

Enterprise Application Integration als Enabler flexibler Unternehmensarchitekturen

Stephan Aier und Marten Schönherr

Technische Universität Berlin, Sekr. FR 6-7, Franklinstr. 28/29, 10587 Berlin,
stephan.aier@tu-berlin.de, schoenherr@sysedv.tu-berlin.de,
WWW home page: <http://www.sysedv.tu-berlin.de/eai>

Zusammenfassung. Ein wesentliches Ziel aktueller IT-Infrastruktur-Projekte ist die nachhaltige Flexibilisierung von Unternehmensarchitekturen. Der Beitrag erörtert, beginnend mit dem Paradigma der Flexibilität von Unternehmensarchitekturen, Möglichkeiten der integrierten Modularisierung von Organisations- und IT-Strukturen im Rahmen eines integrierten Architekturmanagements.

1 Problemstellung – Notwendigkeit flexibler Unternehmensarchitekturen

Das System Unternehmung ist den vielschichtigen Veränderungen seiner Umwelt ausgesetzt. Die drei wesentlichen Umweltdimensionen sind *Komplexität*, *Dynamik* und *Abhängigkeit* [1]. Insbesondere die Dimensionen Komplexität und Dynamik [2] verstärkt durch aktuelle Entwicklungen wie Globalisierung und Technisierung [3], stellen Unternehmungen vor Entscheidungs- und Handlungsprobleme zu deren Lösung sie grundsätzlich zwei Möglichkeiten der Reaktion haben [4]:

1. Sie können Maßnahmen zur Einwirkung auf die Umwelt mit dem Ziel ergreifen, Dynamik und Komplexität durch Verringerung der Abhängigkeiten zu reduzieren.
2. Oder sie können die Anpassungsfähigkeit der Unternehmung erhöhen.

Im vorliegenden Beitrag wird der zweite Punkt – die Erhöhung der Anpassungsfähigkeit – weiter untersucht. Dazu werden nach der Charakterisierung von Flexibilität als Gestaltungsziel die Architekturkomponenten Organisation und IT zueinander in Bezug gesetzt, um dann einen Ansatz zur Modularisierung der Organisations- und IT-Architektur im Rahmen von Enterprise Application Integration (EAI) darzulegen.

2 Flexibilität als Gestaltungsziel

Die Erkenntnis, dass „Unternehmenswandel von einem in größeren zeitlichen Abständen zu organisierenden Ereignis zu einem Dauerzustand geworden ist bzw. werden muss“ [5], ist heute weithin akzeptiert. Die Aufgabe besteht nun darin, Unternehmungen für diesen Wandel zu flexibilisieren.

KRÜGER differenziert den Wandel von Unternehmungen nach Wandlungsbedarf, Wandlungsbereitschaft und Wandlungsfähigkeit. [5] Die *Wandlungsfähigkeit* stellt den Kern der Betrachtungen dieses Beitrag dar.

Um mit intern und extern ausgelösten Wandlungsbedarfen effektiv umzugehen, sind in den Unternehmungen entsprechende Strukturen und organisatorische Instanzen zur Institutionalisierung von Wandel zu schaffen.¹

Neben institutionellen Maßnahmen werden im Folgenden strukturelle Maßnahmen zur Erhöhung der Wandlungsfähigkeit – Maßnahmen zur Flexibilisierung – diskutiert. Der Begriff der *Flexibilität* entstammt dem Lateinischen und bedeutet Veränderbarkeit, Beweglichkeit oder Biegsamkeit. Ein System ist dann flexibel, wenn einem Wandlungsbedarf ein in angemessener Zeit aktivierbares Wandlungspotenzial im System gegenübersteht [4,9]. Der Wandlungsbedarf enthält eine sachliche und eine zeitliche Dimension. Die zeitliche Dimension fordert zum einen das Reaktionsvermögen einer Unternehmung als auch die Fähigkeit antizipativer Anpassung. HILL/FEHLBAUM/ULRICH bezeichnen dies als Produktivität zweiter Ordnung [10]. KIESER/KUBICEK identifizieren folgenden Ausprägungen der Struktur von Unternehmungen als geeignet, deren Flexibilität zu erhöhen [4]:

- geringe Spezialisierung auf Stellen und Abteilungsebene
- starke Dezentralisierung
- flache Hierarchien
- Minimierung der Stärke zentraler unterstützender Abteilungen (Stäbe)
- einfache, d. h. keine umfassenderen Matrixstrukturen

Die genannten Ausprägungen zielen auf eine Entkopplung der Strukturen und Prozesse durch eine Reduktion von Schnittstellen. Im Folgenden werden diese Gedanken weiterentwickelt, um durch Modularisierung auf verschiedenen Ebenen der Unternehmung, Entkopplung und flexible Rekonfiguration von Architekturen zu ermöglichen.

3 Organisation und IT

Heute ist es nicht ausreichend ausschließlich formale organisatorische Gestaltungsaspekte bei der Flexibilisierung von Unternehmungen zu betrachten. Vielmehr ist es notwendig auch die informations- und kommunikationstechnologischen Aspekte zu berücksichtigen und beide in einem integrierten Architektur-Ansatz zu verbinden. Umgekehrt macht es wenig Sinn, Informationssysteme einzuführen oder zu ändern, ohne die Wechselwirkungen mit den Aufgaben und Prozessen die sie unterstützen zu berücksichtigen [11,12]. EAI integriert hier nicht nur IT-Systeme sondern ist vor allem der Auslöser, die Domänen Organisation und IT integriert zu gestalten und gemeinsam weiterzuentwickeln.

In der Wissenschaft hat diese Diskussion um die gegenseitigen Abhängigkeiten von IT und Organisation der Unternehmung eine lange Tradition, in der

¹ Bspw. kontinuierliche Prozessverbesserungen [6] durch ein betriebliches Vorschlagswesen [7] oder Kaizen [8].

sowohl die Technologie (*technological imperative*), die Organisation (*organizational imperative*) als auch komplexe Wechselwirkungen zwischen beiden (*emergent perspective*) [13] als treibende Faktoren beschrieben wurden [14,15,16,17].² FRESSE folgend, wird hier nicht davon ausgegangen, dass Informationstechnologie und Unternehmensorganisation in einem deterministischen Ursache-Wirkungs-Zusammenhang stehen [3]. Vielmehr wird angenommen, dass IT eine den Gestaltungsspielraum des Organisators erweiternde Option darstellt [19].

4 Modulare Unternehmensarchitekturen

Unter einer Unternehmensarchitektur wird hier das Zusammenwirken technologischer, organisatorischer und psychosozialer Aspekte bei der Entwicklung und Nutzung von betrieblichen soziotechnischen Informationssystemen verstanden [20]. Im Folgenden werden vor allem die technologischen und organisatorischen Domänen als IT- und Organisationsarchitekturen betrachtet. Ein aktuell diskutiertes Mittel zur Flexibilisierung von IT- und Organisationsarchitekturen ist die Modularisierung. Unter Zuhilfenahme der Systemperspektive werden nachfolgend allgemeine Charakteristika von Modulen beschrieben, welche der Organisations- und IT-Literatur entnommen sind und hier für beide Bereiche gleichrangig gelten sollen.

Ein *Modul* besteht aus zwei Teilen, der *Modulschnittstelle* und dem *Modulrumpf*. Die Modulschnittstelle enthält dabei die Spezifikation der Leistungen des Moduls die für seine Umwelt zur „Benutzung“ notwendig sind. Der Modulrumpf implementiert die spezifizierten Leistungen. *Modularisierung* bedeutet die Strukturierung eines Systems in kleine, teilautonome Subsysteme. Die Komplexitätsreduktion ergibt sich dabei aus der Subsystembildung innerhalb des Systems Unternehmung [21]. Die Subsystembildung wirkt komplexitätsreduzierend, da sie zum einen die subsysteminterne Komplexität im Sinne der Kapselung vor der Subsystemumwelt verbirgt, zum anderen durch die Reduktion auf wenige bekannte Schnittstellen eine Entkopplung der Subsysteme bewirkt.

Die Flexibilisierung der Architektur ergibt sich durch die nun leichtere Rekonfigurationsmöglichkeit der entkoppelten Module. Komplexitätsreduktion kann somit als Voraussetzung für die Flexibilisierung angesehen werden. Bei der Modularisierung sind folgende Gestaltungsziele zu beachten [22,23]:

- Abstraktion von der Implementierung
- Kapselung im Sinne des Verbergens der internen Funktionsweise
- Austauschbarkeit
- Wiederverwendbarkeit
- Zeitliche Gültigkeit
- Orthogonalität (im Sinne von „sich gegenseitig nicht beeinflussend“)
- Überschneidungsfreiheit
- Vollständigkeit (Abgeschlossenheit)

² Eine Übersicht über Studien sowie die Analyse von Ergebnissen finden sich in [18].

- Wohldefinierte Schnittstellen
- Schnittstellenminimalität³
- Generizität

Durch die Anwendung dieser Prinzipien werden Module geschaffen, die potenziell kombinierbar, wiederverwendbar und gut änderbar sind [22]. Im Folgenden wird nun dargestellt, welche Ansätze sich bei der Modularisierung der Organisation und der IT ergeben.

4.1 Modularisierung der Organisation

Die Modularisierung der Organisation, wird im Folgenden wird auf der Ebene der Gesamtunternehmung als Makroebene und der Ebene der Geschäftsprozesse als Mikroebene betrachtet. Mit der Definition einer Makro- und einer Mikroebene werden zwei Ziele verfolgt:

1. Festigung und zugleich Flexibilisierung der Strukturen
2. Definition überschaubarer und managebarer Module

Die Makroebene hat eine stark strukturierende Wirkung auf die Organisation. Die Module der Makroebene, die sogenannten Makromodule, sollen über einen längeren Zeitraum stabil sein. Die Makroebene bildet damit ein Bezugssystem als Grundlage für die Gestaltung der Module der Mikroebene innerhalb der jeweiligen Makromodule. Die Module der Mikroebene hingegen sollen die Dynamik der wechselnden Prozessanforderungen durch entsprechende Rekonfiguration widerspiegeln. Ziel ist es also, eine bezuggebende, stabile, organisationsinvariante Ebene – die Makroebene – und eine dynamische, flexibel konfigurierbare Ebene – die Mikroebene – zu bilden. Nachfolgend werden die Charakteristika der Makro- und der Mikroebene dargestellt.

Modularisierung auf Makroebene Um eine direkte Abbildung strategieorientierter Organisationsstrukturen auf die IT-Infrastruktur zu erreichen, werden mehr als die klassischen ablauf- bzw. aufbauorganisatorischen Sichtweisen benötigt. Das Ziel der Definition der Makro-Module besteht in der Betrachtung der Gesamtorganisation mit Hilfe geeigneter Kriterien, um genau diesen Schritt ableiten zu können. Im Folgenden werden in Erweiterung der o. g. allgemeinen Kriterien spezielle Kriterien zur Bildung von Makromodulen genannt:

- Ähnlichkeit der Prozesse innerhalb eines Moduls
- Ähnlichkeit des notwendigen Prozess-Know-hows
- Minimierung der externen Abhängigkeiten
- Maximierung der internen Kohärenz
- Unabhängigkeit von operativen Prozessanpassungen

³ In Bezug zur Systemtheorie bedeutet Schnittstellenminimalität die Reduktion der Relationen des Subsystems zu seiner Umwelt und somit Maximierung – im Sinne von Verschiebung – der Relationen in den Subsystemen.

- Branchenspezifische Best Practices für die Abgrenzung der Module

Die aufgezählten Kriterien werden z. T. im Bereich der Componentware verwendet, um die Granularität von Softwarekomponenten zu bestimmen. Die analoge Verwendung auf Makroebene führt zumindest zu ähnlichen Methoden, mit denen später ein Match zwischen Makro-, Mikro- und Softwarearchitekturebene erreicht werden kann. Die definierten Makro-Module stellen keine Handlungsanweisung für eine Reorganisation dar. Sie verfolgen vielmehr das Ziel, eine besondere Sichtweise auf die Organisation zu gewähren. Diese Sichtweise unterstützt die Modularisierung der jeweiligen IT-Systeme, die die Prozesse eines Makro-Moduls unterstützen. In Abbildung 1 wird anhand einer klassischen Matrixorganisation die sich nach Sparten und Funktionsbereichen aufstellt gezeigt, wie Makro-Module durch die oben aufgeführten Kriterien gebildet werden. Dabei stellen die farbig gleich hervorgehobenen Bereiche jeweils ein Makro-Modul dar. Beispielsweise wurden innerhalb der verschiedenen Vertriebsorganisationen zwei Bereiche als Makro-Modul definiert. Grund dafür könnte sein, dass sich die Prozesse in beiden Bereichen stark ähneln und ähnlicher Informationen zur Ausführung bedürfen.

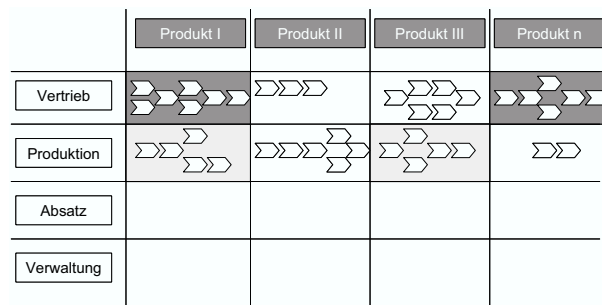


Abb. 1. Definition von Makro-Modulen in einer Matrixorganisation

Auf der Ebene der Makromodule sollen Strukturähnlichkeiten die Grundlage für die Definition bilden. Es können situativ weitere Kriterien zu den oben genannten bestimmt werden. Prozesse laufen i. d. R. modulübergreifend, d. h. es wird keine Abgrenzung nach Anfang bzw. Ende von Prozessketten vorgenommen.⁴

Modularisierung auf Mikroebene Innerhalb der definierten Makromodule sollen jetzt die Strukturähnlichkeiten genauer betrachtet werden. Für die IT-Systemwelt sind vor allem die innerhalb der Module vorhandenen Geschäftsprozesse relevant. Hier müssen alle Prozessschritte verglichen und die

⁴ Vgl. zu praktischen Umsetzungen ähnlicher Konzepte bei der Credit Suisse [24], bei der HypoVereinsbank [25].

strukturgleichen identifiziert werden. Ein Geschäftsprozess setzt sich prinzipiell aus verschiedenen Schritten zusammen. Die Wahrscheinlichkeit, dass sich in einer Menge ähnlicher Prozessketten identische Schritte befinden, ist sehr hoch. Selbst wenn sich die Teilprozesse nur sehr leicht unterscheiden, kann betrachtet werden, ob es im Sinne einer Komplexitätsreduktion möglich ist, die tatsächlichen Geschäftsprozesse so zu modifizieren, dass sie möglichst viele identische Teilprozesse verwenden.

GERYBADZE leitet aus den Methoden der modularen Produktgestaltung Kriterien für die Bestimmung des Grades der Modularisierbarkeit von Prozessen ab. Eine gute Modularisierbarkeit ergibt sich nach GERYBADZE, wenn [26]:

- sich für jede Aktivität eine genau definierte Funktion des Gesamtsystems definieren lässt,
- die Qualität des Ergebnisses dieser Aktivität genau bestimmt werden kann,
- für den jeweiligen Output Preise bzw. Verrechnungspreise bestimmt werden können und
- die Schnittstellen sehr genau definiert werden können.

Neben der generellen Frage der Modularisierbarkeit ist die Bestimmung der *optimalen Modulgröße* entscheidend für die erfolgreiche Umsetzung der Modularisierung auf Prozessebene. Zu große Module können die erhoffte Flexibilität der Organisation verhindern, da die Modularisierung ja gerade kleine, flexibel integrierbare Fragmente schaffen soll, die beweglicher sind als eine gesamte monolithische Struktur. Zu große Module können durch ihre Komplexität die Beherrschbarkeit der Prozesse innerhalb des Moduls einschränken. Andererseits bedeuten zu viele kleine Module eine stärkere Arbeitsteilung und Zergliederung. Je kleiner die Module, desto spezialisierter müssen naturgemäß die Prozesse innerhalb eines Moduls sein. Um allgemeine Kriterien zur Schaffung einer optimalen Modulgröße zu definieren, müssen die Charakteristika der Modularisierung betrachtet werden. Die zu beachtende *Mindestgröße eines Moduls* ergibt sich aus den Aktivitäten für ein klar definierbares (Zwischen-)Produkt. Die *maximale Modulgröße* wird durch die Beherrschbarkeit der Komplexität innerhalb eines Moduls bestimmt. Die Komplexität darf nicht so groß sein, dass die Verantwortlichen das Modul nicht mehr steuern können.

Realisierungsprinzipien Die Realisierung einer modularen Organisation ist stark von den situativen Rahmenbedingungen abhängig. Wesentliche Einflussfaktoren sind die *Größe* der Organisation, ihr *Angebotsprogramm*, ihre *Internationalisierung* und *Kultur*, die bestehende *Strukturierung* sowie die *Organisationsumwelt* [4]. Trotz dieser situativen Abhängigkeit lassen sich einige grundsätzliche Realisierungsprinzipien finden:

Das Hauptprinzip ist, *kleine Organisationseinheiten* zu bilden. Sie müssen groß genug sein, zusammengehörige Prozesse zu einem Objekt (z. B. Produkt, Produktgruppe etc.) zu umfassen. Sie dürfen aber den Umfang und die Komplexität betreffend die Aufnahmegrenzen und Problemlösungskapazitäten des Menschen nicht überschreiten.

Die *Prozessorientierung* kommt durch die Forderung zum Ausdruck, die Module durchgängig an den Prozessen zur Erstellung von Leistungen zu orientieren. Die Prozessorientierung zeigt, dass die Modularisierung nicht auf eine funktionsorientierte, sondern auf eine objektorientierte Strukturierung abzielt.

In engem Zusammenhang mit der Prozessorientierung steht die *Kunden-* bzw. *Marktorientierung*. Hiernach sollen sich alle Wertschöpfungsaktivitäten der Module an den externen und/oder internen Kundenanforderungen orientieren. Dies resultiert aus dem Hauptziel, der flexiblen Anpassung an die Erfordernisse des Marktes.

Integriertheit der Aufgaben: Die Prozesse in einem Modul sollen weitestgehend ihrer Art nach zusammengehören, um die Abgeschlossenheit der in einem Modul konzentrierten Prozesse zu gewährleisten.

Nicht-hierarchische Koordinationsformen: Die Koordination autonomer Handlungseinheiten, die nicht mehr in einem hierarchischem Verhältnis zueinander stehen, erfordert neue Formen der Zusammenarbeit, welche nicht mehr auf Fremdsteuerung, sondern auf Selbststeuerung basieren.

4.2 Modularisierung der IT

Ausgehend von der dargelegten modularen Organisationsarchitektur sind adäquate IT-Architekturen zu definieren. Die abgestimmte Gestaltung von Strategien, Prozessen und technischen Infrastrukturen ist ein klassisches Thema der Wirtschaftsinformatik [27]. Dazu wurden verschiedene Architekturkonzepte entwickelt, die eine Homologie zwischen Organisation und IT zum Ziel haben [28,29,30]. Diese Konzepte beruhen jedoch auf der Annahme, über Implementierung und Neueinführung von IT zu sprechen. Um möglichst nachhaltige Architekturen zu schaffen, sollte das Paradigma der *strukturellen Analogie* zwischen Organisation und IT so weit wie möglich aufrecht erhalten werden [20].

EAI vs. Service Oriented Architecture (SOA) Serviceorientierung und Systemintegration wurden viele Jahre unabhängig voneinander diskutiert. EAI-Plattformen integrieren heterogene Systemlandschaften zentral auf Prozess-, Methoden- und Datenebene.⁵ Je stärker die Integrationsprojekte allerdings mit objektorientierten Verfahren implementiert werden, desto größer erscheint die Nähe zur Serviceorientierung. Die dienstorientierte Anwendungsintegration ist mit der Integration auf Interface- und Methodenlevel vergleichbar. Sie stellt eine alternative Sichtweise dar, die versucht, „gewrappte“ Module aus Altanwendungen und bereits serviceorientiert implementierte Neuanwendungen innerhalb einer SOA zu integrieren. Eines der wichtigsten Ziele der Serviceorientierung ist die Wiederverwendung bestehender Komponenten. Voraussetzung dafür ist ein zentrales Servicemanagement, dass Funktionen wie Service-Life-Cycle, Serviceverteilung und die Versionierung bereitstellen muss [32].

⁵ Vgl. zu den Begriffen User Interface Integration, Data Level Integration, Application Interface Integration, Method Integration, Service based Integration sowie Process Integration [31].

Da die meisten Entwicklungswerkzeuge inzwischen eine Serviceorientierung unterstützen, werden Anwendungen in Zukunft sehr viel einfacher als Services entwickelbar sein. Altanwendungen sind jedoch oft schwer in als Service abgrenzbare Module aufzuspalten. Neben Entwicklungsumgebungen die eine Neuimplementierung mit Servicecharakter aktiv unterstützten bieten einige Integrations-Suiten inzwischen ebenfalls Tools, die das Wrappen von Altanwendungen in Services erleichtern. Die Kosten dieser initialen Umwandlung können hoch sein und oft widerspricht der Eingriff in den Quellcode den strategischen Richtlinien der IT-Verantwortlichen, die vor allem monolithische Legacies unberührt lassen wollen [33]. Bei der Betrachtung einer Gesamt-IT-Architektur muss von zu integrierenden Altanwendungen und geplanten Neuimplementierungen ausgegangen werden. Sowohl der zentrale EAI-Ansatz als auch die dezentrale SOA stellen Methoden zu Verfügung, Lösungen für das beschriebene Spannungsfeld zu implementieren. EAI und SOA sind daher harmonisierende Bestandteile einer Gesamt-IT-Architektur.

Problemfelder der technischen Modularisierung Im Kontext der Systemintegration muss allerdings beachtet werden, dass eines der primären Ziele der EAI, die Reduktion der Punkt-zu-Punkt-Integrationsszenarien nicht durch eine ähnlich komplexe SOA ersetzt wird. Zwei Hauptprobleme bei der Anbindung von Altanwendungen sind standardisierte Schnittstellen und die Granularität der Services. Bei der serviceorientierten Integration bestehender Anwendungen kommt es auf die Granularität der als Services verpackten Funktionen an. So macht es Sinn, diejenigen Funktionen als Services zugänglich zu machen, die allgemein benötigt und somit wieder verwendet werden sollten. Sie sollten eine komplette Arbeitseinheit ausführen sowie in ihrer Funktion und im Ergebnis gut beschreibbar sein [34]. Mit größter Wahrscheinlichkeit liegen die Altanwendungen jedoch monolithisch bzw. in falscher Granularität vor [35]. Das Problem der Granularität stellt sich auch für Anwendungen, die in Programmiersprachen geschrieben wurden, die mehr oder weniger direkt unter Anwendung spezieller Tools in Webservices umgewandelt werden können. Obwohl sich die Verzeichnisse, die Services beschreiben aus Definitionen von Schnittstellen bereits bestehender Technologien verteilter Softwaresysteme (Java-Klassen, JavaBeans, CORBA-Objekte, Visual Basic-Klassen, C#) erzeugen lassen, sind diese zumeist ebenfalls nicht in der für Services geeigneten Granularität definiert [36].

5 Fazit

Viele Publikationen mit technischem Fokus diskutieren derzeit dezentrale Architekturen zur Integration komplexer IT-Infrastrukturen. Unternehmen greifen das Thema der Wiederverwendung erneut auf, obwohl es vor einigen Jahren unter dem Namen der Business Process Repositories, die konzipiert wurden, um rekonfigurierbare Geschäftsprozesse zu implementieren, fast völlig aus der Fachdiskussion verschwunden waren. Performantere Plattformen begründen teilweise diese unerwartete Renaissance. Das Thema EAI trägt ebenfalls zur aktuellen

Diskussion bei, die auf Unternehmensarchitekturebene geführt wird. Dabei geht es im ersten Schritt um die fachliche Definition der Module (bzw. Services), nachfolgend werden erst Technologien zur Implementierung eine Rolle spielen. In diesem Sinne konzentriert sich der Beitrag vor allem auf die Modularisierung auf fachlicher Ebene und deren Überführung in eine technische Ebene. Details der technischen Implementierung stehen nicht im Vordergrund.

Literatur

1. Jurkovich, R.: A core typology of organizational environments. *Administrative Science Quarterly* **19** (1974) S. 380–394
2. Krystek, U.: Vertrauen als Basis erfolgreicher strategischer Unternehmensführung. In: D. Hahn/B. Taylor, *Strategische Unternehmensplanung – Strategische Unternehmensführung*. 8 edn. Physica, Heidelberg (1999) S. 266–288
3. Frese, E.: *Grundlagen der Organisation: Konzept – Prinzipien – Strukturen*. 8 edn. Gabler, Wiesbaden (2000)
4. Kieser, A., Kubicek, H.: *Organisation*. 3 edn. De Gruyter, Berlin, New York (1992)
5. Krüger, W.: Management permanenten Wandels. In: H. Glaser, E.F. Schröder, A. v. Werder, eds. *Organisation im Wandel der Märkte*. Gabler, Wiesbaden (1998) S. 227–249
6. Österle, H.: *Business Engineering: Prozeß- und Systementwicklung*. 2 edn. Volume 1. Springer, Berlin et al. (1995)
7. Thom, N.: *Betriebliches Vorschlagswesen: ein Instrument der Betriebsführung und des Verbesserungsmanagements*. 5 edn. Lang, Berlin et al. (1996)
8. Imai, M.: *Kaizen*. Ullstein, Frankfurt a. M., Berlin (1993)
9. Gronau, N.: Modellierung von Flexibilität in Architekturen industrieller Informationssysteme. In: H. Schmidt, *Modellierung betrieblicher Informationssysteme*. Proceedings der MobIS-Fachtagung, Siegen (2000) S. 125–145
10. Hill, W., Fehlbaum, R., Ulrich, P.: *Organisationslehre 1: Ziele, Instrumente und Bedingungen der Organisation sozialer Systeme*. 5 edn. Haupt, Bern, Stuttgart, Wien (1994)
11. Kaib, M.: *Enterprise Application Integration: Grundlagen, Integrationsprodukte, Anwendungsbeispiele*. DUV, Wiesbaden (2002)
12. Derszteler, G.: *Prozessmanagement auf Basis von Workflow-Systemen*. Josef Eul, Lohmar, Köln (2000)
13. Markus, M., Robey, D.: Information technology and organizational change: Causal structure in theory and research. *Management Science* **34** (1988) S. 583–589
14. Leavitt, H., Whisler, T.: Management in the 1980s: New information flows cut new organization flows. *Harvard Business Review* **36** (1958) S. 41–48
15. Applegate, L.M., Cash, J.I., Miles, D.Q.: Information technology and tomorrow's manager. *Harvard Business Review* **66** (1988) S. 128–136
16. Rockart, J.F., Short, J.E.: It in the 90's: Managing organizational independence. *Sloan Management Review* (1989) S. 7–17
17. Burgfeld, B.: *Organisationstheorie und Informationstechnologie*. DUV, Wiesbaden (1998)
18. Lewin, A.Y., Hunter, S.D.: Information Technology & Organizational Design: A Longitudinal Study of Information Technology Implementations in the U.S. Retailing Industrie, 1980-1996. In: H. Glaser, E.F. Schröder, A. v. Werder, eds. *Organisation im Wandel der Märkte*. Gabler, Wiesbaden (1998) S. 251–286

19. Frese, E.: Theorie der Organisationsgestaltung und netzbasierte Kommunikations-effekte. In: E. Frese, H. Stöber, eds. E-Organisation. Gabler, Wiesbaden (2002) S. 191–241
20. Gronau, N.: Wandlungsfähige Informationssystemarchitekturen – Nachhaltigkeit bei organisatorischem Wandel. Gito, Berlin (2003)
21. Krcal, H.C.: Systemtheoretischer Metaansatz für den Umgang mit Komplexität und Nachhaltigkeit. In: R. Leisten and H.-C. Krcal, Nachhaltige Unternehmensführung – Systemperspektiven. Gabler, Wiesbaden (2003) S. 3–30
22. Rombach, D.: Software nach dem baukastenprinzip. Fraunhofer Magazin (2003) S. 30–31
23. Lang, K.: Gestaltung von Geschäftsprozessen mit Referenzprozeßbausteinen. Gabler, Wiesbaden (1997)
24. Hagen, C.: Integrationsarchitektur der Credit Suisse. In: S. Aier, M. Schönherr, eds., Enterprise Application Integration – Flexibilisierung komplexer Unternehmensarchitekturen. Gito, Berlin (2003) S. 61–83
25. Weber, H.-W.: Wege zur Entwicklung von Wertschöpfungsnetzen: Konzeption einer modularen IT-Architektur. <http://www.sysedv.tu-berlin.de/eai/eaitag200302> Vortrag auf dem EAI-Expertentag, Berlin (2003)
26. Gerybadze, A.: Strategisches Management und dynamische Konfiguration der Unternehmens-Umwelt-Beziehungen. In: R. Leisten and H.-C. Krcal, Nachhaltige Unternehmensführung – Systemperspektiven. Gabler, Wiesbaden (2003) S. 83–100
27. Wall, F.: Organisation und betriebliche Informationssysteme – Elemente einer Konstruktionslehre. Gabler, Wiesbaden (1996)
28. Krcmar, H.: Bedeutung und ziele von informationssystem-architekturen. Wirtschaftsinformatik **32** (1990) S. 395–402
29. Pohland, S.: Globale Unternehmensarchitekturen – Methode zur Verteilung von Informationssystemen. Weißensee-Verlag, Berlin (2000)
30. Scheer, A.W.: Architektur integrierter Informationssysteme. Springer, Berlin et al. (1991)
31. Aier, S., Schönherr, M.: Flexibilisierung von Organisations- und IT-Architekturen durch EAI. In: S. Aier, M. Schönherr, eds., Enterprise Application Integration – Flexibilisierung komplexer Unternehmensarchitekturen. Gito, Berlin (2003) S. 1–59
32. Lublinsky, B., Farrell, M.: 10 misconceptions about web services. EAI Journal (2003) S. 30–33
33. Apicella, M.: Side by side in perfect harmony? InfoWorld (2002)
34. Narsu, U., Murphy, P.: Web services adoption outlook improves. (2003) Giga Information Group.
35. Erlikh, L.: Integrating legacy into extended enterprise: Using web services. (2003) Relativity Technologies.
36. Newcomer, E.: Understanding Web services: XML, WSDL, SOAP and UDDI. Addison-Wesley Longman, Amsterdam (2002)