

Reihe: Telekommunikation @ Mediendienste · Band 14

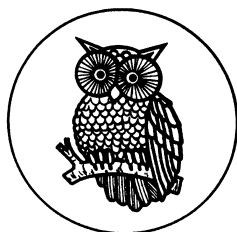
Herausgegeben von Prof. Dr. Dr. h. c. Norbert Szyperski, Köln, Prof. Dr. Udo Winand, Kassel, Prof. Dr. Dietrich Seibt, Köln, Prof. Dr. Rainer Kuhlen, Konstanz, Dr. Rudolf Pospischil, Brüssel, Prof. Dr. Claudia Löbbcke, Köln, und Prof. Dr. Christoph Zacharias, Köln

PD Dr.-Ing. habil. Martin Engelien
Dipl.-Inf. Jens Homann (Hrsg.)

Virtuelle Organisation und Neue Medien 2002

Workshop GeNeMe2002
Gemeinschaften in Neuen Medien

TU Dresden, 26. und 27. September 2002



JOSEF EUL VERLAG
Lohmar · Köln

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Virtuelle Organisation und Neue Medien 2002 / Workshop GeNeMe 2002 – Gemeinschaften in Neuen Medien – TU Dresden, 26. und 27. September 2002. Hrsg.: Martin Engeliien ; Jens Homann. – Lohmar ; Köln : Eul, 2002

(Reihe: Telekommunikation und Medienwirtschaft ; Bd. 14)

ISBN 3-89936-007-9

© 2002

Josef Eul Verlag GmbH

Brandsberg 6

53797 Lohmar

Tel.: 0 22 05 / 90 10 6-6

Fax: 0 22 05 / 90 10 6-88

<http://www.eul-verlag.de>

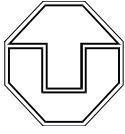
info@eul-verlag.de

Alle Rechte vorbehalten

Printed in Germany

Druck: RSP Köln

Bei der Herstellung unserer Bücher möchten wir die Umwelt schonen. Dieses Buch ist daher auf säurefreiem, 100% chlorfrei gebleichtem, alterungsbeständigem Papier nach DIN 6738 gedruckt.



Technische Universität Dresden
Fakultät Informatik • Institut für Angewandte Informatik
Privat-Dozentur Angewandte Informatik

PD Dr.-Ing. habil. Martin Engelen

Dipl.-Inf. Jens Homann

(Hrsg.)

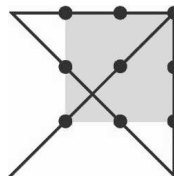


an der

Fakultät Informatik der Technischen Universität Dresden

in Zusammenarbeit mit der
Gesellschaft für Informatik e.V.,
GI-Regionalgruppe Dresden

gefördert von der Klaus Tschira Stiftung
gemeinnützige Gesellschaft mit beschränkter Haftung



am 26. und 27. September 2002

in Dresden

<http://pdai.inf.tu-dresden.de/geneme>

Kontakt: Thomas Müller (geneme@pdai.inf.tu-dresden.de)

D.7. Anwendungsintegration entlang der Geschäftsprozesse mittels Workflow-Management-System "KontextFlow"

Martin Halatchev

Technische Universität Dresden

Abstract

Die Entstehung und die Existenz virtueller Unternehmen ist in der Regel zweckgebunden, d.h. das Verfolgen eines gemeinsamen Zieles ist eine notwendige Voraussetzung. Wichtige organisatorische und technische Maßnahme bleibt dabei die Verknüpfung der Geschäftsprozesse einzelner organisatorischer Einheiten in einem übergeordneten Gebilde. Jede Einheit (gewöhnliches Unternehmen bzw. Teile davon) organisiert und automatisiert ihre Vorgänge nach eigenen Regeln unter Einsatz verschiedener Systeme. Würde man Vorgänge mehrerer Organisationen miteinander verknüpfen, so entsteht zwangsläufig der Bedarf zur Integration vorhandener Anwendungen. Dieses ist ein klassisches Problem im Rahmen der Einführung von E-Procurement-, Supply-Chain- und anderer Systeme. In diesem Artikel wird nicht diskutiert, wie und mit welchen Mitteln übergeordnete Geschäftsprozesse zustanden kommen können, sondern man beschränkt sich auf den Teilaspekt „Anwendungsintegration“.

Eines der Forschungsfelder der Privatdozentur PDAI ist die Arbeitsorganisation und Steuerung von Produktions- und Office-Prozessen. Im Zuge der Forschungsarbeiten wurde das Workflow-Management-System "KontextFlow" entwickelt, das eine Steuerung von Vorgängen menschlicher Akteure ermöglicht. Das Systemkonzept erlaubt allerdings das Einbeziehen weiterer Akteure-Arten (zum Beispiel: fremde Anwendungen). Um dieses effektiv umzusetzen, sind spezielle SW-Bausteine notwendig, die die "Kopplung" der Akteure an das "KontextFlow" ermöglichen. Diese Bausteine werden im Umfeld der PDAI als "Application Gateways" bezeichnet und werden in diesem Artikel diskutiert.

1. Einführung in die Integrationsproblematik

1.1 Allgemeine Möglichkeiten zur Anwendungsintegration

Generell kann man zwischen Ablauf- und Datenintegration unterscheiden. Zwei Anwendungen sind dann isoliert, wenn keine „Ohne-Medien-Bruch“-Integration stattfindet.

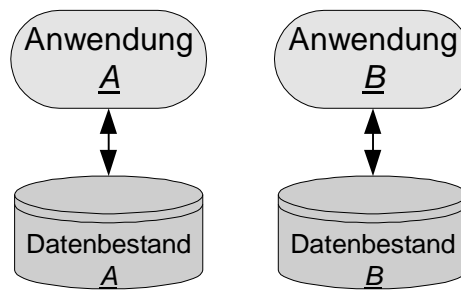


Abb. 8: Isolierte Anwendungen (Ausgang)

Eine Ablaufintegration erfolgt über die API's der jeweilige Anwendungen. Die gegenseitigen Zugriffe erfolgen, gemäß der API-Implementierung, meistens jedoch über Middleware (COM/DCOM, CORBA, RMI, Messaging, Web-Services etc.). Die eigenen Datenstände bleiben unberührt.

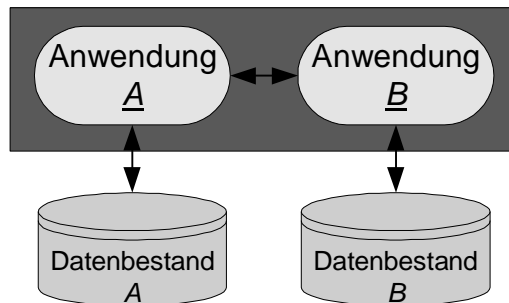


Abb. 9: Ablaufintegration

Das in der Praxis meist praktizierte Schema ist die Datenintegration. Die Anwendungen tauschen dabei Daten über eine Schnittstelle aus, die nicht unbedingt zu den Kernfunktionen des Systems zählt. Je nach Implementierung kann die Schnittstelle „on-line“ (direkter Datentransfer von/zu „fremden“ Anwendung möglich) oder „off-line“ (kein direkter Datentransfer von/zu „fremden“ Anwendung möglich -> Zwischenschritte notwendig) sein.

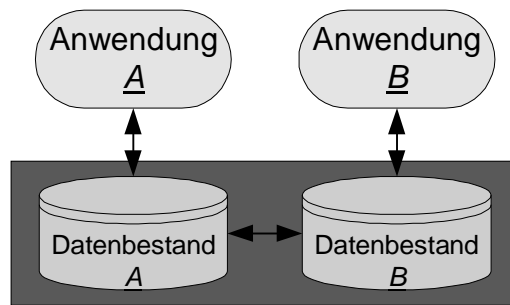


Abb. 10: Datenintegration

Nach Bedarf können auch Mischformen (Daten- und Ablaufintegration) realisiert werden.

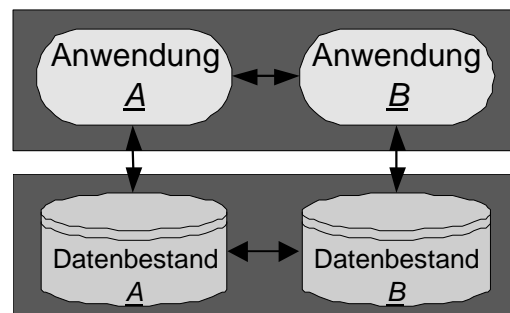


Abb. 11: Daten- und Ablaufintegration

1.2 Prinzipielle Möglichkeiten zur Realisierung der Datenintegration

Im Folgenden erfolgt eine Ausführung der "Black-Box"-, "Grey-Box"- und "White-Box"-Konzepte der Datenintegration.

Bei der "Black-Box"-Integration findet ein Abgleich der Datenbestände der zu integrierenden Anwendungen statt. Die Check-In-/Check-Out-Mechanismen sind Teil einer Integrationsanwendung bzw. eines Moduls.

Vorteile:

1. Die Legacy-Systeme bleiben in ihren Funktionalitäten unberührt.
2. Check-In/Check-Out-Mechanismen müssen nicht on-line erfolgen (Beispiel: Datenabgleich findet einmal in der Woche statt).

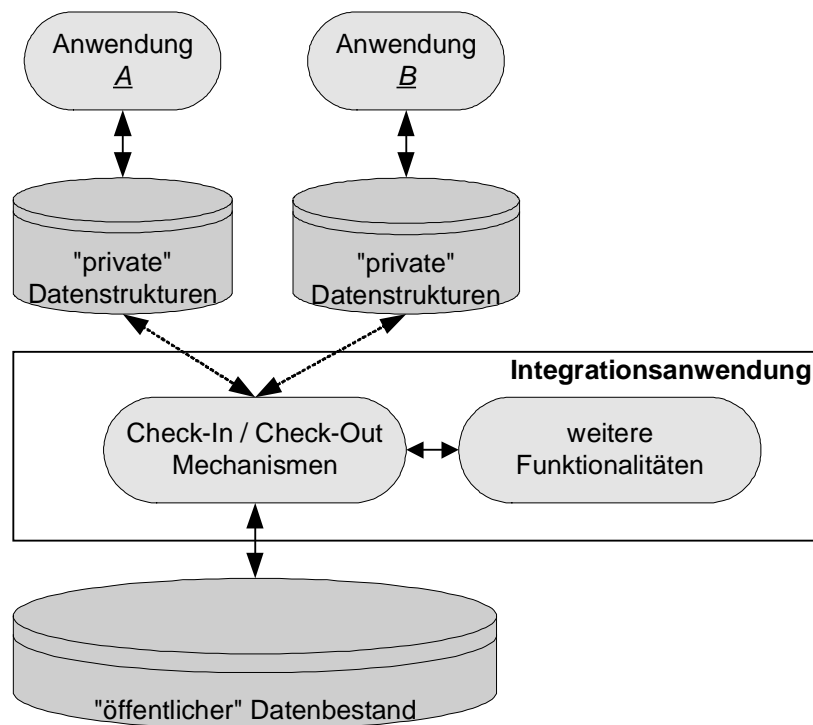


Abb. 12: Black-Box-Integration

Nachteile:

1. Mehraufwand bei der Realisierung der Integrationsanwendung (Implementierung der Check-In/Check-Out-Mechanismen).
2. Mehraufwand für die Legacy-Systeme – Zusatzmodule für die Zugriffe auf die Check-In/Check-Out-Mechanismen der Integrationsanwendung notwendig.

In dieser Variante verzichten alle Legacy-Systeme auf ihre eigenen (privaten) Datenbestände. Statt dessen werden alle Daten von der Integrationsanwendung verwaltet. Die Zugriffe erfolgen über eine einheitliche Datenverwaltungskomponente.

Vorteile:

1. Strukturelle und inhaltliche Dateninkonsistenzen ausgeschlossen.
2. Einheitlichkeit.

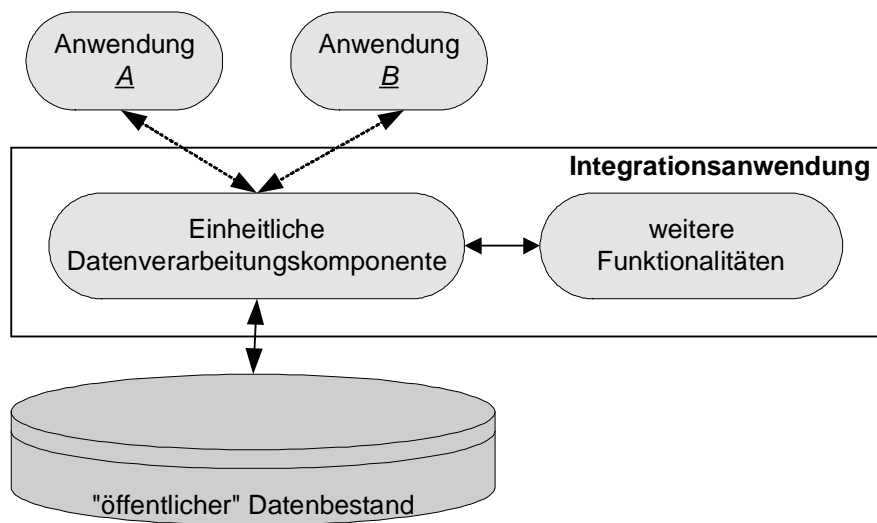


Abb. 13: Grey-Box-Integration

Nachteile:

1. Mehraufwand bei der Realisierung der Integrationsanwendung (einheitliche Datenverwaltungskomponente).
2. Mehraufwand für die Legacy-Systeme – Re-Implementierung der Datenzugriffsfunktionen.
3. Organisatorisch schwer durchzusetzen (wenn überhaupt).

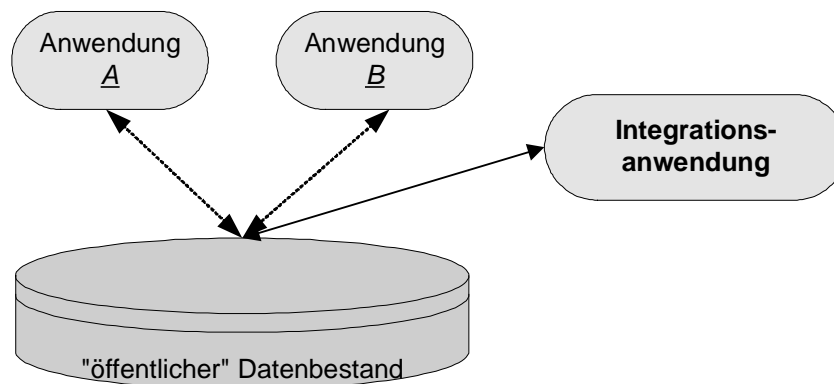


Abb. 14: White-Box-Integration

Ähnlich der „Grey-Box“-Integration verzichten auch hier alle Legacy-Systeme zu Gunsten der Integrationsanwendung auf ihre eigenen (privaten) Datenbestände. Jedes System passt seine Datenzugriffsmodule (Sub-Systeme, Schichten etc.) an die Anforderungen der neuen Datenquelle an.

Vorteile:

1. Strukturelle und inhaltliche Dateninkonsistenzen ausgeschlossen.
2. Einheitlichkeit.

Nachteile:

1. Mehraufwand für die Legacy-Systeme – Re-Implementierung der Datenzugriffsfunktionen.
2. Organisatorisch schwer durchzusetzen (wenn überhaupt).
3. Zugriffe aller Systeme über eigene Zugriffsmittel birgt viele Risiken.

2. Enterprise Application Integration (EAI)

Unternehmen sind häufig mit der Frage konfrontiert "Wie vereinbart man am besten die steigende Anzahl von Anwendungen mit dem bestehenden Zwang zur Gestaltung ganzheitlicher Prozesse?". In solchen Fällen bietet der Ansatz der Enterprise Application Integration (EAI) vielversprechende Hilfe. Die Anwendungsintegration vollzieht man meist durch den Austausch von Nachrichten über einen logischen Bus bzw. Hub, der diese umformatiert (übersetzt) und weiterleitet. Die auf EAI basierenden Lösungen bauen auf bekannten Middleware-Technologien wie Messaging, Transaktions-Engines, Web-Services, Komponentenmodelle (COM/DCOM), Kommunikationsstandards (CORBA) etc. auf. Neu in EAI ist die Vereinigung dieser Middleware-Techniken zu mächtigen, integrierten Produktplattformen, die auch als eine Brücke zu System-Management-Systemen verstanden werden können. Im Gegensatz zu früheren Integrationsversuchen, die meistens eine Punkt-zu-Punkt-Integration realisierten (vgl. EDI) und sich dabei auf Datenintegration beschränkten, ist EAI ein zentralistischer Ansatz, der eine Integration auf Daten-, Anwendungs- und Prozess-Ebene nicht nur in einem Unternehmen, sondern auch bei Kunden und Partnerfirmen verspricht.

Wichtiger Standard im Umfeld von EAI ist die von Sun¹ veröffentlichte *Java Connection Architecture* (JCA). Dieser definiert eine Schnittstelle zur Integration von Informations- und Datenbanksystemen mittels sogenannter *Resource Adapter* und ist Bestandteil der hauseigenen J2EE-Plattform.

¹ <http://java.sun.com/>

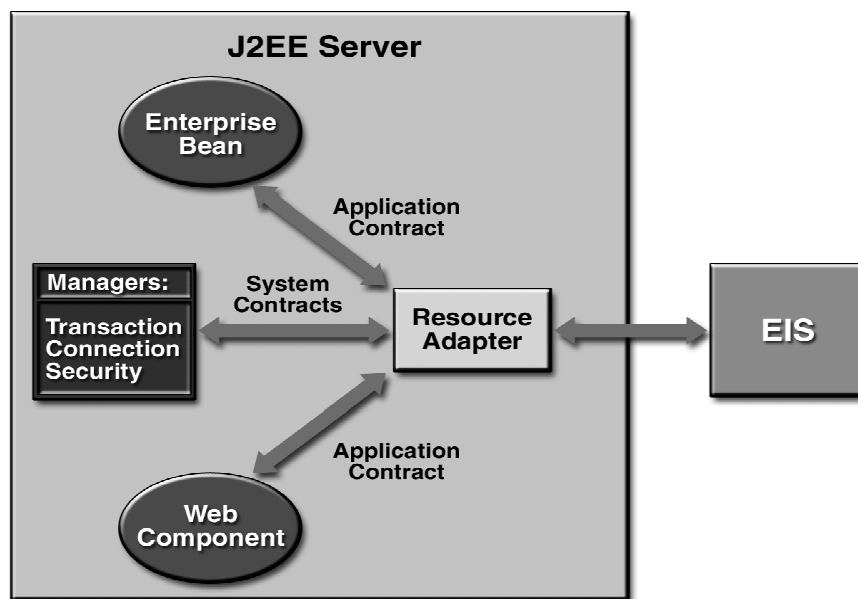


Abb. 15: Resource Adapter (Quelle: Sun Microsystems, Inc.)

Ein Adapter ist eine J2EE-konforme Komponente, die für die Integration eines "externen" Informationssystems zuständig ist. Der Adapter stellt über *contracts* definierte API's zur Verfügung, die das Ansprechen einer "externen" Ressource (System) ermöglichen. Mit Hilfe sogenannter *system contracts* können wichtige Dienste, wie *connection* bzw. *connection pool management*, *transaction management*, *security management* etc. des *application server* (soweit eingesetzt) beansprucht werden. Die eigentliche Anwendungsschnittstelle wird über die *application contracts* definiert. Sie ist entweder standard-konform (*common client interface*) oder spezifisch (*specific interface*).

Die technische Komplexität und der hohe Stellenwert der Technologie erweisen sich aber auch als Nachteil der EAI. Die Geschäftsprozesse werden zwar berücksichtigt, nicht aber in einer Form, die den Einsatz eines Workflow-Management-Systems (WfMS) ermöglichen würde. Aspekte der Prozessoptimierung und Simulation bleiben im Umfeld von EAI meist unbehandelt.

Man kann ein WfMS generell als eine Integrationsplattform verstehen. Anders als bei EAI lautet aber das Motto: "nur so viel wie notwendig integrieren".

3. Das Workflow-Management-System als Integrationsplattform

Die Anbindung an fremde bzw. Legacy-Systeme kann sich aufgrund der Schnittstellen-Problematik als sehr aufwändig erweisen. Dafür können zwei wesentliche Gründe genannt werden:

1. Unterschiedliche Basistechnologien
2. Unterschiedliche Daten-Interpretationen vom jeweiligen System

Das Hinzufügen einer oder mehrerer logischer Schichten in bzw. um die betreffenden Anwendungen ermöglicht in der Regel einen Ausgleich der technologischen und semantischen Systemunterschiede. Die Realisierung kann beispielsweise auf einem synchronen (zum Beispiel: Application-Gateway) oder einem asynchronen Ansatz (zum Beispiel: mobile Agenten) basieren. Beide Ansätze bieten eine eigenständige Schnittstellen- bzw. Anbindungslösung. Die Implementierung erfolgt auf der Basis der verteilten Modelle und Techniken (COM/DCOM, CORBA, RMI, Web-Services etc.).

Eine Abkopplung der Verknüpfungslogik vom Kernsystem wird auch von der WfMC² empfohlen.

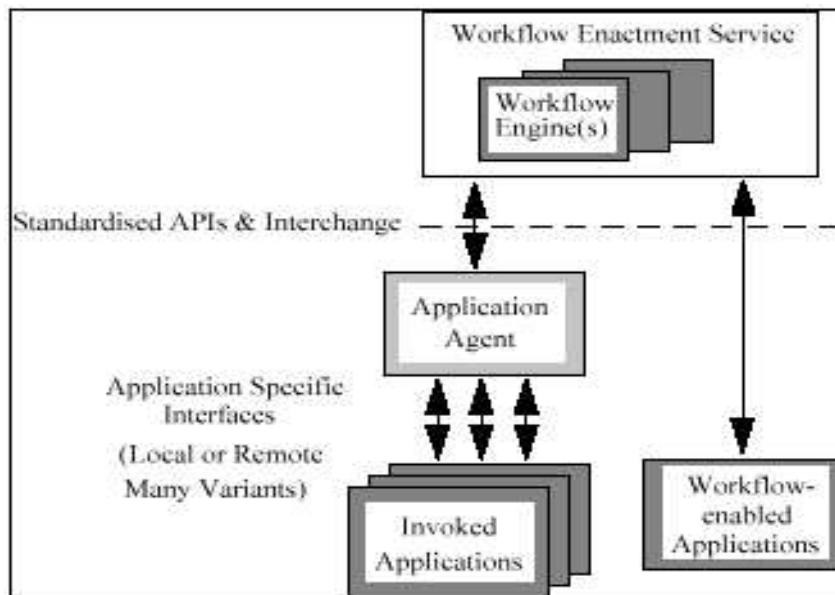


Abb. 16: Anbindung externer Systeme (vgl. Interface 2 der WfMC³)

² vgl. WfMC - Workflow Management Coalition

Die aufgeführten "Application / Tool Agents" werden zwar als Verbindungselemente dargestellt, eine ausführliche Erläuterung der Architektur und Funktionsweise erfolgt von der WfMC aber nicht.

Die im Rahmen des Workflow-Managements entstehenden Integrationsaufgaben und -probleme fallen unter den sogenannten Operationsaspekt⁴. Diese Prozesssicht beschreibt, in welcher Form externe Werkzeuge an das System angebunden werden und auf welche Weise die Kommunikation mit ihnen realisiert wird. Externe Werkzeuge können Software-Module bzw. Systeme jeglicher Art sein.

Ein Akteur ist das arbeitsausführende Element innerhalb eines Workflow-Management-Systems (WfMS). Workflow-Akteure können daher generell in zwei Kategorien gruppiert werden:

1. „menschliche“ Akteure
2. fremde Anwendungen

Eine Aufgabe kann als eine Aktorsicht auf einer Aktivität verstanden werden. Die Aufgabe kann sich auf eine (Work Item) oder mehrere (Work List) Aktivitäten beziehen und wird vom System einem Akteur zugewiesen. Sollte eine Aktivität von mehreren Akteuren ausgeführt werden, bekommt jeder der beteiligten eine eigene Aufgabe zugeteilt.

Unterschiedliche Aktivitäten können unterschiedliche Ausführungen verlangen. Die Ausführungsart ist entscheidend dafür, welcher Akteur(en) die Aktivität ausführen kann (können). Diese Zuordnung bedarf bei den „menschlichen“ Akteuren einer vorangehenden Gruppierung in bestimmte *Rollen*. Mit dem Rollenkonzept lassen sich verschiedene Gliederungskriterien (zum Beispiel nach Kompetenzen, Fähigkeiten oder Verantwortlichkeiten) darstellen. Rollen können in Beziehungen wie Zugehörigkeit, Hierarchie, Vertretung, Zuständigkeit, Kompetenz etc. zueinander stehen.

Die Frage "Welche Akteure einer Rolle sollen eine Aktivität ausführen?" ist wegen der "schwachen" Organisationsbindungen gerade im Kontext der virtuellen Gemeinschaften

³ <http://www.wfmc.org/standards/docs.htm>

⁴ Jablonski, Böhm, Schulze: Workflow-Management, Entwicklung von Anwendungen und Systeme, dpunkt.verlag, Heidelberg, 1997

sehr interessant. Es sind also Mittel in Form von Zuordnungsstrategien notwendig, die eine angemessene Aufgabenverteilung ermöglichen. Eine Zuordnungsstrategie definiert daher ein Auswahlverfahren, das mit Hilfe von Regeln unter Einbeziehung bestimmter Parameter eine Menge von passenden Aktoren ermittelt, die für die Ausführung der Aktivität zuständig ist. Die Wahl erfolgt zur Laufzeit ("run time") durch die Interpretation der Regeln unter Berücksichtigung der Parameterwerte (*indirekte* Zuweisung). Der Gebrauch von Zuordnungsstrategien ist allerdings nicht zwingend. Würde man auf diese verzichten, so müssten die passenden Aktoren *direkt* (ohne Rollenauflösung), d.h. während der Modellierung ("build time"), der Aktivität zugeordnet werden. Als *statisch* werden diejenigen Zuordnungsstrategien bezeichnet, die unter Angabe der Rolle(n) eine direkte Aktorenauswahl ermöglichen. Im Gegensatz zur direkten Zuordnung findet eine Rollenauflösung statt.

4. „Application-Gateway’s“ im Umfeld von „KontextFlow“

„KontextFlow“ ist ein web-fähiges System, das die wesentlichen Aspekte des Workflow-Managements, wie Organisation, Funktion, Verhalten, Information und Operation, behandelt.

In „KontextFlow“ werden beide Akteurenarten („menschliche“ Aktoren und fremde Anwendungen) unterstützt. Die Aufgabenzuweisung erfolgt anhand der Zuordnungsstrategien, die einem der folgenden Grundtypen entsprechen:

1. HUMAN – Strategien für „menschliche“ Aktoren
2. MACHINE – Strategien für fremde Anwendungen
3. MIXED – Strategien für „menschliche“ Aktoren und fremde Anwendungen

Die Anbindung an die fremden Anwendungen erfolgt mittels spezieller SW-Bausteine, die „Application Gateways“ genannt werden. Basierend auf den Zuordnungsstrategien übergibt das System „KontextFlow“ die auszuführenden Aufgaben an die entsprechenden Gateways, die als eine Art Stellvertreter für die fremden Anwendungen fungieren.

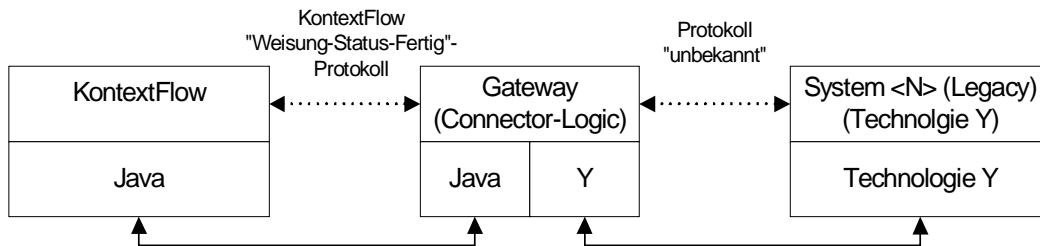


Abb. 17: Application-Gateway

Das "Gateway" ist ein abstraktes, selbständiges Modul, das in die „KontextFlow“-Umgebung integriert ist. Dieses kann als Programm, Dienst, Agent etc. realisiert werden. Das Modul implementiert, ähnlich dem OSI-Referenzmodell, zwei Schichten. Die untere "technologische" Schicht

- besitzt eine Java-Schnittstelle zum "KontextFlow",
- und "wrap" Implementierungen diverser technischer Schnittstellen (CORBA, COM/DCOM, Web-Services etc. oder Datenschnittstellen).

In der zweiten "Connector-Logic"-Schicht findet eine Übersetzung der Interaktionsprotokolle zwischen „KontextFlow“ und der zu integrierenden Anwendung statt. In Bezug auf die Vorgangsteuerung ist hauptsächlich die Interaktion "Zuweisung und Abarbeiten von Aufgaben" interessant. Diese Interaktion wird seitens „KontextFlow“ und des "Application-Gateway" mittels eines Protokolls, welches die beiden Partner kennt, unterstützt. Die Interaktionsprotokolle von der anzubindenden Anwendung sind vorerst unbekannt und müssen bei Bedarf implementiert werden.

Mit Hilfe von "Application-Gateways" läßt sich eine Anwendungsintegration leicht bewerkstelligen. Voraussetzung dafür ist das Definieren eines Integrations- bzw. Koordinationsvorganges in der „KontextFlow“-Engine, welche eine beliebige Prozesskomplexität erlaubt. Damit wäre man in der Lage, übergreifende Workflows zu spezifizieren und über den verfügbaren Anwendungen ausführen zu lassen.

Im Gegensatz zu Sun's *Java Connector Architektur (JCA)* (vgl. Kapitel 2) besteht kein Bedarf, die Schnittstellen der zu integrierenden Anwendungen zwecks "Standard-Konformität" zu ändern. Sobald diese ein Interface besitzt, erfolgt die Implementierung des Interaktionsprotokolls bei dem entsprechenden "Application-Gateway".

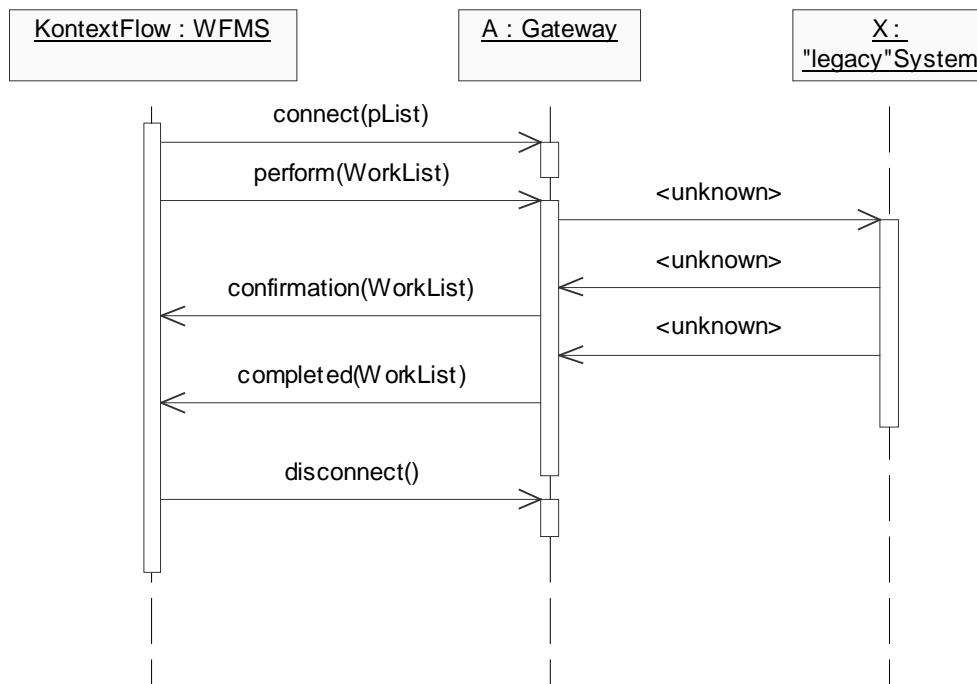


Abb. 18: Übersetzung der Interaktionsprotokolle

Ein weiterer Nachteil der JCA ist die unidirektionale Kommunikation des dort vorgestellten *Resource Adapter*. Dieses macht eine Implementierung komplexer Protokolle praktisch unmöglich.

5. Zusammenfassung und Ausblick

In diesem Artikel wurde die Anwendungsintegration als notwendiger Schritt zur Schaffung der technischen Voraussetzungen für die Bildung virtueller Organisationsstrukturen diskutiert. Nach einer Einführung in die Integrationsproblematik wurden die gängigen Ansätze der *Enterprise Application Integration (EAI)* sowie die *Java Connector Architektur (JCA)* als einige der aktuellsten Standards vorgestellt.

Workflow-Management-Systeme (WfMS) werden primär zur Automatisierung der Ausführung von Abläufen eingesetzt. Soweit vom System unterstützt, können auch Anwendungen als ausführende Entitäten eines Ablaufes fungieren (vgl. Operationsaspekt des Workflow-Managements). Diesbezüglich existieren auch Standards, wie zum Beispiel Interface 2 der *Workflow Management Coalition (WfMC)*. Der Architekturempfehlung der WfMC folgend, ermöglicht das web-fähige Workflow-Management-System „KontextFlow“ das Einbeziehen fremder Anwendungen als

Workflow-Aktoren in die Prozessausführung. Dieses erfolgt durch die Unterstützung spezieller Aufgabenzuordnungsstrategien (HUMAN, MACHINE, MIXED). Die eigentliche Interaktion übernehmen dabei die sogenannten „Application Gateways“. Im Gegensatz zu den komplexen EAI-Lösungen, deren voller Funktionsumfang nur in seltenen Fällen ausgeschöpft wird, kann mit Hilfe von „KontextFlow“ nur so viel „integriert“ werden, wie tatsächlich notwendig ist.

Die künftigen Arbeiten konzentrieren sich auf die Entwicklung von Aufgabenzuordnungsstrategien für „nicht-menschliche Aktoren“ sowie auf die Entwicklung passender Standard-„Applikation-Gateways“, die eine Anbindung von gängigen Systemen in den Bereichen Office und ERP ermöglichen würden.

6. Literatur

- [ArFaHä95] *Arnold; Faisst; Härtling*: Virtuelle Unternehmen als Unternehmenstyp der Zukunft, In: HMD, Nr. 185, 1995
- [ArHä95] *Arnold; Härtling*.: Virtuelle Unternehmen: Begriffsbildung und Diskussion, Arbeitspapiere der Reihe Informations- und Kommunikationssysteme als Gestaltungselement Virtueller Unternehmen, Nr.3/1995
- [EnSt89] *Engelien; Stahn*: Software-Engeneering: CAMARS-Technologie, Akademie-Verlag Berlin, 1989
- [FoChiBA] *Fox, Chionglo, Barbuceanu*: The Integrated Supply Chain Management System, <http://www.ie.utoronto.ca/EIL/iscm-descr.html>
- [GaHa95] *Ganeshan, Harrison*: An Introduction to Supply Chain Management, http://silmaril.smeal.psu.edu/misc/supply_chain_intro.html
- [HaKö98] *Halatchev; Közle*: Workflow-Management in virtuellen Unternehmen; *Engelien/Bender*: Gemeinschaften in Neuen Medien , Eul, Koeln/Lohmar, Germany, 1998
- [Jab97] *Jablonski at al*: Workflow-Management: Entwicklung von Anwendungen und Systemen; Facetten einer neuen Technologie, 1. Auflage, Heidelberg: dpunkt-Verlag, 1997

-
- [MaSch97] *Maurer; Schramke: Workflow-Management-Systeme in virtuellen Unternehmen, Universität Mainz, Arbeitspapiere WI Nr. 11/1997*
- [NgDo98] *D.T Nguyen; V.T Do.: Virtuelle Gemeinschaft - Infrastruktur und Technologie; Engelen/Bender: Gemeinschaften in Neuen Medien, Eul, Koeln/Lohmar, Germany, 1998*
- [Sie96] *Sieber, P.: Die Internet-Unterstützung Virtueller Unternehmen, Arbeitspapiere der Reihe Informations- und Kommunikationssysteme als Gestaltungselement Virtueller Unternehmen, Nr.6/1996*
- [Sun] *Java Connector Architecture (JCA):*
<http://java.sun.com/j2ee/connector/index.html>
- [WfMC] *Workflow Management Coalition: <http://www.wfmc.org/>*