

All copyrights of this article are held by the author/s

To cite this publication

we suggest to use the APA (American Psychological Association) standard:

[Author's Last Name, Initials, (year), Title of work. Retrieved month, day, year, from source.]

You downloaded this document for free
from the



The NetAcademy is an internet research platform organizing the accumulation, dissemination and review of scientific research and publications world wide.

For any use of this document which is not strictly private, scholarly work, please contact the NetAcademy editors respectively the MCM institute at the below address.

The NetAcademy ®
www.netacademy.org
NA.editors@netacademy.org

The NetAcademy project has been developed at the
Institute for Media and Communications Management MCM
University of St. Gallen
Blumenbergplatz 9
CH-9000 St. Gallen, Switzerland
www.mcm.unisg.ch

The NetAcademy - Managing Internet Peer Review Process In a Multi-agent Framework

Lei Yu Beat F. Schmid

Institute for Media and Communications Management
University of St. Gallen
Mueller Friedberg Str. 8 , CH-9000 St. Gallen
Switzerland
Email: lei.yu@unisg.ch , beat.schmid@unisg.ch

Abstract

This paper explores managing Internet peer review process in a Multi-agent framework. We model the peer review process as a set of relating roles such as author, editor, reviewer, etc. Each role is defined in terms of goals, qualifications, obligations, permission, relations to other roles, protocols, etc. We adopt protocols to govern the interaction between roles. Roles are assigned to agents based on the evaluation of their capabilities. Coordination of workflow is achieved by communication between agents. This work shows us that the methodology for agent oriented and role based workflow management is a powerful modeling and design technique.

1. Introduction

The Internet is already moving rapidly towards providing a wide range of services to connected users. Some of these services are on-line replacements of traditional services (e.g. help desk, customer feedback and survey, shopping for merchandise, etc.). The Internet offers any individual the ability to publish and exchange information with anyone in the world as if they were in the same village - the "global village". This makes it possible to create communities of people with a common interest where distance is irrelevant.

The project NetAcademy aims to provide an Internet based platform for scientific community to aid the creation, exchanging and dissemination of scientific knowledge (<http://www.netacademy.org/>). The international Journal "Electronic Markets" is under its own NetAcademy (<http://www.electronicmarkets.org/>). Here the online Journal publishes research articles on Electronic Markets that have been accepted for print publication. Authors can online submit their articles. However, what happened behind the scenes, i.e., the internal workings of the peer review process, is not addressed.

Essentially, the peer review process is a quality control mechanism for the academic journal. Editorial peer review in the past has been secretive (because only the Editor can see the whole process), narrow (because each article is usually reviewed by only two or three people) and somewhat arbitrary (because the choice of reviewers is limited by the Editor's knowledge and the time constraints of the potential reviewers) [Bingham et al. , 1998].

Consider the Peer Review Process. We assume the review process is to select reviewers, distribute the paper(s) to the selected reviewers, have the reviewers perform the reviews and collaborate in producing a joint review document, and finally forward it to the authors. This is **an ad hoc workflow** because it involves: (1) negotiation for selecting the reviewers, and (2) collaboration between the reviewers for producing a joint review. The ad hoc workflow tasks involve human coordination, collaboration, or co-decision. Thus, the ordering and coordination of tasks are not automated but are instead controlled by humans. Furthermore, the task ordering and coordination decisions are made while the workflow is performed.

However, the current generation workflow management systems tend to operate with a central workflow engine which monitors all events in the system. They offer limited support and minimal flexibility during process enactment [Jennings et al., 1996]. With agent technology, a completely new paradigm evolved which promises to tackle the issues of distributed workflow systems more adequately [Hawryszkiewicz and Gorton, 1998] [Kaldoudi et al., 1997]. It looks at cases where agents have to coordinate themselves in order to achieve their work. In contrast to traditional workflow systems, no plan of the work to be done (process or workflow description) is existent before the work starts.

In this paper, we design the Internet peer review process management system in a multi-agent framework by applying the agent oriented and role based workflow management methodology [Yu and Schmid, 1999]. At first, we discuss and redesign the Internet peer review process in section 2. Section 3

is dedicated to the conceptual model for agent oriented and role based workflow management. We apply the role and protocol based analysis in the Internet peer review process in detail in section 4 and 5. The Multi-agent framework for Internet Peer Review Process Management System is proposed and discussed in section 6 and 7. Finally, the work is summarized in section 8.

2. The Internet Article Peer Review Process of NetAcademy

We intend to develop an Internet peer review and publish system for the NetAcademy. The redesigned Internet Peer Review Process is described as follows:

Step 1: An author submits an article to the executive editor, completing with paper identifier (authors, title, etc.), an abstract, a set of keywords.

Step 2: If the executive editor accepts it, then he places the article on the journal's Website.

Step 3: Executive editor selects reviewers for the article according to procedures described below and appoints the copyeditor for the article.

Suppose the editor aims to ensure quality by having each submission reviewed by three referees. The editor then broadcasts to the network of referees that it has a review job to do with a proposed resource allocation of 3 weeks, and invites bids to undertake the work. If enough reviewers respond favorably, then three of them are chosen and are appointed.

If insufficient reviewers accept the initial terms, then the editor selects what it considers to be a set of appropriate referees based on the referee's areas of expertise and the submission's keywords, and engages in one-to-one negotiation with each in the attempt to get a commitment to review the submission. At first the editor sends a proposal to potential reviewers. The terms of the proposal maybe "weak" than the last request. The receiver may reject it, accept it, or provide a new offer to the editor. This continues until either enough reviewers have been found or the set of potential reviewers has been exhausted. In the latter case an alert is raised.

Step 4: Executive editor invites the author, selected reviewers and copyeditor to participate in the Online discussion. Access by password is restricted to these participants.

Step 5: Review process conducted as online discussion [Bingham et al., 1998]. Executive editor moderates the discussion. He introduces and concludes the main stages of review. (1) Discussion of the first submission, concluding with the editor's decision to reject, or definitive instructions for revising the article (3 weeks); (2) Discussion of the authors' revision, concluding with the editor's decision to accept or reject, or to revise again (2 weeks); (3) Discussion of the copyedited article, concluded when the copyedited article is finalized and approved by the authors (1 week). Commissioned reviewers post their reviews to the discussion list associated with the article; The author observes the discussion of the paper and can respond

to comments with explanations, alternative arguments or proposals for revising their article; Copyeditor prepares accepted articles for publication.

Step 6: After revision, acceptance and editing, the article and its associated discussion list are published for open review. Readers can contribute comments; open review is concluded when the author and editor agree that no further changes are desirable (4 weeks).

Step 7: The article is allocated to a print-issue of the Journal and published in print. The on-line Magazine is updated.

The infrastructure technology currently used by ad hoc WFMSs ranges from "enhanced" electronic mail to group calendar and conference systems. Based on these infrastructure, we are about to handle the reviewer selection, communication and the commitments automatically through agent technology. The Internet Peer Review System (IPRS) is designed as a distributed multi-agent systems based on World Wide Web. To do this, the author, editor, and each referee have their own representative, a personal agent, which enables them to participate in this distributed peer review process.

3. Towards Agent Oriented and Role Based Workflow Management

In a previous work [Yu and Schmid, 1999], we view a business process as a collection of autonomous, problem solving agents which interact when they have interdependencies. Corresponding to the agent oriented view of a business process, we model a business process as a set of relating roles.

A *role* refers to the expected behavior patterns an agent must perform. Roles are defined in terms of goals, obligations, concurrency constraints, permissions, qualifications, relationships to other roles and protocols. An *obligation* is a prescription that a particular behavior is required. An obligation is fulfilled by the occurrence of the prescribed behavior. It defines the functionality of a role. A *permission* is a prescription that a particular behavior is allowed to occur. Roles are assigned to agents based on the evaluation of qualification and capabilities. Once a role is assigned to an agent, the agent inherits the obligations and permissions specified in that role.

Ongoing interactions between roles often fall into typical patterns called protocols. A *protocol* is specified by a set of rules governing the conversation among agents. A conversation is a set of interactions in order to ask for a service or request or update information. A simple interchanged message has the following attributes: speech-act, agent communication language, knowledge representation language, synchronization, sender, receiver, and ingredients, etc. The messages exchanged are represented as performatives (KQML speech acts) of the agent communication language. Coordination of workflow is achieved by communication between agents.

We present the conceptual architecture of workflow to illustrate the definitions. The architecture shown in Figure 1 builds upon the general ideas. It lays out the workflow conceptual entities and their relationship.

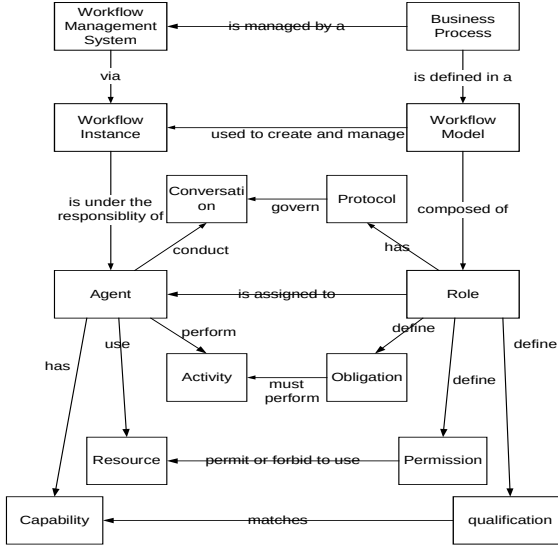


Figure 1 Conceptual Model of Agent Oriented and Role Based Workflow Systems

The vision of agent oriented and role based workflow management established the framework for the practical work. In the next sections we apply them in NetAcademy to develop the Internet Peer Review System.

4. Roles Determination And Specification

At first, the goals of a workflow must be defined and decomposed into sub-goals. The goal of the Peer Review Process is to control the quality of a publication. It may be decomposed into these sub-goals: submit an article, register an article, find reviewers for the article, distribute the article to the reviewers, review the article, conduct online discussion, make a review decision, revise the article, copyedit the article, etc.

Then these goals are assigned to roles. The above sub-goals may be assigned into elementary roles or roles like author, reviewer-selector, potential reviewer, reviewer, executive editor, copyeditor, etc. Although some names of roles are similar to those of organizational positions, we distinguish among elementary roles, roles, and positions. A position usually covers a number of roles. It is equal to an aggregate role in [Pitt et al, 1995]. For examples, the roles like reviewer-selector, the executive editor, copyeditor may be aggregated into one position "editor" filled by Dorian Selz in reality.

A role definition has these attributes:

Name: a name distinguishing the role from others

Goals: One or more goals the agent playing the role is responsible for.

Qualifications: One or more necessary preconditions or skills required to achieve the goals.

Relationships: one or more relations with other roles in a workflow. The relations from one role to another are categorized along two dimensions that is important for agent decision-making : authority and cooperation relations.

Obligations: a set of one or more obligations. The attribute defines the functionality of a role. For an example, an obligation of the role "reviewer-selector" is to allocate review-time to a potential reviewer. It is defined as follows.

Table 1 Obligation "review-time allocation"

Attribute	Value
Name	Reviewer-selection
Obligated roles	Reviewer-selector
Authority roles	Editor
Benefited roles	Author
Trigger	Insufficient reviewers accept the initial terms to review an article
Actions	Choose a potential reviewer for the article, Allocate a review-time to the potential reviewer
Constraints	Review-time <= 4 weeks
Cost	middle

Concurrency constraints of a role specify the mandatory sequencing, synchronization and parallelism of obligated actions. We define some basic operators to obligations in Table 2, where O1 and O2 are obligations.

Table 2 Obligation Operators

O1 ; O2	; denotes sequential composition
O1 O2	denotes non-deterministic choice
O1 & O2	& denotes parallel composition
[O1]	[] denotes O1 is optional
<O1>	<> denotes O1 is necessary
O1*	* denotes O1 occurs 0 or more times
O1+	+ denotes O1 occurs 1 or more times

Permissions: a set of permissions or prohibitions. A permission is a prescription that a particular behavior is allowed to occur. A prohibition is a prescription that a particular behavior must not occur.

Protocols: all protocols in which the role plays a part.

As an example, we define the role "Reviewer-selector" in Table 3.

Table 3 Definition of Role "Reviewer-selector"

Attribute	Value
Name	Reviewer-selector
Goals	G1: appoint three reviewers for a submission
Qualifications	C1: Know a set of potential reviewers C2: Have a submission for a peer review C3: Be able to match the content of an article with expertise of reviewers

Relationships	R1 IS_AUTHORIZED_BY editor R2 COOPERATE_WITH potential-reviewer R3 AUTHORIZE committed-reviewer
Obligations	O1: initial call for review O2: allocate a review-time O3: find a potential reviewer O4: negotiate with a potential reviewer O5: appoint a potential reviewer as a reviewer
Concurrency Constraints	O1 ; [O2 & O3 ; O4]* ; O5+
Permissions	PM1: Access submissions and reviewers database PM2: Free to contact with potential reviewers PM3: Allocate a review-time PM4: Cancel a review request PM5: One should not review his own submission
Protocols	PT1: Initial-Call-for-Review PT2: Negotiation for a review

5. Model And Specify Roles

Interaction Protocols

Protocols can be graphically modeled by means of state-based techniques like Finite State Machine (FSM). A protocol consists of these attributes:

Name: a unique name of the protocol

Description: brief description of its functions in a workflow;

Initiators: the role(s) responsible for starting the interaction;

Responders: the role(s) with which the initiator interacts;

States: a set of states of the FMS represent the states a conversation can be in. There is a distinguished initial state any conversation starts in, and a terminating state that when reached signals the termination of the conversation.

Transition rules: a set of rules control the workflow between these states.

Speech acts: a set of speech acts used in the protocol.

For an example, the protocol “Negotiation for a review” is modeled in Figure 2.

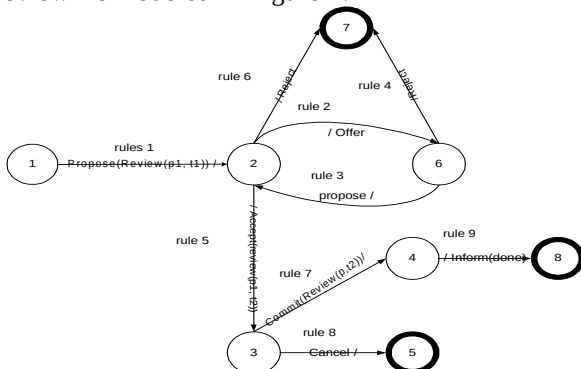


Figure 2 Negotiation Protocol for a Review

This protocol is specified as follows:

Name: Negotiation-for-review protocol.

Description: reviewer-selector negotiate with a potential reviewer for an agreement on review terms

Initiator: Reviewer-selector

Responder: a potential reviewer

There are 8 states in this state transition diagram:

State 1: Start state.

State 2: The reviewer-selector has made a review proposal to the potential reviewer.

State 3: The proposal has been accepted by the potential reviewer.

State 4: The reviewer-selector has committed the reviewer to review the article.

State 5: The reviewer-selector has cancelled the review proposal. END State.

State 6: The potential reviewer has not accepted the proposal and made a new offer to the selector.

State 7: The reviewer rejected the proposal or the selector rejected the offer. END State.

State 8: The review has been finished and the editor has been informed about it by the reviewer. END State.

When a protocol is realized in active database management system, a state transition diagram can directly be mapped into Event-Condition-Action (ECA) rules [Berndtsson, 1998]. A set of ECA rules specify how an agent in a given state receives a messages of specified type, does local actions (e.g., updating local data), sends out messages, and switches to another state. The use of ECA rules rather than the standard production rules is important because ECA rules have a clear distinction between events and conditions. The semantics of a ECA rule is : when ECA-events occur, ECA-conditions are evaluated; and, if these conditions are evaluated to true, ECA-actions are executed. For examples, the transitions in Figure 2 are mapped into the following ECA rules:

Rule1 (State 1 → State 2):

ON Insufficient reviewers accept the terms of the initial call for review

IF the reviewer-selector find a potential reviewer

THEN he has made a proposal of review terms and sends it to the potential reviewer.

Rule2 (State 2 → State 6):

ON the potential reviewer receives the proposal

IF he will not accept it and will make a new offer

THEN he makes an offer and sends the offer to the reviewer-selector

Rule3 (State 6 → State 2):

ON the selector receives the offer

IF he decides not to accept the offer and will make a new proposal

THEN he makes a new proposal and sends it to the potential reviewer

Rule 4 (State 6 → State7) :

ON the selector receives the offer

IF he decides to reject the offer

THEN he sends a rejection to the potential reviewer

Rule 5 (State 2 → State 3) : ...

Because we model the protocols by using Formal Description Techniques, the inconsistency of the rules can be checked.

The negotiation protocol of the case study encompass primitives from the most common phase of negotiation such as Inform, Query, Commit, Propose, Offer, Accept, Reject, Cancel. The initial call for review protocol has “Request” primitive. A “request” is non-negotiable: a negotiable request is a “Propose”, which allows the receiver to make a counter-offer using the “Offer”. For examples, the reviewer-selector sends this message to a potential reviewer proposing review an article within 3 weeks. It can be represented as:

```
(propose
:sender ?editor
:receiver ?potential-reviewer
:language predicate-logic
:ontology NetAcademy
:content review( ?potential-reviewer,
article1 ) ^ has(article1,title1,abstract1,keywords1)
^ allocated-review-time( article1, 3-weeks ) )
```

The ontology is a formal description of the problem domain. It maps these symbols to a well-understood meaning for the problem domain. For one agent to properly understand the meaning of a message from another agent, both agents must ascribe the same meaning to the symbols (constants) used in that message.

6. A Proposed Multi-agent Framework for IPRPMS

We design the Internet Peer Review Process Management Systems (IPRPMS) as a Multi-agent system. The core of the IPRPMS is the workflow enactment service which consists of several autonomous cooperating agents: the workflow manager, workflow engines, personal agents, actor agents. One workflow instance in IPRPMS is illustrated in Figure 3.

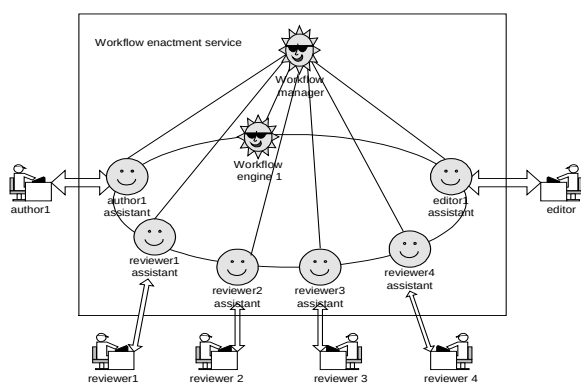


Figure 3 One Workflow Instance In the Internet Peer Review Process Management System

The workflow manager is the central element of the framework. It is used only to forward requests from a client application (which can be a user of an event) to an appropriate workflow engine. Its main purpose is to realize the network of individual agents

and initialize the communication among them. However, the main body of messages is communicated directly among the individual agents.

The core of the processing is carried out by the distributed **workflow engines**. They form the main part of the workflow enactment service. Each workflow engine is responsible for managing a single workflow instance. The workflow engine is a state transition machine, where individual activity instance of a workflow change state in response to external events (e.g., completion of an activity) or to specific control decisions taken by a workflow engine (e.g., navigation to the next activity step within a process).

Every workflow participant, either a human or an information system, has its own representative within the system, i.e., the personal agent or actor agent. A **personal agent** acts on behalf of a participant, e.g., author assistant, editor assistant, reviewer assistant, etc. An **actor agent** takes the role of an actor and is invoked by a workflow engine. The major role of this agent is to compile a worklist of all tasks assigned to the particular participant by different engines and manage the execution of these tasks in an autonomous way. Each Personal Agent or Actor Agent is responsible for the execution of the individual tasks that constitute a workflow.

Once a workflow engine has been assigned for the management of a particular workflow instance, the client application communicates instantiation data to the workflow engine. The engine decomposes the workflow and looks through the manager for the appropriate personal or actor agent that will undertake the management of each individual task. Further communication and data interchange between workflow engine and personal or actor agent is direct.

In the peer review process, we assume the following elementary roles are aggregated into an editor role: register an article, reviewer-selector, distributed article to reviewers, conduct the online discussion, make a review decision, and copyediting an article, etc. Then we have 3 personal agent types: author-assistant, editor-assistant, reviewer-assistant. This arises also naturally from the persons involved in a paper peer review process.

Author assistant: It manipulates author's work items. It initializes a peer review process when the author submits an article to the NetAcademy. It alerts the author when it receives a rejection or revision request message from editor-agent, or when a new comments have been made on his article. It delivers author's revision version and comments to editor-agent in the online review discussion, etc.

Editor assistant: It manipulates editor's work items. Responsible for receiving the coming messages to the editor; Alert the editor to make an initial decision when a submission is received; Selects three reviewers for a submission; Collect the committed reviews and publish them on the web; Maintain the potential reviewers database; Alert the editor to make a review decision when it receives all reviews; Remind the editor of deadlines, etc.

Reviewer assistant: The reviewer's communication assistant in the NetAcademy. Alert reviewer if a coming message arrives; Reply a review request; Negotiate with editor agent to reach an agreement for a review; Remind reviewer of deadlines for a committed review; Transfer review reports to the editor agent; etc.

The proposed framework follows not only the WfMC's reference model of workflow but also an agent oriented view of a business process. It leads to an effective agent oriented implementation.

7. Agent Oriented Implementation

The above agent types can be easily implemented by Shoham's agent architecture whose behavior of is consequence of its mental states such as beliefs, intentions, capabilities, goals, commitments, etc [Shoham, 1993]. One available platform is AgentBuilder which has extended Shoham's agent architecture [Reticular Systems, 1999]. Because the roles and protocols specification only provide high-level information and we must provide additional detail in this step to complete construction of the agents.

However, extra efforts have to be made because the Internet Peer Review System must support WWW-enabled workflow. To access workflow engine services, a web browser must be installed on the users' machine. This client machine then communicates with the Web server, which offers the workflow services, using the Hypertext Transfer Protocol (HTTP). The Web server presents workflow run time objects such as workflow manager, workflow engines, personal agents, and work items that are available to the user on Hypertext Markup Language (HTML) documents. The Web server accesses the agents' functions using the Common Gateway Interface (CGI). Now we are about to implement a prototype of the IPRPMS as a part of the NetAcademy (<http://www.netacademy.org/>).

8. Conclusion

In this paper has designed the Internet peer review system in a Multi-agent framework. The procedure integrates peer review with the electronic publication and workflow management, allowing rapid publication after rigorous peer review. It preserves the chief virtue of the traditional peer review system (expert appraisal before publication), but brings a new openness to the procedure that we hope will enhance its accuracy and fairness. Through agent technology, it is obvious the cooperation and communication among editor, authors and reviewers are enhanced and improved. Therefore, the review quality and major delay time will be guaranteed and improved. Moreover, this application shows us that the methodology for agent oriented and role based workflow management is a powerful modeling and design technique. It takes a distributed, and hence more robust, flexible, and scaleable approach for workflow management. We believe Workflow and Internet are more effective in combination than

alone. Many similar applications in E-Commerce are easily be imaged.

Acknowledgements. The first author wishes to thank Dr. Salome Schmid-Isler for her helpful discussion on the project NetAcademy.

References

- [Berndtsson, 1998] M. Berndtsson. Active Capability Support For Cooperative Strategies in Cooperative Information Systems. Ph.D. Dissertation. Faculty of Science, University of Exeter, UK, July 1998
- [Bingham et al., 1998] C. Bingham, C. Higgins, R. Coleman, M. B Van Der Weyden. The Medical Journal of Australia Internet peer-review study. *The LANCET*, Vol. 352, August 8, 1998, pp441-445
- [Kaldoudi et al., 1997] E. Kaldoudi, M. Zikos, E. Leisch, S. C. Orphanoudakis. Agent-based Workflow Processing For Functional Integration And Process Re-engineering In The Healthcare Domain. *The 15th International EuroPACT Meeting (EuroPACS'97)*. Sept. 25-27, Pisa, Italy, 1997, pp247-250
- [Hawryszkiewicz and Debenham, 1998] I. Hawryszkiewicz and J. Debenham. A Workflow System Based on Agents. In *Proceedings of 9th International Conference on Database and Expert Systems Applications; DEXA'98*, Vienna, Austria, Lecture notes in computer science; Vol. 1460, Springer, August 1998
- [Jennings et al., 1996] N. R. Jennings, P. Faratin, M. J. Johnson, T. J. Norman, P. O'Brien, M. E. Wiegand: Agent-based Business Process Management. *International Journal of Cooperative Information Systems*, 5 (2&3), 1996, 105-130.
- [Pitt et al., 1995] J. Pitt, M. Anderton and J. Cunningham. Normalized Interactions Between Autonomous Agents: A Case Study In Inter-organizational Project Management. In *Proceedings of the International Workshop on the Design of Cooperative Systems*. January 1995, Antibes-Juan-les-Pins, France, pp76-95
- [Reticular Systems, 1999] Reticular Systems, Inc. AgentBuilder – An Integrated Toolkit for Constructing Intelligent Software Agents. *User's Guide*, Version 1.1, <http://www.agentbuilder.com>, January 19, 1999.
- [Shoham, 1993] Y. Shoham. Agent-oriented Programming. *Artificial Intelligence*, 60 (1993), pp51-92
- [Yu and Schmid, 1999] L. Yu and B. F. Schmid. A Conceptual Framework For Agent Oriented and Role Based Workflow Modeling. *Presented at CAiSE'99 Workshop on Agent Oriented Information Systems (AOIS'99)*, Heidelberg, Germany, June 14-15, 1999