

# Ganzheitliche Unterstützung für die simultane und virtuelle Produktentwicklung

Dr. rer. pol. Veit Köppen <sup>1)</sup>, Dipl.-Inf. Stephan Vornholt <sup>2)</sup>, Dipl.-Inf. Ingolf Geist <sup>3)</sup>,  
Prof. Dr. rer. nat. habil. Gunter Saake <sup>4)</sup>

<sup>1)</sup> Center for Digital Engineering, Otto-von-Guericke-Universität Magdeburg, Deutschland, vkoeppen@ovgu.de,  
+49 (0) 391 67 19350

<sup>2)</sup> Fakultät für Informatik, Otto-von-Guericke-Universität Magdeburg, Deutschland, vornholt@iti.cs.uni-magdeburg.de

<sup>3)</sup> Fakultät für Informatik, Otto-von-Guericke-Universität Magdeburg, Deutschland, ingolf.geist@ovgu.de

<sup>4)</sup> Fakultät für Informatik, Otto-von-Guericke-Universität Magdeburg, Deutschland, saake@iti.cs.uni-magdeburg.de

## Zusammenfassung

Die durchgehende Nutzung von computergestützten virtuellen Produktmodellen in der Produktentwicklung ermöglicht kürzere Entwicklungszeiten. Dabei werden komplexe Produkte in handhabbare Subsysteme zerlegt und in einem gemeinschaftlichen Prozess von einzelnen Fachdisziplinen virtuell geplant, modelliert und simuliert. Die Prozesse der virtuellen Produktentwicklung nutzen unterschiedliche Werkzeuge zur digitalen Konstruktion und Analyse des Verhaltens der Produkte. Jedes Werkzeug benötigt spezielle Daten und verwaltet eigene Datenstrukturen. Ein integriertes Produktdatenmodell, welches bestehende Datenstrukturen in einem gemeinsamen Komponentenmodell abbildet, ermöglicht eine gleichzeitige Datenbearbeitung mehrerer Fachdisziplinen. Für diese simultane Produktentwicklung ist es notwendig, flexible Prozesse nachvollziehbar zu verwalten. In dieser Arbeit werden unterschiedliche Datenmodelle zur Unterstützung der virtuellen Produktentwicklung vorgestellt. Um den Produktentwicklungsprozess leichter nachvollziehbar zu gestalten, werden Metadaten genutzt. Sie eignen sich, anfallende und zusätzlich durch die Entwickler generierte Informationen zu Produktmodelldaten abzubilden und einander zuzuordnen. Dazu werden einzelne Entwicklungsschritte des Prozesses und deren Ergebnisse den Produktmodellen zugeordnet sowie mit zusätzlichen Informationen angereichert. Die verschiedenen Lösungsansätze im Bereich der integrierten Produktmodelle und Metadatenverwaltung werden in dieser Arbeit betrachtet, bewertet und integriert zusammengefasst. Das Ergebnis ist ein Softwaresystem, welches flexible, simultane Workflows unterstützt, indem es heterogene Daten komponentenorientiert integriert und Produktdatenmodelle mit Metadaten versieht. Als Ergebnis wird ein webgebundener Prototyp auf der Basis einer gemeinsamen Datenbank vorgestellt.

Schlüsselwörter: Virtuelle Produktentwicklung, Datenverwaltung, Datenaustauschprozess, Metadaten

## 1. Einleitung

Die konsequente und durchgehende Nutzung computergestützter Methoden in den Phasen *Planung* und *Entwicklung* des Produktlebenszyklus [1] ermöglicht eine Verkürzung von Entwicklungszeiten komplexer Produkte, beispielsweise in der Luftfahrts- und Automobilindustrie. Die methodische Überführung eines Konstruktionsprozesses in einen rechnerinternen Ablauf, bei gleichzeitiger Darstellungsmöglichkeit des realen Verhaltens der zu entwickelnden Objekte, wird als virtuelle Produktentwicklung (VPE) bezeichnet [2]. Aktivitäten in der VPE nutzen Systeme zur Konstruktion sowie verschiedene Simulationswerkzeuge zur frühzeitigen Analyse des potentiellen Produktverhaltens. Für eine integrierte und effiziente Softwarelösung zur Unterstützung der VPE ist es notwendig, die Prozesse abzubilden und zuzuordnen. Dazu werden Produktdaten mit semantischen Daten angereichert, um das Re-Engineering zu vereinfachen. Einzelne Entwicklungsschritte müssen flexibel in den Prozess eingegliedert und Teilschritte sowie Ergebnisse einander zugeordnet werden. VPE ist durch einzelne Phasen gekennzeichnet, in denen die computergestützte, virtuelle Entwicklung von Produkten jeweils unterschiedliche Zielvorgaben hat (bspw. Simulation, Visualisierung und Rapid Prototyping). Zentrale Domänen sind die Konstruktion, mechanische und elektrische Modellierung sowie Mehrkörper-, mechatronische oder Finite-Element-Simulationen. Zur Bearbeitung einer domänenspezifischen Aufgabe benötigt der jeweils nächste Schritt als Grundlage die Daten der vorhergehenden Lösungen. Obwohl die Aufgabe des Informationsaustausches durch Schnittstellen zur Datenübertragung in virtuellen Umgebungen vereinfacht wird, treten typische Schnittstellenherausforderungen auf. Dazu zählen insbesondere die Vermeidung von Informations- und Datenverlust, aber auch die Notwendigkeit eines Informationsaustausches und eine stetige

Nachvollziehbarkeit auf Daten- und Modellebene. Eine Beschreibung der Abhängigkeit von vorhergehenden Versionen und Varianten ist ebenso notwendig wie die Berücksichtigung von verworfenen Modellen.

Eine weitere Herausforderung stellt die Datenkonsistenz dar. Im Entwicklungsprozess werden Daten und Parameter der einzelnen Modelle häufig auch in den späteren Phasen geändert oder verfeinert. Damit ist die gespeicherte Version der vorangegangenen Entwicklungsschritte bereits veraltet, wenn Ergebnisse oder Änderungswünsche eintreffen. Diese Inkonsistenz auf Datenebene erschwert Veränderungen und die Erstellung neuer Varianten.

Zur Unterstützung des Datenmanagements in VPE werden daher nach einer kurzen Einführung in den Stand der Technik (Abschnitt 2) in diesem Beitrag Konzepte vorgestellt, die Referenzlösungen für die VPE darstellen, um Ingenieuren die simultane Bearbeitung von virtuellen Modellen zu erleichtern. Die Anforderungen sind hierbei: Modelldaten müssen in eine gemeinsame Struktur integriert, mit zusätzlichen Informationen angereichert und Teilkomponenten verschiedener Modelle einander zugeordnet sowie durch eine geeignete Softwareumgebung unterstützt werden. Dazu wird ein ganzheitliches Konzept vorgestellt, welches die domänenspezifischen Aufgaben verknüpft (Abschnitt 3). Produktmodelle und ihre Parametrisierungen werden in einem komponentenorientierten integrierten Produktmodell genutzt, um Datenkonsistenz zu gewährleisten und domänenübergreifende Parameter und Teilmodelle definieren zu können (Abschnitt 3.1). Weiterhin ermöglicht eine Prozessschicht Vor- und Nachbereitung der Daten und eine Aufschlüsselung der Abhängigkeiten anhand eines flexiblen Prozessmodells (Abschnitt 3.2). Anwender verschiedener ingenieurwissenschaftlicher und angegliederter Disziplinen können die (semi-)automatische Zuordnung der Teilmodelle grafisch unterstützt nutzen. Damit werden eine Vielzahl an Erweiterungen für Workflowmanagement- (WFM) und Produktdatenmanagementsysteme (PDM) zur Unterstützung der VPE bereitgestellt (Abschnitt 4). Diese Arbeit schließt mit einer Zusammenfassung der erzielten Ergebnisse und einem Ausblick über zukünftige Entwicklungen und Möglichkeiten (Abschnitt 5).

## **2. Virtual Engineering – Begriff und Anwendungen**

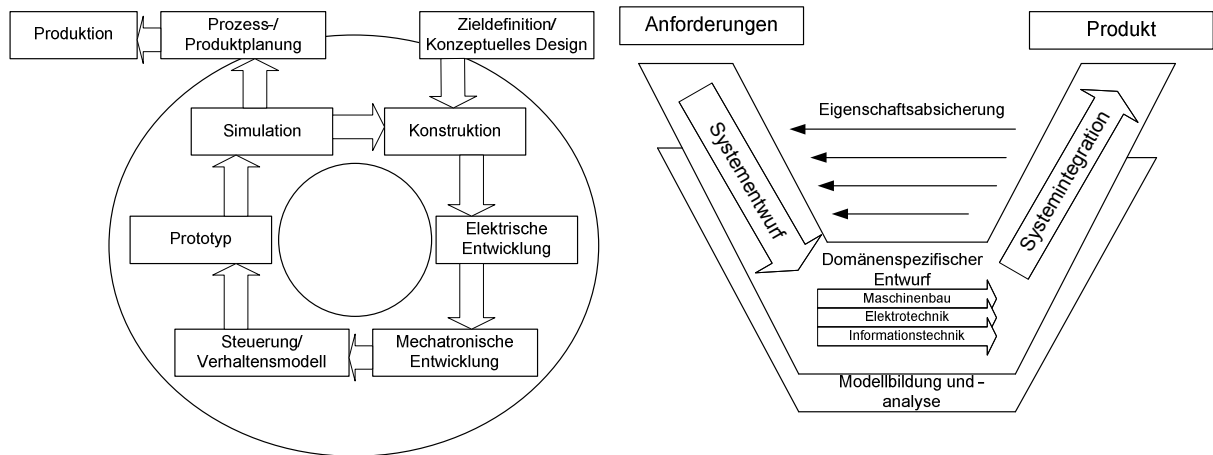
Der Begriff virtuelle Produktentwicklung wird u.a. definiert als “die methodische Überführung eines Konstruktionsprozesses in einen rechnerinternen Ablauf mit gleichzeitiger Darstellungsmöglichkeit des wirklichen Verhaltens der zu entwickelnden Objekte wird Virtualisierung genannt“ [3]. Wohingegen Abramovici die "Virtuelle Produktentwicklung" als durchgehend rechnerunterstützte Produktentwicklung unter intensiver Anwendung von Simulations- und Verifikationstechniken auf der Basis digitaler, realitätsnaher Modelle definiert [4].

Gemein ist den Definitionen, dass die VPE zum Test von realen Gegenständen rechnerinterne und für das Verhalten geeignet abstrahierte Produktmodelle geschaffen werden, welche sich für mehrfache Tests unter verschiedenen Bedingungen (Parametern) eignen. Ein parametrisiertes Produktmodell wird auch als Instanz oder Variante bezeichnet.

### **2.1 Virtueller Entwicklungsprozess**

Methoden und Werkzeuge des VPE ermöglichen den Aufbau einer durchgängig digitalen Beschreibung, Modellierung, Simulation und Optimierung von Produkten über den gesamten Produktlebenszyklus. Dabei werden insbesondere die frühen Phasen der Konstruktion und Analyse/Simulation durch computergenerierte Modelle unterstützt (siehe Abbildung 1a). Da die einzelnen Phasen der Entwicklung von Produkten unterschiedliche Zielvorgaben besitzen, gibt es verschiedene vom Produktlebenszyklus abgeleitete Vorgaben an die unterstützenden Systeme.

Die Entwicklung von Produkten ist ein iterativer Prozess, der mehrere Domänen umfasst, z.B. Maschinenbau, Elektrotechnik und Informationstechnik (siehe Abbildung 1b). Üblicherweise werden die Prozessschritte aufeinanderfolgend oder in manchen Fällen parallel ausgeführt. So wird für eine Mehrkörpersimulation zuerst ein mechanisches Modell erzeugt. Werden zusätzlich elektrische Antriebe definiert, können beide zusammen in mechatronischen Simulationen eingesetzt werden. Zur Bearbeitung der eigenen Aufgabe benötigt der jeweils nächste Schritt als Grundlage die Informationen der vorhergehenden Lösungen. Auf der einen Seite werden viele verschiedene Modelle erstellt, weitergeleitet, geändert und wieder verworfen. Auf der anderen Seite muss der Entwicklungsprozess nachvollziehbar gestaltet sein, damit eventuelle Fehler oder Änderungen am Produktmodell sowie die Gründe dafür erkannt werden können. Eine Beschreibung der Abhängigkeiten von vorhergehenden Versionen, Varianten und Domänen ist daher notwendig. Ebenso müssen auch fehlerhafte Modelle, die bereits in den Prozess eingeflossen sind, berücksichtigt werden. Hier ist eine Zuordnung von Produktmodellen und den entsprechenden Nachfolgern nötig. Beispielsweise muss ein Produktmodell, das in einer ersten Simulation nicht die gewünschten Ergebnisse erzielt hat, grundlegenden Änderungen im mechanischen Modell unterzogen werden. Alle Schritte der vorhergehenden Prozesse müssen sich wiederholen.



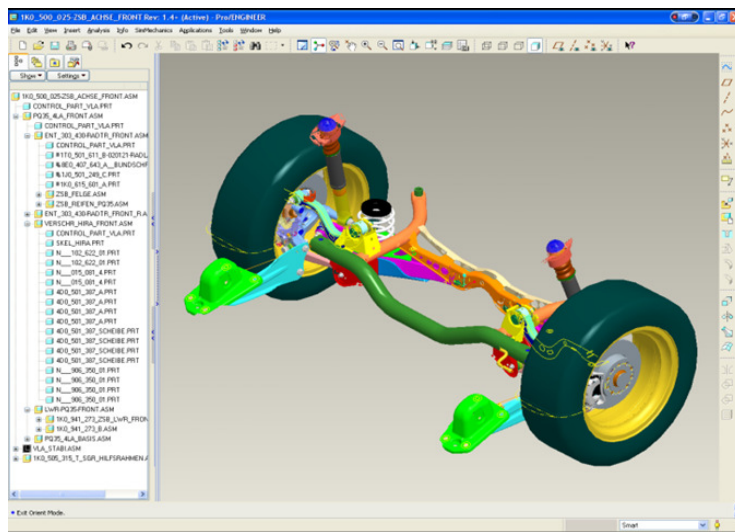
a) Einordnung in den Produktlebenszyklus

b) Vorgehensmodell nach VDI Richtlinie 2206 [5]

**Abbildung 1: Einordnung in den Produktlebenszyklus und Vorgehensmodell**

## 2.2 Virtuelle Produktdaten

Eine häufig genutzte Datenrepräsentation in der virtuellen Produktentwicklung stellen Visualisierungen dar. Dabei ist die virtuelle Realität (VR) mit dem Ziel, das Objekt möglichst realitätsnah abzubilden, die bekannteste Form der Visualisierung. Auch andere Simulationen wie thermischer Einfluss oder Spannungen können mittels VR visualisiert werden. Dabei bilden die Rohdaten die Grundlage für den Informationsaustausch und die Algorithmen, welche die Daten aufbereiten. Abbildung 2a zeigt, dass das visuelle Bild im Vordergrund steht. Links ist die Struktur der Daten zu sehen, welche die einzelnen Teile (Komponenten) sowie ihre Verbindungen abbilden. Die Daten liegen entweder in einem nativen Format oder einen Austauschformat (neutral oder systemgebunden) vor. Abbildung 2b zeigt einen Ausschnitt einer XML basierten Modellbeschreibung.



a) Datenstruktur & Visualisierung

```
<property name="originalName" value="ARM3.2"/>
<property name="mass" value="20.3494871671806" />
<property name="massUnits" value="kg" />
<property name="inertia"
value="0.211048003945875,0,0,-1.21972744404619E-18,0.16881
186938978,0,133709514359284" />
<property name="inertiaUnits" value="kg*m^2" />
<property name="volume" value="0.00753684709895577" />
<property name="volumeUnits" value="m^3" />
<property name="centerOfGravity" value="5.99389770378761E-16" />
<property name="centerOfGravityUnits" value="m" />
<node name="ACHSE_REVOLUTE_6">
<scs>
<position>
<translation x="0" y="-0.204000003635883" z="0" />
<rotation x="0" y="0" z="0" />
</position>
</scs>
<property name="axistype" value="revolute" />
<property name="axis" value="0,1,0" />
<node name="ARM4" shape="ARM4.wrl">
<scs>
<position>
<translation x="0" y="0.284000009298325" z="0" />
<rotation x="0" y="0" z="0" />
</position>
</scs></xml version="1.0" encoding="UTF-8"?>
```

b) Parametrisiertes XML-Datenmodell

**Abbildung 2: Struktur, Visualisierung und Daten eines virtuellen Prototypen**

**Abhängigkeiten und Inkonsistenz der Daten:** Eine Herausforderung im Datenaustausch zwischen den Werkzeugen stellt die Konsistenzsicherung der Produktdaten unterschiedlicher Prozessschritte dar. Bei einem zielgerichteten Prozess werden die Modelle der einzelnen Schritte in den späteren Phasen geändert oder verfeinert. Die frühen Versionen der Produktmodelle beinhalten diese Änderungen späterer Domänen jedoch nicht. Die daraus resultierende Inkonsistenz der Daten erschwert Veränderungen und die Erstellung neuer Varianten. Beispielsweise wird die Masse eines Körpers in verschiedenen Modellen und Simulationen eines Prozesses genutzt. In den unterschiedlichen Simulationen können verschiedene Massen eingestellt werden und bilden dann neue Varianten, die nicht mit dem ursprünglichen Modell übereinstimmen.

**Ausgewählte Schritte des Prozesses:** Grundlage des VPE bilden die Phasen Konstruktion (CAD-Systeme) und Entwicklung (CAE-Systeme) sowie die Simulation. Die computergestützten Werkzeuge in diesen Bereichen haben sich unabhängig voneinander entwickelt und bieten für ihre Domäne geeignete Sichten, Unterstützungsmechanismen, Austauschformate zu abhängigen Modellen und notwendige Daten und Datenstrukturen. Die Basis des VPE bilden dabei:

- Konstruktion:
  - *Mechanische Modelle* werden üblicherweise als 2D- oder 3D-Solidmodelle erstellt. Es wird die technische Beschreibung des Modells entwickelt.
  - Ein *elektrisches Modell* fügt elektrische Antriebe und Steuerungen dem mechanischen Modell hinzu und verfeinert so das Modell.
- Simulationen identifizieren Schwachstellen oder Mängel und geben Information über mögliche Verbesserungen:
  - *Mehrkörpersimulationen* werden u.a. zur Kollisionsüberprüfung, Verhaltensüberprüfung (Bewegung) und für kinematische Tests genutzt.
  - *Mechatronische Simulationen* verwenden elektrische Antriebe und Steuerungen, um das Verhalten eines Modells zu simulieren.
  - Die *Finite Element Analyse* ist für Tests zum elastischen Verhalten (bspw. Frequenzeinfluss, thermisches Verhalten, Biegespannung) geeignet.

**Metadaten im VPE-Prozess:** Metadaten enthalten beschreibende Informationen und bereiten somit Aussagen über die Eigenschaften von Datensätzen, deren Struktur und inhaltliche Zusammenhänge auf<sup>1</sup>. Übliche Metadaten, die im Entwicklungsprozess anfallen und oft in den Dateien gespeichert werden sind zum Beispiel Urheber und Änderungsdatum. Diese Daten können genutzt und erweitert werden, um eine geeignete Nachvollziehbarkeit der Prozesse zu ermöglichen. Dabei werden erst durch semantische Beschreibungen aus den Daten Informationen, welche ebenso integriert werden wie spezielle domänenspezifische Eigenschaften der Daten [6]. Diese Integration muss dabei sowohl für die funktionalen (Masse, Temperatur) als auch für nicht-funktionale (Sicherheit, Qualität) Anforderungen repräsentiert werden [7].

Neben den Metadaten, die im Prozess gespeichert werden, müssen die VPE-Metadaten auch Abhängigkeiten in Teilmodellen der Prozessschritte, Daten und deren Abhängigkeiten erfassen. Innerhalb eines Modells gibt es verschiedene Teilmodelle und auch Abhängigkeiten von Teilmodellen, die untereinander erfasst werden müssen.

### 2.3 Aufbereitung und Austausch der Daten

Zur Unterstützung der Interoperabilität stehen eine Reihe von Werkzeugen bereit. Eine grobe konzeptuelle Einteilung unterscheidet zwischen Schnittstellen- und Integrationslösungen (z.B.: [8]). Standardformate wie (IGES [9], JT [10] oder STEP [11]) bieten solche Schnittstellen und ermöglichen durch definierte Standards den Austausch und die Wiederverwendung von Produktdaten. Es kann zwischen den grafischen Formaten, welche sich auf die Form des Produkts konzentrieren, und Produktdaten, welche auch Kenngrößen, wie das Gewicht, übergeben unterschieden werden. Weitere Formate sind zwischen den einzelnen Werkzeugen im Einsatz, um die bestehenden Standards zu erweitern und zusätzliche Kenngrößen auszutauschen.

Zur Optimierung von Arbeitsprozessen werden Metadatenmanagementsysteme oder gemeinsame Speicherplattformen verwendet, um den Entwicklungsprozess zu steuern. Dazu werden Produktdatenmanagement (PDM) oder Produktlebenszyklusmanagement (PLC) Systeme eingesetzt (z.B.: [12]). Die Möglichkeiten dieser Systeme unterscheiden sich deutlich. Sie reichen vom einfachen Versionsmanagementsystem bis zum fest definierten Ablauf mit Prozessteilnehmern, Abhängigkeiten und übergeordneten Sichten sowie den Anhang von weiteren Kenngrößen und Daten, wie rechtlichen Bestimmungen. Eine Herausforderung beim Systemeinsatz stellt die Komplexität der Lösung dar, welche auf die eigenen Bedürfnisse angepasst werden muss. So muss bei Einsatz eines Systems ein Standardprozess modelliert oder nach eigenen Anforderungen definiert werden. Deswegen werden leichtgewichtige Lösungen als Alternative oder integrierbarer Bestandteil vorgeschlagen, siehe beispielsweise [13]. Diese konzentrieren sich auf Nachvollziehbarkeit und Unterstützung von flexiblen Strukturen. Eine Übersicht über verschiedene Kategorien von Unterstützungssystemen und deren Eigenschaften findet sich in [14].

### 2.4 Ziele

Um eine integrierte Datenmanagementlösung zur simultanen VPE zu entwickeln, ist eine ganzheitliche Betrachtungsweise der benötigten Anforderungen notwendig [15]. Für die Aufgabe verschiedene, integrierte Lösungen für die VPE zur Verfügung zu stellen und dabei die simultane Arbeit von Ingenieuren zu unterstützen, eignet sich eine webgebundene Lösung auf der Basis einer gemeinsamen Datenbank. So können Daten

<sup>1</sup> <http://www.giswiki.org/wiki/Metadaten>

unabhängig vom System und simultan bearbeitet werden. Hier müssen die Daten importiert, (semi-)automatisch mit zusätzlichen Metadaten angereichert und die Teilmodule anderen Modulen zugeordnet werden. Weiterhin muss eine Prozesssicht die Vor- und Nachbereitung der Daten und ihre Abhängigkeiten anhand eines Prozessmodells darstellen und verfeinert die einzelnen Daten zuordnen können.

Die Ergebnisse (Modelle und Parameterwerte) der einzelnen Schritte im Entwicklungsprozess müssen nachvollziehbar sein. Dazu müssen die vorhergehenden Schritte abgebildet und ihrer Domäne zugeordnet werden können. Dies gilt auch für Zwischenergebnisse und -schritte. Daten, welche innerhalb eines Prozessschrittes genutzt werden, werden häufig geändert und sind zudem für die folgenden Arbeitsschritte nicht von Interesse. Diese Schritte dürfen nicht automatisch integriert werden, sondern benötigen eine Freigabe durch die Prozessverantwortlichen.

Die Dokumentation des Entwurfs werden während des gesamten Entwurfsprozesses kontinuierlich gepflegt und ergänzt. Sie stellen die Beschreibung des gesamten Prozesses und der Ergebnisse sowie der Abhängigkeiten dar. Informationen liegen in heterogenen Formaten vor, d.h. in verschiedenen Datenmodellen und unterschiedlichen Strukturen. Deshalb müssen verschiedene Strukturen aufgebrochen und möglichst feingranular mit ihren Abhängigkeiten gespeichert werden. Zusammenfassend sind folgende Eigenschaften für eine ganzheitliche Datenmanagementlösung zu unterstützen:

- Datenmodell:
  - Integration von neuen Domänen und Partialmodellen,
  - Instanzbildung,
  - Änderungen der Modelle weiterleiten,
  - Sichten ermöglichen und
  - Konsistenz sicherstellen.
- Metadatenmodell:
  - Prozessverwaltung,
  - Nutzerintegration und Kommunikation,
  - nachverfolgbare Versionen und Varianten sowie
  - Unterstützung von Flexibilität und Performanz.

### 3. Architektur & Konzept

Das in dieser Arbeit vorgestellte Konzept des Daten- und Metadatenmodells zur Integration, Konsistenzsicherung und flexiblen Modellverwaltung hat verschiedene Prozessschritte von der Konzeption über technische Beschreibungen und ihre grafischen Modelle bis zur Auswertung der Simulationen im Fokus. Eine Integration weiterer Domänen kann zusätzliche Anforderungen an die Architektur stellen, siehe zum Beispiel [16]. Es wurde eine offene Architektur für das Informationssystem entworfen, welche auf bereits existierenden Systemen in CAD-Entwicklungsumgebungen aufsetzt. Die Architektur besteht aus drei Schichten (vgl. hierzu Abbildung 3):

- Die *PDM-Schicht* nimmt die Daten der verschiedenen Domänen auf, um die Akzeptanz in der Wirtschaft zu erhöhen und eine Änderung von gewohnten Arbeitsabläufen zu vermeiden. Diese Schicht wird bereits durch kommerzielle Softwaresysteme zur Verfügung gestellt und ist damit nicht Bestandteil dieser Arbeit.
- Die *Metadatenschicht* speichert Annotationen und Prozesse aus verschiedenen Systemen, um die Metadaten zu komplettieren. Diese Schicht beschreibt Daten aus verschiedenen Domänen einheitlich, ermöglicht die Synchronisation zwischen Daten und Metadaten und bietet die Möglichkeit der Definition von bereichsübergreifenden Abhängigkeiten. Mit Hilfe von Annotationen können zusätzlich benutzerdefinierte Informationen den Objekten zugeordnet werden. Die Aufgaben der Schicht lassen sich in Metadatenverwaltung und Prozessverwaltung untergliedern.
- Die dritte Schicht ist die *Komponentenschicht* auf der Grundlage des Komponentenmodells, welche Modelle, Abhängigkeiten und Metadaten persistent speichert. Sie übernimmt die Konsistenzkontrolle und ermöglicht eine ganzheitliche Sicht auf die Daten sowie den Austausch von Parametern und Teilobjekten.

In der prototypischen Umsetzung wird kein vollständiges PDM-System als PDM-Schicht eingesetzt, sondern es werden nur die genannten, grundlegenden Funktionalitäten genutzt.

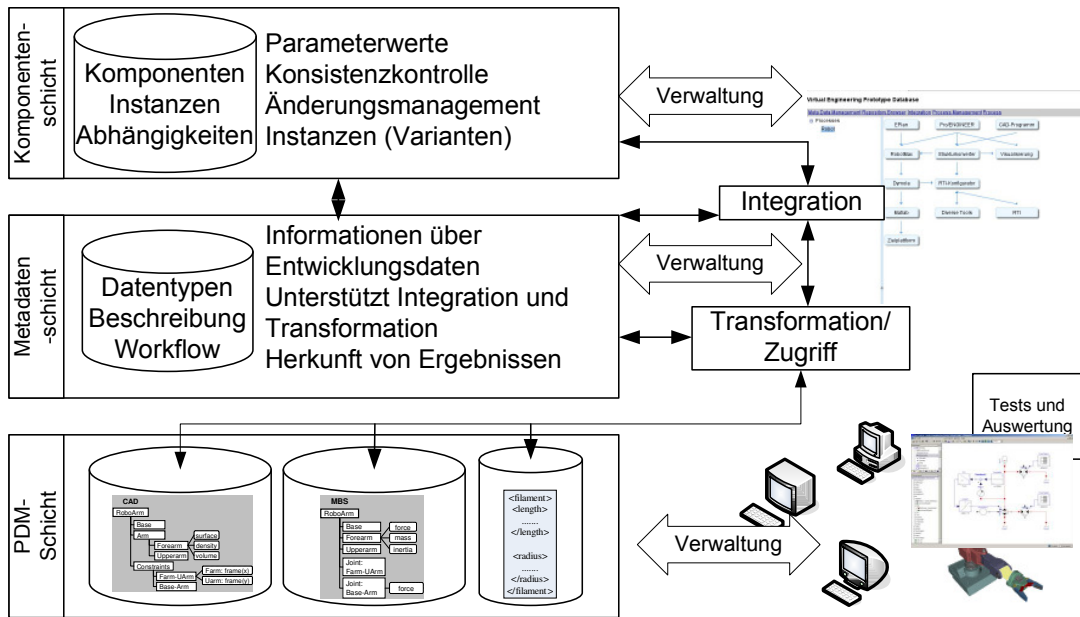


Abbildung 3: Architectureinbindung der Referenzdatenmodelle

### 3.1 Komponentenschicht

Grundlage des Datenmodells der Komponentenschicht sind Komponenten, welche domänenübergreifende Datenstrukturen bereitstellen und verschiedene Datenformate integrieren sowie explizit Abhängigkeiten verwalten. Abbildung 4a zeigt das Datenmodell als UML-Diagramm. Komponenten kapseln hierbei die separaten domänenspezifischen Modelle. Ein Domänenmodell beschreibt Verhalten, Form und Funktion eines virtuellen Prototyps in einer Entwicklungsdomäne. Modelle beinhalten Parameter und Ports (Verbindungen). Modelle unterschiedlicher Domänen, welche dieselben Objekte beschreiben, werden zu Komponenten zusammengefasst. Eine Instanz ist ein voll parametrisiertes Objekt eines Modells oder einer Komponente. Über die Schnittstelle kann ein Parametersatz (Parameterwerte) an die Teilkomponenten oder die Modelle weitergeleitet werden.

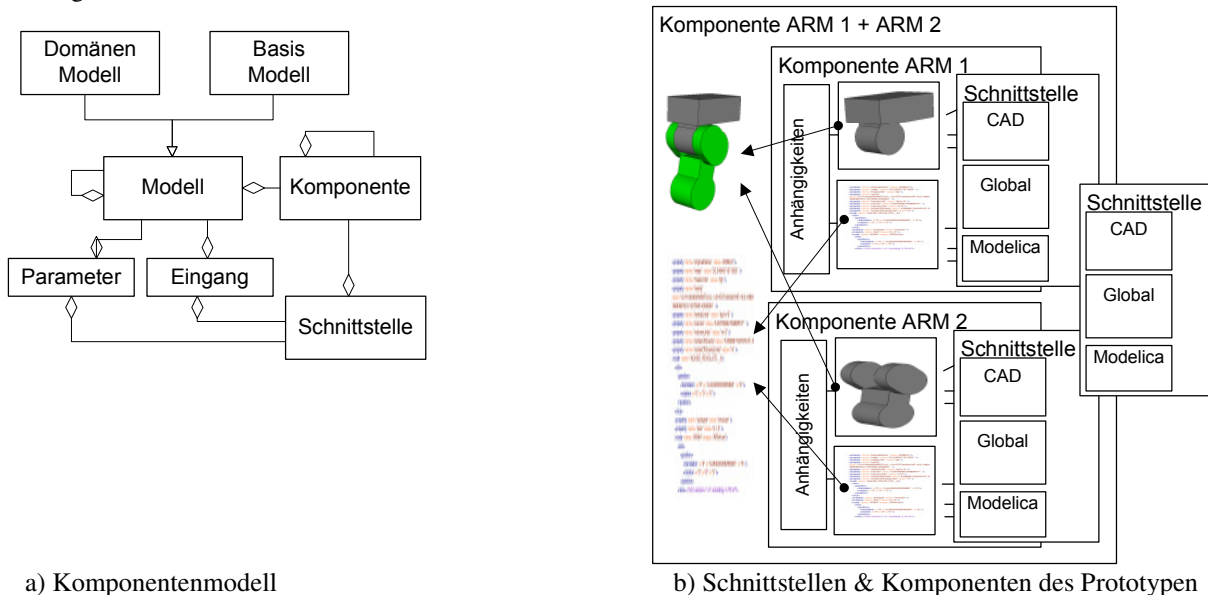


Abbildung 4: Komponentensorientierte Verwaltung und Speicherung heterogener Modelle

Da Parameterwerte in unterschiedlichen Domänen voneinander abhängen und konsistent gehalten werden müssen, ist eine Zuordnung und gemeinsame Darstellung einer Komponente notwendig (vergleiche Abbildung 4b). Hierzu werden Komponentenschnittstellen auf Basis von Mappings bzw. Transformationen definiert. Die

gemeinsame Schnittstelle bietet eine einfache und wiederverwendbare Instanziierung, ermöglicht aber auch verschiedene Sichten auf die Daten.

Das Datenmodell stellt Operationen zur Manipulation der Komponenten bereit. Sie ermöglichen eine flexible Verbindung von Teilkomponenten, die Synchronisation von domänenspezifischen Anwendungen und eine Konsistenzprüfung während des VPE. Das Komponentenmodell ermöglicht eine Wiederverwendung von domänenübergreifenden Verbindungen und eine flexible Integration von Domänen. Es beschreibt Komponenten als auch Instanzen und Änderungen. Weitere Details zu den umgesetzten Konzepten, Datenmodell und Operationen sind in [17] und [18] zu finden.

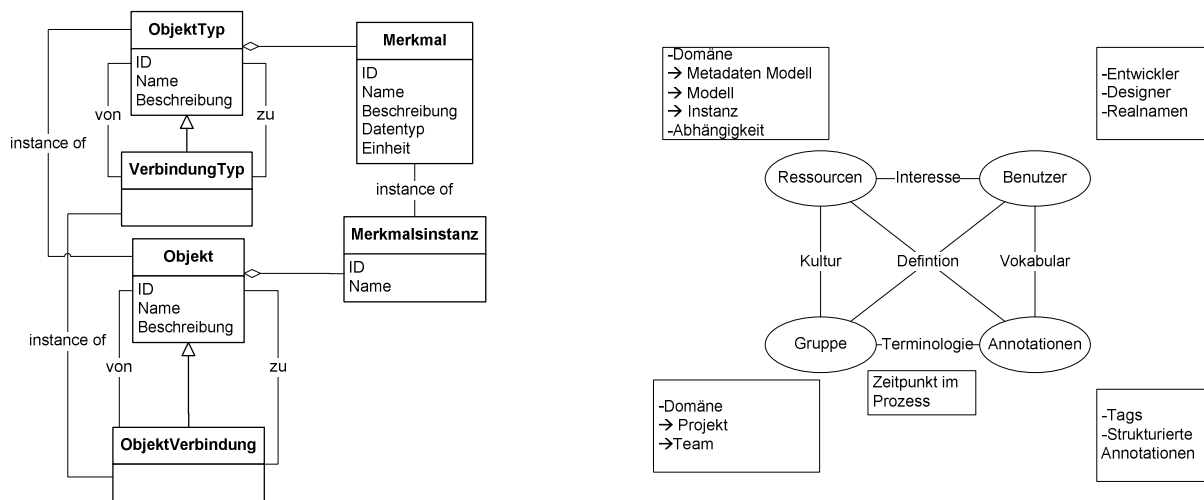
### 3.2 Metadatenmodelle

Innerhalb des VPE entwickelt eine Vielzahl von Ingenieuren aus unterschiedlichen Fachdisziplinen mit eigenen Fachbegriffen einen gemeinsamen virtuellen Prototypen. Um zusätzliche Informationen über die Herkunft von Simulationsergebnissen zu erhalten und den Prozess nachvollziehbar zu gestalten, muss das Metadaten-Repository die Informationen zu Entwicklungsdaten für alle Fachbereiche zur Verfügung stellen. Damit stellt die einheitliche Beschreibung der Daten die Basis für die explizite Darstellung aller Abhängigkeiten im VPE dar.

Es werden die folgende Arten von Metadaten unterschieden [19]: *Strukturelle Metadaten* zur Beschreibungen der lokalen Datenmodelle und -formate, *Workflow Metadaten*, für Datenflüsse und -transformationsschritte, *Deskriptive Metadaten* für die inhaltliche Beschreibungen der Ressourcen mithilfe von Tags (frei definierten Begriffen), Texten und/oder standardisierten Taxonomien, *Lineage Metadaten* beinhalten zusammengefasste Informationen wie ein Ergebnis zustande kommt (Ausgangsdaten, Transformationsschritte) sowie *technische Metadaten* mit Informationen über genutzte Systeme und Protokolle.

Um die Nachvollziehbarkeit von Prozessen zu gewährleisten, ist eine strukturierte Verwaltung der virtuellen Prototypen und Simulationen sowie deren Ergebnissen in einem strukturierten Metadatenrepository notwendig. Für die gemeinsame Verwaltung von Informationen aus unterschiedlichen Fachbereichen, ohne die Daten zu verändern, ist eine Link-Datenbank zur Integration von Modellen und ihren Kopplungen sinnvoll. Eine Möglichkeit diese an die Datenbank anzubinden, ohne die Daten selbst zu verändern, ist in [20] als eine "lightweight"-Datenstruktur beschrieben.

Abbildung 5a zeigt die Metadatenstruktur zur Integration von verschiedenen Domänen. Die Grundlage der Metadatenmodelle bildet diese Metadatenstruktur, mit deren Hilfe Objekttypen, beispielsweise CAD-Daten mit STEP-Typen, deren Merkmale (Parameterbeschreibungen) sowie verschiedenartige Verbindungstypen zwischen Objekten beschrieben werden. Verbindungstypen repräsentieren Arten von Objektbeziehungen. Dabei werden zwei Arten von Beziehungen unterschieden: innerhalb einer Domäne bzw. eines Entwicklungssystems und zwischen Objekttypen unterschiedlicher Entwicklungsdomänen. Instanzen der definierten Typen beschreiben somit die Informationen in den Quellen. Das Metadatenmodell dient als Beschreibung der Daten. Die eigentlichen Instanzen verbleiben in den Quellen.



a) Metadatenstruktur zur Integration von verschiedenen Domänen b) Folksonomy Konzept

### Abbildung 5: Metadaten zur Integration und gemeinsamen Verschlagwortung

Damit dieses Wissen der Entwickler über mehrere Fachbereiche genutzt werden kann, wird ein Metadatenmodell zur freien Verschlagwortung von Entwicklungsergebnissen benötigt. Dieses nutzt einen an Folksonomies angelehnten Ansatz (Abbildung 5b) für die gemeinschaftliche Verwaltung von Informationen [21].

Da sich fortlaufend neue Werkzeuge in den einzelnen Domänen etablieren, neue Domänen wie Simulationsmöglichkeiten hinzukommen sowie bessere Schnittstellen zur Verfügung stehen, muss auch ein

flexibler Prozessablauf unterstützt und die Ergebnisse in Abhängigkeit zu diesen Daten generiert werden. Ein Metadatenmodell zur Erfassung von komplexen Systemen und ihren Schnittstellen wird in [22] als Softwaresystem vorgestellt.

#### 4. Implementierung

Die entwickelten Datenmodelle zur Unterstützung des VPE wurden zur Evaluation in einem gemeinsamen Prototyp umgesetzt. Als Schnittstelle zwischen Daten und Metadaten fungiert ein Service, der die Metadaten aus der Datenbank mittels einer JDBC-Schnittstelle bzw. die Modelldaten aus dem SVN-Server mittels einer in Java programmierten SVN-Schnittstelle anfragt, weiterverarbeitet und an den Client überträgt. Dieser visualisiert die Daten anhand der vom Google Web Toolkit (GWT) unterstützten Weboberfläche (GUI). Die Datenverwaltung in der Datenbank sowie die Verarbeitung kann in die Bereiche Metadatenverwaltung, Dateiverwaltung und Komponentendatenbank zerlegt werden. Die dazugehörigen Dateien der Modelle sind auf einem SVN-Server hinterlegt. In der objekt-relationalen PostgreSQL-Datenbank sind die Datenbankschema als relationale Modelle umgesetzt. Die Webserviceschnittstelle soll die Ankopplung des Systems an unterschiedliche Anwendungen ermöglichen, ist im Prototyp aber auf die Einbindung von Dateien begrenzt. Kern der Implementierung und der Architektur stellt der Client dar.

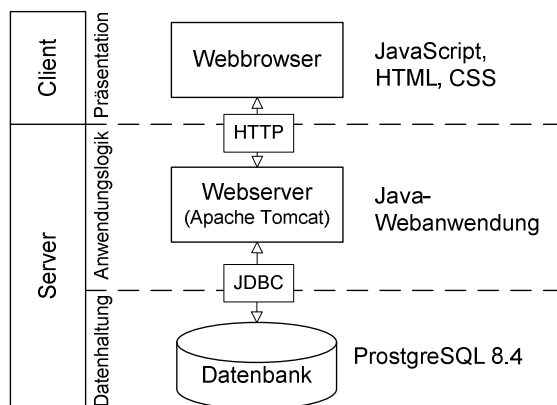


Abbildung 6: Architekturanbindung

#### Visualisierung zur Benutzerunterstützung

Wichtige Metadaten müssen automatisch aus verschiedenartigen Dateiformaten extrahiert und mit zusätzlichen Informationen dargestellt werden. Metadaten stellen dabei die aus der Struktur gewonnenen Unterteilungen, eine weitgehende Versions- und Variantenverwaltung als auch manuell editierbare Informationen dar. Um Anwender verschiedener ingenieurwissenschaftlicher Disziplinen dabei zu unterstützen, muss die Zuordnung einfach gehalten und weitestgehend visuell erfolgen.

Verschiedene Prozessabläufe müssen dargestellt werden und die einzelnen Teilschritte hervorgehoben werden können. Dadurch wird auf einzelne Daten zu einer Domäne verwiesen. Der Nachfolgeprozess sowie die Vorgänger einer Datei müssen dem Anwender ersichtlich sein. Dabei kann es mehrere Nachfolger und Vorgänger eines Schrittes geben.

Die Visualisierung der Produktdatenprozesse beinhaltet die Herausforderung, dass nicht alle Teilschritte über visuell darstellbare Modell verfügen. Diese werden jedoch benötigt, um fachfremden Entwicklern eine Orientierung zu geben. Um dennoch ein geeignetes Bild für die Unterstützung des Anwenders bereitzuhalten, werden entsprechende Teilschritte der letzten möglichen Visualisierung verwendet. Die Produkte werden in einzelnen Prozesseinheiten verwaltet, um mehrere Prozesse gleichzeitig unterstützen zu können.

In Abbildung 7 ist die grafische Darstellung eines Beispielprozesses bestehend aus fünf Prozessschritten abgebildet. Die Abhängigkeiten der Prozessschritte untereinander werden aus den, in der Datenbank hinterlegten, Metadaten generiert und als Übersicht dargestellt. Für den Nutzer sind die verschiedenen Abhängigkeiten der Prozessschritte untereinander sowie deren mögliche Eingabe- und Ausgabeformate ersichtlich. Nach Auswahl eines bestimmten Prozessschrittes gelangt der Nutzer zur Übersicht über die in dem Prozess enthaltenen Modelltypen und deren dazugehörigen Modelle. Aus der Datenbank wird dazu eine Auswahl an Metadaten angefragt.

Einerseits ist die Versionsgeschichte und somit die Nachvollziehbarkeit der Bearbeitung des Modells gewährleistet, andererseits wird die Verknüpfung des Modells mit Vorgänger- und Nachfolgermodellen, die sich aus den ausgeführten Prozessschritten ergibt, dargestellt.

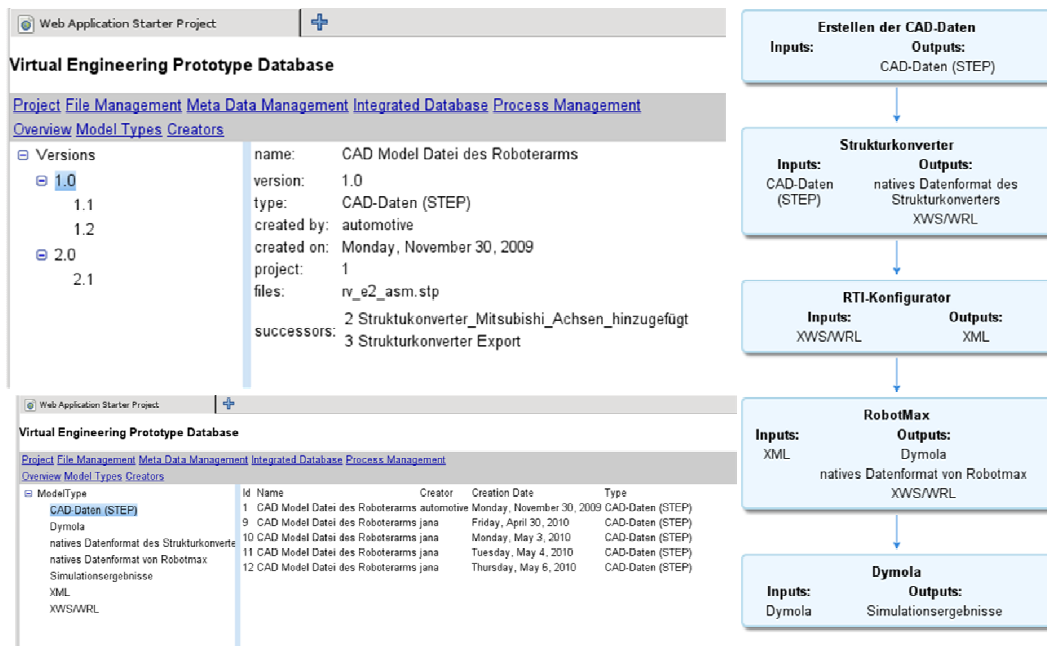


Abbildung 7: Prozessschritte und Instanzen (Screenshot)

## 5. Zusammenfassung und Ausblick

In dieser Arbeit wurden die Herausforderungen an ein integriertes Produktdatenmodell vorgestellt. Es wurde gezeigt, dass nur ein ganzheitlicher Ansatz mit verschiedenen Schichten und einer geeigneten Metadatenverwaltung eine Unterstützung der Entwickler über den gesamten VPE-Prozess hinweg die Nachvollziehbarkeit der Entwicklungsvorgänge ermöglicht.

Die entwickelte Datenmanagementlösung besteht aus drei Schichten:

Die *PDM-Schicht* nimmt die Rohdaten der verschiedenen Domänen auf. Somit unterstützt die Architektur etablierte Systemlandschaften und vermeidet eine Änderung von gewohnten Arbeitsabläufen. Dies führt zu einer erhöhten Akzeptanz in der Wirtschaft.

Die *Metadatenschicht* speichert und verwaltet Annotationen und Prozesse aus verschiedenen Systemen, um die Metadaten zu komplettieren. Durch die Integration werden Daten aus verschiedenen Domänen einheitlich beschrieben, eine Synchronisation zwischen Daten und Metadaten automatisch ermöglicht und die Möglichkeit der Definition von bereichsübergreifenden Abhängigkeiten geboten. Mit Hilfe von Annotationen können zusätzliche benutzerdefinierte Informationen (Metadaten) den Objekten zugeordnet werden. Diese Daten werden an die Komponentenschicht übergeben. Die Metadatenschicht lässt sich grob in Metadatenverwaltung und Prozessverwaltung untergliedern.

In der dritten Schicht werden die Daten erfasst und verwaltet. Dies erfolgt auf der Grundlage des Komponentenmodells, welches die Modelle, Abhängigkeiten und Metadaten speichert. Die *Komponentenschicht* ermöglicht eine Konsistenzkontrolle und eine ganzheitliche Sicht auf die Daten sowie den Austausch von Parametern und Teilobjekten über Domänengrenzen hinweg.

Die entwickelten Referenzlösungen ermöglichen einen Einsatz in existierenden, kommerziellen Softwaresystemen (z.B. als Ergänzung zu oder integrierter Bestandteil von PDM-Systemen) ebenso wie die Weiterentwicklung von Einzellösungen.

### Danksagungen

Diese Veröffentlichung entstand in Kooperation mit dem Verbundprojekt Competence in Mobility (COMO C1-3201201 und C3-320120) und wurde durch das deutsche Ministerium für Bildung und Forschung innerhalb des Projektes ViERforES-II (Nr. 01IM10002B) unterstützt. Der Inhalt dieser Veröffentlichung steht in alleiniger Verantwortung der Autoren und spiegelt somit in keiner Weise die Meinung der Europäischen Union oder des Ministeriums für Bildung und Forschung wieder.

## Literaturverzeichnis

- [1] Köppen, V.; Siegmund, N.; Soffner, M. und Saake G.: An Architecture for Interoperability of Embedded Systems and Virtual Reality. IETE Technical Review 26 (5), S. 350-356, 2009.
- [2] Eigner, Martin: IT-Lösungen für den Produktentwicklungsprozess. In: Handbuch Unternehmensorganisation: Strategien, Planung, Umsetzung. Berlin, Heidelberg, Springer-Verlag Berlin Heidelberg, S. 247–260, 2009.
- [3] Spur, G.; Krause, F.-L.: Das virtuelle Produkt - Management der CAD-Technik. Carl Hanser Verlag, 1997.
- [4] Abramovici, M.: Rechnerintegrierte Produktenwicklung. Vorlesungsskript, Lehrstuhl für Maschinenbauinformatik, Ruhr-Universität Bochum, Sommersemester 2003
- [5] VDI 2206: Entwicklungsmethodik für mechatronische Systeme. Berlin, Beuth Verlag GmbH, Juni 2004.
- [6] Vornholt, S. und Mencke, S.: Ontologies for the Virtual Engineering Process. In Michael Schenk, editor, Tagungsband – Virtual Reality und Augmented Reality zum Planen, Testen und Betreiben technischer Systeme, VR und AR–Automotive 11. IFF-Wissenschaftstage, Seiten: 183–190. Fraunhofer, IFF, June 2008.
- [7] Köppen, V. und Saake, G.: Einsatz von Virtueller Realität im Prozessmanagement. . Industrie Management, 26(2):49–53, 2010.
- [8] Mocko, G. M.; Fenves, S. J.: A Survey of Design – Analysis Integration Issues. Technical Report NISTIR 6996, National Institute Of Standards and Technology, 2003.
- [9] Nagel, R. N. ; Braithwaite, W. W.; Kennicott, P. R.: Initial Graphics Exchange Specification (IGES), Version 1.0, N.B.S. Report NBSIR 80-1978 (R), Januar 1980.
- [10] Cyon Research: The Business Case for a Common Data Distribution Platform: A Look at UGS' JT: White Paper. 2006.
- [11] Fowler, J.: STEP for Data Management, Exchange and Sharing. Technology Appraisals, 1995.
- [12] Anderl, R. und Malzacher J.: SimPDM - Simulationsdatenmanagement-Standard nach Mass. CAD CAM, 2009(1-2):38–41, 2009.
- [13] Bettaieb, S.; Noel, F.: A generic Architecture to Synchronise Design Models issued from Heterogeneous Business Tools: Towards more Interoperability between Design Expertises. Engineering with Computers, 2007.
- [14] Vornholt, S.; Geist, I. und Li, Y.: Categorisation of Data Management Solutions for Heterogeneous Data in Collaborative Virtual Engineering. In Proceedings of the First International Workshop on Digital Engineering, IWDE '10, Seiten: 9–16, ACM, New York, NY, USA, 2010.
- [15] Kreimeyer, M.; Herfeld, U.; Deubzer, F.; Lindemann, U.: Effiziente Zusammenarbeit von Konstruktions- und Simulationsabteilungen in der Automobilindustrie. Competence in Design and Development 02(01), Juni 2006.
- [16] Vornholt, S. und Köppen, V.: Data-driven and Integrated Engineering for Virtual Prototypes. In The 3rd International Multi-Conference on Engineering and Technological Innovation: IMETI, S. 164–168, 2010.
- [17] Vornholt, S. und Geist, I.: Flexible Integration Model for Virtual Prototype Families. In Proceedings on the 5th International Conference on Product Lifecycle Management (PLM08). Inderscience, S. 67-76, Juli 2008.
- [18] Vornholt, S. und Geist, I.: Interface for Multidisciplinary Virtual Prototype Components. In 19th International Workshop on Database and Expert Systems Applications (DEXA), Turin, Italy, Seiten: 85–89. IEEE Computer Society, 2008.
- [19] Geist, I. und Vornholt, S.: Metadata Repositories for Virtual Engineering. In Michael Schenk, editor, Tagungsband – Virtual Reality und Augmented Reality zum Planen, Testen und Betreiben technischer Systeme, VR und AR–Automotive 11. IFF-Wissenschaftstage, Fraunhofer, IFF, S. 149–158, Juni 2008.
- [20] Geist, I.; Vornholt S.: Eine Link-Datenbank zur Integration von Virtual Engineering-Daten. In Proceedings of the 21. GI-Workshop on Foundations of Databases (Grundlagen von Datenbanken), Rostock-Warnemünde, Mecklenburg-Vorpommern, Germany, June 2-5, 2009, Volume: CS-02-09 of Preprints aus dem Institut für Informatik, Seiten: 45–49. Universität Rostock, 2009.
- [21] Geist, I. und Vornholt, S.: Management of User-Defined Meta Information for Virtual Product Development. In 19th International Workshop on Database and Expert Systems Applications (DEXA 2008), Turin, Italy, S: 80–84, IEEE Computer Society, 2008.
- [22] Vornholt, S.; Geist, I.; Stoye, M., Köppen, V. und Saake, G.. Datenmodell zur flexiblen Verwaltung von Datenaustauschprozessen in der virtuellen Produktentwicklung. In 10. Magdeburger Maschinenbau-Tage "Forschung und Innovation", Magdeburg, 2011.