

Konzept zur zielgerichteten, ontologiebasierten Wiederverwendung von Produktmodellen

Philipp Kestel¹, Thomas Luft¹, Claudia Schon², Patricia Kügler¹,
Thomas Bayer¹, Benjamin Schleich¹, Steffen Staab²,
Sandro Wartzack¹

¹ *Lehrstuhl für Konstruktionstechnik (KTmfk); Friedrich-Alexander-Universität Erlangen-Nürnberg*

² *Institute for Web Science and Technologies (WeST); Universität Koblenz-Landau*

Abstract

The reuse of product and process models during the product development process leads to considerable savings of costs and time. Such models are available in digital form in data and knowledge bases. However, the successful reuse of these models also requires the removal of model elements that do not apply to the new product. These removal processes are not yet supported by procedure models or IT systems. Up to now, the direct deletion of model elements by product developers leads to inconsistent and fragmentary knowledge bases or retention of unnecessary data. Thus, the aim of this contribution is to facilitate the reuse of product and process models by adding methods for the efficient removal of obsolete model elements to the existing IT systems.

Keywords: Product Data Management, Knowledge-based Engineering, Ontology-based knowledge representation, Intentional Forgetting

1 Motivation und Zielsetzung

Durch die Wiederverwendung bereits vorhandener Produkt- und Prozessmodelle lassen sich im Produktentwicklungsprozess (PEP) Kosten- und Zeitaufwand erheblich reduzieren. Derartige Modelle stehen aufgrund der heutzutage vorhandenen Computerunterstützung meist in digitaler Form in umfangreichen Datenbanken und Wissensbasen zur Verfügung. Allerdings müssen die wiederzuverwendenden Produkt- und Prozessmodelle bei der Entwicklung neuer Produkte angepasst werden. In den entsprechenden IT-Systemen müssen hierzu Modellelemente (wie Anforderungen, Lösungsprinzipien, Geometrielemente und Prozessschritte), welche für das neue Produkt nicht mehr zutreffen, entfernt oder ersetzt werden. Beim Beispiel VW E-Golf (mit Elektromotor) basieren z. B. große Teile des Produkt- und Entwicklungsprozessmodells auf dem Konzept des VW Golfs mit Verbrennungsmotor. Die hierbei erforderlichen Anpassungen gehen jedoch weit über das bloße Löschen von Daten hinaus. Beim Entfernen von Modellelementen müssen z. B. die resultierenden Leerstellen geeignet repräsentiert werden. Das Wegfallen des Verbrennungsmotors im E-Golf führt u. a. dazu, dass die Abwärme des Motors nicht mehr als Heizquelle dienen kann. Die Leerstelle „Fahrgastzelle beheizen“ muss somit durch ein Ersatzelement repräsentiert werden, so dass diese Leerstelle im späteren Entwicklungsprozess berücksichtigt werden kann. Diese und viele weitere komplexe Prozesse des Entfernens bzw. „absichtlichen Vergessens“ in Daten- und Wissensbasen werden als IF-Mechanismen (engl. Intentional Forgetting) bezeichnet und betreffen nicht nur die CAD-Modelle sondern sämtliche Produktmodelle, in allen Konkretisierungsgraden (wie u. a. Anforderungs-, Funktions- und Prinzipmodelle) sowie alle zugehörigen Prozessschritte. Für die immer komplexer werdenden Produkt- und Prozessmodelle stellt die Anwendung dieser Mechanismen in IT-Systemen eine immense Herausforderung dar. In derzeitigen IT-Systemen führen ad-hoc Löschungen des Entwicklers zu inkonsistenten, lückenhaften und unübersichtlichen Datenbeständen bzw. zur Erhaltung nicht mehr erwünschter Informationen.

In diesem Beitrag wird daher ein Konzept zur Unterstützung und Nutzbarmachung der beschriebenen IF-Mechanismen vorgestellt, um die bestehenden IT-Systeme und Prozesse aus der Produktentwicklung entsprechend zu erweitern. Zur Erarbeitung dieses Konzepts wird der bisherige, implizite (ggf. unbewusste) Einsatz der IF-Mechanismen in der Produktentwicklung analysiert und geklärt, welche Voraussetzungen bei einem expliziten, gezielten Einsatz der Mechanismen erfüllt sein müssen. Dabei werden Arbeitsschritte in der Produktentwicklung betrachtet, die bereits bei der Wiederverwendung von Produktmodellen bzw. beim Entfernen von Modellelementen zum Einsatz kommen, jedoch nicht effizient und methodisch in den IT-Systemen

unterstützt werden. Ausgehend von den Untersuchungen wird eine ontologiebasierte Wissensbasis entwickelt [15], die entsprechende IF-Operatoren unterstützt und effizient im PEP zur Verfügung stellt. Diese IF-Operatoren dienen beispielsweise dem kontrollierten Entfernen von Elementen in Produkt- und Prozessmodellen und der Abbildung von Leerstellen. Für die Erreichung dieser Ziele findet eine interdisziplinäre Zusammenarbeit zwischen dem Institut für Web Science and Technologies (WeST) an der Universität Koblenz-Landau und dem Lehrstuhl für Konstruktionstechnik (KTmfk) an der Friedrich-Alexander-Universität Erlangen-Nürnberg statt.

2 Stand der Technik und Wissenschaft

Nachfolgend wird der für diesen Beitrag relevante Stand der Forschung hinsichtlich der kontextsensitiven Wissensbereitstellung in der Produktentwicklung und der ontologiebasierten Wissensrepräsentation kurz erläutert.

2.1 Kontextsensitive Wissensbereitstellung

Für eine systematische Produktentwicklung stehen verschiedene Ansätze (z. B. VDI 2221 oder Modellraum des Konstruierens) zur Verfügung, die den PEP in mehrere Phasen einteilen. In den einzelnen Phasen (z. B. Planen, Konzipieren, Entwerfen, Ausarbeiten) wird das zu entwickelnde Produkt mittels verschiedener, jeweils geeigneter Produktmodelle beschrieben. Produktmodelle früherer Phasen (z. B. für Anforderungslisten oder Funktionsstrukturen) sind hierbei deutlich abstrakter als Produktmodelle späterer Phasen (z. B. parametrisierte CAD oder -Simulationsmodelle). Innerhalb der einzelnen Phasen können Teilaspekte eines Produktmodells durch Variation bzw. Restriktion (z. B. geometrische Rahmenbedingungen, die erst in späteren Phasen behandelt werden können) oder durch Zerlegung bzw. Aggregation modifiziert werden (z. B. bei der Festlegung wesentlicher Module). Durch die in den Produktmodellen vorgesehenen Entwicklungsschritte erfolgt die Erstellung von granularen, verknüpften Elementen des Produktmodells. Wissensbasierte Systeme bilden die Zusammenhänge zwischen den Produktmodellelementen und den Prozessmodellen digital ab, z. B. durch Matrizen [6] oder Graphen [5]. In den Entwicklungsmodellen ist das Intentional Forgetting von Modellelementen bisher nicht vorgesehen. Die IF-Mechanismen müssen daher im PEP verortet und die Modelle entsprechend erweitert werden. Aufgrund einer adäquaten Granularität sowie hinreichenden Verbreitung werden die IF-Mechanismen mit dem Vorgehensmodell der VDI 2221 [19] verknüpft.

Im Rahmen des PEPs ist insbesondere die Erfassung, Formalisierung und Integration von relevantem Wissen in die verwendeten CAx-Systeme eine

wesentliche Herausforderung [7]. Dabei stellt das „Wissensbasierte Konstruieren“ (engl. Knowledge Based Engineering - KBE) hilfreiche Methoden und Werkzeuge bereit, auf die Produktentwickler zurückgreifen können [7]. STOKES definiert KBE als die Nutzung von fortschrittlichen Softwaretechniken, um Produkt- und Prozesswissen wiederzuverwenden [17]. Der Einsatz von KBE-Anwendungen erlaubt u. a. die Standardisierung (z. B. Wiederverwendung vorhandener Baugruppen), Automatisierung (z. B. durch assoziative Anpassung von Konstruktionen) und Qualitätssteigerung und -sicherung (z. B. durch computerunterstützte Anwendung formalisierter Expertenregeln) in der Entwicklung [8]. Allerdings liegt der Fokus bei der wissensbasierten Konstruktion darauf, Wissen zu erheben und in den Entwicklungsprozess zu integrieren. Der Aspekt des Intentional Forgetting ist bisher nicht vorgesehen.

Zur Repräsentation von Wissen werden syntaktische (Regeln der Anordnung) und semantische (Regeln der Bedeutung) Vorschriften zur Beschreibung von Sachverhalten verwendet [14]. In der logikorientierten Repräsentation werden Objekte und Beziehungen in Aussagen beschrieben, deren Inhalt richtig oder falsch sein kann. Durch Verknüpfung mehrerer dieser Aussagen, kann komplexes Wissen beschrieben werden. Aus dem so repräsentierten expliziten Wissen kann implizites Wissen inferiert werden. Constraint-Repräsentationen werden verwendet um Zusammenhänge zwischen Größen darzustellen (z. B. Berechnung der Getriebeübersetzung aus An- und Abtriebsdrehzahl). Bei der regelorientierten Repräsentation kommen Regeln zum Einsatz, die sich aus einem Bedingungsteil (Prämisse) und einem Aktionsteil (Konklusion) zusammensetzen, gemäß einer Wenn-Dann-Beziehung [7]. In der objektorientierten Repräsentation werden Objektstrukturen genutzt, um Wissen darzustellen. Klassen und Vererbung können z. B. für die Repräsentation ähnlicher, sich wiederholender Objekte (z. B. von Bauteilen) eingesetzt werden.

Neben KBE-Anwendungen werden Produktentwickler auch durch wissensbasierte Assistenzsysteme unterstützt. So wurde zur Unterstützung des fertigungsgerechten Konstruierens das Konstruktionssystem mfk entwickelt [2] und in [16] um eine kontextsensitive Wissensbereitstellung erweitert. Eine Weiterentwicklung der Methodik des Konstruktionssystems mfk stellen die Assistenzsysteme in [10, 12, 20] dar, die durch Sammlung, Verarbeitung und gezielte Bereitstellung von Wissen die Entwickler in ihren Tätigkeiten fachspezifisch unterstützen. Der Fokus des Konstruktionssystems mfk liegt auf Dreh-, Blech- und Gussteilen [20]. Weitere Assistenzsysteme befassen sich mit der Unterstützung semantischer Modellierung und Analyse von Informationsflüssen in der Eigenschaftsabsicherung von Produkten [21], strukturmechanischer Finite-Elemente-Analysen im Automobil- und Stahlbau [4, 13] sowie der fertigungsgerechten Gestaltung blechmassivumgeformter Bauteile [1].

2.2 Ontologiebasierte Wissensrepräsentation

In der Informatik beschreibt der Ontologiebegriff ein formales Wissensmodell, mit dem die Struktur eines Systems modelliert werden kann. In [18] wird eine Ontologie als eine explizite, formale Spezifikation einer gemeinsamen Konzeptualisierung eines abgegrenzten Bereiches definiert. Dabei bezeichnet eine Konzeptualisierung eine abstrakte und vereinfachte Sicht auf die zu repräsentierende Welt, in der die vorkommenden Objekte nach bestimmten Kriterien in Klassen eingeteilt werden. Zudem wird in der Konzeptualisierung beschrieben, welche Attribute einzelne Klassen haben und mit welchen Relationen sie miteinander verbunden sind. Eine Ontologie, die ein Getriebe repräsentiert, enthält z. B. eine Klasse *Bauteil*, die die Klassen *Lager*, *Welle* und *Zahnrad* als Unterklassen hat. Weiterhin ist die Klasse *Lager* unterteilt in die beiden Unterklassen *Gleit-* und *Wälzlager*. Die Klasse *Zahnrad* weist ein Attribut auf, das ihre Nabengeometrie spezifiziert und die Klasse *Passfeder-verbinding* ist über die Relation *hatBauteil* mit ihren Komponenten verbunden. Die in der obigen Definition geforderte Explizitheit der Konzeptualisierung bedeutet, dass sie in einer eindeutigen Form, die wenig Freiraum für Fehlinterpretationen lässt, festgehalten wird. Die Forderung, dass die Konzeptualisierung formal sein soll, zeigt auf, dass sie von Computern automatisch verarbeitet werden kann. In der Regel werden Ontologien mit Hilfe von speziellen Repräsentationssprachen, wie der Web Ontology Language (OWL) [9], festgehalten, deren klarer Vorteil ihre genau spezifizierte Semantik ist. Auch in diesem Beitrag kommt OWL zum Einsatz. Ein weiterer Vorteil der Repräsentation in OWL ist die Möglichkeit, mit Hilfe eines Reasoners automatisch Schlussfolgerungen aus dem repräsentierten Wissen zu ziehen. Aus der Definition, dass das *Lager* eine Unterklasse der Klasse *Bauteil* ist und dass das *Gleitlager* wiederum eine Unterklasse des *Lagers* ist, lässt sich folgern, dass das *Gleitlager* auch eine Unterklasse der Klasse *Bauteil* ist und somit auch alle Eigenschaften dieser Klasse erbt. Solche und wesentlich kompliziertere Schlussfolgerungen können automatisch von Reasonern durchgeführt werden. Damit tragen Reasoner dazu bei, dass nicht nur die explizit repräsentierten, sondern auch die implizit repräsentierten Informationen genutzt werden können. Bereits im Modellierungsprozess der Ontologie kann die Einbindung eines Reasoners sehr hilfreich sein, um frühzeitig Fehler und Inkonsistenzen zu identifizieren. Beispielsweise kann ein Reasoner zu jedem Zeitpunkt der Modellierung alle gültigen Schlussfolgerungen aus der aktuellen Version der Ontologie berechnen, was die Identifikation von ungewollten Schlussfolgerungen und Widersprüchen ermöglicht. Viele Reasoner sind in der Lage für alle Schlussfolgerungen Begründungen zu berechnen, die aus minimalen Teilmengen der Ontologie bestehen, aus denen die Schlussfolgerung gezogen werden kann. Diese Begründungen sind für den Entwickler der Ontologie sehr wert-

voll, um durch Modellierungsfehler auftretende ungewollte Schlussfolgerungen oder Inkonsistenzen aus der Ontologie zu entfernen.

Obwohl es einige Ansätze gibt, aus OWL-Ontologien Informationen zu entfernen, fehlt es jedoch bisher an Arbeiten zu IF-Operatoren. Beispielsweise fand das temporäre Ausblenden von Wissen und die Repräsentation von Leerstellen nach dem Vergessen von Teilen einer Ontologie in der Forschung wenig Beachtung. Eigene Vorarbeiten beschäftigen sich mit der Veränderung des instanzbasierten Teils einer Ontologie [3] sowie mit dem Verwenden von sogenannten Provenance Informationen zum Entfernen von Inkonsistenzen aus Ontologien [11]. Unter dem Begriff Provenance werden zusätzliche Informationen wie Zeitstempel, Herkunft und Glaubwürdigkeit zu einzelnen Teilen einer Ontologie zusammengefasst, die sich für die Berechnung von Veränderungen einer Ontologie als sehr wertvoll erwiesen haben.

Die Verwendung von IF-Operatoren in einer Ontologie kann, wie auch bei anderen Veränderungen einer Ontologie, leicht zu Inkonsistenzen führen. In einer vereinfachten Ontologie für Getriebe wird z. B. repräsentiert, dass eine *Welle-Nabe-Verbindung* eine *Keil-* oder *Zahnwellenverbindung* ist. Darüber hinaus wird modelliert, dass *Keilwellenverbindung* und *Zahnwellenverbindung* disjunkte Klassen sind. Wird nun aus einem Getriebe zunächst die Keilwellenverbindung über einen IF-Operator entfernt, dann eine Zahnwellenverbindung hinzugefügt und schließlich die entfernte Keilwellenverbindung über einen anderen IF-Operator wiederhergestellt, kommt es zu einer Inkonsistenz, da die Welle-Nabe-Verbindung nun gleichzeitig durch eine Zahnwellenverbindung und eine Keilwellenverbindung in der Ontologie repräsentiert wird. Inkonsistenzen in Ontologien haben den Nachteil, dass Reasoner mit den Ontologien nicht mehr richtig arbeiten können, da sich aus einer inkonsistenten Ontologie beliebige Aussagen folgern lassen. Dies führt zu unerwünschtem Verhalten der Reasoner. Deshalb ist es wichtig, dass die entwickelten IF-Operatoren auch immer die Konsistenz der resultierenden Ontologie sicherstellen.

3 Verwendete Methoden und Vorgehensweise

Zur umfassenden Analyse der implizit im Entwicklungskontext angewandten IF-Mechanismen werden Anwendungsszenarien aus der Lehre, Forschung und Industrie betrachtet. Hierzu kommen indirekte Wissensakquisitionsmethoden, wie Dokumentenanalysen, Interviews und Debriefings, zum Einsatz [14]. In der Lehre werden Veranstaltungen am KTmfk betrachtet, in der studentische Teams als konkurrierende Ingenieurbüros agieren und unter Anleitung von wissenschaftlichen Mitarbeitern industrielle Aufgabenstellungen bearbeiten. Ziel ist die Neuentwicklung innovativer, technischer Systeme, wie z. B.

stufenloser E-Bike-Getriebe. In der Forschung dient die Prüfstandskonstruktion als Anwendungsszenario, wie z. B. die Entwicklung von Tassenstößelverschleißprüfständen. Da alle Prüfstände am KTMfk nach VDI 2221 entwickelt werden, sind deren Produktmodelle und Entwicklungsprozesse ausführlich dokumentiert. Zudem dient die Entwicklung von Türmodulteilen (z. B. Fensterhebern) bei einem Automobilzulieferer als industrielles Anwendungsszenario.

Anhand dieser Anwendungsszenarien werden der Kontext und die Voraussetzungen für einen expliziten, computerunterstützten Einsatz der IF-Mechanismen erfasst. Hierbei werden alle Produktentwicklungsphasen berücksichtigt und somit Produktbeschreibungen in jeglichen Konkretisierungsgraden untersucht. In Bild 1 (oben) ist der analysierte PEP einer Getriebestufe nach VDI 2221 dargestellt. Die aus den jeweiligen Entwicklungsphasen resultierenden und analysierten Produktbeschreibungen (vgl. Bild 1) reichen von den erhobenen Anforderungen (im Rahmen der Produktplanung), den hieraus abgeleiteten Funktionsstrukturen und identifizierten Wirkprinzipien bzw. Wirkstrukturen (Konzeptphase) bis hin zu den resultierenden CAD-Modellen und Fertigungsunterlagen (Entwurfs- und Ausarbeitungsphase).

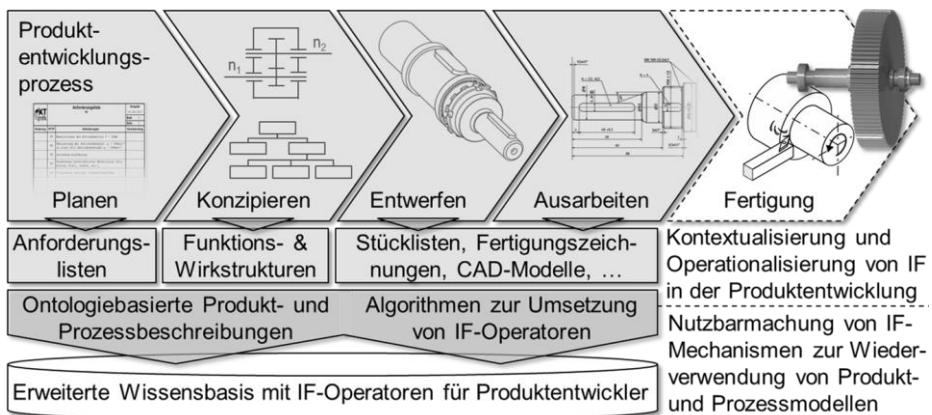


Bild 1: Analyse und IT-Unterstützung von IF-Mechanismen

Wie in Bild 1 (unten) dargestellt, müssen diese informale Produkt- und Prozessbeschreibungen in formale (rechnerverarbeitbare), ontologiebasierte Modelle überführt werden. Für einen effizienten Einsatz sollen diese Ontologien zukünftig automatisiert aus den im PEP bereits vorliegenden Datenbeständen generiert werden, wie u. a. aus CAD-Modellen. Schließlich wird auf Grundlage der Ontologien eine Wissensbasis aufgebaut, in der IF-Operatoren die effiziente Wiederverwendung der Produkt- und Prozessmodelle ermöglicht.

chen. Entsprechende Algorithmen zur Umsetzung der IF-Operatoren werden am Institut WeST für die ontologiebasierte Wissensbasis entwickelt.

4 Konzept zur Nutzbarmachung von IF-Mechanismen

Im Folgenden wird das Konzept zur Umsetzung einer ontologiebasierten Wissensbasis vorgestellt, in der durch IF-Operatoren die Wiederverwendung von Produkt- und Prozessmodellen unterstützt wird.

4.1 Ontologiebasierte Produkt- und Prozessbeschreibung

Zur Veranschaulichung der ontologiebasierten Produkt- und Prozessbeschreibung ist in Bild 2 der Ausschnitt einer vereinfachten Ontologie für eine Getriebestufe dargestellt. Für dieses Beispiel werden vorwiegend die Entwurfs- und Ausarbeitungsphasen des PEPs betrachtet, in denen bereits die Konstruktion der Baugruppen und Komponenten entsprechend der Anforderungen im CAD-System erfolgt. Für die Abbildung von Produkt- und Prozessbeschreibungen aus vorgelagerten Phasen (z. B. der Wirkstrukturen in der Konzeptphase) werden entsprechend erweiterte Ontologien benötigt.

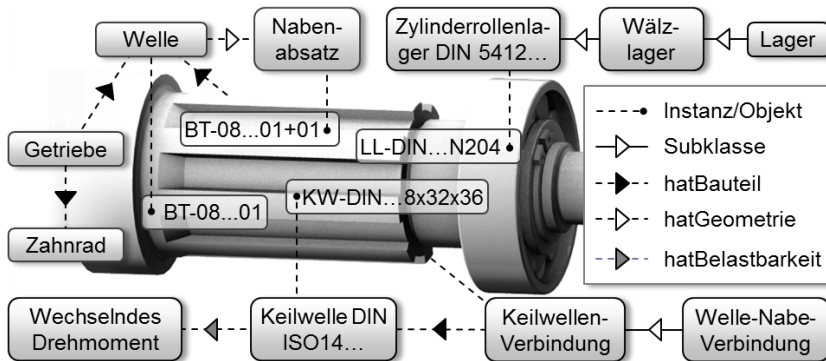


Bild 2: Ontologiebasierte Modellierung einer Getriebestufe

Wie in Abschnitt 2.2 beschrieben, werden Klassen für die Ontologie der Getriebestufe in Bild 2 definiert und deren Eigenschaften und Attribute auf Subklassen vererbt (z. B. auf das *Zylinderrollenlager* als Subklasse des *Wälzlagers*). Zudem werden über Relationen Zusammenhänge zwischen den Klassen abgebildet. In Bild 2 verbindet z. B. die Relation *hatBauteil* das Getriebe oder die *Keilwellenverbindung* mit den zugehörigen Komponenten und die Relation *hatBelastbarkeit* die *Keilwelle* mit ihrer spezifischen Belastbarkeit. Bei den Klassen, zugehörigen Subklassen und Relationen handelt es sich um ter-

minologisches Wissen. Zudem werden zu den Klassen und Subklassen konkrete Instanzen definiert (z. B. das Loslager *LL-DIN5412-N204* oder die Keilwelle *KW-DINISO14-8x32x36* in Bild 2 mit spezifisch ausgeprägten Dimensionierungsparametern). Die Instanzen repräsentieren die tatsächlichen, individuellen Elemente des zu entwickelnden Produkts, also in dieser Phase z. B. die dimensionierten Komponenten und Geometrieelemente im CAD-Modell oder ein geeignetes, spezifisches Wirkprinzip in der Konzeptphase. Für Bauteil-Instanzen, die sich nicht anhand von Normteilbezeichnungen kennzeichnen lassen, wird eine eindeutige Sachnummer nach Richtlinien des KTmfk bzw. nach Richtlinien der industriellen Kooperationspartner angegeben. Damit wird der Abgleich der Ontologie mit bestehenden Produktdokumentationen aus vergangenen und aktuellen Entwicklungsprozessen erleichtert (z. B. mit Fertigungszeichnungen und Stücklisten der entwickelten Prüfstände oder Unterlagen der genannten Lehrveranstaltungen). Ferner wurden diese Richtlinien zur Kennzeichnung der Wellenabsätze in den Ontologien erweitert.

4.2 Spezifikation und Operationalisierung von IF-Mechanismen

Für Ontologien aus der Produktentwicklung werden nun fünf IF-Operatoren in Anlehnung an Aspekte des menschlichen Vergessens konzeptioniert, die eine Wiederverwendung von Produkt- und Prozessmodellen unterstützen.

Vergessen des inferierten Wissens: Im Produktmodell sind häufig Generalisierungen festgelegt, z. B. könnte zu einem früheren Zeitpunkt festgelegt worden sein, dass alle Getriebe Zahnräder haben. Wird nun aber ein Getriebe entwickelt, das anstelle von Zahnrädern Riemen verwendet, so müssen die Zahnräder aus der Repräsentation des Getriebes entfernt werden. Dafür kann kein Datenbankeintrag gelöscht werden, sondern inferiertes Wissen muss entfernt werden. Die Herausforderung ist hier, zu überprüfen, wie sich das Vergessen inferierten Wissens auf Produktmodelle auswirkt und in welchem Maß die Beschreibung der Produktstruktur ggf. restrukturiert werden muss.

Erinnern an das Vergessen: Der Einsatz von IF-Mechanismen muss einem kontextbezogenem Management im PEP unterliegen. Das Entfernen der Modellelemente soll dabei reversibel bleiben und die eingesetzte IF-Operation darf nicht vergessen werden – sei es zur Planung, Kontrolle oder Dokumentation. Hierzu muss die Frage, wer, wann, was, zu welchem Zweck verändert hat, dauerhaft beantwortbar sein und die Reversibilität unterstützt werden.

Temporäres Vergessen: Da Produktentwickler die Vielzahl von Elementen in Produktmodellen nicht gleichzeitig entwickeln und berücksichtigen können, sind bestimmte Elemente vorübergehend auszublenden. Hierzu sind die Ele-

mente hinsichtlich ihrer Bedeutung im jeweiligen Entwicklungsschritt zu bewerten und zu priorisieren. Temporär zu entfernende Elemente sind so lange auszublenden, bis deren (relative) Priorität in späteren Entwicklungsschritten wichtiger wird. So können z. B. bei der Konstruktion einer Getriebestufe die Lager zunächst vernachlässigt und später bei der Gehäusekonstruktion wieder berücksichtigt werden. Hierfür werden geeignete Priorisierungsmethoden und Kriterien für die Produktentwicklung entworfen. Diese Methoden und Kriterien werden in der Wissensrepräsentation genutzt, um Informationen inklusive ihrer Inferenzen temporär aus- und einblenden zu können.

Repräsentation von Leerstellen: Wird ein Element in einem Produkt- oder Prozessmodell obsolet, muss dieses ggf. durch ein anderes Element ersetzt werden, ohne dass bekannt ist, um welches Element es sich konkret handelt. So muss z. B. bei einer Anforderungsänderung von "wechselndes Drehmoment übertragbar" zu "einseitiges Drehmoment übertragbar" eine Keilwellenverbindung entfernt werden. Dies darf jedoch nicht ersatzlos erfolgen, da weiterhin eine Welle-Nabe-Verbindung benötigt wird. Die Leerstelle *Welle-Nabe-Verbindung* muss also so repräsentiert werden, dass sie im späteren PEP behandelt und z. B. durch eine Passfederverbindung gefüllt werden kann. Hierzu werden Kriterien definiert, mit denen solche Leerstellen identifiziert und kontextsensitiv behandelt werden können. Die Wissensrepräsentation muss diese Unterspezifikation abbilden und das spätere Füllen ermöglichen.

Kaskadierendes Vergessen: Beim Entfernen von Modellelementen sollten nicht nur diese Elemente selbst, sondern auch bestimmte, damit verknüpfte Elemente und Entwicklungsschritte gelöscht werden. So sollten beim Entfernen der Keilwellenverbindung auch damit verbundene Elemente gelöscht werden (wie z.B. die Keilwelle zusammen mit der Keilnabe). Hierzu werden strukturelle Kriterien entwickelt und als Teil der Löschoptionen realisiert. Diese Kriterien geben an, wo das kaskadierende Entfernen enden muss, damit nicht unbeabsichtigt zu viele Elemente des Produktmodells gelöscht werden.

5 Ausblick

Das Ziel der weiteren Forschungsarbeiten ist die Nutzbarmachung der IF-Mechanismen im PEP für die Wiederverwendung von Produkt- und Prozessmodellen. Hierzu werden die konzeptionierten IF-Operatoren umgesetzt und die bestehende ontologiebasierte Wissensbasis für die genannten Anwendungsszenarien aus der Prüfstandskonstruktion, den Lehrveranstaltungen und der Industrie erweitert. Abschließend wird die VDI-Richtlinie 2221 durch die erarbeiteten IF-Mechanismen ergänzt und durch die Anwendung des zu entwickelnden Assistenzsystems im Rahmen der Anwendungsszenarien validiert.

Danksagung

Die Autoren danken der DFG für die Förderung des Projektes EVOWIPE im Rahmen des SPP 1921 und die Unterstützung der hier vorgestellten Arbeit.

Literatur

- [1] Breitsprecher, T.; Wartzack, S.: "Architecture and realization of a self-learning engineering assistance system for the use within sheetbulk metal forming", In: Hansen, P.; Rasmussen, J. (Hrsg.): Proc. of the 9th Norddesign, Design Society, Aalborg, 2012.
- [2] Finkenwirth, K.-W.: "Fertigungsgerecht Konstruieren mit CAD – Konzept eines Konstruktionssystems zur Informationsverarbeitung mit CAD-Systemen". Diss., FAU Erlangen-Nürnberg, 1990.
- [3] Furbach, U.; Schon, C.: "Semantically guided evolution of SHI aboxes", In: Galmiche, D.; Larchey-Wendling, D. (Hrsg.), Proc. of Automated Reasoning with Analytic Tableaux and Related Methods, France, 2013, Vol. 8123 of LNCS, Springer, 134-148.
- [4] Kestel, P.; Schneyer, T.; Wartzack, S.: "Feature-based approach for the automated setup of accurate, design-accompanying Finite Element Analyses", In: Marjanovic, D.; et al. (Hrsg.): Proc. of the 14th Int. Design Conf., Design Society, Dubrovnik, 2016.
- [5] Luft, T.; Ewringmann, N.; Wartzack, S.: "Application and validation of the matrix-based product description in a case study by using the software Looméo", In: Moroni, G. et al. (Hrsg.); Proc. of the 24th CIRP Design Conf., Elsevier, Milano, 2014.
- [6] Luft, T.; Krehmer, H.; Wartzack, S.: "An advanced procedure model for property-based product development", In: Lindemann, U. et al. (Hrsg.): Proc. of the 19th Int. Conf. on Engineering Design, Design Society, Seoul, 2013.
- [7] Luft, T.; Roth, D.; Wartzack, S.; Binz, HG.: "Die neue VDI-Richtlinie 5610 Blatt 2 Wissensbasierte Konstruktion", In: Binz, HG. et al. (Hrsg.): SSP 2017, Fraunhofer IAO, Stuttgart, 2015.
- [8] Lutz, C.: „Rechnergestütztes Konfigurieren und Auslegen individualisierter Produkte“, Diss., Technische Universität Wien, 2011.

-
- [9] "OWL 2 web ontology language document overview (second edition)", Technical report, W3C, 2012. <http://www.w3.org/TR/2012/REC-owl2-overview-20121211>.
- [10] Puri, W.: "Semantische Funktionsmodellierung als Basis für effizientes Product Life Cycle Management", Diss., FAU Erlangen-Nürnberg, 2003.
- [11] Schenk, S.; Dividino, R.; Staab, S.: "Using Provenance to Debug Changing Ontologies", Web Semantics, Special issue on "Semantic Web Dynamics", Bd. 9 (2011) Nr. 3, S. 284-298.
- [12] Schön, A.: "Konzept und Architektur eines Assistenzsystems für die Mechatronische Produktentwicklung", Diss., FAU Erlangen-Nürnberg, 2000.
- [13] Sprügel, T. C.; Hallmann, M.; Wartzack, S.: "A Concept for FE Plausibility Checks in Structural Mechanics", Proc. of the NAFEMS World Congress, San Diego, 2015.
- [14] Spur, G.; Krause, F.-L.: "Das virtuelle Produkt: Management der CAD-Technik", Hanser, München, 1997.
- [15] Staab, S.; Studer, R.: "Handbook on Ontologies", Springer, Berlin, 2009.
- [16] Storath, E.: "Kontextsensitive Wissensbereitstellung in der Konstruktion". Diss., FAU Erlangen-Nürnberg, 1996.
- [17] Stokes, M.: "Managing engineering knowledge". Wiley-VCH, 2001.
- [18] Studer, R.; Benjamins, R.; Fensel, D.: "Knowledge engineering: Principles and methods", Data & Knowledge Engineering, 25 (1-2), 1998.
- [19] VDI 2221: "Methodik zum Konstruieren und Entwickeln technischer Systeme und Produkte", Beuth, Berlin, 1993.
- [20] Wartzack, S.: "Predictive Engineering - Assistenzsystem zur multikriteriellen Analyse alternativer Produktkonzepte", Diss., FAU Erlangen-Nürnberg, 2001.
- [21] Westphal, C.: "Ein Beitrag zur semantischen Modellierung und Analyse von Informationsflüssen in der Produkteigenschaftsabsicherung", Diss., FAU Erlangen-Nürnberg, 2014.