

Informationssystemdesign für Produktionssysteme der Zukunft

Gerald Lichtenegger, Wolfgang Vorraber, Christoph Wolfsgruber, Siegfried Vössner¹

Institut für Maschinenbau- und Betriebsinformatik/ Technische Universität
Graz

1. Vorgehensmodell zur technologieunabhängigen Architekturgestaltung von Informationssystemen
 - 1.1. Analysephase
 - 1.2. Synthesephase
 - 1.3. Designphase
 - 1.4. Evaluierungsphase
2. Praxisbeispiel
 - 2.1. Allgemeines
 - 2.2. Vorgehensweise
3. Zusammenfassung
4. Literatur- und Quellenangaben

¹ Korrespondenzautor: voessner@tugraz.at

Abstract

Aus Sicht der Informatik begann die dritte industrielle Revolution bereits in den 1950er Jahren mit dem Übergang von analoger, elektrisch/elektronischer und mechanischer zu digitaler Technologie. Im ausgehenden zwanzigsten Jahrhundert ermöglichten Fortschritte der Informations- und Kommunikationstechnologie zudem die großräumige Vernetzung dieser zur Verfügung stehenden Ressourcen. Nachdem die Produktionstechnik in der Praxis wie auch die Produktionswissenschaften diesen technologischen Möglichkeiten nur sehr zögerlich Beachtung schenkten, entstand in den letzten Jahren ein sehr starkes Interesse an global vernetzten Produktionssystemen. Besonders die Anwenderseite sah, ermutigt durch Technologieanbieter in diesem Bereich, große Einsparungs- bzw. Produktivitätspotenziale. Dabei wurde jedoch auch das Fehlen von Konzepten und Strategien offenbar. In diesem Bereich ist die Wissenschaft aufgerufen, solche mit einem Anwendungsfokus zu entwickeln und wissenschaftlich zu untermauern.

Im folgenden Kapitel werden wir ein neues, auf die beschriebenen aktuellen Anforderungen angepasstes, Vorgehensmodell zur technologieunabhängigen Architekturgestaltung von Informationssystemen basierend auf den Prinzipien des Requirements Engineerings vorstellen. Dieses Vorgehensmodell eignet sich besonders zur IT-Unterstützung komplexer, soziotechnischer Prozesse, in welchen Menschen und Maschinen synergetisch zusammenwirken. Hierbei werden ausgehend von einer partizipativen Prozessanalyse Verbesserungsansätze identifiziert, deren IT-basierte Umsetzungspotenziale bewertet und priorisiert. Davon ausgehend werden dann technische Lösungen konzipiert sowie im Rahmen von Proofs of Concepts evaluiert. Dabei wird im Bereich der Systemgestaltung (Design-Phase) besonders auf neue Gestaltungskonzepte und Werkzeuge eingegangen, die insbesondere auf Struktur- und Netzwerkeffekte durch unterschiedliche Nutzenfunktionen der beteiligten Partner (Value Networks) sowie Sicherheitsaspekte Rücksicht nehmen.

Die hier vorgestellten Konzepte wurden in vielen realen Projekten erprobt, angewandt und umgesetzt. Als Beispiel zur Veranschaulichung des Vorgehensmodells wird im Anschluss ein Projekt aus der Automobilindustrie gezeigt.

1 Vorgehensmodell zur technologieunabhängigen Architekturgestaltung von Informationssystemen

Das in Vössner und Lichtenegger (2011) und Lichtenegger (2015) beschriebene Vorgehensmodell zur technologieunabhängigen Architekturgestaltung von Informationssystemen gliedert sich, wie in Bild 1 dargestellt, in vier Phasen: *Analyse-*, *Synthese-*, *Design-* und *Evaluierungsphase*. In allen diesen Phasen werden die Bedarfsträger und Endbenutzer in Form von Workshops und Feldversuchen sowie bei der Implementierung bzw. Integration in die Produktivsysteme aktiv mit einbezogen (Partizipativer Ansatz). Obwohl die Phasen zwar chronologisch nacheinander erfolgen, sind Rückkoppelungsschleifen ein wichtiger Bestandteil des Modells. Die in den einzelnen Phasen angewendeten Lösungsansätze basieren unter anderem auf dem von Haberfellner et al. (2015) vorgestellten Systems-Engineering-Vorgehensmodell, welches wiederum auf vier Grundgedanken aufgebaut ist. So wird bei der Problemanalyse und Strukturierung „Top-Down“ - vom „*Groben zum Detail*“ vorgegangen. Lösungsalternativen werden nach dem Prinzip des „*Denkens in Varianten*“ erstellt und bewertet. Der dritte Grundgedanke des Systems-Engineerings erfordert es, den Prozess der Systementwicklung und -realisierung nach zeitlichen Gesichtspunkten zu gliedern („*Phasenablauf*“). Die Anwendung eines methodischen Vorgehenskonzeptes, quasi eine Art Arbeitslogik zur Lösung der Probleme, die als formaler Leitfaden dient („*Problemlösungszyklus*“), stellt das vierte Grundprinzip dar.

Ein weiteres, wichtiges Prinzip des in Bild 1 skizzierten Modells ist eine möglichst späte Berücksichtigung von technologischen Möglichkeiten und Einschränkungen („*technology penetration point*“). Bis zum Beginn der Designphase sollen sämtliche Analysen und Synthesen möglichst ohne Bezug zur technischen Realisierbarkeit ausgeführt werden, sodass unabhängig und unbeeinflusst von aktuellen technischen Möglichkeiten, Nutzenhebel erkannt, bewertet und damit zu priorisiert werden können. Erst in der Design-Phase wird auf technische Limitationen bzw. Möglichkeiten eingegangen, um damit möglichst effizient die identifizierten Nutzenhebel zu realisieren.

Im Folgenden werden die einzelnen Schritte des Frameworks näher erläutert.

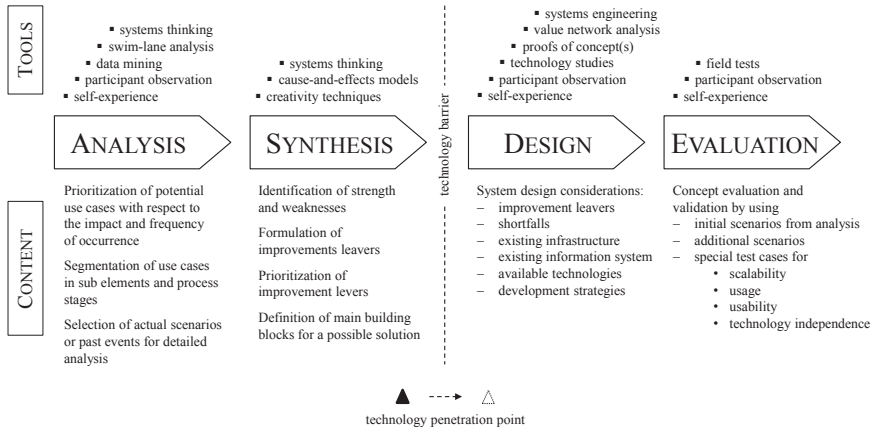


Bild 1: Vorgehensmodell zur technologieunabhängigen Architekturgestaltung von Informationssystemen. (Vössner und Lichtenegger 2011, Abbildung aus Lichtenegger 2015)

1.1 Analysephase

In der ersten Phase werden zunächst das Problemfeld genauer untersucht und gewisse Hauptbereiche identifiziert, zu denen weitere Untersuchungen angestellt werden. Diese Sondierung erfolgt mithilfe festzulegender Relevanzkriterien, die je nach Anwendungsgebiet spezifisch erarbeitet werden müssen (z.B. wirtschaftliche oder gesellschaftliche Relevanz). Im nächsten Schritt werden konkrete Einsatzfälle der einzelnen Hauptbereiche in Hinblick auf organisatorische, prozessuale, technische, rechtliche und gesellschaftliche Aspekte analysiert. Diese Analyse wird einerseits im Rahmen von interdisziplinären Workshops mit Experten der Bedarfsträger und den wissenschaftlichen Partnern sowie durch das Aufarbeiten bestehender Berichte, Dokumentationen und Statistiken durchgeführt. (Vössner und Lichtenegger 2011, Vorraber et al. 2015)

1.2 Synthesephase

In der Synthesephase werden zu den untersuchten Anwendungsfällen Verbesserungspotentiale identifiziert. Anschließend werden für diese Verbesserungspotentiale sogenannte „Nutzenhebel“ formuliert. Nutzenhebel sind Konzepte, mit denen die identifizierten Verbesserungspotentiale umgesetzt werden können, sie

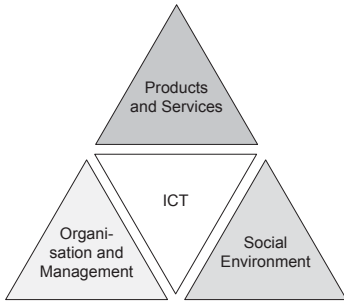
können als wichtige Attribute des Zielsystems aufgefasst werden. Ein wichtiges Merkmal der Nutzenhebel ist die komplette Unabhängigkeit von den möglichen späteren technischen Umsetzungen, damit sollen etwaige Denkbarrieren vermieden werden. Nachdem die Nutzenhebel identifiziert wurden, werden sie in einem nächsten Schritt nach ihrer Wichtigkeit für das Erreichen des Zieles gereiht. (Vössner und Lichtenegger 2011, Vorraber et al. 2015)

1.3 Designphase

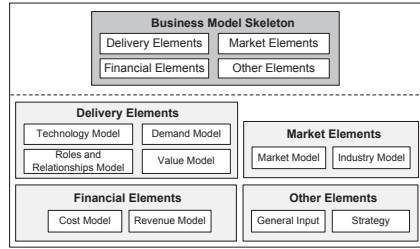
Diese Phase beschäftigt sich mit dem Entwurf und der Umsetzung von meist technischen Lösungen für die identifizierten Nutzenhebel in Form von Proof of Concepts (PoC). Beim Design der PoC werden organisatorische, prozessuale, technische, rechtliche, gesellschaftliche und wirtschaftliche Aspekte berücksichtigt. Prinzipiell soll mit den PoC die voraussichtliche Umsetzbarkeit der entworfenen Lösungen evaluiert werden. Diese Überprüfung erfolgt durch die exemplarische Anwendung der PoC durch die Endanwender. Für die erfolgsversprechendsten PoC werden im nächsten Schritt Prototypen erstellt, welche einen wesentlich höheren technischen Reifegrad als die PoC aufweisen. (Vössner und Lichtenegger 2011, Vorraber et al. 2015)

Für den Entwurf solcher (IT-basierten) Dienste existieren vor allem im kommerziellen Bereich einige nützliche Service Design Frameworks (siehe Bild 2).

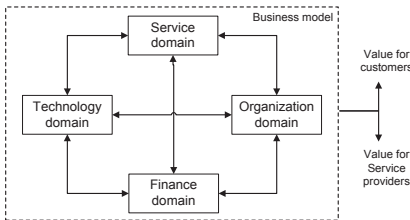
Diese Frameworks strukturieren den Design-Prozess grob in die Hauptbestandteile „Technik“, „Organisation“, „Service“ und „Finance“ und bieten für diese Kategorien Handlungsempfehlungen und Werkzeuge zur Planung neuer Systeme an. Grundprämisse ist dabei meist die Erstellung eines kommerziell erfolgreichen Dienstes, der dem Dienstanbieter möglichst viele monetäre Vorteile bringt. (Vorraber 2012). Zur Analyse von dynamischen Effekten zwischen diesen einzelnen statischen Bestandteilen können sogenannte Wertschöpfungsnetzwerke herangezogen werden.



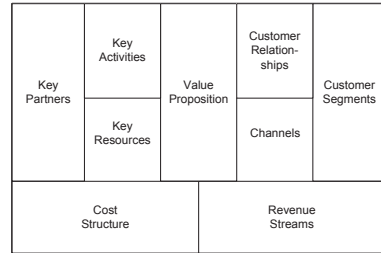
a. Fribourg ICT-Management Framework (Teufel et al. 2004)



b. BIZTEKON Business Model Skeleton (Gjerde et al. 2007)



c. STOF Model (Bouwman et al. 2008)



d. Business Model Canvas (Osterwalder & Pigneur 2010)

Bild 2: Überblick über aktuelle Service Design Frameworks. (Vorraber 2012)

Im folgenden Kapitel wird die von Vorraber (2012) entwickelte „V²“ Notation zur Darstellung von Value Networks, welche eine Erweiterung der Notation von Biem und Caswell (2008) darstellt, näher erklärt (siehe auch Vorraber und Vössner 2010, Vorraber und Vössner 2011).

1.3.1 V² Value Network Notation

Diese Notation beinhaltet als Grundbaustein Entitäten, welche entweder spezielle Personen oder gewisse Rollen repräsentieren.

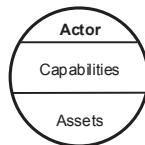


Bild 3: Entität der V² Value Network Notation. (Vorraber 2012)

Wie aus Bild 3 ersichtlich besteht eine Entität in der V² Value Network Notation gemäß Biem und Caswell (2008) aus drei Teilen:

- *Agent* (Akteur): Bezeichnung des Teilnehmers
- *Capabilities* (Fähigkeiten): Dynamische Aspekte – Prozesse und Fähigkeiten
- *Assets* (Ressourcen/Ausstattung): Statische Aspekte – Anlageobjekte (z.B.: Wissen, IT-Systeme)

In Bild 4 sind die im Value Network möglichen Austauschbeziehungen dargestellt. Ein „Provision Link“ weist auf eine Werterbringung in Richtung des Endkunden hin und wird mit einer durchgezogenen Linie dargestellt. Mit einer gestrichelten Linie wird ein „Revenue Link“ dargestellt, der die Werterbringung entgegen der Richtung des Endkunden markiert (Vorraber 2012).



Bild 4: Austauschbeziehungen (Vorraber 2012)

Der Wert der zwischen den einzelnen Entitäten ausgetauscht wird, wird durch Transferobjekte dargestellt. In Bild 5 sind ein Teil der fünf Grundtypen von Biem und Caswell (2008) (Produkt, Marke, Service, Koordination und Information) und die beiden Erweiterungen von Vorraber (2012) (Geldwert und immaterieller Wert) dargestellt.

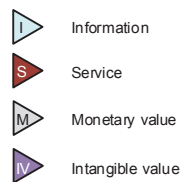





Bild 5: Transferobjekte (Abbildung modifiziert (Vorraber 2012))

Information: Das Transferobjekt Information repräsentiert den Austausch von Information als eine Art von austauschbarem Gut.

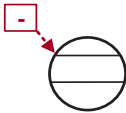
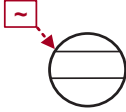
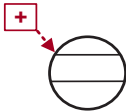
Service: Nach (Biem und Caswell 2008) stellt dieses Transferobjekt eine Dienstleistung eines Diensteanbieters bei einem Dienstempfänger dar. Es kann sich dabei entweder um materielle oder immaterielle Güter handeln.

Tabelle 1: Kategorisierung und Notation der endogenen Motivationen. (Vorraber 2012)

Defensiv (-)	Tasks haben geringste Priorität	
Neutral (~)	Tasks werden zeitgerecht und kollaborativ erbracht	
Aktiv (+)	Tasks haben höchste Priorität	

Exogene Einflüsse stellen externe Kräfte dar (z.B. der Führungsetage), welche die Teilnahme des Agenten entweder erschweren oder erleichtern. In Tabelle 2 ist die Kategorisierung und die Notation dieser Motivationen dargestellt. (Vorraber 2012)

Tabelle 2: Kategorisierung und Notation der exogenen Motivationen (Vorraber 2012)

Defensiv (-)	Externe Kraft hat eine negative Einstellung gegenüber der Wertschöpfungsaktivität	
Neutral (~)	Externe Kraft verhindert nicht, fördert aber auch nicht	
Aktiv (+)	Externe Kraft fördert aktiv die Wertschöpfungsaktivität	

Wie in Bild 6 ersichtlich, können mit Hilfe dieser Notation dynamische (positive sowie negative) Netzwerkeffekte sichtbar gemacht werden und damit die strategische Planung von Diensten unterstützt werden (Vorraber 2010).

1.4 Evaluierungsphase

In der abschließenden Phase wird der erstellte Prototyp auf die Umsetzbarkeit der in der Analyse- und Synthesephase identifizierten Nutzenhebel überprüft. Diese Evaluierung erfolgt durch Feldbeobachtungen und Experteninterviews. (Vössner und Lichtenegger 2011, Vorraber et al. 2015)

2 Praxisbeispiel

Im folgenden Beispiel wird das bisher beschriebene Vorgehensmodell auf eine Problemstellung aus der Praxis angewandt. Das Beispiel ist aus Wolfsgruber und Lichtenegger (2015) übernommen und dort im Detail beschrieben.

2.1 Allgemeines

Mit den Visionen von Industrie 4.0 eng in Verbindung steht der mittlerweile Jahrzehnte alte Traum von einer digitalen, papierlosen Produktion. Das Aufkommen numerisch gesteuerter Werkzeugmaschinen (NC) in den 1940er Jahren bzw. des Computer Integrated Manufacturing (CIM) Konzepts in den späten 1970er Jahren und von leistungsfähigen Enterprise Resource Planning Systemen (ERP) in den 1990er Jahren hat viele Unternehmen ermutigt, frühzeitig diesen Traum in die Realität umsetzen zu wollen, was allerdings bisher nur in den seltensten Fällen durchgängig gelungen ist. Vielmehr führte die ständig steigende Verfügbarkeit von Daten sowie die durchaus wünschenswerte Professionalisierung von Mitarbeitern im Bereich des Einsatzes von Standardsoftwareprodukten (wie beispielsweise Tabellenkalkulationsprogramme) vor allem im operativen Bereich von Unternehmen zu einer Fülle an heterogenen Lösungen. Hinzu kommt das seit Jahren breite und immer noch steigende Angebot an mobilen Endgeräten, die in ihrer Fähigkeit zur Darstellung von komplexen Informationsinhalten kaum hinter einer professionellen Office-Umgebung zurück bleiben. Unter dem Schlagwort Enterprise Mobility lässt sich hier branchenübergreifend ein Trend zum Einsatz mobiler „Informationsknoten“ beobachten. Eine Umfrage im Rahmen der Studie „Produktionsarbeit



<http://www.springer.com/978-3-658-12096-2>

Industrial Engineering und Management
Beiträge des Techno-Ökonomie-Forums der TU Austria
Biedermann, H. (Hrsg.)
2016, X, 285 S. 66 Abb., Softcover
ISBN: 978-3-658-12096-2