Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2003 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2003

The GRITICKA Ontology for Modeling Multiagent-Based Integrative Business Information Systems

Hong Zhang
Southwest Missouri State University

Rajiv Kishore
State University of New York at Buffalo

Ram Ramesh
State University of New York at Buffalo

Raj Sharman
State University of New York at Buffalo

THE GRITICKA ONTOLOGY FOR MODELING MULTIAGENT-BASED INTEGRATIVE BUSINESS INFORMATION SYSTEMS

Hong Zhang
Southwest Missouri State University
hoz565f@smsu.edu

Rajiv Kishore
State University of New York at Buffalo
rkishore@buffalo.edu

Ram Ramesh
State University of New York at Buffalo
rramesh@buffalo.edu

Raj Sharman
State University of New York at Buffalo
rsharman@buffalo.edu

Abstract

While multiagent technology has been widely applied in various business areas, there is no unified and coherent multiagent system architecture and conceptual grammar for modeling integrative business information systems. In this paper we propose a special-purpose ontology for Multiagent-based Integrative Business Information System (MIBIS) called the GRITICKA ontology that provides a modeling grammar for the MIBIS universe. We hope that this ontology will help in developing ontology-driven MIBIS applications with a high degree of reuse.

Keywords: Multiagent systems, ontology, integrative business information systems, reuse

Introduction

While multiagent technology has been widely applied in various business areas, there is no unified and coherent multiagent system architecture and conceptual grammar for modeling integrative business information systems. A consequence of this situation is that system developers have to craft individual application systems from scratch each time. Ontological approaches for multiagent system development not only allow knowledge-level reuse, but also improve system quality by providing guidance for system developers to select appropriate system components. The main contribution of this paper is the development of a special-purpose ontology termed GRITICKA for the Multiagent-based Integrative Business Information Systems (MIBIS) universe. We develop this ontology by integrating relevant and necessary concepts pertinent to both the domains of multiagent systems and integrative business information systems. The main motivation for developing GRITICKA ontology is that this ontology will help in developing ontology-driven MIBIS systems with a high degree of reuse. Our approach is similar to efforts such as the Smart Object Model (Vaishnavi, et al., 1997) developed for modeling of complex operations management systems, the SEAM model (Bajaj and Ram, 2002) developed for modeling of workflow systems, the REA Ontology (Geerts and McCarthy, 2000) developed for enterprise accounting information systems, and the knowledge-based approach proposed by Sugumaran et al. (2000) for modeling and developing business information systems. However, the GRITICKA ontology provides a comprehensive set of foundation constructs, their semantics, and their relationships for modeling systems in the MIBIS universe, which other existing ontologies and modeling grammars do not provide. Further, our ontology specifies mechanisms for integrating the individual GRITICKA constructs for the MIBIS universe.

Multiagent-Based Integrative Business Information Systems

An *Integrative Business Information System* (IBIS) is a complex information system that integrates multiple work *systems* in a single information system. For example, an ERP system is an IBIS because it integrates a number of work systems including customer order management, inventory management, product pricing, and logistics management in a single information system. An IBIS supports and coordinates the tasks performed by multiple actors who are individually neither capable nor knowledgeable about all the different tasks in the entire workflow supported by the integrative system (Kishore, et al., 2003). A *multiagent system* is defined as a loosely coupled network of problem solvers that interact to solve problems that are beyond the individual capabilities or knowledge of each problem solver (Durfee and Lesser, 1989). While IBIS systems support a number of organizational actors who interact with one another while performing tasks as part of business processes to achieve their own as well as agreed upon and shared business goals, multiagent systems are essentially "communities" of autonomous and problem-solving agents that interact using computer networks. Multiagent systems architecture, therefore, appears to be quite appropriate for conceptualizing and implementing IBIS systems. This paper applies the multiagent systems paradigm for developing IBIS systems and refers to such information systems as MIBIS, short for **M**ultiagent-based **I**ntegrative **B**usiness **I**nformation **S**ystems.

Integrating multiagent systems architectures with integrative business information systems, a MIBIS entity possess the following characteristics: 1) a MIBIS entity is a distributed multiagent system; 2) there is no global control in the MIBIS entity; 3) agents within a MIBIS entity coordinate their tasks through conversational interactions; and 4) agents play organizational roles within the MIBIS entity to solve organizational problems and meet business goals.

Ontology-Driven Development of Multiagent-Based Integrated Business Information Systems (MIBIS)

Ontology – or the nature of being, has been a subject matter of study in the field of philosophy for a long time and it provides a description of the essential properties and relations of *all* beings in *the* universe. In recent times, however, the notion of ontology has been expanded in the field of artificial intelligence which now uses the term to refer to not one but multiple ontologies, each of which refers to the specification of knowledge about entities and their relationships in a limited knowledge universe only (Chandrasekaran, et al., 1999). Regarded as an engineering artifact, an ontology in IS has recently been defined as "a representational vocabulary for a discourse universe that is complete with appropriate syntax and semantics, and which can be utilized to model domains of interest within the discourse universe" and have been classified into general-purpose and special-purpose ontologies (Kishore, et al., Forthcoming). General-purpose ontologies are used to describe unbounded universes of discourse whereas special-purpose ontologies are used to describe bounded universes of discourse. For example, UML and ER modeling are general-purpose ontologies because they can be used to model any information system whereas the REA ontology (Geerts and McCarthy, 2000) and the Smart Object Model (Vaishnavi, et al., 1997) are used to model enterprise accounting information systems and complex operations management systems, respectively. While general-purpose ontologies such as UML provide a rich vocabulary and strong syntax for aiding the analysis and design activities during the course of development of new IS, they lack the semantics that are specific to the context of the MIBIS bounded universe, creating the need for a special-purpose ontology for this discourse universe. For instance, literature shows that most enterprise and workflow models lack the semantics to concisely represent specific activities, tasks, business processes, business goals, and organization structures of a business (Scheer, 1999; Weigand and Heuvel, 2002). This is quite true for the MIBIS universe as well, because domain-specific semantics, such as autonomy, reactivity, distributed control, interactions, etc., are not supported adequately by general-purpose ontologies. Further discussion about the efficacy of individual general-purpose ontologies for the MIBIS universe is beyond the scope of this paper due to word-count limitations.

The GRITICKA Ontology

The GRITICKA ontology extracts core concepts from both IBIS and multiagent systems literature to form a foundation ontology for the MIBIS bounded universe. It is an acronym for the concepts are central to the MIBIS universe: **G**oal, **R**ole, **I**nformation, **T**ask, **I**nteraction, **C**apability, **K**nowledge, and **A**gent. This ontology is not only descriptive in that it describes the essential orthogonal dimensions that are at the root of the MIBIS universe, it is also a prescriptive ontology that provides a representational language, constructs, and semantics for guiding the analysis and design of MIBIS systems and is discussed below.

The Environment-Level Architecture

At the environmental level, a MIBIS is regarded as a black box, interacting with other entities in the business environment. The environment-level concepts are defined to specify the scope and user requirements for a MIBIS system. The GRITICKA ontology provides the concepts of MIBIS Entity, Non-MIBIS Entity, User, and Goal at this level (Figure 1). A MIBIS entity refers to a multiagent-based integrative business information system. A non-MIBIS entity refers to traditional information systems that do not possess intelligence to deal with dynamic environments. A user is a human user who utilizes the MIBIS entity. Each MIBIS entity is associated with one or more business goals. Goals are regarded as a fundamental concept for MIBIS requirement specification because MIBIS entities exist to meet specific business goals. Further, systems developed based on goals are claimed to be more stable than those based on functions, processes or information structures that often change with time (Deloach, et al., 2001; Kendall, et al., 1998). Also, goal-oriented analysis aids in the elicitation and elaboration of requirements, relating system requirements to organizational and business contexts, clarifying requirements with stakeholder groups, and dealing with conflicts among them (Yu and Mylopoulos, 1998). Therefore, a MIBIS entity is represented as a goal tree at this level. Interactions among MIBIS entities, non-MIBIS entities, and users are modeled to help in the construction of the goal tree. Goal being one of the key constructs necessary for modeling a MIBIS entity is part of the foundation GRITICKA ontology for the MIBIS universe.

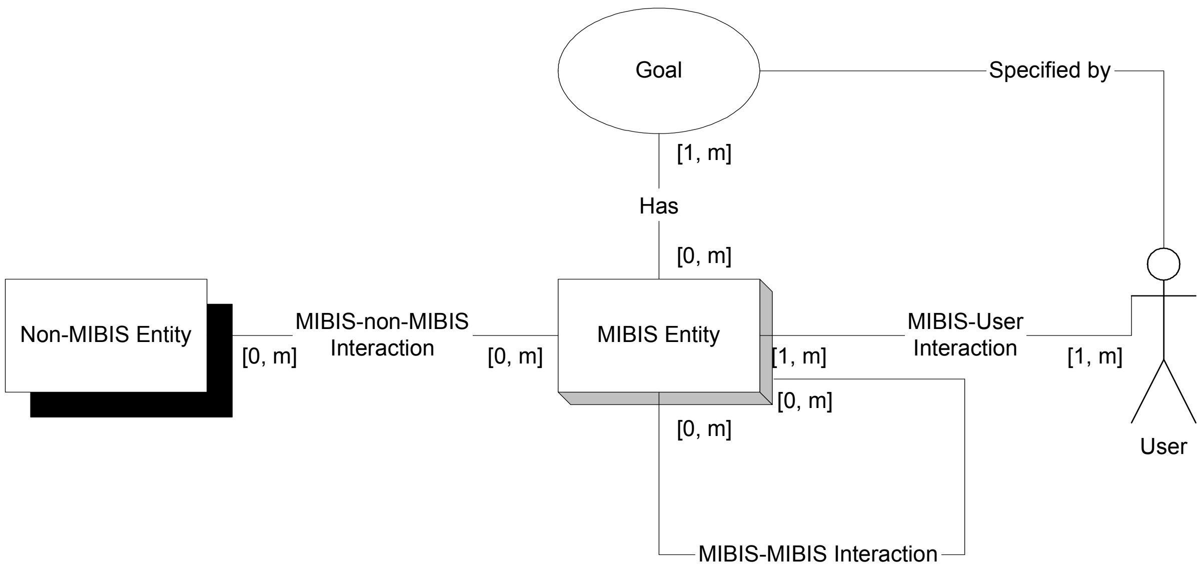

Figure 1. The Environmental-Level Architecture of MIBIS Entities

The System-Level Architecture

At the system level, a MIBIS entity is a white box and the ontological categories at this level capture the concepts that are essential to model individual agents within the MIBIS entity and their interactions with each other during the course of task performance for achievement of business goals specified at the environmental level. These concepts include Goal, Role, Information infrastructure, Task, Interaction, and Agent (see figure 2).

The goal construct provides the bridge between the environmental- and system-level architectures. A system-level goal is essentially a leaf goal in the goal tree that is created at the environmental level. The leaf goals are assigned to roles. Role, defined as “a collection of duties and rights” (Biddle and Thomas, 1966), is identified as a central construct in the GRITICKA ontology because work in organizations is usually assigned to roles as opposed to specific named individuals (Basu and Kumar, 2002). On one hand, usage of role in multiagent systems modeling has advantages of design focus, reusability, and flexibility (Cabri, 2001). For example, the Gaia methodology and MESSAGE/UML both deploy role concept in analysis phase (Caire, et al., 2001; Wooldridge, et al., 2000). On the other hand, role is a basic construct in IBIS systems modeling (Basu and Kumar, 2002; Kumar, 1999; Kumar, et al., 2002). The role construct in the GRITICKA ontology is essentially an abstraction for the tasks that are necessary to be performed to achieve an individual goal, the interactions that need to occur with other roles to successfully perform the required tasks, and the information that needs to be accessed or will be generated during the course of performance of those tasks. Just as in the organizational context where an abstract role is assigned to one or more physical human agents (e.g., the abstract role of an inventory manager may be assigned to a physical human agent named John Q. Batman), an abstract role in the MIBIS universe is assigned to and implemented by physical software agents.

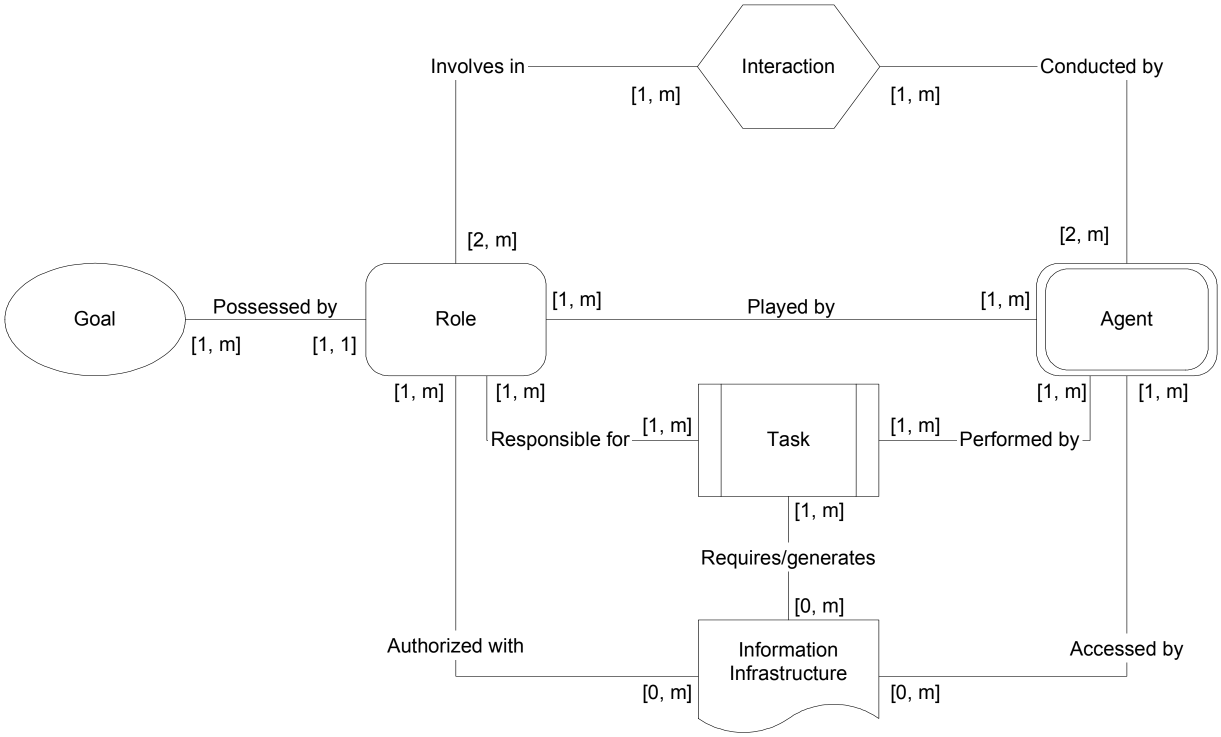

Figure 2. The System-Level Architecture of MIBIS Entities

A task is an activity that is performed by a role to achieve one or more goals. A task can be as simple as a single activity or as complex as a complete business process. A task has both functional and behavioral aspects to it. The functional perspective describes the content of a task, i.e., what a role should perform. The behavioral perspective specifies how a task should be performed including its sequence and timing, its frequency and repetitiveness, and whether it is optional. Tasks can be decomposed and participate in *aggregation/decomposition* relationship with other tasks. Tasks also participate in *precedence and concurrence* relationships with other tasks to describe interdependencies among them. If task A *precedes* task B, then task A must be performed before task B. If task A is *concurrent* with task B, then task A and B must be performed in parallel. If task A is *not concurrent* with task B, then either task A or task B is performed, but they can not be performed at the same time. Because tasks are the means by which system goals are accomplished and they define the duty of a role, they are included as a foundation construct in the GRITICKA ontology for the MIBIS universe.

Interaction is the mechanism by which interdependencies among roles are coordinated. Interdependencies among roles arise due to interdependencies between tasks performed by different roles, simultaneity constraints, task/subtask relationships, and shared resources (Malone and Crowston, 1994). An interaction in the GRITICKA ontology is defined as a coordinated sequence of speech acts or communicative acts aimed at defining and reaching a goal (Dietz, 2001; Winograd, 1987). Speech acts view human natural language as actions, such as requests, suggestions, commitments, acceptance, cancellation, etc, which may affect individual's future courses of action. An interaction in GRITICKA has the following properties: *initiator, responder, states, transition rule,* and *communicative act*. An initiator is the role that starts an interaction with a communicative act. A responder is the role addressed by the initiator and the one expected to respond to the communicative act utilized by the initiator. A state captures the current state of affairs between a pair of initiator and responder and includes such states as message sent, waiting for response, etc. A transition rule governs the flow of interactions from one state to the next. Interactions are also included in the GRITICKA ontology as a foundation construct because this is a central mechanism for coordination in both multiagent systems and integrative business information systems.

Information infrastructure refers to the information resources available in a MIBIS system and may pertain to both the functional and the non-functional aspects of the MIBIS system. The functional information resources pertain to the business functionality that is provided by the MIBIS system whereas the non-functional information resources pertain to operational, performance, and information assurance aspects of the MIBIS system. Information resources are represented as regular entities described by attributes and may participate in *association, generalization,* or *aggregation/composition* types of relationships with other information resources. Information infrastructure is obviously a key aspect of any information system and the MIBIS universe is no exception. Therefore, we include information infrastructure as a foundation construct in the GRITICKA ontology.

An agent is the entity that physically implements a role. An agent inherits the goals, duties, and rights from the roles it plays and actually performs the tasks, engages in interactions with other agents, and accesses and updates the required information infrastructure. Figure 2 above depicts the GRITICKA constructs and their logical relationship in graphic symbols at the system level.

The Agent-Level Architecture

A software agent is a computer system that is situated in a particular IT environment and is capable of autonomous action in this environment in order to meet its design objectives (Wooldridge and Jennings, 1995). Software agents exhibit properties of autonomy, reactivity, pro-activeness, and social ability. The ontological categories at the agent level capture the constructs that are necessary to model the internal architecture of individual agents within a MIBIS entity.

The abstract roles defined at the MIBIS system level may be assigned to different physical agents each of who may implement the same role in a different manner due to their unique internal knowledge structure and capabilities. Knowledge is information possessed in the mind of individuals (Alavi and Leidner, 2001). Knowledge is also defined as a justified belief that increases an entity's capacity for effective action (Huber, 1991; Nonaka, 1994). Therefore, agent knowledge is defined as the set of beliefs an agent holds and the rules it uses to form new beliefs. Beliefs are facts an agent keeps about itself, the environment it resides in, other agents in the system and their beliefs, and the problem domain. Deduction rules are a set of rules which enable an agent to reason and deduct new beliefs about the itself and its environment.

Capability is defined as an agent's knowledge about actions including the agent's current actions, commitments, problem solving strategies, and social strategies. Agent's current actions and commitments describe what actions an agent is performing and what commitments it has made for task performance. Problem solving strategies and social strategies allows an agent to form different plans to perform tasks or to communicate with other agents. Before an agent can be assigned to a role or roles, it must have capabilities to fulfill the duties defined by the role/roles. Figure 3 summarizes the logical relationships among agent, capability, and knowledge infrastructure.

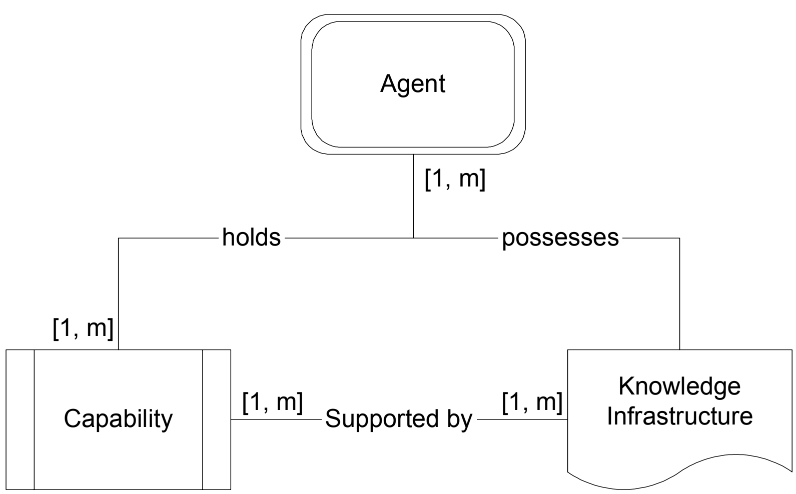

Figure 3. The Agent-Level Architecture of MIBIS Entities

Conclusion and Future Directions

In this paper, we propose a special-purpose ontology GRITICKA for MIBIS. While some other ontologies deploy similar concepts, our ontology differs from them in detail semantics and the universe focused. For example, the organization ontology for the TOVE project (Fox, et al., 1998) is developed for enterprise modeling and therefore introduces role hierarchy for control relationship in the enterprise. Our ontology focuses on integration of multiple work systems, and therefore we only consider peer-to-peer relationship in the role concept. SEAM (Bajaj and Ram, 2002) focuses on traditional workflow modeling by deploying the concepts of State, Entity, and Activity, while ours distinguish Entity into Agent, and Information to allow more flexible modeling of adaptive business systems.

The GRITICKA ontology provides a start point for ontological study of MIBIS. We are currently involved in 1) developing formal definitions for the GRITICKA constructs; 2) developing a GRITICKA-based MIBIS development methodology and design principles; 3) developing lower-level GRITICKA constructs to facilitate reuse; and 4) developing software tools that implement the GRITICKA ontology and development methodology to support MIBIS analysis, design, and development.

References

Alavi, M., and Leidner, D.E. "Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues," *MIS Quarterly* (25:1), 2001, pp. 107-136.

Bajaj, A., and Ram, S. "SEAM: A State-Entity-Activity-Model for a Well-Defined Workflow Development Methodology," *IEEE Transactions on Knowledge and Data Engineering* (14:2), 2002, pp. 415-431.

Basu, A., and Kumar, A. "Research Commentary: Workflow Management Issues in e-Business," *Information Systems Research* (13:1), 2002, pp. 1-14.

Biddle, B.J., and Thomas, E.J. *Role Theory: Concepts and Research*, John Wiley & Son, Inc, 1966.

Cabri, G. "Role-based Infrastructures for Agents," *Proceedings of the 8th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS 2001)*, Bologna, 2001.

Caire, G., Leal, F., Chainho, P., Evans, R., Francisco Garijo, J., Gomez, J.P., Kearney, P., Stark, J., and Massonet, P. "Agent Oriented Analysis using MESSAGE/UML.," *Proceedings of the Agent-Oriented Software Engineering (AOSE)*, Montreal, CA, 2001, pp. pp. 101-108.

Chandrasekaran, B., Josephson, J.R., and Benjamins, V.R. "What Are Ontologies, and Why Do We Need Them?," *IEEE Intelligent Systems* (14:1), 1999, pp. 20-26.

Deloach, S.A., Wood, M.F., and Sparkman, C.H. "Multiagent Systems Engineering," *International Journal on Software Engineering and Knowledge Engineering* (11:3), 2001, pp. 231-258.

Dietz, J.L.G. "DEMO: Towards a Discipline of Organisation Engineering," *European Journal of Operational Research* (128:2), 2001, pp. 351-363.

Durfee, E.H., and Lesser, V.R. "Negotiating Task Decomposition and Allocation Using Partial Global Planning," In *Distributed Artificial Intelligence*, L. Gasser and M. Huhns (eds.), 2, Morgan Kaufmann, San Francisco, CA, 1989, pp. 229-244.

Fox, M.S., Barbuceanu, M., Gruninger, M., and Lin, J. "An Organisation Ontology for Enterprise Modeling," In *Simulating Organizations: Computational Models of Institutions and Groups*, L. Gasser (ed.) AAAI/MIT Press, Menlo Park CA, 1998, pp. 131-152.

Geerts, G., and McCarthy, W.E. "The Ontological Foundation of REA Enterprise Information Systems," Michigan State University), 2000.

Huber, G. "Organizational Learning: The Contributing Processes and the Literatures," *Organization Science* (2:1), 1991, pp. 88-115.

Kendall, E.A., Palanivelan, U., and Kalikivayi, S. "Capturing and Structuring Goals: Analysis Patterns," *Proceedings of the Third European Conference on Pattern Languages of Programming and Computing*, Bad Irsee, Germany, 1998.

Kishore, R., Zhang, H., and Ramesh, R. "Integrative business information systems and multiagent architectures: Current approaches and future directions," Working Paper State University of New York, 2003.

Kishore, R., Zhang, H., and Ramesh, R. "Ontologies, Frameworks, and Systems: A Helix-Spindle Model for Ontological Engineering," *Communications of the ACM*), Forthcoming.

Kumar, A. "Dynamic routing and operational controls in workflow management," *Management Science* (45:2), 1999, pp. 253-272.

Kumar, A., Aalst, W.M.P.v.d., and Verbeek, E.M.W. "Dynamic Work Distribution in Workflow Management Systems: How to Balance Quality and Performance," *Journal of Management Information Systems* (18:3), 2002, pp. 157-193.

Malone, T.W., and Crowston, K. "The Interdisciplinary Study of Coordination," *ACM Computing Surveys* (26:1), 1994, pp. 87-119.

Nonaka, I. "A Dynamic Theory of Organizational Knowledge Creation," *Organization Science* (5:1), 1994, pp. 14-37.
Scheer, A.-W. *ARIS-Business Process Modeling*, Springer, Berlin, 1999.
Sugumaran, V., Tanniru, M., and Storey, V.C. "Supporting reuse in systems analysis," *Communications of the ACM* (43:11es), 2000, pp. 312-322.
Vaishnavi, V.K., Buchanan, G.C., and Jr, W.L.K. "A Data/Knowledge Paradigm for the Modeling and Design of Operations Support Systems," *IEEE Transactions on Knowledge and Data Engineering* (9:2), 1997, pp. 275-291.
Weigand, H., and Heuvel, W.-J.v.d. "Cross-Organizational Workflow Integration Using Contracts," *Decision Support Systems* (33), 2002, pp. 247-265.
Winograd, T. "A Language/Action Perspective on the Design of Cooperative Work," *Journal Of Human-Computer Interaction* (3:1), 1987, pp. 3-30.
Wooldridge, M., and Jennings, N.R. "Intelligent Agents: Theory and Practice," *Knowledge Engineering Review* (10:2), 1995, pp. 115-152.
Wooldridge, M., Jennings, N.R., and Kinny, D. "The Gaia Methodology for Agent-Oriented Analysis and Design," *Autonomous Agents and Multi-Agent Systems* (3:3), 2000, pp. 285 - 312.
Yu, E., and Mylopoulos, J. "Why Goal-Oriented Requirements Engineering," *Proceedings of the 4th International Workshop on Requirements Engineering: Foundations of Software Quality*, Pisa, Italy, 1998, pp. 15-22.