

TCO-as-a-Service – Servicebasierte Lebenszyklusrechnung für hybride Leistungsbündel

Jörg Becker

Westfälische Wilhelms-Universität Münster

European Research Center for Information Systems (ERCIS)

becker@ercis.uni-muenster.de

Daniel Beverungen

Westfälische Wilhelms-Universität Münster

European Research Center for Information Systems (ERCIS)

daniel.beverungen@ercis.uni-muenster.de

Ralf Knackstedt

Westfälische Wilhelms-Universität Münster

European Research Center for Information Systems (ERCIS)

ralf.knackstedt@ercis.uni-muenster.de

Oliver Müller

Westfälische Wilhelms-Universität Münster

European Research Center for Information Systems (ERCIS)

oliver.mueller@ercis.uni-muenster.de

Steffen Müller

Westfälische Wilhelms-Universität Münster

steffen.mueller@ercis.uni-muenster.de

1 Bedeutung der Lebenszyklusrechnung für hybride Leistungsbündel

Aufgrund der Rückläufigkeit bzw. Stagnation des Sachleistungsgeschäfts und zunehmenden Preis- und Wettbewerbsdrucks sehen sich Investitionsgüterhersteller zunehmend veranlasst, Sachleistungen mit Dienstleistungen zusammen in aufeinander abgestimmten Leistungsbündeln anzubieten (vgl. Böhmman, Krcmar 2007; Fitzsimmons, Fitzsimmons 2001). Von der Integration von Sach- und Dienstleistungen versprechen sich Unternehmen unter anderem monetäre Vorteile, eine Differenzierung vom Wettbewerb, die Erhöhung der Kundenzufriedenheit und -bindung, die Möglichkeit des Angebots von Komplettlösungen und eine Stärkung des Unternehmensimage (vgl. Howells 2003; Kersten, Zink, Kern 2006).

Dienstleistungen, die das Sachleistungsangebot ergänzen, können ausgehend von der Vorkauf- und Kauf- über die Nutzungs- bis zur Nachnutzungsphase unterschiedliche Abschnitte des bzw. den gesamten Lebenszyklus einer Sachleistung betreffen. Maschinen- und Anlagenbauer verbinden das Angebot ihrer Sachleistungen z. B. mit Beratungsleistungen zur Gestaltung des Sachleistungseinsatzes, mit Schulungen zur Qualifikation des Bedienpersonals, mit Wartungs- und Instandhaltungsdienstleistungen sowie Ersatzteilmanagement zur Sicherstellung der Verfügbarkeit der Sachleistung, mit Finanzierungsangeboten, bereits beim Kauf zugesicherten Entsorgungsgarantien oder der Übernahme des Anlagenbetriebs selbst.

Um Kunden die wirtschaftlichen Vorteile des Kaufs eines Leistungsbündels darstellen zu können, bedarf es aus theoretischer Sicht einer Betrachtung sämtlicher über den Lebenszyklus des Leistungsbündels anfallenden direkten und indirekten Kosten auf Seiten des Kunden. Kunden von Investitionsgüterherstellern fragen eine solche Lebenszyklusrechnung zunehmend nach. Die Investitionsgüterhersteller selbst sind aufgrund der durch diese Berechnungen geschaffenen Transparenz und des damit derzeit verbundenen Aufwands noch zurückhaltend, dieser Anforderung nachzukommen. Erste Beispiele der Unternehmenspraxis legen allerdings die Prognose nahe, dass die Lebenszyklusrechnung für Leistungsbündel zunehmend an Bedeutung gewinnen wird. Gestützt wird diese Aussage auch dadurch, dass in der betriebswirtschaftlichen Forschung eine Reihe von Konzepten zur lebenszyklusorientierten quantitativen Planung, Steuerung und Kontrolle entwickelt wurden. Hierzu zählen in chronologischer Reihenfolge (vgl. Seewöster 2006 und die dort analysierte Literatur): Life Cycle Costing (1978), Produktlebenszyklusorientierte Ergebnisrechnung (1988), Product-Life-Cycle-Cost Management (1994), Produktlebenszyklusorientierte Planungs- und Kontrollrechnungen (1994), Product Life Cycle Management (1995), Lebenszykluskostenrechnung (1996), Lebenszyklusrechnung (1996), Lebenszyklusorientiertes Kos-

ten- und Erlösmanagement (1999), Life Cycle Target Costing (2000), Prototypgestützte Lebenszyklusrechnung (2002), Lebenszyklusrechnung unter Integration eines lebenszyklusbezogenen Zielkostenmanagements (2005) und Controlling von Life Cycle Cost-Verträgen produzierender Dienstleister (2006).

Der vorliegende Beitrag untersucht die Fragestellung, wie eine serviceorientierte Softwareunterstützung der Lebenszyklusrechnung gestaltet werden kann. Im Kern einer solchen Softwareunterstützung stehen Services, die es ermöglichen, die relevanten Daten eines konkreten Leistungsbündels zu übergeben und Informationen bezüglich der Lebenszykluskosten des übergebenen Leistungsbündels zu erhalten. Folglich würden die Services als Input ein konkretes Leistungsbündelmodell (inkl. u. a. Struktur des Bündels, Leistungen des Bündels, Eigenschaften und Eigenschaftsausprägungen der Leistungen, geplanter Lebenszyklus) erwarten, auf Basis dieses Inputs und hinterlegter Algorithmen sämtliche über den Lebenszyklus anfallenden Zahlungen (Kosten und Erlöse) ableiten, terminieren und verrechnen und schließlich zu Kennzahlen verdichtete Informationen zur Unterstützung der vorliegenden Investitionsentscheidung als Output liefern.

Der Beitrag adressiert mit dieser Entscheidungsunterstützung einen wichtigen, aber bewusst eingeschränkten Bereich anderer lebenszyklusbezogener Planungs-, Steuerungs- und Kontrollkonzepte. Er lässt sich über die folgenden Dimensionsausprägungen von anderen Ansätzen abgrenzen (vgl. Seewöster 2006): Rechnungsgegenstand sind Leistungsbündel, die sowohl aus Sach- als auch aus Dienstleistungen bestehen. Die Fristigkeit der Rechnung ist langfristig. Die Rechnung umfasst die gesamte Nutzungsdauer des Leistungsbündels beim Kunden. Die Rechnung ist dynamisch in dem Sinne, dass einzelne Werte im Laufe des Rechnungszeitraums variieren können. Es wird nicht allein – wie im statischen Fall – eine repräsentative Periode, z. B. anhand von Durchschnittswerten, berechnet. Der Einsatzzeitpunkt der Rechnung ist die Entscheidung des Kunden über die Anschaffung des Leistungsbündels. Dabei können dem Kunden alternative Leistungsbündel jeweils unter Nutzung derselben Services berechnet werden. Primärer Rechnungszweck ist damit die Fundierung einer langfristigen Investitionsentscheidung aus Kundensicht. Die Rechnung weist damit deutliche Parallelen zur Investitionsrechnung auf.

Die hier präsentierte Konzeption der Services zur Lebenszyklusrechnung basiert deshalb auf zwei wesentlichen Grundlagen (vgl. Abschnitt 2). Zur Bereitstellung der Datenbasis wird ein Konzept zur Modellierung von Leistungsbündeln benötigt und zur Erstellung der Entscheidungsunterstützung wird ein geeignetes Investitionsrechnungsverfahren gewählt. Der gewählte Modellierungsansatz wird in Abschnitt 2 vorgestellt, für die Durchführung der Lebenszyklusrechnung selbst wird auf das Konzept der vollständigen Finanzplänen zurückgegriffen (vgl. Grob 1989; Grob, Lahme 2004). Auf der

Basis dieser Grundlagen werden die notwendigen Services identifiziert und spezifiziert (vgl. Abschnitt 3). Die Kopplung der Services mit Produktdatenmanagementsystemen, die einen Großteil der notwendigen Daten des vorgestellten Ansatzes verwalten, wird am Beispiel der Referenzimplementierung des PLM Services 2.0-Servers des PDM Implementor Forums des ProSTEP iViP Vereins diskutiert (vgl. Abschnitt 4). Der Ausblick stellt den Gegenstand zukünftiger Entwicklungsarbeit vor (vgl. Abschnitt 5).

2 Konzeptionelle Grundlagen

2.1 Modellierung hybrider Leistungsbündel

Leistungsbündel können Sach- und Dienstleistungen integrieren. Für die ingenieurmäßige Beschreibung von Sachleistungen haben eigene Standards wie z. B. STEP (vgl. ISO 10303-41: Fundamentals of Product Description and Support; ISO 10303-42: Geometric and Topological Representation; ISO 10303-46: Visual Presentation) (vgl. Anderl, Trippner 2000; ProSTEP 2007) große Verbreitung gefunden. Im Zuge der Etablierung der Disziplin Service Engineering wurde die modellbasierte Spezifikation, bspw. in Form des Service Blueprinting Ansatzes (vgl. Shostack 1982), auf Dienstleistungen übertragen. Diese beiden Entwicklungsstränge werden durch Ansätze zur Modellierung hybrider Leistungsbündel vereint (vgl. Becker, Beverungen, Knackstedt 2008). Die Ansätze unterscheiden sich insbesondere dahingehend, ob sie Sachleistungen und Dienstleistungen streng getrennt betrachten, ob sie eine Modularisierung der Leistungsbündelbestandteile vorsehen, inwieweit sie Regeln zur Konfiguration von Leistungsbündeln beinhalten und ob sie ökonomische Daten, wie sie für die Lebenszyklusrechnung unumgänglich sind, berücksichtigen.

Die nachfolgende Spezifikation von Services zur Lebenszyklusrechnung basiert auf einem Modellierungsansatz, dessen konzeptioneller Sprachaspekt in Abbildung 1 dargestellt ist. Auf eine Vorstellung des repräsentationellen Sprachaspekts kann hier verzichtet werden, da von diesem bei der Nutzung der zu entwickelnden Services nicht notwendigerweise Gebrauch gemacht wird. Es sei allerdings darauf hingewiesen, dass die verwendete Sprache auch in dieser Hinsicht spezifiziert ist und durch ein Modellierungswerkzeug unterstützt wird.

len, die ihre Kombinationsmöglichkeiten definieren. Module werden durch die Zuordnung von Outcomes näher spezifiziert. Outcomes stellen Leistungen, d.h. das vermarktbare Ergebnis einer betrieblichen Faktorkombination, dar. Outcomes können in Strukturen angeordnet werden, d.h. Leistungen können aus untergeordneten Teilleistungen bestehen. Auf eine disjunkte Unterscheidung zwischen Sach- und Dienstleistungen wird aufgrund der zunehmend problematischen Abgrenzung verzichtet (vgl. Teboul 2006; Vargo, Lusch 2004). Outcomes werden durch Attributes beschrieben. Bei Leistungen mit überwiegendem Sachleistungsanteil bieten sich physikalische, mechanische oder technische Eigenschaften an. Bei Dienstleistungen finden funktionale (Welchen Zweck erfüllt die Dienstleistung?) und nicht-funktionale Eigenschaften (z. B. Preis, Qualität, Verfügbarkeit) Anwendung. Outcomes werden ferner zu LifecycleStages und Intervals zugeordnet, um die Definition der zeitlichen Struktur eines Leistungsbündels zu ermöglichen. Innerhalb der zeitlichen Dimension erfolgt zudem eine Zuordnung von Resources, Activities, Business Units, Jobs und Cost Centers, wodurch eine anbieterseitige, monetäre Bewertung von Leistungen ermöglicht wird. Als letztes relevantes Konstrukt stellt die ValueBundleInstance eine konkrete, durch den Kunden (Customer) durchgeführte Konfiguration (Configuration) eines generischen ValueBundleTypes dar. Es stellt die primäre Datenbasis für die in Abschnitt 3 beschriebenen Services zur Lebenszyklusrechnung dar.

2.2 Vollständige Finanzpläne

Der vollständige Finanzplan (VoFi) ist eine Methode des Investitionscontrollings (vgl. im Folgenden Grob, Lahme 2004). In einem VoFi werden sämtliche einem Investitionsobjekt, wie bspw. einem hybriden Leistungsbündel, zurechenbaren Zahlungen (Ein- und Auszahlungen) einschließlich der monetären Konsequenzen finanzieller sowie weiterer investiver Maßnahmen (z. B. Re- und Ergänzungsinvestitionen) explizit dargestellt und zeitlich terminiert. Wie in Abbildung 2 gezeigt, werden zu jeder Periode eines VoFi Zahlungen (Payment), Eigenkapital (EquityCapital), Kredite (Loan), Anlagen (Investment) sowie Steuerzahlungen (Tax) erfasst und miteinander zu Bestandssalden (Balance) verrechnet. Zudem können im VoFi periodenunabhängig Kennzahlen (KeyFigure), wie beispielsweise die Total Cost of Ownership (TCO), berechnet werden.

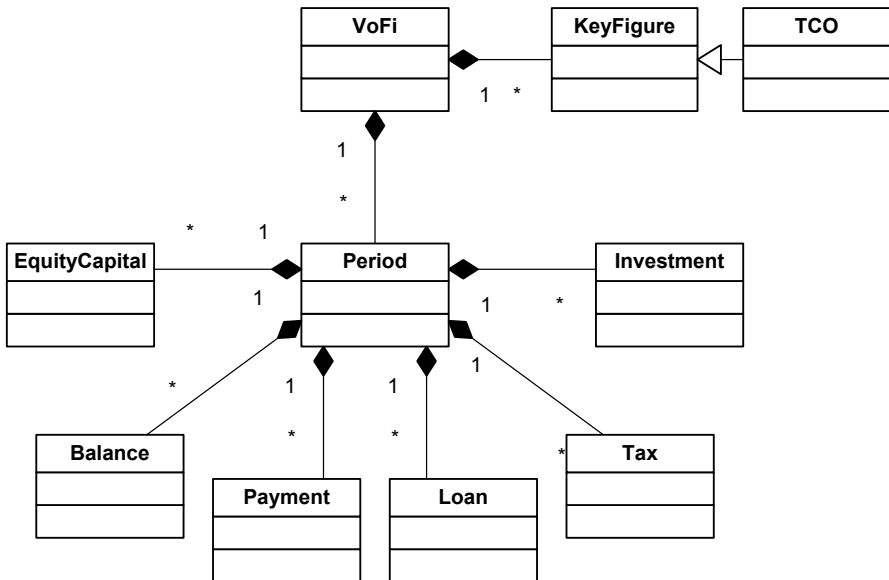


Abbildung 2. Vereinfachtes UML-Diagramm des vollständigen Finanzplans (vgl. Dewanto 2007)

3 Spezifikation der Web Services

3.1 Überblick

Der hier vorgestellte Ansatz zur Lebenszyklusrechnung lässt sich in drei Phasen einteilen, mittels derer sich auch die notwendigen Services gliedern lassen (vgl. Abbildung 3).

Voraussetzung für die Lebenszyklusrechnung ist, dass ein Kunde ein oder mehrere konkrete Leistungsbündelinstanzen aus dem vom Anbieter definierten Möglichkeitsraum abgeleitet hat. Die Konfiguration des Leistungsbündels liegt außerhalb des Fokus dieses Beitrags, was durch die kreisförmige Darstellung dieser Phase symbolisiert wird. Dieser Bereich liegt Das Modell des Leistungsbündels wird als Input an den Service `ValueBundle-to-PaymentsSequence` übergeben. Anhand der statischen (Module und Leistungen) und zeitlichen (Lebenszyklus und Intervalle) Struktur des Leistungsbündels sowie der Leistungseigenschaften (insb. Zeit- und Mengengerüst sowie Preise) und ihrer konkreten Eigenschaftswerte leitet der Service die originäre Zahlungsfolge (d. h. Abfolge der direkt mit der Anschaffung des

Leistungsbündels verbundenen Zahlungen) des übergebenen Leistungsbündels ab. Im Anschluss wird die originäre Zahlungsfolge an den Service `PaymentsSequence-to-TcoVoFi` übergeben. Dieser ermittelt und verrechnet aus der originären Zahlungsfolge derivative Zahlungen (d. h. Abfolge der durch die originäre Zahlungsfolge entstehenden Zahlungen, z. B. Kapitalkosten, Steuern). Dazu sind zusätzliche, vom Leistungsbündel unabhängige, Informationen bezüglich der finanziellen Rahmenbedingungen zu übergeben. Durch die Verrechnung der originären und derivativen Zahlungsfolgen über den Lebenszyklus des Leistungsbündels entsteht ein vollständiger Finanzplan. Der Service stellt Operationen bereit, um sowohl den vollständigen Finanzplan als auch zu Kennzahlen, insb. die Total Cost of Ownership, verdichtete Informationen bezüglich des übergebenen Leistungsbündels abzurufen. Auf Basis dieser Informationen kann der Kunde einen systematischen und fundierten Vergleich unterschiedlicher Leistungsbündel vornehmen kann.

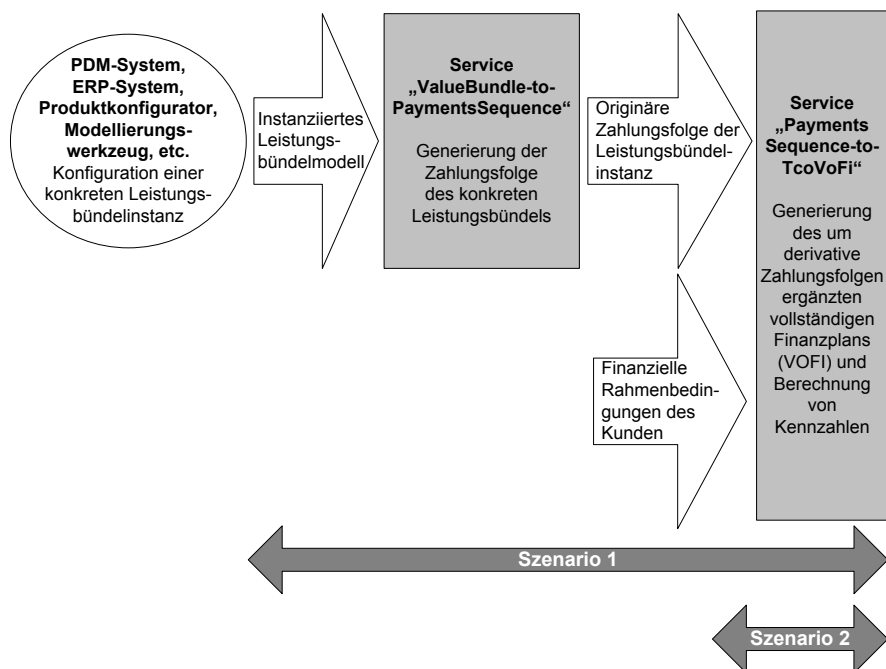


Abbildung 3. Phasen der Lebenszyklusrechnung

3.2 Web Service ValueBundle-to-PaymentsSequence

Tabelle 1 gibt eine Übersicht der fachkonzeptionellen Spezifikation des Service `ValueBundle-to-PaymentsSequence` mitsamt seiner Operationen sowie Inputs und Outputs. Der Service ermittelt aus dem übergebenen instanziierten Leistungsbündelmodell (`ValueBundleType`, `ValueBundle`

Instance, Outcomes, Attributes, LifecycleStages, Intervals) die originäre Zahlungsfolge (Periods, Payments).

Die Inputs und Outputs des Service sowie die interne Logik soll im Folgenden näher betrachtet werden. Die Schnittstelle des Service erwartet als Input eine Instanziierung eines konkreten Leistungsbündelmodells. Das bedeutet, dass wesentliche Teile der in Abbildung 1 gezeigten Klassen vom Quellsystem zu Objekten instanziiert und an einen XML-Parser übergeben werden müssen. Das erzeugte XML-Dokument im erwarteten Format kann dann an den Service übergeben werden. Neben den Objekten `ValueBundleType` und `ValueBundleInstance` sind zudem sämtliche referenzierten Instanzen der Objekte `Outcome`, `Attribute`, `LifecycleStage` und `Interval` zu parsen und an die Service-Schnittstelle weiterzureichen. Sind die benötigten Daten an den Service übergeben, so werden zunächst die Intervalle und Lebenszyklusphasen der Leistungsbündelinstanz in eine einheitliche Zeitreihe (Periods) umgerechnet. Im Anschluss werden die einzelnen Leistungen des Bündels und deren Teilleistungen durchlaufen und die jeweiligen zahlungsrelevanten Leistungseigenschaften identifiziert und deren Eigenschaftswerte (Payments) auf der zuvor erstellte Zeitreihe terminiert und verrechnet. Der entstehende assoziative Array aus Periods und Payments stellt die originäre Zahlungsfolge des Leistungsbündels und den Output des Service `ValueBundle-to-PaymentsSequence` dar.

Web Service	Operation	Input	Output
ValueBundle-to-PaymentsSequence	getOriginal_PaymentsFrom_ValueBundle	ValueBundleType, ValueBundle_Instance, Outcomes, Attributes, LifecycleStages, Intervals	Periods, Payments

Tabelle 1. Signatur des Web Service ValueBundle-to-PaymentsSequence

3.3 Web Service PaymentsSequence-to-TcoVoFi

Tabelle 2 gibt eine Übersicht der fachkonzeptionellen Spezifikation des Service `PaymentsSequence-to-TcoVoFi` mitsamt seiner Operationen sowie Inputs und Outputs. Der Service ermittelt aus einer zu übergebenden originären Zahlungsfolge (Periods, Payments) einen vollständigen Finanzplan (VoFi) sowie die Kennzahl Total Cost of Ownership (Tco).

Die Service-Spezifikation wird im Folgenden näher betrachtet. Die Operation `getTcoVoFiFromOriginalPayments` erwartet als Input zum einen die

originäre Zahlungsfolge eines Leistungsbündels (*Periods*, *Payments*). Diese kann entweder durch den vorgelagerten *Service ValueBundle-to-PaymentsSequence* (Szenario 1) oder durch ein beliebiges anderes System erstellt worden sein (Szenario 2). Zum anderen sind die finanziellen Rahmenbedingungen (*EquityCapitals*, *Loans*, *Investments*) des Kunden, welche unabhängig vom Leistungsbündel sind, zu übergeben. Diese umfassen das verfügbare Eigenkapital, geplante Kredite inklusive der Attribute Betrag, Disagio, Laufzeit und Sollzinsen sowie Standardanlagen inklusive ihrer Rendite – jeweils mit Periodenbezug (Grob, Lahme 2004). Nach Übergabe der Inputs wird die Zeitreihe der Zahlungsfolge durchlaufen und die dort spezifizierten Zahlungen werden mit den gerade beschriebenen finanziellen Rahmenbedingungen saldiert. So entsteht ein vollständiger Finanzplan (*VoFi*), der einen Teil des Outputs des Service darstellt. Der zweite Teil des Outputs, die Total Cost of Ownership (*Tco*), wird durch die Verrechnung des Saldos der letzten Periode des Leistungsbündellebenszyklus mit den kalkulatorischen Zinsen auf das eingesetzte Eigenkapital ermittelt.

Service	Operation	Input	Output
Payments _Sequence-to- _TcoVoFi	getTcoVoFi _FromOriginal _Payments	Periods, Payments, EquityCapitals, Loans, Investments	VoFi, Tco

Tabelle 2. Signatur des Web Service *PaymentsSequence-to-TcoVoFi*

4 Abgleich mit PLM Services 2.0

Als Datenquellen für die Lebenszyklusrechnung kommen neben speziellen Modelleditoren, welche die in Abschnitt 2.1 erläuterten Sprachkonstrukte unterstützen, gemäß ihrem intendierten Anwendungsbereich insbesondere Produktdatenmanagement(PDM)-Systeme in Frage. PDM-Systeme dienen der unternehmensübergreifenden, datentechnischen Integration der Informationen über Produkte aus verschiedenen Systemen, wie z. B. CAx-Werkzeugen und ERP-Systemen (vgl. Eigner, Stelzer 2001). Produkte werden in PDM-Systemen durch sogenannte produktdefinierenden Daten, die in ihrer Gesamtheit das sog. Produktmodell bilden, abgebildet (vgl. Scheer 2006). Die Informationen gliedern sich in sämtliche über den Lebenszyklus anfallenden Stamm- und Strukturdaten (*Items*) und Dokumente (*Documents*), wie z. B. CAD-Zeichnungen und Textdokumente (vgl. Eigner, Stelzer 2001).

Im Folgenden soll untersucht werden, inwieweit PDM-Systeme tatsächlich als Datenquelle für den hier vorgestellten Ansatz geeignet sind. Als exemplarisches PDM-System dient dazu die Referenzimplementierung des „PLM Services 2.0“-Servers des PDM Implementor Forums der ProSTEP iViP

Association, welcher einen Zugriff auf Produktdaten über standardisierte Web Services ermöglicht. Die Spezifikation der „PLM Services 2.0“ erfolgt durch die Object Management Group (OMG) (für eine ausführliche Beschreibung der PLM Services 2.0, des Clients und des Servers der Referenzimplementierung siehe (vgl. Feltes, Lämmer 2005)). Die Analyse in Tabelle 3 beschränkt sich auf die für die in Abschnitt 3 konzipierten Services relevanten Objekte der UML-Diagramme aus Abbildung 1 und 2.

HLB Objekt	PLM Services Package	PLM Services Class (Auszug)	Abdeckungsgrad
ValueBundle Type	Configuration _management	Complex_product, Product_structure _relationship, Product_class, Product_component, Product_identification	●
ValueBundle Instance	Configuration _management	Physical_instance	●
Outcome	Part _identification	Item, Item_version, Design_discipline _item_definition	●
Outcome Structure	Part _structure	Assembly_definition, Assembly_component _relationship	●
Attribute	Properties	Cost_property, Quality_property, Feature_parameter	●
Lifecycle Stage	Part _identification	Application_context. Life_cycle_stage	◐
Interval	Nicht vorhanden		○

Tabelle 3. Abgleich des vorgestellten Ansatzes mit den PLM Services 2.0 (vgl. OMG 2009)

Wie aus Tabelle 3 erkenntlich wird, werden die statischen Konstrukte des HLB-Ansatzes vollständig durch die PLM Services abgedeckt. Leistungsbündel auf Typ- und Instanzebene inklusive ihrer hierarchischen sowie nicht hierarchischen Strukturen lassen sich ohne Probleme durch die Klassen des Configuration_management Pakets abbilden. Selbiges gilt für einzelne Leistungen, die sich durch die Klassen des Part_identification Pakets abbilden lassen, sowie Leistungsstrukturen, die sich durch die Klassen des

Part_structure Pakets abbilden lassen. Attribute können ohne weiteres durch das Properties Paket umgesetzt werden. Im PLM Services Standard fehlen jedoch weitgehend Konstrukte zur Repräsentation der dynamischen Struktur eines Leistungsbündels über seinen Lebenszyklus hinweg. Während der präsentierte Ansatz erlaubt, sowohl für Leistungsbündel als auch einzelne Leistungen Lebenszyklen und Intervalle auf beliebigen Granularitätsstufen zu definieren, ermöglicht der PLM Services Standard lediglich die Zuordnung von einzelnen Leistungen (Items) zu den drei vordefinierten Lebenszyklusphasen Design, Manufacturing und Recycling. Zudem gleicht diese Zuordnung eher eine Bildung von Sichten zu reinen Übersichtszwecken als einer Definition von zeitlichen Strukturbeziehungen.

5 Ausblick

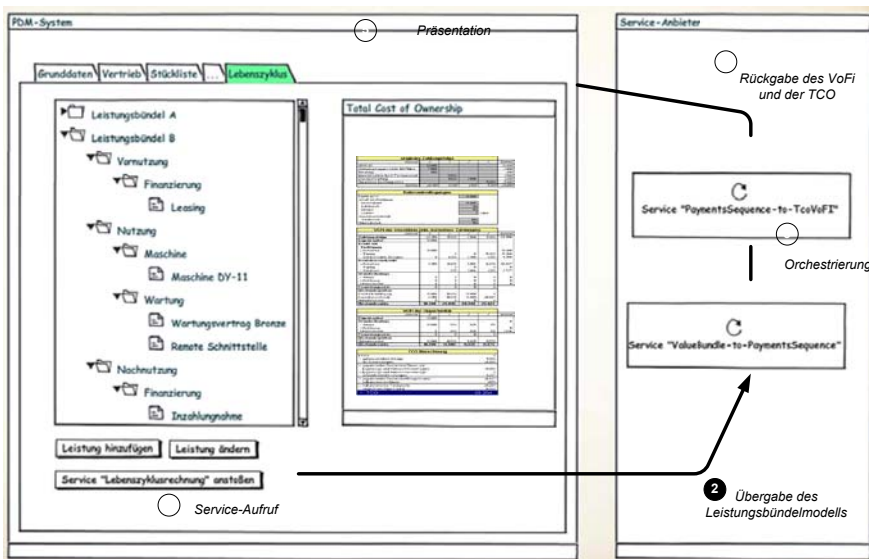


Abbildung 4. Integration der externen Services in die Oberfläche von PDM-Systemen

Es wurde ein Konzept zur servicebasierten Lebenszyklusrechnung für Leistungsbündel vorgestellt. Aktuell erfolgt die prototypische Implementierung der konzipierten Services auf Basis der in Abschnitt 4 angesprochenen Referenzimplementierung der PLM Services 2.0. Dabei gilt es neben der eigentlichen Evaluation des präsentierten Ansatzes die identifizierten Lücken im PLM Services 2.0 Standard zu schließen. Die erwarteten Erkenntnisse können Entwicklungspotenziale für anstehende Versionen des Standards aufzeigen. Zudem sollen die Services in noch folgenden Arbeiten, wie in Abbildung 4

illustriert, in bestehende Informationssysteme zur Stammdatenpflege eingearbeitet werden. Dabei ist angedacht, die Services an geeigneten Stellen in die Oberflächen dieser Systeme zu integrieren und die Übergabe der Input- als auch Output-Parameter zu automatisieren.

Danksagung

Dieser Beitrag wurde ermöglicht durch die Förderung der BMBF-Projekte „FlexNet“ (Flexible Informationssystemarchitekturen für hybride Wertschöpfungsnetzwerke; Förderkennzeichen 01FD0629) im Rahmen des Programms „Innovationen mit Dienstleistungen“ und „ServPay“ (Zahlungsbereitschaften für Geschäftsmodelle produktbegleitender Dienstleistungen; Förderkennzeichen 02PG1010) im Rahmen des Programms „Forschung für die Produktion von morgen“. Wir danken an dieser Stelle auch besonders den Projektträgern Deutsches Zentrum für Luft- und Raumfahrt (DLR) und dem Forschungszentrum Karlsruhe (PTKA-PFT) für die Betreuung.

Literaturverzeichnis

- Anderl R., Trippner D. (2000): STEP: standard for the exchange of product model data, Wiesbaden.
- Becker J., Beverungen D., Knackstedt R. (2008): Reference Models and Modeling Languages for Product-Service Systems – Status quo and Perspectives for Further Research. In: Proceedings of the 41st Hawaii International Conference on System Sciences (HICSS'08), Waikoloa, Big Island, Hawaii.
- Becker J., Beverungen D., Knackstedt R., Müller O. (2008): Modeling, Customer-Specific Configuration and Calculation of Value Bundles. In: Proceedings of the 14th Americas Conference on Information Systems (AMCIS'08), Toronto, Canada.
- Böhmman T., Krcmar H. (2007): Hybride Produkte: Merkmale und Herausforderungen, In: Bruhn M., Stauss B. (Hrsg.): Wertschöpfungsprozesse bei Dienstleistungen – Forum Dienstleistungsmanagement. Wiesbaden, S. 239-255.
- Dewanto B. L. (2007): Anwendungsentwicklung mit Model Driven Architecture – Dargestellt anhand vollständiger Finanzpläne. Berlin.
- Eigner M., Stelzer R. (2001): Produktdatenmanagement-Systeme – Ein Leitfaden für Product Development und Life Cycle Management. Berlin.
- Feltes M., Lämmer L. (2005): PLM Services Standardization – A Leap forward in Product Data Communication. In: Proceedings of the ProSTEP iViP Science Days, S. 28-39.

- Fitzsimmons J. A., Fitzsimmons M. J. (2001): *Service Management – Operations, Strategy, and Information Technology*. Bosten.
- Grob H. L. (1989): *Investitionsrechnung mit vollständigen Finanzplänen*. München.
- Grob H. L., Lahme N. (2004): Total Cost of Ownership-Analyse mit vollständigen Finanzplänen. *Controlling*, 16 (3), S. 157-164.
- Howells J. (2003): Innovation, Consumption and Services: Encapsulation and the Combinatorial Role of Services. *Service Industries Journal*, 24 (1), S. 19-36.
- Kersten W., Zink T., Kern E.-M. (2006): Wertschöpfungsnetzwerke zur Entwicklung und Produktion hybrider Produkte: Ansatzpunkte und Forschungsbedarf, In: Blecker T., Gemünden H. G. (Hrsg.): *Wertschöpfungsnetzwerke*. Festschrift für Bernd Kaluza. Berlin, S. 189-201.
- OMG (2009): *Product Lifecycle Management Services*, <http://www.omg.org/spec/PLM/2.0/PDF/>.
- ProSTEP (2007): *ProSTEP iViP: Architektur und Aufbau*, <http://www.prostep.org/de/standards/was/ausbau>.
- Scheer C. (2006): *Kundenorientierter Produktkonfigurator: Erweiterung des Produktkonfiguratorkonzeptes zur Vermeidung kundeninitiiertter Prozessabbrüche bei Präferenzlosigkeit und Sonderwünschen in der Produktspezifikation*. Berlin.
- Seewöster T. (2006): *Controlling von Life Cycle Cost-Verträgen produzierender Dienstleister*. Berlin.
- Shostack G. L. (1982): How to design a service. *European Journal of Marketing*, 16 (1), S. 49-63.
- Teboul J. (2006): *Service is Front Stage: Positioning Services for Value Advantage*, Palgrave Macmillan. Basingstone.
- Vargo S. L., Lusch R. F. (2004): The Four Service Marketing Myths. Remnants of a Goods-Based, Manufacturing Model. *Journal of Service Research*, 6 (4), S. 324-335.