

Rothenberge EMISA Report 1997

Bericht vom Strategie-Treffen in Rothenberge

unter Mitwirkung von W. Deiters, J. Desel, Ph. Hanschke, S. Jablonski,
H. Jasper, M. Jeusfeld, J. Küng, K. Lautenbach, U. Lipeck,
A. Oberweis, D. Steinbauer, R. Studer, M. Weske

1. Einführung
 - 1.1 Hintergrund
 - 1.2 Teilnehmer
 - 1.3 Übersicht über den Bericht
2. Zur Situation der Informatik
3. Profil der Fachgruppe EMISA
 - 3.1 Positionierung von EMISA
 - 3.2 Grundsätzliches Ziel von EMISA
 - 3.3 Abgrenzung
 - 3.4 Organisatorische Aspekte
 - 3.5 Spezifische Aufgaben und Themen für EMISA
 - 3.5.1 Lebenszyklus großer Informationssysteme
 - 3.5.2 Methoden und Modelle für Informationssysteme
 - 3.5.3 Kooperation, Koordination und Prozesse (Workflows)
4. Themen
 - 4.1 Zentrale Themen der EMISA
 - 4.1.1 Softwarearchitekturen großer Informationssysteme
 - 4.1.2 Konstruktionslehre für große Informationssysteme
 - 4.1.3 Aktive Informationssysteme
 - 4.1.4 Multimediale Informationssysteme
 - 4.1.5 Informationssysteme und Workflow-Management
 - 4.1.6 Formale Grundlagen des World-Wide Web
 - 4.2 Spezielle Themen zur Basis von Informationssystemen
 - 4.2.1 Grundlagen objekt-relationaler Datenbanken
 - 4.2.2 Transaktionssysteme
 - 4.2.3 Zum Begriff „Information“
5. Zusammenfassung und Ausblick

1. Einführung

1.1 Hintergrund

Vom 19. bis 21. Februar 1997 fand ein Strategie-Treffen der GI-Fachgruppe 2.5.2 EMISA im Landhaus Rothenberge der Universität Münster in Weitringen bei Münster zum Thema „Quo vadis EMISA“, statt. Das Ziel war, über die Fachgruppen-Zukunft und die unseres Gebietes nachzudenken, um daraus idealerweise Impulse, Anregungen, Vorstellungen etc. für die weitere Fachgruppen-Arbeit zu entwickeln.

Eingeladen zu diesem Treffen waren die Mitglieder der Fachgruppenleitung, die jeweils einen weiteren Gast zu dieser Veranstaltung mitbringen konnten und sollten. Jeder Teilnehmer hatte ein Positionspapier vorzubereiten und vor Ort vorzutragen, in welchem Teilnehmer aus dem universitären Bereich versuchen sollten, die folgende Frage zu beantworten:

Wenn Sie 5 weitere Projekte finanziert bekämen (zusätzlich zu denen, die in Ihrer Gruppe bereits laufen), mit welchen Themen würden Sie sich befassen?

Teilnehmer aus der Industrie und Wirtschaft sollten sich mit folgender Frage auseinandersetzen:

Wenn Sie 5 Universitätskooperationen vom Staat finanziert bekämen, zu welchen Themen würden Sie diese wählen?

Die Vielzahl der dabei angesprochenen Themen ist im vorliegenden Bericht reflektiert. Neben Sachthemen wurden auch strategische Aspekte diskutiert, die die Arbeit der EMISA als GI-Fachgruppe betreffen; auch diese werden im folgenden wiedergegeben.

1.2 Teilnehmer

Die folgenden Herren haben an dem Treffen (Leitung: G. Vossen) teilgenommen:

W. Deiters, Fraunhofer-Institut Dortmund
J. Desel, Uni Karlsruhe
Ph. Hanschke, sd&m Ratingen
S. Jablonski, Uni Erlangen
H. Jasper, Hochschule Vechta
M. Jeusfeld, RWTH Aachen
R. Kaschek, Uni Klagenfurt
J. Küng, Uni Linz
K. Lautenbach, Uni Koblenz
A. Oberweis, Uni Frankfurt
E. Ortner, TH Darmstadt
D. Steinbauer, GEZ Köln
R. Studer, Uni Karlsruhe
H. Thoma, Novartis Basel
G. Vossen, Uni Münster
M. Weske, Uni Münster

1.3 Übersicht über den Bericht

In Kapitel 2 beschreiben wir kurz die aktuelle Situation der Informatik, speziell aus der Sicht des Teilgebietes der Informationssysteme. In Kapitel 3 wird das Profil der EMISA dargestellt, so wie es sich in den zahlreichen und zum Teil kontrovers geführten Diskussionen herausgestellt hat. In Kapitel 4 stellen wir ein breites Spektrum fachlicher Themen vor, die EMISA für die Fachgruppenarbeit als relevant erachtet. In Kapitel 5 geben wir eine kurze Zusammenfassung sowie einen Ausblick auf die mögliche weitere EMISA-Arbeit.

„Katalogwissenschaft,, zu verstehen scheint, die ausgewählt und konfiguriert, sich um Informatikmethoden zu wenig kümmert und diese im Studium nur mangelhaft vermittelt.

Allerdings muß aus Sicht der Praxis heute auch von folgendem ausgegangen werden:

- Das in der Informatik beschäftigte Personal ist mit den Konzepten und Grundlagen der Informatik oft kaum vertraut,
- Client/Server-Systeme mit replizierten Daten dominieren (Verteilte Datenbanken im engeren Sinne gibt es selten),
- Probleme mit Entwicklungsprojekten (Management und Vorgehensmodelle), der Anwendungs-Architektur in Unternehmen und einigemassen akzeptable Kaufpreise führen zu einer Dominanz von Standardsoftware.
- Die Blood Research Group hat ermittelt, daß von vorhandener PC-Leistung heute weniger als 1 % der Möglichkeiten genutzt wird; von dem, was genutzt wird, werden dazu noch etwa 70 % für Maus und Tastatur verbraucht.

Am diesjährigen Weltwirtschaftsforum in Davos diskutierten Spitzenvertreter aus Wirtschaft und Politik mit 2000 Personen über die Auswirkungen des unbegrenzten Datenaustausches auf die Geschäftswelt, die Finanzmärkte und die Gesellschaft. Europa wurde hierbei ein schlechtes Zeugnis ausgestellt. Es laufe Gefahr, in Rückstand zu geraten, obwohl eine gute informationstechnische Infrastruktur vorhanden sei. Die Nutzung der Informations- und Kommunikationstechnik ist in USA und Ostasien intensiver, Marktchancen werden in USA schneller wahrgenommen als in Europa.

Beim Strategie-Treffen war man sich einig, daß eine Flexibilisierung der Arbeitsmärkte, die sozial nicht verträglich ist, jedoch nicht als Lösung dieser Schwäche angestrebt werden darf. Die Wirtschaft darf nicht zum einzigen Focus unserer heutigen Gesellschaft werden.

3. Profil der Fachgruppe EMISA

3.1 Positionierung von EMISA

EMISA ist zu verstehen als berufsständische Vereinigung von Personen aus der Forschung, der Lehre und der Praxis im Umfeld von Informationssystemen. Sie versteht sich als Interessenvertretung im deutschsprachigen Raum für Weiterbildung, Forschung und Kontakte. Die EMISA befaßt sich mit Anwendungssystemen und deren Basistechniken, auch mit deren Struktur, Aufbau, Komponenten, Abläufen, Algorithmen etc., also ihren „Innereien“. Nicht weiter verfolgen will sie Themen wie Ad-hoc-Ansätze ohne grossen Neuigkeitswert („how I did it,„) sowie Themen, die um ihrer selbst willen abgehandelt werden („l'art pour l'art,„). Die EMISA sollte neue Themen belegen und bearbeiten sowie regelmäßig ihr strategisches Ziel überdenken und gegebenenfalls modifizieren.

EMISA positioniert sich als Fachgruppe in der Informatik im Bereich und Umfeld des *Information Systems Engineering*, das die Herstellung, Nutzung und Wartung von Informationssystemen zum Inhalt hat. Information Systems Engineering schlägt nicht nur eine Brücke von

2. Zur Situation der Informatik

Die folgenden Tendenzen sind heute weltweit festzustellen:

- Die Anzahl installierter Informationssysteme steigt ständig (durch neue Anwendungsgebiete, breiteren Zugang, z. B. durch WWW),
- weiter zunehmend ist die Vernetzung von Informationssystemen (etwa zum Zwecke der Delegation von Anfragen, verbunden i.a. mit der Gefahr von Inkonsistenzen),
- im Zusammenhang mit dieser Globalisierung werden die sozialen Aspekte der Vermögenskonzentration immer ausgeprägter (siehe Buch „Die Globalisierungsfalle,„),
- das Reverse Engineering nimmt immer breiteren Raum ein (etwa zum Erschließen alter Informationssysteme für neue Zwecke oder für Data Mining und Knowledge Discovery),
- die Freiräume der Bürger erweitern sich, insbesondere durch die Möglichkeiten der ständigen Erreichbarkeit.

Speziell aus der Sicht der Informationssysteme läßt sich des weiteren sogar feststellen:

- Die Menge des on-line verfügbaren Wissens steigt rapide an (z. B. durch POS-Daten, Case Libraries, Web-Seiten, Multimediadokumente).
- Die Daten- bzw. Wissensquellen sind heterogen (unstrukturiert, halbstrukturiert, strukturiert), gegebenenfalls weltweit verteilt, partiell inkonsistent und zeitlichen Veränderungen unterworfen.
- Die klassische Trennung in Datenbanksysteme, Informationssysteme und wissensbasierte Systeme wird mittelfristig hinfällig (komplexe sowie erweiterbare Datenmodelle, Vererbungskonzept, multimediale Information, immer größere Wissensbasen).
- Information bzw. Wissen erhält zunehmende Bedeutung als

- Produktionsfaktor,
- Produktionskomponente (Komponenten-Software kommt immer öfter vor),
- Produkt, das man verkaufen kann (z. B. elektronische Börse, Kundenprofile für gezieltes Marketing).

Hieraus läßt sich als erstes Fazit ableiten, daß die hohe Informationsflut die Entwicklung intelligenter Lösungen erfordert (Integration von KI-Methoden in IS-Anwendungen, Agentenkonzept paßt in natürlicher Weise in sich herausbildende Szenarien) und daß der strategische Einsatz von Information zunehmend zum Wettbewerbsfaktor wird.

Aus Sicht des Faches Informatik ist es daher nicht verwunderlich, daß sich Informatikanteile in die unterschiedlichsten Fachgebiete hinein entwickelt haben und weiter entwickeln. Gewisse Defizite werden in der Wirtschaftsinformatik erkannt, die sich manchenorts als eine Art

der Soziologie, der Psychologie, der Philosophie (Ethik), der Pädagogik und der Wirtschaftswissenschaft zur Informatik, zur Mathematik, zur Kommunikationstechnik etc.. Es integriert die Teile dieser Disziplinen, die notwendig sind zum Aufbau einer Lehre von der Konstruktion, dem Betrieb und der Wartung von Informationssystemen sowie zur Vermittlung und Anwendung dieser Lehre. Information Systems Engineering arbeitet mit Lebenszyklusmodellen der Informationssysteme.

Das Themengebiet der EMISA wird im wesentlichen von den Fachinformatiken (z.B. der Wirtschaftsinformatik), den Basistechnologien (z.B. den Datenbanksystemen) und dem Software-Engineering (Softwaretechnik) aufgespannt. Beachtet werden zudem die Einflüsse der Sozialwissenschaften auf die Konstruktion von Informationssystemen. Es ist der Diskussionsrunde bewusst, daß es sich bei Informationssystemen um soziotechnische Systeme handelt und daß neben Methoden der Informatik auch Ansätze anderer Wissenschaften Gegenstand der Arbeit von EMISA sind.

Für die Entwicklung von Informationssystemen wird Programmierung unwichtiger, wichtiger wird dagegen die Modellierung (Grund: konfigurierbare Standard-Software, Workflow-Management-Systeme). Bei künftigen Informationssystemen wirken Standard-Software-Komponenten sowie individuell erstellte Anwendungsteile zusammen, sie enthalten Datenbank-, Workflow-, KI- sowie Groupware-Komponenten. Diese Disziplinen werden in der Forschung derzeit noch isoliert voneinander betrachtet. Integrierende Modellierungs-, Entwicklungs- und Systemkonzepte fehlen. Viele Aspekte lassen sich im Rahmen von Vorgehensmodellen und Vorgangssprachen integriert behandeln mit einer Untergliederung in

- Benutzer-Fachsprachen
 - Diagrammsprachen
 - Spezifikationssprachen
 - Programmiersprachen
 - Verbindungssprachen
 - Beobachtungssprachen
- etc.

Zu beachten ist, daß Sprachen im Anwendungsentwicklungsprozeß eine spezifische Verwendung finden. Benutzer-Fachsprachen treten in den frühen Entwicklungsphasen als „Vertragsprachen“, (fachliche Normsprachen) zwischen Entwicklern und Benutzern auf. Hierzu gehören auch Beschreibungssprachen für Anforderungen, die an der Sprache des Benutzers orientiert sind. Konstruktions- und Entwicklungssprachen werden bei Qualitätssicherungsmaßnahmen als „Beurteilungssprachen“, (Verifikation) eingesetzt. Durch umfassende Analyse der Verwendungsarten von Sprachen im Informationssystem-Lebenszyklus können die Vorgänge innerhalb der Entwicklungsphasen genauer festgelegt werden.

3.2 Grundsätzliches Ziel von EMISA

Das grundsätzliche Ziel unserer Fachgruppe kann wie folgt umschrieben werden:

Herstellung, Anwendung und Vermittlung einer Konstruktionslehre für Informationssysteme, die an den Anwendungsgebieten (Fachsprachen) orientiert ist.

Als Eckpfeiler einer solchen Konstruktionslehre kommen in Betracht:

- Entwicklungsinstrumente aus Anwendungs-Entwicklungsumgebungen, bestehend aus Vorgehensmodell, Methoden und Werkzeugen, Normteilen und Standards, Projektmanagement, Qualitätssicherung, Repository-System und Vorgangssteuerungssystem.
- Entwicklungsergebnisse, die als Musteranwendungen einzelner Branchen (z. B. Handel, Versicherungen, Produktionsunternehmen) oder als Komponenten-Sammlungen mit hoher Variantenzahl zur Entwicklung von Anwendungssystemen aus Bauteilen lehrstoffartig behandelt werden.
- Qualitätssicherung, die als Aushängeschild einer Ingenieurwissenschaft dient mit einem umfassenden Repertoire an Methoden zur Begründung der Entwicklungsleistung und der Entwicklungsergebnisse.

Als Ziele für EMISA im weiteren Sinn wären zu nennen:

- Verbindung mit dem Software-Engineering (Softwaretechnik),
- Verbindung mit „Fachinformatiken“, (z. B. der Wirtschaftsinformatik),
- Verbindung mit einem Technologie-Bereich (z. B. Datenbanksysteme).

Als „Fachinformatiken“, bezeichnen wir Forschungs- und Lehrgebiete, die in den Fachwissenschaften (z. B. Bauingenieurwesen, Betriebswirtschaftslehre, Biologie, Chemie, Mathematik, Medizin) mit dem Rechner und mit Informatik-Methoden als Arbeitsmittel arbeiten.

3.3 Abgrenzung

Hinsichtlich der Darstellung der Fachgruppe nach außen ist festzustellen: Hinter EMISA steckt kein allgemein anerkanntes Teilgebiet der Informatik. Begriffe wie *Information Systems Engineering* oder *Data Engineering* sind hierzulande kaum geläufig. Sie könnten durch EMISA in die Diskussion gebracht werden. Trotzdem ist für EMISA - bedingt durch die inhaltlichen Bezüge zu anderen Disziplinen und Fachgruppen - eine Zusammenarbeit mit anderen Gruppen wichtig, die sich auch mit Informationssystemen befassen. Die Lehre der Konstruktion, Nutzung und Wartung von Informationssystemen muß sich stärker präsentieren, insbesondere interdisziplinär und integrativ.

Man kann eine Abgrenzung zwischen EMISA, Fachgruppen der Wirtschaftsinformatik und des Software-Engineering wie folgt diskutieren: Die Abgrenzung zwischen EMISA und Fachgruppen der Wirtschaftsinformatik liegt in der Unterscheidung zwischen der Konstruktion einer Anwendungslösung und ihrem Gebrauch begründet. Während sich die Interessen der EMISA-Mitglieder an der Konstruktion von Informationssystemen orientieren, organisieren Wirtschaftsinformatiker häufig primär den effizienten Gebrauch von Informationssystemen. Die Abgrenzung zwischen EMISA und Fachgruppen des Software-Engineering kann man durch die Unterscheidung zwischen Inhalt und Form einer Anwendungslösung aufzeigen. EMISA-Mitglieder setzen sich mit formalen Methoden und den Inhalten der Anwendungsbereiche gleichrangig auseinander. Der Einfluß formaler Methoden auf die Inhalte einer Anwendung - und der Einfluß der Inhalte auf formale Methoden - wird als Arbeitsgegenstand angesehen. Im Software-Engineering liegt das Schwergewicht der Arbeit auf der Entwicklung formaler und semi-formaler Methoden. Die fachspezifischen Inhalte einer Anwendung werden als Angelegenheit der Anwender betrachtet.

- die Kommunikation mit dem Anwender als Implementierungsgrundlage sowie zur Analyse (auch zum Zwecke der Wartung)
- das Arbeitsteam: Modellierer sowie Teams unterschiedlicher Struktur und Qualifikation

Die EMISA soll Empfehlungen geben können, etwa und speziell hinsichtlich einer Konstruktionslehre. Formale Modelle müssen dabei nicht gesondert hervorgehoben werden, sondern sie sind Bestandteil der Betrachtungen.

3.5.3 Kooperation, Koordination und Prozesse (Workflows)

Informationssysteme arbeiten in Zukunft vermehrt mit vernetzten - teilweise mobilen - Komponenten, mit unterschiedlichen Informationsquellen, mit synchronen und asynchronen Kooperationsdiensten. Diesem muss EMISA Rechnung tragen durch die verstärkte Beschäftigung mit verteilten Systemen (im Sinne von Interoperabilität). Oberste Maxime hierbei ist die autonome Entwicklung und der autonome Betrieb der (heterogenen) Subsysteme. Das Informationssystem wandelt sich zu einem Informationsnetz, dessen Knoten irgendwie geartet sein können.

Für die Kopplung von „Informations-Knoten“, kommen je nach Anforderung diverse Optionen in Betracht. Die loseste Form der Kopplung ist Electronic Data Interchange mit Datenaustauschformaten (z.B. mit dem UN-Standard EDIFACT). Eine stärkere Kopplung wäre mit dem Einsatz von Middleware wie CORBA gegeben. Noch stärker sind Netze mittels Workflow-Management gekoppelt, eventuell sogar mit einer Hierarchie von Workflow-Management-Systemen (WFMS). Dabei können Knoten im WFMS ihrerseits wieder WFMS sein. Anwendungen von Information-Netzen sind u.a. Electronic Market, virtuelle Unternehmen, gemeinsame Projekte unterschiedlicher Unternehmungen (die autonomen Subsysteme können dann auch Groupware-Systeme sein). Wenn zentral entwickelte Programme mittels Internet (Java) ausgeführt werden, liegt eine vergleichbare Situation vor.

EMISA sollte sich im Rahmen von Veranstaltungen dieser Thematik annehmen, sie nicht nur von der Anwendungsseite beleuchten, sondern auch Ansätze und Konzepte zur Implementierung und Realisierung derartiger Systeme betrachten. EMISA sollte sich sowohl mit der Technologie als solcher als auch mit dem Thema Entwicklungsmethoden und Vorgehensmodelle für solche Systeme auseinandersetzen. Dabei muß die lokale Entwicklung und Problemlösung von Informationsknoten auch berücksichtigt werden.

Dieses Gebiet hat Zukunft in der Informatik, derzeit sind die Probleme noch groß und die Entwicklung steht weitestgehend am Anfang.

4. Themen

4.1 Zentrale Themen der EMISA

4.1.1 Softwarearchitekturen großer Informationssysteme

Große Informationssysteme bestehen aus einer Vielzahl funktionaler Einheiten, z.B. zur Verwaltung von Daten oder zur Unterstützung anwendungsspezifischer Vorgänge. Mit klassischer

3.4 Organisatorische Aspekte

EMISA soll in Zukunft besser identifizierbar werden, der Fachgruppe würde ein großes sowie ein kleines Treffen pro Jahr gut tun. Außerdem ist eine Konzentration auf wenige Themen für einen überschaubaren Zeitraum erforderlich. Ferner sind Workshops sowie Arbeitstreffen zu spezifischen Themen nötig.

3.5 Spezifische Aufgaben und Themen für EMISA

3.5.1 Lebenszyklus großer Informationssysteme

Das Gebiet „Beherrschung großer Informationssysteme“, gewinnt in der Informatik zusehends an Bedeutung. Für EMISA stellt sich diese Thematik als Entwicklungs-, Betriebs- und Managementproblematik dar. In Anlehnung an VLDB (Very Large Data Bases) ist die Bezeichnung VLIS (Very Large Information Systems) angemessen. Zur Klärung der Frage, was große Informationssysteme sind, gilt es beispielweise „Anzahl der Instanzen“, „Anzahl der Benutzer“, „Anzahl der Querbeziehungen zwischen Modulen“, „Anzahl der Schnittstellen“, „Anzahl der Modellierungskonzepte“, oder die „Legacy-Problematik“, zu beachten. Die Beherrschbarkeit solcher Informationssysteme kann mit Repositories und Standardisierung durch Zerlegungsmethodiken bei der Analyse und Synthese oder durch Schichtenbildung und klare Architekturkonzepte gelingen.

Ein weiteres Feld für große Informationssysteme eröffnet sich durch die Bereitschaft des privaten und öffentlichen Sektors, sich zu vernetzen. Darüber hinaus erwachsen aus den im Aufbau begriffenen thematischen Netzwerken wie Telearbeit oder elektronische Märkte zusätzliche Anwendungsbereiche.

3.5.2 Methoden und Modelle für Informationssysteme

Im Bereich der Entwicklung von Informationssystemen ist absehbar, daß die eigentliche Programmierarbeit immer weniger und unwichtiger wird. Im Zusammenhang mit den zunehmend eingesetzten konfigurierbaren betriebswirtschaftlichen Standard-Softwaresystemen und Workflow-Management-Systemen spielt die Modellierung, d.h. die Erstellung von Objekt- und Ablaufschemata, die den Systemen zugrunde liegen, eine immer wichtigere Rolle. Damit sollte auch die Bedeutung des Information Systems Engineering zunehmen. Voraussetzung ist, daß man sich mit diesen Fragestellungen auch in wissenschaftlichen Arbeitskreisen auseinandersetzt und das Feld nicht Anderen (z.B. Unternehmensberatern) überläßt.

Die Betrachtungsgegenstände der EMISA in diesem Bereich sind

- das Informationssystem: Die Art des Informationssystems, dessen Umgebung und dessen Zweck
- die Beschreibungsmittel, insbesondere Modellierungssprachen und eine Konstruktionslehre

Architektur sind sie aus hierarchisch angeordneten Softwarekomponenten aufgebaut, z. B. auf der Basis von Datenbanksystemen.

Durch neuere, insbesondere objektorientierte Programmierparadigmen, Vernetzung dezentraler Organisationseinheiten und die Unterstützung umfangreicher Anwendungsprozesse sind Anforderungen an die Softwarearchitektur entstanden, die mit hierarchischen Ansätzen nicht zufriedenstellend gelöst werden können. Daher nutzen moderne Anwendungssysteme Konzepte wie Client/Server-Architekturen und universelle Objektschnittstellen, etwa „Object Linking and Embedding“, (OLE). Weitergehende Entwicklungen kennen sogenannte „Object Broker“, die verteilte, modulare und erweiterbare objektorientierte Softwarearchitekturen zum Ziel haben. Zukünftige Anwendungssysteme werden eventuell aus vordefinierten, parametrisierbaren Bausteinen zusammengestellt (Componentware), die in weitestmöglichen Zeilen verfügbar sein können und die auf der Grundlage einer universellen objektorientierten Infrastruktur kommunizieren. Standardsoftware kann ein solcher Baustein sein. Solche Konzepte, deren Entwicklung sowie deren Vor- und Nachteile sollten beleuchtet werden. Profile zur Charakterisierung von Anwendungskomponenten und Informationsquellen sowie die wechselseitige Anpassung von Anforderungen und angebotenen bzw. konfigurierten Funktionalitäten sind hier als Beispiele zu nennen.

Bei der Architektur geschäftsprozessweiter oder gar unternehmensweiter Anwendungen (Architektur im Großen) ist häufig die Größe und damit auch Komplexität des Gesamtsystems eines der zentralen Probleme. Das Kernproblem besteht darin, die Anwendungen in überschaubare Teilsysteme zu untergliedern, ohne die Komplexität in das Zusammenspiel der Teilsysteme zu verlagern. Es müssen Organisationsprinzipien für große Anwendungssysteme (jenseits von Schichtenarchitekturen) gefunden werden und diese angemessen (beispielsweise durch Pattern, Bausteine oder Frameworks) unterstützt werden können. Workflowsysteme sind ein populäres Beispiel. Andere Beispiele sind Pipeline- oder Blackboardarchitekturen. Welche weiteren Prinzipien gibt es, und was sind ihre Anwendungsbereiche? Wie können sie unterstützt werden? Was ist ihre Wirkung auf den Betrieb und die Weiterentwicklung des Systems?

Das folgende Szenario von Anforderungen ist (mit gewissen Abwandlungen) in verschiedenen Projekten zu finden: Teilfunktionen in einem verzweigten Gesamtplan stehen in einer festen Reihenfolgebeziehung. Ein gewisser Anteil aller Vorgänge kann vollautomatisch abgewickelt werden, der verbleibende Anteil jedoch nicht. Welcher Anteil dies ist, kann nicht ohne weiteres im Voraus festgestellt werden. Vielmehr gibt es asynchrone Eingriffe in das System von Teilfunktionen aus den verschiedensten Gründen. Sachbearbeiter beobachten den Bearbeitungsprozess, greifen nach ihrem Ermessen ein und können jeden Bearbeitungsschritt nachträglich korrigieren. Diese Eingriffe erfolgen beispielsweise, um Änderungsanforderungen eines Kunden einzubringen oder weil eine der Ableitungen in einer der Teilfunktionen auf die Datenkombination eines Vorgangsobjektes nicht abwendbar ist. Wesentlich ist, daß ein bereits einmal durchgeführter Schritt nur dann nochmals durchgeführt werden soll, wenn sich an den Eingangsparametern zu diesem Schritt tatsächlich etwas geändert hat.

Aufgrund der asynchronen Eingriffe und der Bedingungen, Funktionen möglichst selten durchzuführen, trifft eine Modellbildung mit einem konventionellen Workflowsystem aufgrund der Komplexität schnell an ihre Grenzen. Ein Lösungsansatz ist die Erweiterung klassischer Workflowsysteme um nicht-deterministische Verzweigungen sowie um die Möglichkeit zur Behandlung von asynchronen Eingriffen in die Vorgangsbearbeitung. Die Diskussion weitere Konzepte, evtl. einer formalen Semantik für ähnliche Aufgaben könnte interessant

sein. Ebenso von Interesse könnte das folgende allgemeinere Probleme sein: Wie berücksichtigt man Abläufe (die Dynamik eines Systems) in einer objektorientierten Architektur?

Geschäftsvorfallobjekte: Ein geschäftsvorfallobjekt ist ein Dialog, der dazu konzipiert ist, einen bestimmten Vorgang (z. B. Kundenneuanlage) zu unterstützen. Nicht-geschäftsvorfallobjekte orientieren sich dagegen beispielsweise am zugrundeliegenden Daten-/Objektmodell und erlauben die Bearbeitung verschiedenartiger Geschäftsvorfälle (z. B. Kunde anlegen, löschen, ändern). In einer objektorientierten Systemarchitektur besitzen die Geschäftsobjekte ein mehr oder weniger explizites Zustandsmodell. Komplexe Geschäftsvorfälle erfordern es, die Bearbeitung in Schritte unabhängig von diesen Zuständen zu gliedern. Die Geschäftsvorfälle besitzen also ihrerseits ein Zustandsmodell. Treffen viele Geschäftsvorfälle auf viele Geschäftsobjekte, so kommt es zu einer kombinatorischen Explosion bei der Zahl der zu betrachtenden Kombination von Zuständen in den Geschäftsobjekten und den Zuständen in den geschäftsvorfallobjekten. Auch hier ist das Grundproblem die Berücksichtigung von Abläufen in einer objektorientierten Architektur. Welche Strategien vereinfachen die Situation?

4.1.2 Konstruktionslehre für große Informationssysteme

Von der Programmierkunst zur Anwendungssystementwicklung: Durch Abstraktion, Modellierung oder Simulation realer Fakten oder Abläufe gelingt es, Komplexität in Anwendungsbereichen besser zu beherrschen. Dies kann durchaus auch zu substantiellen Erkenntnisgewinnen in den Fachwissenschaften der Anwendungsbereiche führen. Eine enge Zusammenarbeit der Informatik mit den Fachdisziplinen ist hierbei für die Konstruktion von Anwendungssystemen zwingend geboten. Eine Frage ist, ob es eine gemeinsame Konstruktionsmethodik für „Fachinformatiken“, (z. B. Wirtschaftsinformatik, Informatik in der Medizin) überhaupt geben kann, und in welchen Bereichen diese Konstruktionsmethodik über das allgemeine „Software-Engineering“, hinaus zu erweitern wäre.

Management von Anwendungssystemen: Der Graben zwischen dem aus Sicht des Projektmanagements verständlicheren, bevorzugten einfachen, linearen Vorgehen bei der Systementwicklung und den Erfordernissen benutzerorientierter Prozesse muß überwunden werden. Hierfür muß die inkrementelle und evolutionäre Systementwicklung mit kleineren Systemteilen und Prototyping plan- und steuerbar gemacht werden. Seitens der Technik gibt es heutzutage für diese Art der Realisierung mehr Möglichkeiten als früher. Das Prototyping darf jedoch nicht zum Feigenblatt für „Quick and Dirty“-Programmierung werden. Wie ist ein Projektmanagement zu gestalten, daß der inkrementelle Entwicklungsprozess optimal unterstützt werden kann? Mit der Einführung von Anwendungssystemen ist das System-Management jedoch noch nicht abgeschlossen. Die Wartung und Pflege von Systemen hat sich mit zunehmender Modularisierung und Verteilung von Systemkomponenten im Netz verstärkt an Modifikationen von Komponenten und ihren gegenseitigen Abhängigkeiten zu orientieren. Reverse Engineering gehört ebenfalls zu diesem Themenkomplex.

Konstruktion „neuartiger“, Informationssysteme: Methoden und Modelle für die Konstruktion von Informationssystemen erfassen diese Systeme noch nicht in ihrer gesamten Breite. Wie sehen z.B. Konstruktionskonzepte für Informationssysteme aus, die unscharfe Daten verarbeiten und unscharfe Antworten geben können. Darunter wird hier verstanden, dass auf eine präzise Frage auch Antworten auf (nicht gestellte) Fragen ähnlicher Bedeutung gegeben werden können. Beispielsweise auf die Frage nach einer Hotelunterkunft in einem bestimmten Ort er-

hält man auch Hinweise auf ähnliche Unterkunstmöglichkeiten in einem Ort in dessen Nähe.

Zur Entwicklungsmethodik für Informationssysteme im World-Wide Web (WWW): Wie soll man Informationssysteme im WWW aufbauen und betreiben? Wie soll der Zugriff von Anwendungssystemen auf heterogene Informationsquellen konstruiert werden? Ist eine Entkopplung von Anwendungssystem und Informationsquellen durch Mediatoren in die Praxis umsetzbar? Man benötigt hierzu beispielsweise Ontologien, Bewertungsmethoden, und Zugriffspläne. Wie sieht eine Konstruktionslehre für interaktive Anfragen auf heterogene Informationsquellen aus? Soll mit einer multimedialen Benutzerschnittstelle (natürliche Sprache, Anfragesprache, Menü, Formulare, graphisch) gearbeitet werden, sind integrierte Wörterbücher denkbar? Die aktive Informations-/Wissenssuche bzw. Filterung in heterogenen Informationsquellen wird hier thematisiert: Benutzerprofile (Kenntnisstand, Informations-/Wissensbedarf), intelligente Suchagenten für Intra/Internet, Informationsaufbereitung und Darstellung, Agenten als Filter (email), Einbettung in entscheidungsunterstützende Systeme (Schwellenwerterkennung, Trenderkennung).

Die zunehmende weltweite Vernetzung bringt die heterogene Informationssystem-Landschaft quasi auf den Schreibtisch des Anwenders. Eine traditionelle Integration (globales Schema, föderierte Datenbank etc.) ist praktisch hoffnungslos, da keine zentrale Kontrolle mehr möglich ist. Offene Fragen: Wie bewältigt man unter diesen Umständen die Heterogenität der zugreifbaren Informationen (Datenstruktur und Semantik)? Wie geht man mit Fehlern um, wenn Teilsysteme kein Zurücksetzen von Transaktionen erlauben? Ideen: Migration zu CORBA-ähnlichen Strukturen, Übernahme und Filterung von heterogenen Daten in eine Art Data Warehouse.

Zur Konstruktion von Workflow-Management-Systemen: Unter einer Informationsfluß-Datenbank stellt man sich eine Meta-Datenbank vor, die die Anforderungen der Anwender an den Informationsfluß in der Unternehmung speichert. Der Vorteil der Repräsentation in einer Meta-Datenbank ist die Möglichkeit, Anfragen bei der Systemanalyse zu stellen. Solche Werkzeuge wurden bisher nur in Software-Engineering-Umgebungen intensiv genutzt. Erkenntnisse aus der Anwendung von Tools wie ARIS zeigen aber, daß es auch außerhalb der traditionellen Software-Entwicklung ein enormes Einsatz-Potential für sie gibt.

Formalisierung von Teilbereichen mit „intelligenten“, Übergängen: Die Objektorientierung erlaubt bei vielen Anwendungen ein hohes Maß an struktureller Durchgängigkeit in verschiedenen Dimensionen:

- Zwischen den Schichten eines Anwendungssystems: Ähnlich strukturierte Klassenstrukturen finden sich in der grafischen Oberfläche, der Schicht der Anwendungslogik und der objektorientierten Datenbank.
- Die Entwicklungsdokumente der verschiedenen Phasen weisen ebenfalls hohe strukturelle Ähnlichkeiten auf: Ein Objekt Auftrag gibt es in der fachlichen Anforderungsdefinition, in der Systemspezifikation, in der Konstruktion und im realisierten System.

Die Menge an Klassen und damit an Teilmasken, Klassenspezifikationen etc. ist oft der hemmende Faktor. Die strukturelle Ähnlichkeit erleichtert es nun, Übergänge zwischen den verschiedenen Phasen und Schichten zu automatisieren. Im Projekt „Janus“, wurde beispielsweise gezeigt, wie aus Objektmodellen Benutzeroberflächen und Datenbankschemata abgeleitet werden können. Wie programmiert man derartige Abbildungen? Vielleicht würde es sich loh-

nen, die Methoden der symbolischen Wissensverarbeitung (Regelverarbeitung, Patternmatching, Defaultreasoning, Constraintverarbeitung) auf dieses Problemfeld anzuwenden. Diese Methoden wurden dafür konstruiert, Wissen und Informationen formal, übersichtlich und flexibel darzustellen und verarbeiten.

Referenzmodelle für die Modellierung von Informationssystemen: Unterschiedliche Modelle für Informationssysteme haben unterschiedliche Schwerpunkte bezüglich ihrer Ausdrucksfähigkeit, ihrer Mechanismen zum modularen Entwurf, ihrer Verfeinerungskonzepte, ihrer Werkzeugunterstützung, ihrer Verbreitung und manches mehr. Mit mathematisch fundierten Konzepten und formalen Modellen aus dem wissenschaftlichen Umfeld können an Referenzmodellen fundamentale Phänomene auf einer abstrakten Ebene beschrieben und erforscht werden. Eigenschaften kommerzieller Produkte – oder wenigstens wichtige Aspekte davon – können auf das Referenzmodell abgebildet und studiert werden.

Formale Spezifikationsmethoden: EMISA darf die formale Spezifikation konzeptueller Modelle nicht aus den Augen verlieren und muß auch in Zukunft ein Forum für die Grundlagenforschung in diesem Bereich sein. Einige Beispiele hierzu kamen in Rothenberge zur Sprache und sind nachstehend kurz skizziert.

Ein Ansatz startet mit Nicht-Standard-Logiken als Ausgangspunkt, die bereits typische Konstrukte für dynamisches Verhalten, Normalverhalten usw. eingebaut haben (modale Logiken wie dynamische, temporale, Default- und deontische Logiken). Komplexitätsanalysen fallen zwar erschreckend aus, aber konsequente (pragmatische, nicht semantische) Modularisierungen helfen, größere Gegenstände zu bewältigen, wenn zudem temporär oder bewußt mit Unsicherheiten oder Inkonsistenzen gearbeitet werden darf. Da inzwischen Logiken bekannt sind, die auch mit nichtkonsistentem bzw. nichtnormativem Verhalten umgehen können, könnte es durchaus Sinn machen, auch für größere Systeme formale Spezifikationen weiterzuentwickeln.

Morphismen von Petri-Netzen – im ursprünglichen, von C. A. Petri geprägten Sinn – sind strukturrepektierende Abbildungen zwischen Petri-Netzen. Insbesondere lassen sich Einbettungen, Vergrößerungen, Faltungen und die Komposition von Netzen mittels Morphismen formalisieren. Diese Transformationen betreffen auch Netze, die auf einer groben konzeptuellen Ebene Systeme beschreiben und keine formale Semantik besitzen. Mit diesem Ansatz könnte die theoretische Fundierung einer werkzeugunterstützten Methodik zur Entwicklung von Petri-Netzmodellen durch strukturerhaltende Transformationen versucht werden. In einer derartigen Umgebung lassen sich bereits in frühen Entwurfsphasen Eigenschaften formulieren und maßschneidern überprüfen.

Das Dualisieren von markierten Petri-Netzen läßt die Existenz einer zweiten Sorte von Marken sinnvoll erscheinen. Diese Marken (T-Marken) liegen auf den Transitionen und bewegen sich gegen die Kantenrichtung. Die Regeln, nach denen sie sich bewegen, liegen fest, weil T-Marken im dualen Netz normale Marken (S-Marken) sind. Interessant ist die Frage, wie die T-Marken zu interpretieren sind und was passiert, wenn S- und T-Marken aufeinander treffen. Eine sinnvolle Interpretation der T-Marken ist, ihnen Verboischarakter zu geben, so daß eine an sich (also durch S-Marken) aktivierte Transition dennoch nicht schalten darf, wenn sie gerade eine T-Markie trägt. Die Folge ist, daß sich S- und T-Marken gegenseitig blockieren. Damit gewinnt man ein völlig neues Instrument zur Steuerung von Systemen.

Die Validierung von Informationssystemen bzw. ihrer formalen Modelle durch Simulation scheidet oft an der Beherrschung der großen Zahl verschiedener Ausführungsfolgen und er-

reichbarer Systemzustände. Die systematische Konstruktion und Analyse der entsprechenden Kausalitätsbeziehungen halbgeordneter Abläufe überwindet teilweise dieses Problem. Hierzu können Modelle von Informationssystemen und ihre Abläufe durch höhere Petrietze beschrieben werden. Die Ablaufetze werden anwendungsgerecht visualisiert, so daß der Anwender diese Beschreibung zur Überprüfung des spezifizierten Systemverhaltens verwenden kann. Hilfreich ist dabei eine explizite Darstellung des Flusses von Daten, Dokumenten und Gütern.

4.1.3 Aktive Informationssysteme

Konventionelle, passive Informationssysteme bieten ihren Nutzern insbesondere die dauerhafte Datenbeschreibung und -verwaltung sowie deren Bereitstellung für zahlreiche Anwendungen. Der Begriff des „aktiven Informationssystems“, charakterisiert unterschiedliche Aspekte von Anwendungssystemen, wobei das Spektrum von Trigger-Sprachen relationaler über Verhaltensbeschreibungen objektorientierter bis hin zu Regelsprachen aktiver Systeme reicht.

Aktive Informationssysteme reagieren selbsttätig auf spezifizierte Ereignisse mit definierten Aktionen. Dies ist eine grundlegende Erweiterung passiver Informationssysteme, die nur explizite Abfrage- oder Update-Operationen ausführen. Sprachmittel zur Beschreibung von Ereignissen und Aktionen sind Trigger und deren Nachfolger, die sogenannten ECA-Regeln. Beide Konzepte ermöglichen eine zentrale Verwaltung von dynamischen Aspekten der jeweiligen Anwendungsgebiete, so daß ihre wiederholte Implementierung in den jeweiligen Anwendungsprogrammen entfällt. Die unterschiedlichen Aspekte sind systematisch zu untersuchen. Dazu gehören Sprach-, Ausführungs- sowie Entwicklungsaspekte. Zusätzlich ist auf Probleme bei der Prüfung von Effektivität und Effizienz aktiver Mechanismen in Anwendungssystemen einzugehen.

Softwaretechnik für aktive Informationssysteme: Bei der Entwicklung aktiver Informationssysteme sind reaktive Aspekte geeignet in die Definitionen und Beschreibungen der unterschiedlichen Phasen des Entwicklungsprozesses zu integrieren. Während eine anwendungsorientierte Analyse reaktive Aspekte in einem einheitlichen Modell darstellen kann, existieren für das Design Freiheitsgrade bei der Zuordnung dieser reaktiven Aspekte zu den unterschiedlichen Funktionseinheiten eines aktiven Informationssystems. Erste Lösungsansätze zur Überführung von Ergebnissen einer objektorientierten Analyse in das Design eines aktiven Informationssystems müssen weiterentwickelt und auf die Praxistauglichkeit überprüft werden. Generell stellt sich die Frage nach einem Regelwerk, mit dem eine geeignete Behandlung der Freiheitsgrade beim Design reaktiver Aspekte möglich wird und das insbesondere die unterschiedlichen Varianten der Realisierung von Ereignis-Aktions-Zusammenhängen berücksichtigt.

4.1.4 Multimediale Informationssysteme

Datenbanken und Informationssysteme haben sich seit vielen Jahren mit der Tatsache auseinandergesetzt, daß sie neben traditioneller Information, insbesondere solcher, die sich in Tabellen ablegen läßt, in zunehmendem Maße hochstrukturierte, aber auch unstrukturierte Information aufzunehmen haben. Inzwischen unterstellt man, daß auch beispielsweise Audio- sowie Video-Information in einer Datenbank ablegbar bzw. aus einem entsprechenden Informationssystem abrufbar sein kann. Es kommt also nicht nur unterschiedlich strukturierte oder

formatierte Information vor, sondern diese stammt im allgemeinen sogar von heterogenen Medien, weshalb man heute einfach von „multimedialen Datenbanken“, oder umfassender von „multimedialen Informationssystemen“, sprechen kann. Es wird erwartet, daß Anwendungen etwa in der Medizin, aber auch in verschiedensten Formen der Aus- und Weiterbildung oder z.B. der Unterhaltungsindustrie („Video-on-Demand,“) in zunehmendem Maße von derartigen Systemen profitieren bzw. hierauf basieren werden.

Für Datenbanken und Informationssysteme ergeben sich im Hinblick auf den angemessenen Umgang mit multimedialer Information nach wie vor Herausforderungen. Von den hier zu behandelnden Fragen bzw. Themenbereichen sind einige wichtige die folgenden:

- **Datmodellierung:** Gibt es angemessene Datenmodelle zur konzeptuellen Beschreibung multimedialer Information sowie Techniken zum Entwurf entsprechender „Schemata,“?
- **Anfragesprachen:** Traditionell kann man auf eine Datenbank entweder per Dialogsprache oder von einer Programmiersprache aus, also über ein API zugreifen. Wie sehen geeignete Dialogsprachen für multimediale Systeme aus? Wie hoch muß ihre Ausdruckskraft angesetzt werden, welche Auswertungskomplexität ist akzeptabel?
- **Systemarchitekturen:** Bei multimedialen Informationssystemen wird es sich i.a. um heterogene Systeme handeln, die eine Reihe unterschiedlicher Peripherie-Hardware an unterschiedlichen Standorten werden bedienen müssen. In traditionellen Datenbank- sowie Informationssystemen haben sich Schichtenarchitekturen bewährt, welche durch eine Reihe von Abstraktions- sowie Funktionsebenen gekennzeichnet sind. Was sind geeignete Abstraktionsebenen einer Schichtenarchitektur für ein multimediales System?
- **Transaktionskonzepte:** Wenn man sich vorstellt, daß ein Videofilm im logischen Kontext einer Transaktion abgespielt (oder auch nur von einem Server abgerufen) wird, so ist das Konzept der Atomarität zu überdenken: Es muß möglich sein, nach einer Unterbrechung durch eine Störung eine laufende Übertragung fortzusetzen, ohne jeweils von vorne zu beginnen. Für andere Medien ergeben sich ähnliche Überlegungen. Was also sind angemessene Transaktionsmodelle und -konzepte für multimediale Daten und Systeme?
- **Benchmarks:** Für bestimmte Anfragen (z.B. im Zusammenhang mit Video) ist eine hohe Performance unabdingbar, während andere Anfragen nicht notwendiger Weise in Realzeit beantwortet werden müssen oder möglicherweise sogar längere Berechnungen auslösen (z.B. bei Bildauswertungen). Welche Arten von Benchmarks eignen sich zur Leistungsmessung in einem Multimediasystem?

4.1.5 Informationssysteme und Workflow-Management

Informationssysteme haben sich in den vergangenen Jahren zu einem Basisbaustein für die sich ausbildende Informationsgesellschaft entwickelt. Es wird dabei erkannt, daß solche Systeme neben der Verwaltung und Verarbeitung von Informationen in zunehmendem Maße eine Bedeutung zur Koordination und Kooperation von Personen in einem vernetzten Umfeld erlangen. Dabei werden zur Zeit noch eine Reihe von Hindernissen gesehen, die einer breiteren Einführung solcher Systeme im Wege stehen.

Workflow-Management: Unter dem Stichwort „Workflow-Management“, wird in den vergan-

genen Jahren an Systemen zur informationstechnischen Unterstützung von Arbeitsabläufen (Geschäftsprozessen) gearbeitet. Diese Systeme steuern – auf der Basis eines Modells von Geschäftsprozessen – den Arbeitsfluß (Workflow) zwischen den Prozeßbeteiligten. Workflows umfassen die koordinierte Ausführung vieler und i.a. vielfältiger Aufgaben durch unterschiedliche Verarbeitungseinheiten, und es sind Prozeduren oder Abläufe, im Rahmen derer Dokumente, Information oder Aufgabenstellungen zwischen Teilnehmern weitergereicht werden mit dem Ziel des Erreichens eines Geschäftsziels. Workflow-Management, also die hard- und software-technische Infrastruktur zur Kontrolle und Koordination vielfacher Workflow-Ausführungen, ist seit einigen Jahren ein wichtiges Thema in vielen Anwendungen. Die Integration von Konzepten für die Fehlerbehandlung, die dynamische Änderbarkeit des Workflows, die Integration von Aspekten der Arbeitswissenschaft und der Organisationslehre auf die Gestaltung des Workflows sind hier interessante, aber noch weitgehend unge löste Probleme.

Zu den systemtechnischen Anforderungen an eine Ausführungsumgebung für Workflows gehören:

- Unterstützung langandauernder Aktivitäten mit und ohne Benutzerinteraktion,
- anwendungsabhängige Korrektheitskriterien für die Ausführung einzelner oder konkurrierender Workflows,
- angemessene Integration mit anderen Systemen, insbesondere mit File-Managern und Datenbanksystemen,
- Zuverlässigkeit und Rücksetzbarkeit in bezug auf Daten,
- zuverlässige Kommunikation von Workflow-Komponenten untereinander,
- Kompensation von Aktivitäten.

Aufgrund der somit zahlreich gegebenen Parallelen zu Fragestellungen, die für systeminterne Abläufe in Datenbanken typisch sind, liegt eine Anwendung des Transaktionskonzeptes nahe, denn dieses liefert Möglichkeiten der Garantie korrekter Ausführungen von multiplen und konkurrierenden Operationen, und zwar sowohl für einfache, systemnahe Operationen als auch für komplexe, benutzernahe Operationen, ferner syntaktische, aber auch semantische Korrektheitskriterien und Möglichkeiten der Behandlung von Fehlern. Es ist daher naheliegend zu untersuchen, inwieweit sich in jüngerer Vergangenheit vorgeschlagene Transaktionsmodelle für die Anwendung auf Workflow-Management-Systeme eignen oder ob neuartige Mechanismen erforderlich sind, die über Transaktionskonzepte hinausgehen.

Es wird vermutet, daß die Betrachtung *transaktionaler Workflows*, also von Workflows mit Transaktionsbasierung, fruchtbar ist, denn Workflows lassen sich in erster Näherung als Verallgemeinerung von Multidatenbank-Transaktionen auffassen. Andererseits muß es möglich sein, jetzt Operationsfolgen zuzulassen, von denen einige Transaktionen, andere elementar sind. Von besonderem Interesse ist ferner die Frage eines angemessenen *Korrektheitsbegriffs*, bei welchem lokale von globaler Korrektheit unterschieden wird.

Bisher hat eine eher „zentralistische Sicht“, auf Geschäftsprozesse vorgeherrscht. So sind zunächst im wesentlichen Anwendungen in der Diskussion, die die Koordination von Abläufen im Büro innerhalb einer Umgebung unterstützen (z.B. Schadenfallabwicklung in einer Versicherung). Für die Zukunft wird erwartet, daß ein enormes Potential für den Einsatz solcher Systeme in einem dezentralen, mobilen Umfeld liegt. Dies setzt Arbeiten für den Fragmentierung von Prozeßmodellen und Interoperabilität von Workflow-Systemen sowie Arbeiten im Bereich mobiler Endgeräte und deren Kopplung an die Koordinations- und Kooperationssoftware voraus. Auf der Basis dezentraler Lösungen lassen sich partnerübergreifende Anwendun-

gen (unternehmensübergreifende Workflows etwa im Zulieferbereich) sowie Anwendungen „außerhalb des Bürobereiches“, entwickeln (etwa im medizinischen Bereich (Krankenhäuser) oder im Baubereich (Baustellen)).

Groupware-Systeme: Neben Workflow-Management Systemen haben sich Groupware-Systeme im Bereich der Koordinations- und Kooperationssoftware etabliert. Damit stehen verschiedene Systeme zur Verfügung, die sich für jeweils spezifische Prozeßarchetypen eignen (Workflow für stark-strukturierte Prozesse, Groupware für schwach strukturierte Prozesse). In der Praxis vermischen sich aber die Anforderungen, d. h. Benutzer benötigen in ihrer Arbeit häufig Unterstützung aus beiden Systemklassen. Zum gegenwärtigen Zeitpunkt stehen Workflow-Management und Groupware Systeme noch isoliert gegenüber. Das kann in bestimmten Anwendungsfällen die Verwendung verschiedener Systeme mit unterschiedlicher Benutzerphilosophie, verschiedenen Benutzeroberflächen etc. bedeuten. Hier ist die Entwicklung eines Dienstbaukastens CSCW gefordert, der verschiedene Formen der Koordinations- und Kooperationsunterstützung zusammenbringt und dem Benutzer je nach Art der Aufgabe, in die er involviert ist, die geeignete Form der Prozeßunterstützung bietet.

Ein Trend der Entwicklung von Informationssystemen ist die Auslagerung spezifischer Informationen in ein dem Informationssystem angegliedertes Modell. Im Fall von Workflow-Management-Systemen beispielsweise wird die Steuerungslogik, aber auch aufbauorganisatorische und datenbezogene Informationen in einem Workflow-Modell hinterlegt. Dieser Trend zur Bildung von Unternehmensmodellen wird zukünftig ein wesentlicher Bestandteil der Informationssystem-Entwicklung sein. Dabei werden auch verschiedene Informationssysteme auf zentrale, d. h. logisch zusammenhängende Unternehmensmodelle zugreifen. Hierbei ergeben sich eine Reihe von Problemen, die von der Wartung solcher Unternehmensmodelle über die Verteilung von Modellbanken bis hin zur Schaffung von standardisierten Schnittstellen für Modellbanken reichen. Darüber hinaus stellt sich die Integration beispielsweise von bereits existierenden Datenmodellen als Problematik. Hier ist eine Migrationsstrategie zu integrierten Unternehmensmodellbanken zu leisten.

Ein weiterer wesentlicher Aspekt für eine adequate Unterstützung von informations- und kommunikationstechnischen Systemen in verteilten Strukturen betrifft den Betrieb großer Netze. Unter diesem Punkt sollen Fragen der Struktur und des Managements großer Netze wie auch Fragen der Sicherheit in großen Netzen verstanden werden. Gerade die öffentlichen Diskussionen zu dem letzten Punkt zeigen in punkto Internet, wie fehlende Sicherheitsmechanismen die Entwicklung und den Einsatz neuer Anwendungen (z.B. „Elektronischer Marktplatz“) verhindern.

4.1.6 Formale Grundlagen des World-Wide Web

Durch das World-Wide Web (WWW) ist es seit einiger Zeit auf einfache Art möglich, Zugang zu Informationen zu erhalten, die über das ganze Internet verteilt sind. Selbst für unerfahrene Benutzer ist es einfach, durch das Web zu navigieren, indem von einem Startdokument aus der Reihe nach Hyperlinks weiterverfolgt werden. Dem großen und schnell wachsenden Informationsangebot im WWW versucht man heute durch die Bereitstellung von Suchwerkzeugen („Suchmaschinen“) zu begegnen, die über die reine Navigation dadurch hinausgehen, daß man gezielter als bei einer sequentiellen Suche zugreifen kann. Eine Suchmaschine liefert im allgemeinen zu einer thematischen Vorgabe in Form eines Schlüsselworts eine Liste von Web-Seiten mit (mehr oder weniger relevanter) Information zu dem ge-

wünschten Thema. Technisch handelt es sich jedoch hierbei lediglich um Indexserver, die oft Verweise aus dem Web einfach lokal vorhalten und zur Beantwortung einer Anfrage diese durchsuchen.

Diese Form der Web-Nutzung bedient sich lediglich gewisser Techniken, die aus dem Bereich des Information-Retrieval adaptiert werden können. Demgegenüber läßt sich auch die Ansicht vertreten, daß das Web als riesige Datenbank aufgefaßt werden kann und aus diesem Grund Datenbank-Techniken, gegebenenfalls geeignet an die Charakteristika des Web angepaßt, eingesetzt werden können. Zu diesen Charakteristika gehören:

- globale Verteiltheit der Information
- dezentrale Organisation des Web
- hohe Eigendynamik der beteiligten Server
- Koexistenz von Daten aus File- sowie aus Datenbanksystemen
- semistrukturierte Information
- im Prinzip unendliche Größe

Ein Studium der formalen Grundlagen des Web befindet sich gerade erst in den Kinderschuhen. Herkömmliche Datenbanktechniken sind nicht ohne weiteres übertragbar, da man es jetzt mit unendlichen Strukturen zu hat. Zu den untersuchenswerten Aspekten in dem Zusammenhang gehören u.a. die folgenden:

- Auffinden eines geeigneten formalen Modells des Web als Datenbank, welches es insbesondere erlaubt, die klassische Unterscheidung zwischen Schema und Instanz beizubehalten.
- Präzisierung des Begriffes einer Anfrage in Rahmen eines solchen Modells, wobei die bekannten Charakteristika wie Berechenbarkeit und Generalizität angemessen zu übertragen sind.
- Entwurf einer beispielsweise auf SQL basierenden Anfragesprache für das Web, welche insbesondere der Möglichkeit eines navigierenden Zugriffs Rechnung trägt.

4.2 Spezielle Themen zur Basis von Informationssystemen

4.2.1 Grundlagen objekt-relationaler Datenbanken

Relationale Datenbankmanagementsysteme verwalten Daten, die ausschließlich einfache, „atomare“, Strukturen besitzen. In vielen Anwendungen sind diese atomaren, „flachen“, Strukturen zur Darstellung der relevanten Sachverhalte unzureichend, so daß dort andere Lösungen notwendig sind. Um auch in diesen Domänen die Vorteile der Datenbanktechnologie nutzen zu können, wurden objektorientierte Datenbankmanagementsysteme (OODBMS) entwickelt, die insbesondere die Modellierung komplexer Strukturen und ihres Verhaltens unterstützen sowie zusätzliche Vorteile bezüglich einer durchgängigen Entwicklungsmethodik bieten.

Trotz ihrer Vorteile konnten sich objektorientierte Datenbanken in größeren, von relationaler Technologie dominierten Anwendungsdomänen bislang nicht durchsetzen. Objekt-relationale Datenbanksysteme werden seit kurzem als die nächste Generation relationaler Systeme diskutiert. In der Essenz geht es darum, die positiven Eigenschaften relationaler Datenbanken

(SQL-Zugriff auf Tabellen bzw. einfach strukturierte Objekte) mit denen objekt-orientierter (angemessene Modellierung komplex strukturierter Daten) zu kombinieren. Etablierte Hersteller sind dabei, ihre relationalen Produkte mit objekt-orientierten Fähigkeiten auszustatten; neue Firmen zielen unmittelbar auf eine solche Kombination von relationalen und objekt-orientierten Aspekten in ihren Systemen. Beispiele der ersten Kategorie sind DB2 (Client/Server-Version, z. B. für Solaris, demnächst auch MVS-Version 5) oder Oracle (Version 8); Beispiele der zweiten sind Iltustra oder UniSQL. Auch im Zusammenhang mit SQL3 wird ein solcher „Merger“, seit längerem diskutiert; hier versteht man darunter konkret die folgenden Aspekte, welche kollektiv auch als die *Major Object-Oriented SQL Extensions* (MOOSE) bezeichnet werden:

- Erweiterung des bisherigen Typsystems von SQL um abstrakte Datentypen (ADTs), wobei unterschieden wird zwischen Werttypen (engl. Value Types) als Verallgemeinerung des Domain-Konzepts von SQL2 und Objekttypen (engl. Object Types) zur Unterstützung von Objekten mit wertunabhängiger Identität.
- ADTs können in ihrer Struktur auf die Typkonstrukturen LIST, SET und MULTiset zurückgreifen.
- Spezialisierung, d.h. ein Typ kann als Untertyp eines oder mehrerer anderer Typen (Obertypen) vereinbart werden, wobei jeder Untertyp genau einen (direkten oder indirekten) „maximalen“, Obertyp hat, welcher seinerseits keinen weiteren Obertyp besitzt. Die Möglichkeit der Bildung von Untertypen gilt gleichermaßen für Wert- und für Objekttypen.
- Tabellenverbände als Form der Spezialisierung für Relationen: Eine Tabelle kann als Untertabelle einer oder mehrerer anderer Tabellen definiert werden.
- Benutzer-definierte Funktionen als Bestandteile eines ADTs, welche in SQL selbst oder als externe Funktionen realisiert werden können.
- Methoden- bzw. Funktionsaufrufe im Rahmen von SELECT-Ausdrücken.

Während es sich auf den ersten Blick um eine naheliegende Maßnahme handelt, die beiden Ansätze (objekt-orientiert und relational) zusammenzuführen, ist jedoch keineswegs klar,

- daß sich die jeweils separat erzielten Ergebnisse zu einzelnen Realisierungsfragen unmittelbar übertragen bzw. einfach zusammensetzen lassen,
- ob sich aus der Symbiose nicht neue Möglichkeiten ergeben, Realisierungen zu erstellen. Konkret erscheinen die folgenden Aspekte untersuchenswert:
- Ausdruckskraft und Komplexität von Anfragesprachen, insbesondere von SQL-Erweiterungen
- Unterstützung von navigierenden Fähigkeiten auf der Ebene der Anfragesprache
- Anfrage-Optimierung

oder zu integrieren, zu betreiben und zu warten und dabei in geeigneter Weise Basistechnologien zu nutzen. Bei diesen Basistechnologien handelt es sich sowohl um Informatik-Methoden (z.B. Petrinetze, Objekt-Orientierung) und -systeme (z.B. Datenbanksysteme) als auch um Methoden anderer Disziplinen.

Große Informationssysteme haben zahlreiche Anwendungsgebiete, z.B. im kaufmännisch-administrativen Bereich, in der Betriebs- und Volkswirtschaftslehre, der Technik, den Naturwissenschaften, im Recht, der Medizin, in der Kunst und vielem mehr. Die EMISA konzentriert sich *nicht* auf einen dieser Bereiche. Sie stimmt allerdings mit dem „Software Engineering-Weltbild“, überein, daß Informationssystementwicklung in die folgenden Teilbereiche zerfällt:

1. Anwendungsbereich
2. Spezifikation
3. Rechnersystem

Die EMISA, die sich mit der Konstruktionslehre für Anwendungssysteme befaßt, liegt thematisch in der Schnittmenge von Basistechnologien (z.B. Datenbanksysteme), Software-Engineering, Fachinformatiken und berührt weitere Gebiete. Sie befaßt sich dabei insbesondere mit Methoden, Techniken und Plattformen sowie Anwendungen einzelner Fachgebiete. Darüber hinaus orientiert sie sich an Sprachen, die im Lebenszyklus eines Informationssystems rekonstruiert werden müssen und die als Mittel der Systementwickler, Systembenutzer und Systembetreiber in die Arbeitsprozesse mit Informationssystemen einfließen.

- Transaktionsverarbeitung unter Ausnutzung semantischer Information
- Speicherungsstrukturen

Bisher ist keine systematische Untersuchung des Einsatzes der vielfältigen Konstrukte objekt-relationaler DBMS vorgenommen worden. Interessant sind hier Orthogonalität und Minimalität der angebotenen Sprachkonstrukte sowie Kompatibilität mit bisherigen Standards. Weiterhin sind Kriterien zur Auswahlentscheidung bezüglich relationaler, objektorientierter oder objekt-relationaler Datenbanken in unterschiedlichen Anwendungsgebieten zu untersuchen und evtl. um eine Methodik zu ihrem Einsatz zu vervollständigen.

4.2.2 Transaktionssysteme

Informationssysteme verwalten große Datenmengen und stellen sie zur gleichzeitigen Nutzung für unterschiedliche Anwendungen zur Verfügung. Dabei ist die Integrität der Datenbestände sowohl bei zeitgleichem Zugriff aus unterschiedlichen Applikationen als auch beim Auftreten von Systemfehlern sicherzustellen. Grundlegende Konzepte und Mechanismen für die Realisierung von Mehrbenutzerfähigkeit und Fehlerbehandlung bieten die bekannten Transaktionsmodelle und -systeme. Die von vielen Systemen angebotenen, geschlossenen (z. B. flachen) Transaktionen bereiten jedoch Probleme bei lang andauernden Aktivitäten, da sie im wesentlichen auf exklusiven Sperren basieren. Zur Erhöhung der Parallelität in derartigen Anwendungsbereichen existieren bereits diverse Vorschläge für weitergehende Transaktionsmodelle, die die zentralen Eigenschaften des flachen Transaktionsmodells - das sind Isolation und Recovery von Transaktionen - beibehalten. Für neuere Anwendungsgebiete, etwa Workflow-Management-, Autoren- und Projektführungssysteme, ergeben sich aus den Eigenschaften dieser Transaktionsmodelle jedoch Einschränkungen für die Kooperation der in einem Prozeß involvierten Agenten. Neuere Transaktionsmodelle, die in derartigen Fällen das Konzept der Isolation durch geeignete Kooperationsmodelle ersetzen, sollten untersucht und bezüglich der Anforderungen in unterschiedlichen Anwendungsbereichen bewertet werden. Interessant ist hierbei auch die Frage ihrer architektonischen Einordnung in Anwendungen, insbesondere ihre Realisierung in DBMS, Middleware-Komponenten (Transaktionsmonitor, Object-Broker) oder Anwendungen wie Workflow-Management-Systemen.

4.2.3 Zum Begriff „Information“,

Der Begriff der Information ist in der Informatik keineswegs gesichert definiert. Die Grunddefinition der Shannonschen Informationstheorie (Theorie der Entscheidungsinformation) ist für die Zwecke der EMISA zu eng. Mit der Auffassung, die die Kommunikationslehre in der Kommunikations-Psychologie kennt, kann EMISA schon eher leben: Hier wird unter Information eine Nachricht (Daten) zusammen mit deren Interpretation verstanden. Die Information ist dabei auch abhängig vom Zeitpunkt und dem Rezipienten der Nachricht. Wie ist die Information im Umfeld von EMISA wirklich definiert?

5. Zusammenfassung und Ausblick

Das Treffen hat dazu beitragen, das eigenständige Profil der EMISA wieder klarzustellen und herauszuarbeiten: Im Prinzip geht es der EMISA darum, Informationssysteme herzustellen