Der Komponentenbegriff der Dienstleistungsdomäne

Martin Böttcher, Stephan Klingner

Abteilung Betriebliche Informationssysteme
Universität Leipzig
Johannisgasse 26
04103 Leipzig
boettcher@informatik.uni-leipzig.de
klingner@informatik.uni-leipzig.de

Abstract: Die Vorteile der Komponentisierung werden bereits in den Wirtschaftsbereichen der Softwareentwicklung und Produktion genutzt. Der vorliegende Artikel beschäftigt sich mit der Anwendung der Komponentisierung in der Domäne der Dienstleistungswirtschaft, den davon zu erwartenden Vorteilen sowie den Besonderheiten der Dienstleistungsdomäne in Bezug auf die Komponentisierung.

1 Einleitung

In den Wirtschafts- und Wissenschaftsbereichen der Softwareentwicklung und Produktion wird der Komponentenansatz bereits umfangreich eingesetzt, um unterschiedliche Vorteile zu erreichen. Auch für die Dienstleistungsdomäne existiert die Forderung nach modular aufgebauten Dienstleistungen [Ba03][BGA03][Gr04]. Trotz allem hat sich bislang weder ein konsistenter Ansatz zur Komponentisierung von Dienstleistungen [Ak04] durchgesetzt, noch haben sich einheitliche Begriffe herausgebildet. Lediglich einzelne Dienstleistungsbranchen mit langer Entwicklungstradition setzen zunehmend den Komponentenansatz um (bspw. Finanzindustrie [BC97][MW02].

Im Folgenden sollen die Komponentenansätze im Produktions- und Softwarebereich betrachtet und auf die Übertragbarkeit auf die Dienstleistungsdomäne analysiert werden. Darauf aufbauend werden die Vorteile einer Komponentisierung im Dienstleistungssektor diskutiert und eine Definition des Komponentenbegriffs dargelegt. Daran anschließend werden die notwendigen Techniken zur Strukturierung von Komponenten im Sinne einer Konfigurierbarkeit spezifiziert.

2 Motivation der Komponentisierung und Komponenten in der Produktion und Softwaretechnik

2.1 Motivation

Sowohl in der Produktion als auch in der Softwaretechnik wird der Ansatz der Komponentisierung bereits angewendet [BC97][Sz02]. Prinzipiell sind viele Wirtschaftsbereiche gekennzeichnet durch zwei gegenläufige Entwicklungen:

1. Es existierte eine zunehmende Komplexität [GS00].

2. Die Marktsituation erfordert die Berücksichtigung von Kundenwünschen (Individualisierung) bei gleichzeitiger Kostenreduzierung, was als Problem des „Mass Customization“ bezeichnet wird [PS00][Gr99].

Sowohl der zunehmenden Komplexität von Produkten als auch den Herausforderungen des Mass Customization kann durch die Modularisierung begegnet werden [Pi99]. Durch die Bereitstellung standardisierter Module und die Komposition dieser zu kundenindividuellen Angeboten können Kostenreduktion, Standardisierung und kundenspezifische Individualisierung erreicht werden [BC97]. Darüber hinaus können Qualitätsverbesserungen, Produktivitätssteigerungen sowie Verkürzungen von Innovationszyklen erreicht werden. Den genannten Vorteilen stehen hohe Anforderungen im Erstellungsprozess von Modulen gegenüber. Die Ersteller modularer Systeme müssen vom Gesamtprodukt und von den Einzelteilen Kenntnis besitzen, um die Regeln für das Zusammenspiel der Module spezifizieren zu können [BC97].

2.2 Komponenten in der Produktionstechnik

Im Kontext der Produktionswirtschaft wird der Ansatz der Modularisierung bereits seit vielen Jahren angewendet [BC97]. So definierte Starr bereits in den 60er Jahren die Notwendigkeit einer Modularisierung in der Produktion durch die Aussage: „The change that we are talking about can be briefly described as the consumer’s demand for maximum productive variety (or maximum choice). To achieve this variety, what I call „modular“ or „combinatorial“ productive capacities – that is, capacities to design and manufacture parts which can be combined in numerous ways – are required.“ (Starr, 1965) Die hierbei definierten Erwartungen konnten insbesondere im Bereich hochvolumige Industrien mit kurzen Product-Life-Cycle erfüllt werden [St65][Pi99].

Im Produktionssektor wurde lange Zeit der Begriff des „Moduls“ verwendet. So definieren Göpfert und Steinbrecher Module als „relativ unabhängige […] Montageeinheiten […], die nur durch wenige aber präzise Schnittstellen miteinander verbunden sind“ [GS00]. Baldwin und Clark sprechen von einer Menge von Elementen, die im Konzert („in concert“) zusammenarbeiten und definieren ein modulares System als „[…] composed of units (or modules) that are designed independently but still function as a whole“ [BC97].

2.2 Komponenten in der Softwaretechnik

Die Softwareentwicklung ist gekennzeichnet durch eine Veränderung über die sogenannte Modularisierung hin zur sogenannten Komponentisierung. Eine derartige Unterscheidung zwischen Modul und Komponente wird für die Domäne der Produktion nicht vorgenommen. Bei der Softwareentwicklung wurde zunächst das Konzept der Modularisierung umgesetzt, bei welchem das Ziel der Komplexitätsreduktion und der Verbindung der Vorteile kundenindividueller und standardisierter Software verfolgt wurde [Sz02]. Ein Softwaremodul wird mit folgenden Eigenschaften belegt: funktionale oder semantische Zusammengehörigkeit, weitgehende Kontextunabhängigkeit, Existenz definierter Schnittstellen sowie einer im qualitativen und quantitativen Umfang existierenden Überschaubarkeit und Verständlichkeit [Ba00b].

Das Konzept der Softwaremodularisierung wurde fortgeführt und der Begriff der Softwarekomponente geprägt. Da dieser Begriff in einem breiten Kontext genutzt wird existiert bislang keine einheitliche Definition und erscheint als solches auch nicht zweckmäßig: „Trying to come up with a general classical definition for software components is not only futile, but harmful“ [CE00]. Die Intention komponentenbasierter Softwareentwicklung besteht in der Kombination von bestehenden Komponenten zu einem neuen Produkt [PW07][Fr99]. Auch soll die Duplikation von Code reduziert und die Wiederverwendung maximiert werden [CE00]. Die Qualität von Software soll bei gleichzeitig deutlich geringeren Kosten nachhaltig verbessert [Fr99], die Produktivität bei der Gestaltung von Anwendungssystemen gesteigert [Sa97][Ga06] und die Wartbarkeit von Anwendungen verbessert werden [Ba00a].

Stellvertretend für unterschiedliche Definitionsansätze sei die folgende Begriffsbestimmung exemplarisch aufgeführt: „A software component is a unit of composition with contractually specified interfaces and explicit context dependencies only. A software component can be developed independently and is subject to composition by third parties.“ [Sz02] Zu den typischen Eigenschaften einer Softwarekomponente gehören unter anderem: a) Plattformunabhängigkeit, b) Wiederverwendbarkeit, c) Zugang über eine Schnittstelle sowie d) Kapselung der Funktionalität [Fr99][Sa97].

3 Komponenten der Dienstleistungsdomäne

3.1 Motivation zur Komponentisierung von Dienstleistungen

Aus Veränderungen der Dienstleistungswirtschaft ergibt sich auch für diese Domäne die Notwendigkeit der Komponentisierung. Anstelle einzelner abgegrenzter Dienstleistungen werden zunehmend komplexe, variantenreiche, industrielle Dienstleistungen am Markt angeboten, so dass Dienstleistungen konfiguriert und gebündelt werden müssen [SP04]. Insbesondere umfangreiche Dienstleistungsportfolios sowie die Notwendigkeit der Standardisierung bei gleichzeitiger Kundenindividualität erfordern die Komponentisierung von Dienstleistungen [AGA03].

Trotz dieser Notwendigkeit erfolgte die Auseinandersetzung mit dem Modularisierungspotential bei Dienstleistungen bislang kaum [Bu02]. Prinzipiell liegen die Vorteile modularer Dienstleistungsarchitekturen in einer „effizienteren Exploration der bestehenden unternehmerischen Ressourcenpotentiale und/oder in einer rascheren und effizienteren Exploration neuer Ressourcenkombinationen“ [Bu02]. Die einzelnen mit einem komponentenorientierten Ansatz verfolgten Ziele lassen sich wie folgt darstellen: a) Aufwandsreduktion, b) Konfiguration, c) Weiterentwicklung und Verbesserung, d) Wiederverwendung sowie e) Übersichtlichkeit und Komplexitätsreduktion.

3.2 Definition des Komponentenbegriffs in der Dienstleistungsdomäne

Die Auseinandersetzung mit dieser Thematik in der wissenschaftlichen Literatur führte zunächst zu einer terminologischen Heterogenität. So werden die Begriffe „Dienstleistungskomponente“ [Bu07][CDG06][TWL08], „Dienstleistungsmodul“ [MMS03][CDG06][SP04][BH04] und „Dienstleistungsbaustein“ [Em05][TS06] grundlegend synonym verwendet und erklären sich meist aus dem verwendeten Kontext, ohne präzise definiert zu werden.

Aus den verschiedenen Definitionsansätzen und den Ausführungen der Komponentenbetrachtung anderer Disziplinen lässt sich folgende Definition einer Dienstleistungskomponente ableiten, welche sich auch an den neuen systemischen Betrachtungsansätzen (bspw. [VL04]) orientiert: Eine Dienstleistungskomponente ist ein Teil eines Dienstleistungssystems und stellt eine definierbare und abgegrenzte Funktionalität dar, welche durch die Interaktion der Dienstleistungssystemelemente erzeugt wird. Durch diese Funktionalität wird eine Teilmenge der Elemente von einem Zustand in einen anderen Zustand überführt. Diese Zustandsänderung stellt einen Mehrwert für den Nachfrager dar. Zu den Elementen einer Dienstleistungskomponente gehören Anbieter, Nachfrager sowie deren Ressourcen. Die Funktionalität der Dienstleistungskomponente stellt mittelbar oder unmittelbar einen Mehrwert für den Nachfrager dar.

Die Idee der Dienstleistungskomponentisierung beinhaltet ebenfalls, dass sich Komponenten aus anderen Komponenten zusammensetzen können. Diese Zusammensetzung ist somit derart gestaltet, dass eine Komponente letztendlich die Summe aus anderen Komponenten darstellt. Somit muss eine Komponente immer aus mindestens zwei weiteren Komponenten zusammengesetzt werden. Komponenten, bei denen sich eine Komponente aus exakt einer anderen Komponente zusammensetzt sind äquivalent und somit redundant. In der wissenschaftlichen Literatur existieren Darlegungen zu den Eigenschaften von Dienstleistungskomponenten bspw. [BJK02][Bu02][Me03][CDG07]. Diese können um die Erkenntnisse aus den oben gegebenen Dienstleistungsdefinitionsansätzen erweitert werden. Daraus ergeben sich folgende Eigenschaften von Dienstleistungskomponenten: a) Geringe Kopplung zwischen Komponenten, b) Abgeschlossenheit einer Komponente, c) Möglichkeit der Dekomposition, d) Wohldefinierte Schnittstellen, e) Kombinierbarkeit zu individuellen Gesamtlösungen und f) Wiederverwendbarkeit.

3.3 Beschreibung von Dienstleistungskomponenten

Für einen komponentenbasierten Ansatz bedarf es zunächst der dedizierten Beschreibung der einzelnen Komponenten sowie der hierfür benötigten Ressourcen. Die einzelnen Komponenten werden mit zwei Gruppen von Eigenschaften beschrieben: funktionale und nicht-funktionale Eigenschaften. Die funktionalen Eigenschaften umfassen alle Eigenschaften, die die Funktionalität einer Dienstleistungskomponente beschreiben. Neben der expliziten Nennung von Zielkennzahlen und der textuellen Beschreibung der Funktionalität kann die Veränderung der Ressourcen sowie die Klassifikation von Dienstleistungskomponenten zur impliziten Darlegung der Funktionalität herangezogen werden.

Nicht-funktionale Eigenschaften sind als Restriktion auf die Funktionalität zu betrachten, ohne selbst Funktionalität zu beinhalten. Zu den nicht-funktionalen Eigenschaften gehören: Anbieter, lokale Verfügbarkeit, temporale Verfügbarkeit, zeitliche Dauer, Preis, Bezahlung und Konsequenzen [BF09]. Im Rahmen der Definition nicht-funktionaler Eigenschaften kann auf verschiedene existierende Standards zurückgegriffen werden (bspw. ISO, DIN, etc.).

Neben funktionalen und nicht-funktionalen Eigenschaften bedarf es der Spezifikation der Ressourcen, welche von den Komponenten benötigt bzw. verändert werden. Hierbei gilt es, zunächst die Ressourcen zu klassifizieren. Dafür werden folgende Eigenschaftsspezifikationen bereitgestellt: Herkunft (interne Ressourcen, externe Ressourcen, Kundenressourcen), Tangibilität (tangibel, intangibel, menschlich) und Mobilität (mobil, immobil). Darüber hinaus werden Attribute zur näheren Spezifikation der Ressource bereitgestellt: Kapazität, Flexibilität, Aktivität (Operator, Operand), Relation zu anderen Ressourcen sowie weitere domänenspezifische Attribute.

3.4 Konfigurationsorientierte Strukturierung der Dienstleistungskomponenten

Um die Arbeit mit Dienstleistungskomponenten zu unterstützen und die Potentiale komponentisierter Dienstleistungen voll auszuschöpfen, bedarf es einer wohldefinierten Strukturierung der Dienstleistungskomponenten (Abbildung 1). Eine unstrukturierte Menge von Dienstleistungskomponenten (Abbildung 1a) wird unter Zuhilfenahme einer graphbasierten Struktur für Konfigurationszwecke geordnet (Abbildung 1b). Aus diesem Graphen kann dann eine kundenindividuelle Konfiguration vorgenommen werden (Abbildung 1c). Die gewählten Komponenten müssen dann instanziiert werden, um den letztendlichen Ablauf der Dienstleistung zu ermöglichen. Hierbei muss sich der Ablauf nach Vorgaben aus dem Konfigurationsgraphen richten (Abbildung 1d).

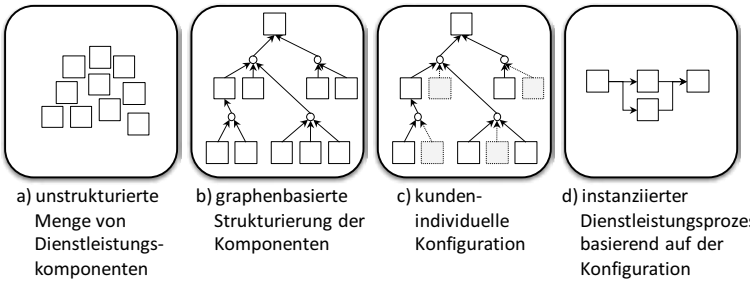

Abbildung 1: Strukturierung und Konfiguration von Dienstleistungskomponenten

Der Konfigurationsgraph setzt sich aus zwei verschiedenen Knotentypen zusammen: Dienstleistungskomponenten und Verbindungsknoten. Die Verbindungsknoten ermöglichen die Spezifikation konfigurationsrelevanter Semantik. Die Verbindung der jeweiligen Knoten erfolgt über gerichtete Kanten. Der Konfigurationsgraph ist derart zu gestalten, dass sich keine Kreise bilden können, um zu gewährleisten, dass eine Komponente sich nicht aus sich selbst zusammensetzt. Zusätzlich können die Knoten um Kardinalitäten ergänzt werden, welche eine präzisere Beschreibung von Regeln für die Konfiguration ermöglichen. Neben Abhängigkeiten, welche sich durch den Konfigurationsgraphen beschreiben lassen, existieren weitere Interdependenzen zwischen den einzelnen Komponenten. Diese können durch die Spezifikation von Regeln definiert werden. Dadurch lassen sich z.B. Aussagen treffen, dass eine Komponente zu wählen ist, falls eine andere Komponente bereits gewählt wurde.

Während bei der Produktionstechnik mit der Idee der Stücklisten die Konfiguration bereits umfangreich abgedeckt ist, unterliegt die Dienstleistungsdomäne der Besonderheit der zeitlichen Abhängigkeit. Neben der Darlegung, wie sich eine Komponente aus anderen Komponenten zusammensetzt, bedarf es der Definition der zeitlichen Abläufe zwischen den Komponenten. Aufbauend auf diesen zeitlichen Vorgaben erfolgt dann die Instanziierung des letztendlichen Prozesses. Um die Flexibilität des Konfigurationsgraphen zu gewährleisten werden die zeitlichen Abhängigkeiten durch die Spezifikation deklarativer Aussagen definiert. Diese unterscheiden sich im Gegensatz zur prozeduralen Darlegung durch die Definition klarer Regeln, an denen sich der letztendliche Ablauf zu orientieren hat. Abbildung 2 fasst die logischen und zeitlichen Abhängigkeiten zusammen.

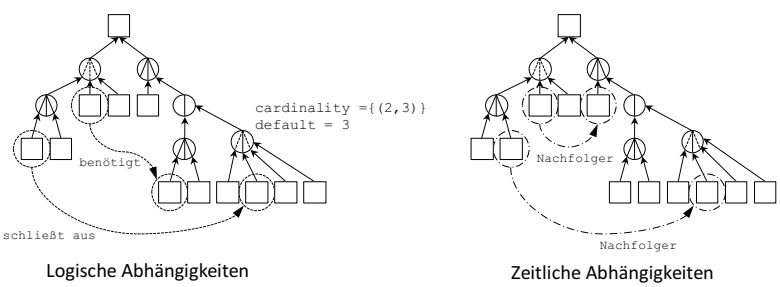

Abbildung 2: Logische und zeitliche Abhängigkeiten im Konfigurationsgraphen

Die Variabilität von Angeboten wird im definierten Graph über die einzelnen Komponenten ermöglicht. Bei der Veränderung einzelner Attribute einer Ressource oder einer Komponente wird eine neue Komponente definiert, die ausgewählt werden kann. Neben dieser konsistenten Betrachtungsweise könnten die Komponenten selbst auch mit einer entsprechenden Variabilität ausgestattet werden, aus welcher dann im Rahmen der Konfiguration zu wählen ist.

Neben der Möglichkeit der kundenindividuellen Konfiguration kann der Ansatz der Komponentisierung von Dienstleistungen auch für die Analyse der Produktivität herangezogen werden. Einerseits ermöglichen die bereits genannten Vorteile auch eine Produktivitätssteigerung per se und andererseits lassen sich für einzelne Komponenten präzisere Kennzahlen identifizieren.

Literaturverzeichnis

[Ak04] Akkermans H. et al.: Value Webs: Using Ontologies to Bundle Real-World Services. IEEE Intelligent Systems, 19, 2004, S 57-66.

[BH04] Bäcker M., Herzog L.: Modulares Dienstleistungskonzept eines mittelständischen Maschinenbauunternehmens. In: Meier H (Hrsg) Dienstleistungsorientierte Geschäftsmodelle im Maschinen- und Anlagenbau: Vom Basisangebot bis zum Betreibermodell. Berlin: Springer, 2004, S 125-140.

[BGA03] Baida, Z.; Gordijn, J.; Akkermans, H.: Service Ontology. Vrije Universität Amsterdam 2003.

[BC97] Baldwin, C. Y.; Clark, K. B.: Managing in an Age of Modularity. Harvard Business Review, 75, 1997, S 84-93.

[Ba00a] Balzert, H.: Lehrbuch der Software-Technik - Software Entwicklung, Heidelberg, Berlin, Spektrum Akademischer Verlag, 2000.

[Ba00b] Balzert, H.: Lehrbuch der Software-Technik - Software- Entwicklung / Software-Management, Software-Qualitätssicherung, Unternehmensmodellierung, Heidelberg, Berlin, Spektrum Akademischer Verlag, 2000.

[BJK02] Böhmann, T.; Junginger, M.; Krcmar, H.: Modular Service Architectures: A Concept and Method for Engineering IT Services. In: 36th Annual Hawaii International Conference on System Sciences, 2002 Hawaii (USA).

[BF09] Böttcher, M.; Fähnrich, K.-P.: Service Systems Modelling. In: Alt R, Fähnrich K-P , Franczyk B (Hrsg) Proceedings First International Symposium on Services Science (ISSS´09). Berlin: Logos, 2009.

[Bu07] Burianek, F. et al.: Typologisierung hybrider Produkte - Ein Ansatz basierend auf der Komplexität der Leistungserbringung. In: Reichenwald R (ed.) Arbeitsberichte des Lehrstuhls für Betriebswirtschaftslehre - Information, Organisation und Management. München: Technische Universität München, 2007.

[Bu02] Burr, W.: Service Engineering bei technischen Dienstleistungen, Wiesbaden, Deutscher Universitäts-Verlag, 2002.

[CDG06] Corsten, H.; Dresch, K.-M.; Gössinger, R.: Modular service production - A coordination-focused analysis. In: IFSAM VIIIth World Congress 2006, 2006 Berlin, Germany.

[CDG07] Corsten, H.; Dresch, K.-M.; Gössinger, R.: Gestaltung modularer Dienstleistungsproduktion. In: Bruhn M , Stauss B (Hrsg) Wertschöpfungsprozesse bei Dienstleistungen. Wiesbaden: Gabler, 2007.

[CE00] Czarnecki, K.; Eisenecker, U.: Generative Programming: Methods, Tools and Application, Addison-Wesley Professional, 2000.
[Em05] Emmrich, A.: Ein Beitrag zur systematischen Entwicklung produktorientierter Dienstleistungen, Universität Paderborn, 2005.
[Fr99] Frank, U.: Component Ware - Software-technische Konzepte und Perspektiven für die Gestaltung betrieblicher Informationssysteme. Information Management & Consulting, 14, 1999, S 11-18.
[Ga06] Gao, T. et al.: A Repository for Component-Based Embedded Software Development. International Journal of Software Engineering & Knowledge Engineering, 16, 2006 S 523-552.
[GS00] Göpfert, J.; Steinbrecher, M.: Modulare Produktentwicklung leistet mehr. Harvard Business Manager, 3, 2000, S 20-31.
[Gr99] Gräßler, I.: Kundenindividuelle Massenproduktion, Berlin, Heidelberg, New York, Springer, 1999.
[Gr04] Grieble, O.: Modellgestütztes Dienstleistungsbenchmarking, Universität des Saarlandes, 2004.
[MW02] Mehlau, J.I.; Wimmer, A.: Produktmodelle im Finanzdienstleistungssektor - Entwicklung eines objektorientierten Meta-Modells. Regensburger Diskussionsbeiträge zur Wirtschaftswissenschaft. Regensburg: Wirtschaftswissenschaftliche Fakultät, 2002.
[MMS03] Meier, H.; Maßberg, W.E.; Schramm, J,: Kundenindividuelle Services auf Basis modularer Dienstleistungen. In: Reinhart G , Zäh M F (Hrsg) Martchance Individualisierung. Berlin et al.: Springer, 2003.
[OEH02] O'Sullivan, J.; Edmond, D.; ter Hofstede, A.H.M.: What's in a Service - Towards Accurate Description of Non-Functional Service Properties. Distributed and Parallel Databases, 12, 2002, S 117-133.
[PS00] Peters, L.; Sadin, H.: IT and the mass customiziation of services: the challenges of implementation. International Journal of Information Management, 20, 2000, S 103-119.
[PW07] Pfeiffer, D.; Winkelmann, A.: Ansätze zur Wiederverwendung von Software im Rahmen der Softwareindustrialisierung am Beispiel von Softwarekomponenten, service-orientierten Architekturen und modellgetriebenen Architekturen. Wirtschaftsinformatik, 49, 2007, S 208-216.
[Pi99] Pine, J.B.: Mass Customization: The Frontier in Business Competition, Boston, MA, Harvard Business School Press, 1999.
[Sa97] Sametinger, J.: Software Engineering with Reusable Components, Berlin, Heidelberg, et al., Springer, 1997.
[SP04] Schramm, J.; Pallentien, K.: Entwicklung und Handhabung modularer Dienstleistungsbaukästen. In: Meier H (Hrsg) Dienstleistungsorientierte Geschäftsmodelle im Maschinen und Anlagenbau - Vom Basisangebot bis zum Betreibermodell. Berlin, Heidelberg: Springer, 2004.
[St65] Starr, M. K.: Modular Production - A New Concept. Harvard Business Review, 43, 1965 S 131-142.
[Sz02] Szyperski, C.: Component software - beyond object-oriented programming, London et al., Addison-Wesley, 2002.
[TS06] Thomas O, Scheer A-W 2006. Customizing von Dienstleistungsinformationssystemen. In: Bullinger H-J , Scheer a-W (Hrsg) Service Engineering - Entwicklung und Gestaltung innovativer Dienstleistungen. Berlin et al.: Springer, S 677-718.
[TWL08] Thomas, O.; Walter, P.; Loos, P.: Product-Service Systems: Konstruktion und Anwendung einer Entwicklungsmethodik. Wirtschaftsinformatik, 50, 2008, S 208-219.
[VL04] Vargo, S.L.; Lusch, R.F.: Evolving to a New Dominant Logic for Marketing. Journal of Marketing, 68, 2004, S 1-17.