeitsauftragsbezogen	work_order => is_controlling => activity	work_order.id, work_order.version_id, work_order.work_order_type, activity.status, activity.id, activity.activity_type	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id", "version_id" oder "work_order_type" der Entität "work_order"
F-5	arbeitsauftragsbezogen	work_order => is_controlling => activity <= item_version <= approval => status => approval_status	work_order.id, work_order.version_id, work_order.work_order_type, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status", die Inhalte im Datenfeld "id", "version_id" oder "work_order_type" der Entität "work_order"
F-6	bauteilebezogen	item <= associated_item <= item_version <= approval => status => approval_status	item.id, item_version.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status", die Inhalte im Datenfeld "id" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"
F-7	bauteilebezogen	item <= associated_item <= item_version => activity	item.id bzw. item_version.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"
F-8	projekt-/teilprojektbezogen	project => work_programm => activity	project.id, activity.status	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id" der Entität "project"

**Tab. A-4: Transformationsmuster für Sachfortschrittsdiagramme (Teil 1)**

<b>Sachfortschritt</b>				
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>	<b>Verdichtungsalgorithmen:</b>
F-9	projekt-/teilprojektbezogen	project => work_programm => activity <= item_version <= is_applied_to <= approval => status => approval_status	project.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status", die Inhalte im Datenfeld "id" der Entität "project"
F-10	vorgangsbezogen	activity	activity.id, activity.activity_type, activity.status	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id" oder "activity_type"
F-11	vorgangsbezogen	activity <= item_version <= approval => status => approval_status	activity.id, activity.activity_type, approval_status.status_ name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status", die Inhalte im Datenfeld "id" oder "activity_type"
<b>Zeit</b>				
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>	<b>Verdichtungsalgorithmen:</b>
Z-1	arbeitsantragsbezogen	work_request <= resolved_request <= activity => actual_start_date => date_time	work_request.id, work_request.version_id, work_request.status, work_request.request_type, date_time.date, date_time.time	die Inhalte des Datenfeldes "date" bzw. "time" entsprechend den Präsentationsanforderungen verdichten (z.B. täglich, wöchentlich, monatlich, jährlich, etc.)
Z-2	arbeitsauftragsbezogen	work_order => is_controlling => activity => actual_start_date => date_time	work_order.id, work_order.version_id, work_order.work_order_type, date_time.date, date_time.time	wie Z-1
Z-3	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	item.id, item_version.id, date_time.date, date_time.time	wie Z-1
Z-4	projekt-/teilprojektbezogen	project => actual_start_date => date_time	project.id, date_time.date, date_time.time	wie Z-1
Z-5	vorgangsbezogen	activity => actual_start_date => date_time	activity.id, activity.activity_type, date_time.date, date_time.time	wie Z-1

**Tab. A-5: Transformationsmuster für Sachfortschrittsdiagramme (Teil 2)**

## Anhang C: Transformationsmuster für Netzpläne

### "Präsentierende Elemente" von Netzplänen und STEP (AP 214-ARM)

Vorgangsname				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
NA-1	Name	activity	activity.activity_type,	Gruppriere die Inhalte der Datenfelder "activity_type" der Entität "activity"
Vorgangsnummer				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
NU-1	Nummer	activity	activity.id	Gruppriere die Inhalte der Datenfelder "id" der Entität "activity"
Abarbeitungsgrad				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
A-1	arbeitsantragsbezogen	work_request	work_request.id, work_request.version_id, work_request.status, work_request.request_type	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status" der Entität "work_request"
A-2	arbeitsantragsbezogen	work_request <= resolved_request <= activity	work_request.id, work_request.version_id, work_request.request_type, activity.status	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status" der Entität "activity" und der Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_order"
A-3	arbeitsantragsbezogen	work_request <= resolved_request <= activity <= item_version <= is_applied_to <= approval => status => approval_status	work_request.id, work_request.version_id, work_request.request_type, approval_status.status_name	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" und der Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_request"
A-4	arbeitsauftragsbezogen	work_order => is_controlling => activity	work_order.id, work_order.version_id, work_order.work_order_type, activity.status, activity.id, activity.activity_type	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status" der Entität "activity" und der Inhalte im Datenfeld "id", "version_id" oder "work_order_type" der Entität "work_order"

Tab. A-6: Transformationsmuster für Netzpläne (Teil 1)

Abarbeitungsgrad				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
A-5	arbeitsauftragsbezogen	work_order => is_controlling => activity <= item_version <= approval =>status => approval_status	work_order.id, work_order.version_id, work_order.work_order_type, approval_status.status_name	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" und der Inhalte im Datenfeld "id", "version_id" oder "work_order_type" der Entität "work_order"
A-6	bauteilebezogen	item <= associated_item <= item_version <= approval =>status => approval_status	item.id, item_version.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"
A-7	bauteilebezogen	item <= associated_item <= item_version => activity	item.id bzw. item_version.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status", die Inhalte im Datenfeld "id" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"
A-8	projekt-/teilprojektbezogen	project => work_programm => activity	project.id, activity.status, activity.id, activity.activity_type	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id" der Entität "project"
A-9	projekt-/teilprojektbezogen	project => work_programm => activity <= item_version <= approval =>status => approval_status	project.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status", die Inhalte im Datenfeld "id" der Entität "project"
A-10	vorgangsbezogen	activity	activity.id, activity.activity_type, activity.status, activity.id, activity.activity_type	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id" oder "activity_type"
A-11	vorgangsbezogen	activity <= item_version <= approval =>status => approval_status	activity.id, activity.activity_type, approval_status.status_name	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" und der Inhalte im Datenfeld "id" oder "activity_type" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item version"

**Tab. A-7: Transformationsmuster für Netzpläne (Teil 2)**

Vorgangsanfang				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
VA-1	arbeitsantragsbezogen	<p><b>Fall 1</b> die Abarbeitung eines Arbeitsantrages hat begonnen: work_request &lt;= resolved_request &lt;= activity =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung eines Arbeitsantrages hat noch nicht begonnen: work_order =&gt; is_controlling =&gt; activity =&gt; planned_start_date =&gt; date_time</p>	work_request.id, work_request.version_id, work_request.status, work_request.request_type, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "work_request" oder unter Berücksichtigung des "status" der Entität "activity"
VA-2	arbeitsauftragsbezogen	<p><b>Fall 1</b> die Abarbeitung eines Arbeitsauftrages hat begonnen: work_order =&gt; is_controlling =&gt; activity =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung eines Arbeitsauftrages hat noch nicht begonnen: work_order =&gt; is_controlling =&gt; activity =&gt; planned_start_date =&gt; date_time</p>	work_order.id, work_order.version_id, work_order.work_order_type, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id", "version_id" oder "work_order_type", der Entität "work_order"
VA-3	bauteilebezogen	<p><b>Fall 1</b> die Abarbeitung der Bauteile hat begonnen: item &lt;= associated_item &lt;= item_version =&gt; activity =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung der Bauteile hat noch nicht begonnen: item &lt;= associated_item &lt;= item_version =&gt; activity =&gt; planned_start_date =&gt; date_time</p>	item.id, item_version.id, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id" der Entität "item" oder unter Berücksichtigung des Datenfeldes "id" der Entität "item_version"
VA-4	projekt-/teilprojektbezogen	<p><b>Fall 1</b> die Abarbeitung des Projektes oder Teilprojektes hat begonnen: activity &lt;= work_programm &lt;= project =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung des Projektes oder Teilprojektes hat noch nicht begonnen: activity &lt;= work_programm &lt;= project =&gt; planned_start_date =&gt; date_time</p>	project.id, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id" der Entität "project"
VA-5	vorgangsbezogen	<p><b>Fall 1</b> die Abarbeitung des Vorgangs hat begonnen: activity =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung des Vorgangs hat noch nicht begonnen: activity =&gt; planned_start_date =&gt; date_time</p>	activity.id, activity.activity_type, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" und "id" oder "activity_type" der Entität "activity"

Tab. A-8: Transformationsmuster für Netzpläne (Teil 3)

Vorgangsende				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
VE-1	arbeitsantragsbezogen	<p><b>Fall 1</b> die Abarbeitung eines Arbeitsantrages ist abgeschlossen: work_request &lt;= resolved_request &lt;= activity =&gt; actual_end_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung eines Arbeitsantrages ist noch nicht abgeschlossen: work_order =&gt; is_controlling =&gt; activity =&gt; planned_end_date =&gt; date_time</p>	work_request.id, work_request.version_id, work_request.status, work_request.request_type, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "work_request" oder unter Berücksichtigung des "status" der Entität "activity"
VE-2	arbeitsauftragsbezogen	<p><b>Fall 1</b> die Abarbeitung eines Arbeitsauftrages ist abgeschlossen: work_order =&gt; is_controlling =&gt; activity =&gt; actual_end_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung eines Arbeitsauftrages ist noch nicht abgeschlossen: work_order =&gt; is_controlling =&gt; activity =&gt; planned_end_date =&gt; date_time</p>	work_order.id, work_order.version_id, work_order.work_order_type, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id", "version_id" oder "work_order_type", der Entität "work_order"
VE-3	bauteilebezogen	<p><b>Fall 1</b> die Abarbeitung der Bauteile ist abgeschlossen: item &lt;= associated_item &lt;= item_version =&gt; activity =&gt; actual_end_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung der Bauteile ist noch nicht abgeschlossen: item &lt;= associated_item &lt;= item_version =&gt; activity =&gt; planned_end_date =&gt; date_time</p>	item.id, item_version.id, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id" der Entität "item" oder unter Berücksichtigung des Datenfeldes "id" der Entität "item_version"
VE-4	projekt-/teilprojektbezogen	<p><b>Fall 1</b> die Abarbeitung des Projektes oder Teilprojektes ist abgeschlossen: activity &lt;= work_programm &lt;= project =&gt; actual_end_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung des Projektes oder Teilprojektes ist noch nicht abgeschlossen: activity &lt;= work_programm &lt;= project =&gt; planned_end_date =&gt; date_time</p>	project.id, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id" der Entität "project"

**Tab. A-9: Transformationsmuster für Netzpläne (Teil 4)**

Vorgangsende				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
VE-5	vorgangsbezogen	<b>Fall 1</b> die Abarbeitung des Vorgangs ist abgeschlossen: activity => actual_end_date => date_time <b>Fall 2</b> die Abarbeitung des Vorgangs ist noch nicht abgeschlossen: activity => planned_end_date => date_time	activity.id, activity.activity_type, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" und "id" oder "activity_type" der Entität "activity"

**Tab. A-10: Transformationsmuster für Netzpläne (Teil 5)**





## Anhang D: Transformationsmuster für Gantt-Diagramme

### "Präsentierende Elemente" von Gantt-Diagrammen und STEP (AP 214-ARM)

Füllstand				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
FS-1	arbeitsantragsbezogen	work_request	work_request.id, work_request.version_id, work_request.status, work_request.request_type	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status" der Entität "work_request"
FS-2	arbeitsantragsbezogen	work_request <= resolved_request <= activity	work_request.id, work_request.version_id, work_request.request_type, activity.status	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status" der Entität "activity" und der Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_order"
FS-3	arbeitsantragsbezogen	work_request <= resolved_request <= activity <= item_version <= is_applied_to <= approval => status => approval_status	work_request.id, work_request.version_id, work_request.request_type, approval_status.status_name	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" und der Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_request"
FS-4	arbeitsauftragsbezogen	work_order => is_controlling => activity	work_order.id, work_order.version_id, work_order.work_order_type, activity.status, activity.id, activity.activity_type	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status" der Entität "activity" und der Inhalte im Datenfeld "id", "version_id" oder "work_order_type" der Entität "work_order"
FS-5	arbeitsauftragsbezogen	work_order => is_controlling => activity <= item_version <= approval =>status => approval_status	work_order.id, work_order.version_id, work_order.work_order_type, approval_status.status_name	Ermittle das Verhältnis zwischen fertigen und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" und der Inhalte im Datenfeld "id", "version_id" oder "work_order_type" der Entität "work_order"

**Tab. A-11: Transformationsmuster für Gantt-Diagramme (Teil 1)**

Füllstand				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
FS-6	bauteilebezogen	item <= associated_item <= item_version <= approval =>status => approval_status	item.id, item_version.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity" die Inhalte im Datenfeld "id" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"
FS-7	bauteilebezogen	item <= associated_item <= item_version => activity	item.id bzw. item_version.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" die Inhalte im Datenfeld "id" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"
FS-8	projekt-/teilprojektbezogen	project => work_programm => activity	project.id, activity.status, activity.id, activity.activity_type	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity" die Inhalte im Datenfeld "id" der Entität "project"
FS-9	projekt-/teilprojektbezogen	project => work_programm => activity <= item_version <= approval =>status => approval_status	project.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" die Inhalte im Datenfeld "id" der Entität "project"
FS-10	vorgangsbezogen	activity	activity.id, activity.activity_type, activity.status, activity.id, activity.activity_type	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity" die Inhalte im Datenfeld "id" oder "activity_type"
FS-11	vorgangsbezogen	activity <= item_version <= approval =>status => approval_status	activity.id, activity.activity_type, approval_status.status_name	Ermittle das Verhältnis zwischen fertigem und gesamten Arbeitsvolumen durch die Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" und der Inhalte im Datenfeld "id" oder "activity_type" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"

**Tab. A-12: Transformationsmuster für Gantt-Diagramme (Teil 2)**

Startdatum				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
SD-1	arbeitsantragsbezogen	<p><b>Fall 1</b> die Abarbeitung eines Arbeitsantrages hat begonnen: work_request &lt;= resolved_request &lt;= activity =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung eines Arbeitsantrages hat noch nicht begonnen: work_order =&gt; is_controlling =&gt; activity =&gt; planned_start_date =&gt; date_time</p>	work_request.id, work_request.version_id, work_request.status, work_request.request_type, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "work_request" oder unter Berücksichtigung des "status" der Entität "activity"
SD-2	arbeitsauftragsbezogen	<p><b>Fall 1</b> die Abarbeitung eines Arbeitsauftrages hat begonnen: work_order =&gt; is_controlling =&gt; activity =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung eines Arbeitsauftrages hat noch nicht begonnen: work_order =&gt; is_controlling =&gt; activity =&gt; planned_start_date =&gt; date_time</p>	work_order.id, work_order.version_id, work_order.work_order_type, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id", "version_id" oder "work_order_type", der Entität "work_order"
SD-3	bauteilebezogen	<p><b>Fall 1</b> die Abarbeitung der Bauteile hat begonnen: item &lt;= associated_item &lt;= item_version =&gt; activity =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung der Bauteile hat noch nicht begonnen: item &lt;= associated_item &lt;= item_version =&gt; activity =&gt; planned_start_date =&gt; date_time</p>	item.id, item_version.id, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id" der Entität "item" oder unter Berücksichtigung des Datenfeldes "id" der Entität "item_version"
SD-4	projekt-/teilprojektbezogen	<p><b>Fall 1</b> die Abarbeitung des Projektes oder Teilprojektes hat begonnen: activity &lt;= work_programm &lt;= project =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung des Projektes oder Teilprojektes hat noch nicht begonnen: activity &lt;= work_programm &lt;= project =&gt; planned_start_date =&gt; date_time</p>	project.id, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id" der Entität "project"
SD-5	vorgangsbezogen	<p><b>Fall 1</b> die Abarbeitung des Vorgangs hat begonnen: activity =&gt; actual_start_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung des Vorgangs hat noch nicht begonnen: activity =&gt; planned_start_date =&gt; date_time</p>	activity.id, activity.activity_type, activity.status, date_time.date, date_time.time	Nimm das niedrigste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" und "id" oder "activity_type" der Entität "activity"

Tab. A-13: Transformationsmuster für Gantt-Diagramme (Teil 3)

Enddatum				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
ED-1	arbeitsantragsbezogen	<p><b>Fall 1</b> die Abarbeitung eines Arbeitsantrages ist abgeschlossen: work_request &lt;= resolved_request &lt;= activity =&gt; actual_end_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung eines Arbeitsantrages ist noch nicht abgeschlossen: work_order =&gt; is_controlling =&gt; activity =&gt; planned_end_date =&gt; date_time</p>	work_request.id, work_request.version_id, work_request.status, work_request.request_type, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "work_request" oder unter Berücksichtigung des "status" der Entität "activity"
ED-2	arbeitsauftragsbezogen	<p><b>Fall 1</b> die Abarbeitung eines Arbeitsauftrages ist abgeschlossen: work_order =&gt; is_controlling =&gt; activity =&gt; actual_end_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung eines Arbeitsauftrages ist noch nicht abgeschlossen: work_order =&gt; is_controlling =&gt; activity =&gt; planned_end_date =&gt; date_time</p>	work_order.id, work_order.version_id, work_order.work_order_type, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id", "version_id" oder "work_order_type", der Entität "work_order"
ED-3	bauteilebezogen	<p><b>Fall 1</b> die Abarbeitung der Bauteile ist abgeschlossen: item &lt;= associated_item &lt;= item_version =&gt; activity =&gt; actual_end_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung der Bauteile ist noch nicht abgeschlossen: item &lt;= associated_item &lt;= item_version =&gt; activity =&gt; planned_end_date =&gt; date_time</p>	item.id, item_version.id, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id" der Entität "item" oder unter Berücksichtigung des Datenfeldes "id" der Entität "item_version"
ED-4	projekt-/teilprojektbezogen	<p><b>Fall 1</b> die Abarbeitung des Projektes oder Teilprojektes ist abgeschlossen: activity &lt;= work_programm &lt;= project =&gt; actual_end_date =&gt; date_time</p> <p><b>Fall 2</b> die Abarbeitung des Projektes oder Teilprojektes ist noch nicht abgeschlossen: activity &lt;= work_programm &lt;= project =&gt; planned_end_date =&gt; date_time</p>	project.id, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" der Entität "activity" und des Datenfeldes "id" der Entität "project"

**Tab. A-14: Transformationsmuster für Gantt-Diagramme (Teil 4)**

Enddatum				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
ED-5	vorgangsbezogen	<b>Fall 1</b> die Abarbeitung des Vorgangs ist abgeschlossen: activity => actual_end_date => date_time <b>Fall 2</b> die Abarbeitung des Vorgangs ist noch nicht abgeschlossen: activity => planned_end_date => date_time	activity.id, activity.activity_type, activity.status, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "status" und "id" oder "activity_type" der Entität "activity"

**Tab. A-15: Transformationsmuster für Gantt-Diagramme (Teil 5)**



## Anhang E: Transformationsmuster für Meilensteintrenddiagramme

### "Präsentierende Elemente" von Meilensteintrenddiagrammen und STEP (AP 214-ARM)

Meilensteindatum				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
M-1	arbeitsauftragsbezogen	work_order => is_controlling => activity => planned_end_date => date_time	work_order.id, work_order.version_id, work_order.work_order_type, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "id", "version_id" und "work_order_type", der Entität "work_order"
M-2	bauteilebezogen	item <= associated_item <= item_version => activity => planned_end_date => date_time	item.id, item_version.id, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "id" der Entität "item" oder unter Berücksichtigung des Datenfeldes "id" der Entität "item_version"
M-3	projekt-/teilprojektbezogen	project => planned_end_date => date_time	project.id, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "id" der Entität "project"
M-4	vorgangsbezogen	activity => planned_end_date => date_time	activity.id, activity.activity_type, date_time.date, date_time.time	Nimm das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "id" oder "activity_type" der Entität "activity"
Zeit				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
Z-1	arbeitsantragsbezogen	work_order => is_controlling => activity => actual_start_date => date_time	work_order.id, work_order.version_id, work_order.work_order_type, date_time.date, date_time.time	Die Inhalte des Datenfeldes "date" bzw. "time" entsprechend den Präsentationsanforderungen verdichten (z.B. täglich, wöchentlich, monatlich, jährlich, etc.)
Z-2	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	item.id, item_version.id, date_time.date, date_time.time	wie Z-1
Z-3	projekt-/teilprojektbezogen	project => actual_start_date => date_time	project.id, date_time.date, date_time.time	wie Z-1
Z-4	vorgangsbezogen	activity => actual_start_date => date_time	activity.id, activity.activity_type, date_time.date, date_time.time	wie Z-1

**Tab. A-16: Transformationsmuster für Meilensteintrenddiagramme**





## Anhang F: Transformationsmuster für die Änderungsquote

### "Präsentierende Elemente" der Änderungsquote und STEP (AP 214-ARM)

Änderungsquote				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
A-1	arbeitsantragsbezogen	work_request	work_request.id, work_request.version_id, work_request.status, work_request.request_type,	Zähle die Datenfelder "version_id" der Entität "work_order" und teile sie durch die Anzahl der Datenfelder "id" der Entität "work_request"
A-2	arbeitsauftragsbezogen	work_order	work_order.id, work_order.version_id, work_order.work_order_type	Zähle die Datenfelder "version_id" der Entität "work_order" und teile sie durch die Anzahl der Datenfelder "id" der Entität "work_order"
A-3	bauteilebezogen	item <= associated_item <= item_version	item.id, item_version.id	Zähle die Datenfelder "id" der Entität "item" und teile sie durch die Anzahl der Datenfelder "id" der Entität "item_version"
Zeit				
Nr.:	Element:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
Z-1	arbeitsantragsbezogen	work_request <= resolved_request <= activity => actual_start_date => date_time	work_request.id, work_request.version_id, work_request.request_type, date_time.date, date_time.time	Das Datenfeld "date" bzw. auch das Datenfeld "time" für die Entität "work_request" entsprechend den Anwenderanforderungen verdichten (z.B. täglich, wöchentlich, monatlich, jährlich, etc.)
Z-2	arbeitsauftragsbezogen	work_order => is_controlling => activity => actual_start_date => date_time	work_order.id, work_order.version_id, work_order.work_order_type, date_time.date, date_time.time	wie Z-1
Z-3	bauteilebezogen	item <= item_version => activity => requestor => date_and_person_organisation => actual_date => date_time	item.id, item_version.id date_time.date, date_time.time	wie Z-1

**Tab. A-17: Transformationsmuster für die Änderungsquote**



## Anhang G: Transformationsmuster für das Gewicht

### "Präsentierende Elemente" des Gewichtes und STEP (AP 214-ARM)

Gewicht				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
G-1	bauteilebezogen	item_version => associated_item => item => material <= described_material <= material_property_ association => associated_property_ value => material_property_value => definition => material_property <= (ABS)property => allowed_unit => unit <= unit_component <= (ABS)value_with_unit => numerical_value	item.id, item_version.id, unit.unit_name, numerical_value.value_ component	Summiere die Inhalte des Datenfeldes "value_component" der Entität "numerical_value" unter Berücksichtigung der Gewichtseinheit, welche sich in dem Datenfeld "unit_name" der Entität "unit" befindet
Zeit				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
Z-1	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	item.id, item_version.id, date_time.date, date_time.time	Das Datenfeld "date" bzw. auch das Datenfeld "time" für die Entität "work_order" entsprechend den Anwenderanforderungen verdichten (z.B. täglich, wöchentlich, monatlich, jährlich, etc.)

Tab. A-18: Transformationsmuster für das Gewicht



## Anhang H: Transformationsmuster für Belastungsdiagramme

### "Präsentierende Elemente" von Belastungsdiagrammen und STEP (AP 214-ARM)

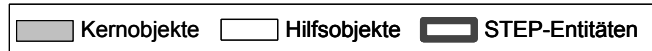
Belastung				
Nr.:	Elemente:	STEP-Entitäten:	STEP-Attribute:	Verdichtungsalgorithmen:
B-1	arbeitsantragsbezogen	work_request	work_request.id, work_request.version_id, work_request.status, work_request.request_type	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "work_request" die Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_request"
B-2	arbeitsantragsbezogen	work_request <= resolved_request <= activity	work_request.id, work_request.version_id, work_request.request_type, activity.status	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity" die Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_request"
B-3	arbeitsantragsbezogen	work_request <= resolved_request <= activity <= item_version <= is_applied_to <= approval => status => approval_status	work_request.id, work_request.version_id, work_request.request_type, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" die Inhalte im Datenfeld "id", "version_id" oder "request_type" der Entität "work_request"
B-4	arbeitsauftragsbezogen	work_order => is_controlling => activity	work_order.id, work_order.version_id, work_order.work_order_type, activity.status	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity" die Inhalte im Datenfeld "id", "version_id" oder "work_order_type" der Entität "work_order"
B-5	arbeitsauftragsbezogen	work_order => is_controlling => activity <= item_version <= approval => status => approval_status	work_order.id, work_order.version_id, work_order.work_order_type, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" die Inhalte im Datenfeld "id", "version_id" oder "work_order_type" der Entität "work_order"
B-6	bauteilebezogen	item <= associated_item <= item_version <= approval => status => approval_status	item.id, item_version.id	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity" die Inhalte im Datenfeld "id" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"
B-7	bauteilebezogen	item <= associated_item <= item_version => activity	item.id bzw. item_version.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity", die Inhalte im Datenfeld "id" der Entität "item" oder die Inhalte im Datenfeld "id" der Entität "item_version"
B-8	projekt-/teilprojektbezogen	project => work_programm => activity	project.id, activity.status	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity" die Inhalte im Datenfeld "id" der Entität "project"
B-9	projekt-/teilprojektbezogen	project => work_programm => activity <= item_version <= is_applied_to <= approval => status => approval_status	project.id, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" die Inhalte im Datenfeld "id" der Entität "project"

Tab. A-19: Transformationsmuster für Belastungsdiagramme (Teil 1)

<b>Belastung</b>				
<b>Nr.:</b>	<b>Elemente:</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>	<b>Verdichtungsalgorithmen:</b>
B-10	vorgangsbezogen	activity	activity.id, activity.activity_type, activity.status	Zähle unter Berücksichtigung des Datenfeldes "status" der Entität "activity" die Inhalte im Datenfeld "id" oder "activity_type"
B-11	vorgangsbezogen	activity <= item_version <= approval =>status => approval_status	activity.id, activity.activity_type, approval_status.status_name	Zähle unter Berücksichtigung des Datenfeldes "status_name" der Entität "approval_status" die Inhalte im Datenfeld "id" oder "activity_type"
<b>Zeit</b>				
<b>Nr.:</b>	<b>Elemente</b>	<b>STEP-Entitäten:</b>	<b>STEP-Attribute:</b>	<b>Verdichtungsalgorithmen:</b>
Z-1	arbeitsantragsbezogen	work_request <= resolved_request <= activity => actual_start_date => date_time	work_request.id, work_request.version_id, work_request.status, work_request.request_type, date_time.date, date_time.time	Die Inhalte des Datenfeldes "date" bzw. "time" entsprechend den Präsentationsanforderungen verdichten (z.B. täglich, wöchentlich, monatlich, jährlich, etc.)
Z-2	arbeitsauftragsbezogen	work_order => is_controlling => activity => actual_start_date => date_time	work_order.id, work_order.version_id, work_order.work_order_type, date_time.date, date_time.time	wie Z-1
Z-3	bauteilebezogen	item <= item_version => activity => actual_start_date => date_time	item.id, item_version.id, date_time.date, date_time.time	wie Z-1
Z-4	projekt-/teilprojektbezogen	project => actual_start_date => date_time	project.id, date_time.date, date_time.time	wie Z-1
Z-5	vorgangsbezogen	activity => actual_start_date => date_time	activity.id, activity.activity_type, date_time.date, date_time.time	wie Z-1

**Tab. A-20: Transformationsmuster für Belastungsdiagramme (Teil 2)**

# Anhang I: Data Dictionary des Konfigurations- und Änderungsmanagementansatzes



## "Data Dictionary" des Konfigurations- und Änderungsmanagementansatzes

Nr.:	Entität:	Attribut:	Beschreibung:	Beispiel:
1	ABS_1		Abstrakte Entität, die selbst nicht instanziiert wird	
1-1		id	Identifikationsnummer des Objektes	WM-98341
1-2		name	Objektname	Meilensteinmuster
1-3		description	Objektbeschreibung	
1-4		status	Reifegrad des Objektes	offen
2	ABS_2		Abstrakte Entität, die selbst nicht instanziiert wird	
2-1		Kategorie	Objekttyp	Berichtszeitpunkt
2-2		Nummer	Identifikationsnummer eines semantischen Abgleichs	Z-3
2-3		Elemente	Namen des Kennlinienelementes	projekt-/teilprojektbezogen
2-4		STEP-Entitäten	Textuelle Beschreibung des STEP-Datenmodellausschnittes, der für den Semantikabgleich mit den Kennlinienelementen benötigt wird	project => planned_end_date => date_time
2-5		STEP-Attribute	Ihre Kombination entspricht der Semantik des Kennlinienelementes	date_time.time, date_time.date
2-6		STEP-Verdichtung	Textuelle Beschreibung der Attributeverdichtung	Das höchste Datenfeld "date" oder "time" der Entität "date_time" unter Berücksichtigung des Datenfeldes "id" der Entität "project"
3	activity		Auszuführende Aktivität	
3-1		activity_type	Aktivitätstyp	Änderung
3-2		id	Identifikationsnummer der Aktivität	A, B, C
3-3		status	Reifegrad der Aktivität	offen
3-4		description	Beschreibung der Aktivität	
4	date_time		Spezifiziert das Datum und die Zeit	
4-1		date	Datum	20.04.2003
4-2		time	Zeit	10:12
5	effectivity		Gültigkeit	
5-1		id	Identifikationsnummer der Gültigkeit	E-52098
6	Entwicklungsstrategie_für_wieder_verwendbare_Transformationsmuster		Beschreibt die Transformationsmusterentwicklung	
7	Leitlinien_zur_Umsetzung_der_Musterinstanzen		Stellt wichtige Grundsätze zur Verwendung der „Weiterverwendbaren_Musterinstanzen“ dar	
8	person		Individuum	
8-1		address	Geschäftsadresse der Person	Bolzmannstr. 15
8-2		person_name	Familienname der Person	Mustermann
8-3		id	Personalnummer	J167674-801
9	Strategie_zur_Anpassung_der_Transformationsmuster		Dokumentiert den Vorgang der Transformationsmusteranpassung	
10	Weiterverwendbare_Musterinstanzen		An eine konkrete PDM-Systemlandschaft angepasste Transformationsmuster	
10-1		PDM-Entitäten	Textuelle Beschreibung des PDM-Datenmodellausschnittes, der für den Semantikabgleich mit STEP benötigt wird	s0WkrOdr => has_WOPPr => WOPPr => has_s7PhyAss => s7PhyAss
10-2		PDM-Attribute	Ihre Kombination entspricht der Semantik der STEP-Attribute	WOPPr.d0DesignTargetDate, s7PhyAss.ProjectName
10-3		PDM-Verdichtung	Textuelle Beschreibung der Attributeverdichtung	Das höchste Datenfeld "d0DesignTargetDate" der Klasse "WOPPr" unter Berücksichtigung des Datenfeldes "ProjectName" der Klasse "s7PhyAss"

Tab. A-21: Data Dictionary (Teil 1)

Kernobjekte   
  Hilfsobjekte   
  STEP-Entitäten

Nr.:	Entität:	Attribut:	Beschreibung:	Beispiel:
11	Wieder_verwendbare_Transformationsmuster		Beschreiben die semantische Zuordnung zwischen STEP und den Überwachungskennlinienelementen	
12	work_order		Arbeitsauftrag	
12-1		id	Identifikationsnummer des Arbeitsantrages	WO-98475
12-2		version_id	Versionsnummer des Arbeitsantrages	in Arbeit
12-3		work_order_type	Arbeitsantragstyp	Änderung
12-4		description	Beschreibung des Arbeitsantrages	
13	work_request		Arbeitsauftrag	
13-1		id	Identifikationsnummer des Arbeitsauftrages	WR-29357
13-2		version_id	Versionsnummer des Arbeitsauftrages	A, B, C
13-3		status	Reifegrad des Arbeitsauftrages	abgeschlossen
13-4		request_type	Arbeitsauftragstyp	Änderung
13-5		description	Beschreibung des Arbeitsauftrages	

**Tab. A-22: Data Dictionary (Teil 2)**



# Literatur

- [AbrGer97] Abramovici, M. Gerhard, D. Engineering Daten Management: Anspruch, Wirklichkeit und Zukunftsperspektiven, In: Industrie Management spezial, GITO-Verlag, 1997, S. 11-15
- [AltNad02] Althoff, K. Nadj, I. Logistikcontrolling im Intranet - Ein Erfahrungsbericht aus der Automobilindustrie, In: Industrie Management, GITO-Verlag, 4/2002, S. 55-58
- [Ande01] Anderl, R. Produktdatentechnologie B: Produktdatenmanagement, Skriptum, 2001
- [Ande03] Anderl, R. Produktdatentechnologie A: CAD-Systeme und CAx-Prozessketten, Skriptum, 2003
- [AndFröKlei00] Anderl, R. Fröhlich, A. Kleiner, S. Integration of Design and computational Engineering for mechatronic products based on Product Data Technology, proceedings, 7<sup>th</sup> European Concurrent Engineering Conference, Leicester, United Kingdom, April 17<sup>th</sup>-19<sup>th</sup>, 2000, S. 107-112
- [AndTri99] Anderl, R. Trippner, D. STEP, B.G. Teubner Stuttgart, 1999
- [Bend01] Bender, B. Zielorientiertes Kooperationsmanagement in der Produktentwicklung, Dissertation, TU-München, 2001

- [Bend03] Bender, K. Automatisierungstechnik im Maschinenwesen, Skriptum, TU-München, 2003
- [BendEtAl01] Bender, K. Jack, P. Koç, A. Péter, I. Megyeri, G. Qualitätssicherung eingebetteter Software: Methoden und Best-Practices, Utz-Verlag, 2001
- [BenKarFis01] Bender, K. Karcher, A. Fischer, F. KEIM - Kontinuierliches Engineering-Informationsmanagement im Product Life Cycle, In: Industrie Management, GITO-Verlag, 6/2001, S. 58-62
- [Bohn94] Bohn, R. E. Measuring and Managing Technological Knowledge, Sloan Management Review, Fall 1994, S. 61-65
- [Bürg96] Bürgel, H. D. et. al.: F&E-Management, Vahlen Verlag, München 1996
- [BulBröWag99] Bullinger, H.-J. Bröcker, A. Wagner, F. Die verteilte Produktentwicklung im Zusammenhang von DMU, VR und EDMS; In: VDI-Gesellschaft Entwicklung, Konstruktion, Vertrieb (Hg): Informationsverarbeitung in der Konstruktion '99. Beschleunigung der Produktentwicklung durch EDM/PDM- und Feature-Technologie, VDI Verlag, Düsseldorf, 1999, S. 3-24
- [Burg02] Burghardt, M. Einführung in Projektmanagement - Definition, Planung, Kontrolle, Abschluss, Siemens, 2002

- [ChaSin04] Chamoni, P. Sinz, E. Data Warehousing und Data Mining - Anwendungen, Technologien, Einsatzerfahrungen, In: Wirtschaftsinformatik, Vieweg Verlag, 1/2004, S. 1
- [DIN 19246] DIN 19246, Messen Steuern Regeln - Abwicklung von Projekten - Begriffe, Beuth Verlag Berlin, 1991
- [DIN 69901] DIN 69901, Projektwirtschaft - Projektmanagement - Begriffe, Beuth Verlag Berlin, 1987
- [DIN 69904] DIN 69904, Projektwirtschaft - Projektmanagementsysteme - Elemente und Strukturen, Entwurf, Beuth Verlag Berlin, 1999
- [DIN EN 13290-1] DIN EN 13290-1, Luft- und Raumfahrt - Raumfahrtmanagement - Allgemeine Anforderungen, Teil 1: Grundsätze und Verfahrensweise, Beuth Verlag Berlin, 1999
- [DIN EN 13290-7] DIN EN 13290-7, Luft- und Raumfahrt - Raumfahrtmanagement - Allgemeine Anforderungen, Teil 7: Kosten- und Zeitplanmanagement, Beuth Verlag Berlin, 2001
- [DIN EN 82045-1] DIN EN 82045-1, Dokumentenmanagement, Teil 1: Prinzipien und Methoden, Beuth Verlag Berlin, 2002
- [DIN EN 82045-2] DIN EN 82045-2, Dokumenten Management, Teil 2: Metadaten und Informationsreferenzmodell, Beuth Verlag Berlin, 2003

- [DobMus00] Doblies, M. Muschiol, M. Integration von EDM/PDM-Systemen über Organisationsgrenzen hinweg, In: Produkte entwickeln im realen Umfeld, VDI Berichte, Nr. 1569, 2000, S. 55-65
- [DomBot03] Dombrowski, U. Bothe, T. Ersatzteilversorgung nach Ende der Serienfertigung, In: Industrie Management, GITO-Verlag, 4/2003, S. 19-27
- [DornEtAl90] Dorninger, C. Janschek, O. Olearczick, E. Röhrenbacher, H. PPS Produktionsplanung und -steuerung, Konzepte, Methoden und Kritik, Verlag Carl Ueberreuter, Wien, 1990
- [Ehr195] Ehrlenspiel, K.: Integrierte Produktentwicklung, München Wien, Springer, 1995
- [ElsDirCor02] Elsweiler, B. Dirlenbach, H. Corsten, A. Systematische Integration von Managementkonzepten, In: Industrie Management, GITO-Verlag, 4/2002, S: 18-21
- [EveEtAl98] Eversheim, W. Schuth, S, Bremer, C. Molina, A.: Globale virtuelle Unternehmen, In: ZWF 3/1998, S. 62-64
- [FisZscTei03] Fischer, M. Zschorn, L. Teich, T. Das Phasenmodell innerhalb des EVCM-Betreiberkonzepts für hierarchielose Produktionsnetze, In: Industrie Management, GITO-Verlag, 4/2003, S. 60-63

- [Gare91] Gareis, R. Projektmanagement im Maschinen und Anlagenbau, Manzsche Verlags- und Universitätsbuchhandlung, 1991
- [GauEbbKall01] Gausemeier, J. Ebbesmeyer, P. Kallmeyer, F. Produktinnovation - Strategische Planung und Entwicklung der Produkte von morgen, Carl Hanser Verlag, München, Wien, 2001
- [Geig00] Geiger, O. Kennzahlenorientiertes Entwicklungscontrolling - Ein ganzheitliches, kennzahlenorientiertes Planungs-, Steuerungs- und Kontrollinstrument zur Analyse des Entwicklungsbereichs industrieller Unternehmen, Dissertation, TU-Braunschweig, Shaker Verlag, 2000
- [Gerh00] Gerhardt, D. Erweiterung der PDM-Technologie zur Unterstützung verteilter kooperativer Produktentwicklungsprozesse, Dissertation, Ruhr-Universität Bochum, Shaker Verlag, 2000
- [GerSch96] Gerhardt, A. Schmied, H. Externes Simultanes Engineering - Der neue Dialog zwischen Kunde und Lieferant, Springer, 1996
- [Glan02] Glander, M. Offene IT-Architektur für verteilte autonome Engineering-Prozesse, TU-München, Utz Verlag München, 2002

- [GrabEtAl00] Grabowski, H. et. all. Zugriff auf Erfahrungswissen durch die automatische Klassifikation von Produkten, In: Produkte entwickeln im realen Umfeld, VDI Berichte, Nr. 1569, 2000, S. 9-23
- [GraBar03] Grauer, M. Barth, T. Kollaboratives virtuelles Prototyping, Industrie Management, 5/2003, S. 29-36
- [GroGen00] Grothe, M. Gentsch, P. Business Intelligence - Aus Informationen Wettbewerbsvorteile gewinnen, Addison-Wesley, 2000
- [Gümb04] Gümbel, C. Neue Wege, In: CADplus - Business + Engineering, 1/2004, S. 40
- [Habe00] Haberstock, P. Executive Information Systems und Groupware im Controlling, Dissertation, TU-Paderborn, Deutscher Universitäts-Verlag, Wiesbaden, 2000
- [Hack89] Hackstein, R. Produktionsplanung und -steuerung (PPS) - Ein Handbuch für die Betriebspraxis, VDI-Verlag, Düsseldorf, 1989
- [Hill02] Hiller, M. Multiprojektmanagement - Konzept zur Gestaltung, Regelung und Visualisierung einer Projektlandschaft, Dissertation, Universität Kaiserslautern, 2002
- [Höfe99] Höfener, C. Methode zur Bewertung des strategischen Nutzens von integriertem Produktdatenmanagement (PDM), Dissertation, Shaker Verlag, 1999

- [Iser99] Isermann, R. Mechatronische Systeme, Berlin, Springer, 1999
- [ISO 10006] ISO 10006, Quality management - Guidelines to quality in project Management, Beuth Verlag Berlin, 1997
- [ISO 10303-1] ISO 10303-1, Produktdatendarstellung und -austausch, Teil 1: Überblick und grundlegende Prinzipien, 1995
- [iViP02] Krause F.-L. Tang, T. Ahle, U. (Hrsg.) Integrierte Virtuelle Produktentstehung - Fortschrittsbericht 2, 2002, S. 7
- [Jenn01] Jenne, F. PDM-basiertes Entwicklungsmonitoring - Ein Beitrag zur Planung und Steuerung von Entwicklungsprozessen, Dissertation, Universität Karlsruhe, 2001
- [JohSch99] Johnson, G. Scholes, K. Exploring Corporate Strategy - 5th Edition, Prentice Hall, 1999
- [Jung99] Jungkunz, R. Knowledge Management - A concept to a data-, information- and knowledge system, Master Dissertation, Liverpool John Moores University, 1999
- [JunKarSch02] Jungkunz, R. Karcher, A. Schnitzler, A. smartPDM - dynamic information support within distributed development processes, Proceedings of the 9th European Concurrent Engineering Conference, Modena, Italy, April 15-17, 2002, S. 47-51

- [JunKarVil02] Jungkunz, R. Karcher, A. Vilsmeier, J. The Eurofighter Programme High Level Interface Definition (HLID) - A STEP based reference model for product data exchange in the aerospace industry, In: ProduktDaten Journal 2/2002, ProSTEP iViP Verein zur Förderung internationaler Produktdatennormen e.V., Darmstadt, 2002
- [JunWir03] Jungkunz, R. Wirtz, J. Projekt Data Warehousing basierend auf PDM, ProduktDaten Journal 1/2003, ProSTEP iViP Verein zur Förderung internationaler Produktdatennormen e.V., Darmstadt, 2003
- [JungWirt03] Jungkunz, R. Wirtz, J. Effizientes Projektmanagement auf Basis eines integrierten Produktmodells, In: Industrie Management, 3/2003, GITO-Verlag, S. 75-78
- [JunWirKar04] Jungkunz, R. Wirtz, J. Karcher, A. PDM-basiertes Projektcontrolling in Produktentwicklungsprojekten der Flugzeugindustrie, Tagungsband des 21. Internationalen Deutschen Projektmanagement Forums, Nürnberg, Oktober 4-7, 2004, S. 277-287
- [Karc02] Karcher, A. PDM- und Engineering Informationssysteme, Skriptum, TU-München, 2002
- [Koca03] Koc, A. Entscheidungsunterstützung zur Planung der Software-Qualitätssicherung in mechatronischen Produkten, Dissertation, TU-München, 2003



- [KooJaeOff01] Koop, H-J. Jaeckel, K. van Offern, A. Erfolgsfaktor Content Management, Vieweg, 2001
- [KraMat97] Krause, F.-L. Mattes, W. Informationstechnik für die Produktentwicklung. In: Grabowski, H. Geiger, K. (Hrsg.): Neue Wege zur Produktentwicklung. Stuttgart, Berlin, Bonn, 1997
- [Kurb98] Kurbel, K. Produktionsplanung und -steuerung, Methodische Grundlagen zu PPS-Systemen, Oldenburg Verlag München Wien, 1998
- [Lang00] Langenberg, L. Firmenspezifische Wissensportale für die Produktentwicklung, Dissertation, Fakultät für Maschinenbau der Ruhr-Universität Bochum, 2000
- [Lind02] Lindemann, U. Entwicklungsmanagement, Skriptum, TU-München, 2002
- [Lind03] Lindemann, U. Methoden der Produktentwicklung, Skriptum, TU-München, 2003
- [LinRei98] Lindemann, U. Reichwald, R. (Hrsg.) Integriertes Änderungsmanagement, Springer, 1998
- [Litk93] Litke, H.-D. Projektmanagement - Methoden, Techniken, Verhaltensweisen, Hanser Verlag, München Wien, 1993

- [LucEve01] Luczak, H. Eversheim, W. Produktionsplanung und -steuerung - Grundlagen, Gestaltung und Konzepte, Springer Verlag, 2001
- [Luka03] Lukas, U. Kooperation in der virtuellen Produktentwicklung durch föderierte PDM-Systeme, In: Industrie Management, GITO-Verlag, 5/2003, S. 33-36
- [Lust99] Lusti, M. Data Warehouse und Data Mining - Eine Einführung in entscheidungsunterstützende Systeme, Springer Verlag, 1999
- [Marx00] Marx, O. PPS-Reorganisation: Sanierung oder Ablösung - empirische Analyse, Erfolgsbeurteilung und Normstrategien, Dissertation, TU-München, 2000
- [Maut02] Mauterer, H. Der Nutzen von ERP-Systemen, Dissertation, Technische Universität Berlin, 2002
- [MayChr01] May, G. Chrobok, R. Priorisierung des unternehmerischen Projektportfolios, In: Zeitschrift für Organisation, 2001, Jg. 70, Heft 2, S. 108-114
- [Mert01] Mertens, P. Information - die Ressource der Zukunft, In: Frankfurter Allgemeine Zeitung, Montag, 20. August 2001, Nr. 192, S. 25
- [MucNic95] Much, D. Nicolai, H. PPS-Lexikon, Cornelsen Verlag, Berlin, 1995

- [Müll95] Müller, K. Management für Ingenieure - Grundlagen, Techniken, Instrumente, Springer-Verlag, 1995
- [Mult00] Multhaupt, M. Data Mining und Text Mining im strategischen Controlling, Dissertation, TU-Braunschweig, Shaker Verlag, Aachen, 2000
- [Murg03] Murgai, M. Dokumenten- und Workflowmanagement, In: eDM Report, Dressler Verlag, Heidelberg, 1/2003, S. 49-52
- [NCO00] NATO CALS Office, NATO Product Data Model, 2000
- [Phil00] Philipp, M. Nutzung von Produktmodellen als institutionelles Gedächtnis - Ein Beitrag zur Methodik in der Produktentwicklung, Dissertation, TU-Darmstadt, 2000
- [Phil02] Philippson, C. Koordination einer standortbezogenen verteilten Produktionsplanung und -steuerung auf der Basis von Standard PPS-Systemen, Dissertation, RWTH Aachen, 2002
- [PraSüd97] Pradel, M. Südmeyer, V. Portfolioplanung als zentraler Bestandteil des Multiprojektcontrollings, In: Zeitschrift für Planung, 1997, Heft 3, S. 291-311, Heidelberg, Physica Verlag
- [PROGRESS02] PROGRESS, Entwicklungsprozesse in der Elektronikindustrie, Abschlussbericht, 2002

- [Rein02] Reinhart, G. Projektmanagement für Ingenieure, Skriptum, TU-München, 2002
- [ReiLinHei96] Reinhart, G. Lindemann, U. Heinzl, J. Qualitätsmanagement, Springer, 1996
- [Ried00] Riedel, D. Standortverteiltes Änderungsmanagement - Eine explorative Analyse über die Gestaltung standortübergreifender Produktänderungen, Dissertation, TU-München, 2000
- [Rind94] Rinza, P. Projektmanagement - Planung, Überwachung und Steuerung von technischen und nichttechnischen Vorhaben, VDI Verlag, Düsseldorf, 1994
- [Rinz94] Rinza, P. Projektmanagement - Planung, Überwachung und Steuerung von technischen und nichttechnischen Vorhaben, VDI Verlag, Düsseldorf, 1994
- [SchBan04] Schwalm, S. Bange, C. Einsatzpotentiale von XML in Business Intelligence Systemen, In: Wirtschaftsinformatik, Vieweg Verlag, 1/2004, S. 1
- [SchBanMer99] Schinzer, H. Bange, C. Mertens, H. Data Warehouse und Data Mining - Marktführende Produkte im Vergleich, Verlag Franz Vahlen München, 1999
- [Sche92] Scheer, A. et. al. Simultane Produktentwicklung, Forschungsbericht 4, Schoder Druck St. Gallen, 1992

- [Schi02] Schichtel, M. Produktdatenmodellierung in der Praxis, Carl Hanser Verlag München Wien, 2002
- [Schie02] Schiegg, H. Qualitätsorientierte Prozessgestaltung der Integrierten Produktentwicklung, Fortschritt-Berichte VDI, Reihe 1, Nr. 361, 2002
- [Schm96] Schmalzl, B. Ein Projektleitsystem im integrierten Produkterstellungsprozess, Dissertation, TU-München, VDI Fortschrittberichte, Nr. 202, VDI Verlag, 1996
- [Schn94] Schneider, U. Mitarbeitertrainings unter der Lupe, Gabler, 1994
- [Schn00] Schneider, D. Unternehmensführung und strategisches Controlling - Überlegene Instrumente und Methoden, Carl Hanser Verlag, 2000
- [Scho00] Schollian, T. Projektportfolio-Management, In: Projektmanagement, 2000, Heft 4, S. 4-16, Köln, Tüv Verlag
- [Schö99] Schöttner, J. Produktdatenmanagement in der Fertigungsindustrie - Prinzip, Konzepte, Strategien, Carl Hanser Verlag, 1999
- [Schw04] Schwarz, H. Data Warehouse, Data Mining und OLAP-Technologien, Skriptum, Fakultät für Informatik, Elektrotechnik und Informationstechnik, Uni Stuttgart, 2004

- [SFB 374] SFB 374, Entwicklung und Erprobung innovativer Produkte - Rapid Prototyping, Uni Stuttgart, 2000, S. 14
- [Soli00] Solingen, R. Product Focused Software Process Improvement: SPI in the embedded software domain, Dissertation, Technische Universität Eindhoven, 2000
- [Stoc89] Stockbauer, H. F&E Controlling, Dissertation, Wirtschaftsuniversität Wien, 1989
- [TFB29-T2] TFB 29-T2, Koordination und Controlling verteilter Produktentwicklungsprozesse, Abschlussbericht, 2004
- [ThoEsc03] Thoben, K.-D. Eschenbächer, J. Die Erweiterung des Produktbegriffs - Konzepte und Praxisbeispiele, In: Industrie Management, GITO-Verlag, 4/2003, S. 48-51
- [TreKinHei02] Treppa, A. Kinkel, S. Heimann, K. Produktcontrolling - Durch computergestützte Informations- und Wissensverarbeitung zu effektiven/effizienten Produktkennzahlen, In: Industrie Management, GITO-Verlag, 4/2002, S. 43-46
- [Trip02] Trippner, D. Vorgehensmodell zum Management von Produktdaten in komplexen und dynamischen Produktentwicklungsprozessen, Dissertation, Universität Karlsruhe, 2002

- [VDI 2219] VDI-Richtlinie 2219, Informationsverarbeitung in der Produktentwicklung - Einführung und Wirtschaftlichkeit von EDM/PDM-Systemen, VDI-Gesellschaft Entwicklung, Konstruktion und Vertrieb, Düsseldorf, 2002
- [VDI 2221] VDI-Richtlinie 2221, Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte, VDI-Gesellschaft Entwicklung, Konstruktion und Vertrieb, Düsseldorf, 1993
- [Vers00] Versteegen, G. (Hrsg.) Das V-Modell in der Praxis - Grundlagen, Erfahrungen, Werkzeuge, dpunkt verlag, 2000
- [Vers02] Versteegen, G. (Hrsg.) Managementtechnologien - Konvergenz von Knowledge-, Dokumenten- und Contentmanagement, Springer Verlag, 2002
- [Vils04] Vilsmeier, J. PDM-Systeme in der industriellen Praxis, Skriptum, TU-München, 2004
- [VilWir01] Vilsmeier, J. Wirtz, J. Datenaustausch im Eurofighter Projekt auf Basis des PDM Schemas, VDMA Conference, 2001
- [Warn01] Warnecke, G. Produktionstechnik in Lebenszyklen, In: Wt Werkstatttechnik, 2001, Jg. 91, S. 470

- [Wehn99] Wehner, T. Wissensmanagement als sozialer Prozess: Die arbeits- und organisationspsychologische Position, In: Proceedings of the science forum „Wissensmanagement und schnelle Produktentwicklung, Sonderforschungsbereich 374, Stuttgart, 1999
- [Wick95] Wicke, J.-M. Controlling von Forschungs- und Innovationsprojekten, Shaker Verlag, 1995
- [Wild98] Wildemann, H. (Hrsg.) Strategien zur Marktführerschaft - die Kundenforderungen von morgen gestalten, Frankfurter Allgemeine Buch, 1998
- [Wild01] Wildemann, H. Reduzierung der Entwicklungszeiten in der Elektronikindustrie, In: Automotive Electronics, 9/2001, S. 28-31
- [Wild03] Wildemann, H. Steuerung von F&E-Projekten und Realisierung einer planmäßigen Produktentwicklung, In: Industrie Management GITO Verlag, 5/2003, S. 37-40
- [Wirt01] Wirtz, J. Ein Referenzmodell zur integrationsgerechten Konzeption von Produktdatenmanagement, Dissertation, TU-München, Utz Verlag München, 2001
- [WirJun03] Wirtz, J. Jungkunz, R. LOTAR - Long Term Archiving and Retrieval of Product Data within Aerospace, Proceedings of ProSTEP iViP Science Days 2003, Dresden, Germany, October 8-9, 2003, S.186-194



- 
- [Xul03] Xu, Linxiang. Wiederverwendbare Modelle zur Maschinensimulation für den Steuerungstest, Dissertation, TU-München, Utz Verlag München, 2003
- [Zwic99] Zwicker, E. Unterstützung der unternehmensübergreifenden Produktentwicklung durch den Einsatz moderner Informationstechnologien, Dissertation, Eidgenössische Technische Hochschule Zürich, VDI Verlag, Düsseldorf, 1999