A Multi-Agent Architecture For Distributed Corporate Memories

Fabien Gandon

A Multi-Agent Architecture For Distributed Corporate Memories

Fabien Gandon
ACACIA research team - INRIA Sophia Antipolis - 2004, route des Lucioles
BP93 - 06902 Sophia Antipolis Cedex - France
email: Fabien.Gandon@sophia.inria.fr

Abstract

This paper presents an approach to design a multi-agent system managing a corporate memory in the form of a distributed semantic web and describes the resulting architecture.

1 Introduction

With our entrance in the information society, organizations had to adapt to the shift in the economy and market rules that followed. Information systems are becoming backbones of organizations and therefore their interest in corporate knowledge management grows stronger. Our research team, ACACIA, studies the semantic web technologies to provide tools, and methodologies to materialize and structure a corporate memory and prepare their exploitation in the form of corporate semantic webs. In parallel, distributed artificial intelligence now proposes the new paradigm of multi-agents system that appears to us as very well suited to deploy a software architecture over this distributed information landscape. We present results from the CoMMA European project [CoMMA, 2000] aiming at implementing a corporate memory management framework based on emerging technologies: agents, ontologies, XML, information retrieval and machine learning techniques. The project is implementing this system in the context of two application scenarios: (1) assisting the insertion of new employees in the company and (2) supporting the technology monitoring processes. In a first part we will present the intervention context and justify the conceptual and technical choices we made. Then we will focus on the approached we followed to design the CoMMA system and the architecture we implemented and tested.

2 The context: a distributed memory

A corporate memory is an explicit, disembodied and persistent representation of knowledge and information in an organization, in order to facilitate their access and reuse by members of the organization, for their tasks [Dieng et al., 2001]. The stake in building a corporate memory management system is the coherent integration of this dispersed knowledge to promote knowledge growth, knowledge communication and in general to preserve knowledge within an organization [Steels, 1993]. This first part presents the motivations of our conceptual and technical choices.

2.1 Integrating emerging technologies

The multi-agent approach relying on loosely coupled software components, is naturally prone to facilitate integration of different technologies in one system. This was an important need for CoMMA, since several emerging technologies have been chosen for the system implementation to address initial observations:
(1) *The memory is, by nature an heterogeneous and distributed information landscape*. Corporate memories are facing the same problem of precision and recall as the Web. The semantic Web [Berners-Lee *et al.*, 1995] is a promising approach where the semantics of documents is made explicit through metadata and annotations to guide later exploitation. We propose to study corporate memories as *corporate semantic Webs* where RDF (Resource Description Framework) is used to semantically annotate the corporate resources.
(2) *The population of the stakeholders of the memory is, heterogeneous and distributed in the corporation*. The system has to interface users with the content of the memory and enable them to exploit or contribute to its content. The CoMMA system relies on machine learning techniques to make agents adaptive to the users and the context. This goes from learning user's preferences, up to information push technologies.
(3) *The tasks as a whole to be performed on the memory are, by nature, distributed and heterogeneous*. Moreover, both the corporate memory and its population of users are distributed and heterogeneous. Therefore, it is interesting to consider a distributed and heterogeneous system such as a Multi-Agents System (MAS) to manage and exploit this information landscape. Programming progresses were achieved through higher abstraction enabling us to model systems more and more complex and MAS are a new stage in abstraction that can be used to understand, to model and to develop a whole new class of distributed systems [Wooldridge *et al.*, 2000]. We believe the MAS paradigm is well suited for designing and deploying a software architecture above distributed information landscapes of corporate memories: on the one hand, individual agents locally adapt to users and resources they are dedicated to ; on the other hand, thanks to cooperating software agents distributed over the network, the whole system capitalizes an integrated view of the corporate memory.

2.2 An annotated world for agents

The article "Agents in Annotated Worlds" [Doyle and Hayes-Roth, 1998] explains that "knowledge can literally be embedded in the world as annotations attached to objects, entities and locations" and thus we obtain "annotated environments containing explanations of the purpose and uses of spaces and activities that allow agents to quickly become intelligent actors in those spaces". Thus *annotated information worlds are, in the current state of the art, a quick way to make information agents smarter* : if the memory becomes an annotated world, agents can use the semantics of the annotations and, through inferences, help users exploit it.

RDF [Lassila and Swick, 1999] uses a simple data model expressed in XML syntax for representing properties of Web resources and their relationships. *We describe the content of documents through semantic RDF annotations and then use and infer from these annotations to efficiently exploit the corporate memory*. RDF annotations can be either internal or external to the document, thus existing documents of the corporate memory may be kept intact and annotated externally. Annotations are based on the O'CoMMA ontology [Gandon, 2001] described and shared thanks to RDF Schema [Brickley and Guha, 2000]. Keyword-based search engines are limited to terms occurrences, the introduction of ontology-based annotations enables agents to access the semantic level. O'CoMMA is the keystone of our system: it is a full resource of the memory and it provides the building blocks for models (user profiles, corporate model), annotations and agent messages, with their associated semantics.

The *enterprise model* is an oriented, focused and somewhat simplified explicit representation of the organization. It gives the system an insight in the organizational context and environment to tune its interactions and reactions. Likewise, the *users' profile* captures all aspects of the user that were identified as relevant for the system behavior. It contains administrative information and topic interests. It positions the user in the organization: role, location and potential acquaintance network. In addition to explicitly stated information, the system derives information from past usage by collecting the history of visited documents and possible feedback from the user, as well as the user's recurrent queries. From this, agents learn some of the user's habits and preferences [Kiss and Quinqueton, 2001]. These derived criterions are used for interface or information push.

2.3 A Multi-Agents Information System

CoMMA adopted the weak notion of agency [Wooldridge and Jennings, 1995] but we do not claim that in our current prototype all our agents are one hundred percent compatible with this definition. The information agents are part of the intelligent agents. A MAS is a loosely coupled network of agents that work together as a society aiming at solving problems that would generally be beyond the reach of any individual agent. A MAS is heterogeneous when it includes agents of at least two types. A Multi-Agents Information System (MAIS) is a MAS aiming at providing some or full range of functionalities for managing and exploiting information resources. Based on these notions, the software architecture of CoMMA is an heterogeneous MAIS ; the application of a MAIS to corporate memories means that the cooperation of agents aims at enhancing corporate knowledge capitalization.

So far a large number of MAIS projects focused on the problem of dynamically integrating heterogeneous sources of information (e.g. InfoSleuth [Nodine *et al.*, 1999]). It comes from the fact that it was one of the problems being addressed in the field of information systems when MAS came to meet them and also because the decentralized nature and local adaptability of agents were assets for wrapping heterogeneous sources. Another interesting type of projects concerns the management of digital libraries (e.g. SAIRE [Odubiyi *et al.*, 1996] and UMDL [Weinstein *et al.*, 1999]). In CoMMA we do not stress the heterogeneous sources reconciliation aspect: documents may be heterogeneous but annotations are represented in RDF and based on a shared ontology formalized in RDFS. CoMMA is focusing on the design of an architecture of cooperating agents, being able to adapt to the user, to the context, and supporting information distribution. The duality the word 'distribution' reveals two important problems to be addressed : (a) 'distribution' means dispersion, that is the spatial property of being scattered about, over an area or a volume ; the problem here is to handle the naturally distributed data, information or knowledge of the organization. (b) 'distribution' also means the act of spreading ; the problem then is to make the relevant pieces of information go to the concerned agent (artificial or human). It is with both purposes in mind that we designed the CoMMA architecture as presented in the following section.

3 Designing the MAS

Our approach shares with the A.G.R. model used in AALAADIN [Ferber and Gutknecht, 1998] and GAIA [Wooldridge *et al.*, 2000] methodologies the concern for an organizational approach where the MAS architecture is tackled, as in a human society, in terms groups, roles and relationships. The manifesto of Panzarasa and Jennings [2001] goes even further, advocating the application of modern organization theory. This type of approach is attractive in the context of an organizational memory and in this section we present our experience in designing the agent society managing the memory, starting from societal level requirements and going down to the agent internal behavior.

3.1 From macroscopic to microscopic

The functional requirements of the system do not simply map to some agent functionalities but influence and are finally diluted in the dynamic social interactions of agents and the set of abilities, roles and behaviors attached to them. This section explains how we went from the system requirements expressed at the societal level down to the point where agent roles can be identified.

Architecture versus Configuration

The MAIS architecture is a structure that portrays the different kinds of agencies existing in an agent society and the relationships among them. A configuration is an instantiation of an architecture with a chosen arrangement and an appropriate number of agents of each type. One given architecture can lead to several configurations. In the case of a multi-agents corporate memory system, the configuration depends on the topography and context of the place where the system is rolled out (organizational layout, network topography, stakeholders location), thus it must adapt to this information landscape and change with it. The architecture must be designed so that the set of possible configurations covers the different corporate organizational layouts foreseeable. The configuration description can be studied and documented at deployment time using adapted UML deployment diagrams to represent hosts (servers, front-end...), MAS platforms, agent instances and their acquaintance graph.

The architectural description is studied and fixed when designing the MAS. The architectural analysis starts from the highest level of abstraction of the system (i.e. the society) and by successive refinements (i.e. nested sub-societies) it goes down to the point where the needed agent roles and interactions can be identified. Considering the system functionalities we identified four dedicated sub-societies of agents: (1) The sub-society dedicated to Ontology and model (2) Annotation-dedicated sub-society (3) User-dedicated sub-society (4) Connection-dedicated sub-society. These societies are analyzed in the following parts.

Organizing sub-societies

Analyzing the resource dedicated sub-societies (Ontology/model, Annotation and Connection Agents) we found that there was a recurrent set of possible internal organizations for these sub-societies:

A *hierarchical organization* distinguishes between two kinds of roles: (1) The representative role: the agent is a mediator between its society and the rest of the MAS. It deals with the external requests. If needed it breaks them up into several sub-requests. It contacts and delegates to resource-dedicated agents. Finally, it compiles the possibly partial results to answer the external requester. (2) The resource-dedicated role: the agent is dedicated to a local resource repository and contributes to solve the requests it receives from the representative as much as it can with its local resources. In this organization, workload is greatly distributed because resource agents only work with the resource they have locally and they leave the fusion work to representative agents that can be placed on another machine that does not necessarily hold a repository of information. However this organization is quite network-consuming.

A *peer-to-peer* organization sets up egalitarian relationships between the roles. Roles are not distributed, but completely redundant: any agent can be contacted from outside the society to handle a request concerning the resource type its society is dedicated to. It will then have to cooperate with its peers in order to efficiently fulfill the request. Agents are specialized only through the content of the local resource repository they are attached to. The workload is less distributed than in the previous case but the network-load may be decreased. There is only one role merging the two previous roles (representative and resource dedicated). Coalitions will be formed to solve external queries.

A *replication organization* is a subtype of the previous case: neither the roles nor the repository content are distributed. Each agent keeps up to date a complete copy of all the resources and is able to solve any request by itself. Therefore the only social interactions that exist are for content updates. The workload is even less distributed than in the previous case and the contents has to be replicated everywhere an agent sits which can be an unacceptable constraint. On the other hand the system is highly redundant, thus more robust, and the network use is minimal when dealing with a request. The only role is the resource-dedicated one.

Depending on the type of tasks to be performed, the size and complexity of the resources manipulated, a sub-society organization will be preferred to another.

Sub-society dedicated to Ontology and Model

The agents from this sub-society are concerned with the ontology and model exploitation aspects of the information retrieval activities. They provide downloads, updates and querying mechanisms for other agents, on the hierarchy of concepts and the description of the organization. For this sub-society, the three types of organizations are conceivable: (1) In a hierarchical organization, we would have a Master role in charge of resolving external queries and an Archivist role in charge of a part or a view of the ontology/model ; (2) in a peer-to-peer society, we would have a cooperative Archivist role; (3) in a replication society each agent would have a complete copy of the ontology/model and could resolve queries by itself.

The last choice is acceptable when the ontology is stable and when a consensus is reached by the users so that the ontological commitment is centralized and the global ontology is updated and propagated over the agent society. This option is implemented in the current prototype of CoMMA. The remaining options are interesting if the ontology/model is large or changes quite often and if the system must support the ontological consensus process ; the agent society can then support the break-up of the ontology/model and maintain the coherence between the different repositories as in the FRODO project [Elst and Abecker, 2001].

Annotation dedicated sub-society

The agents from this sub-society are concerned with the exploitation of the annotations structuring the corporate memory, they will search and retrieve the references matching the users' queries. Here, only the two first types of organization are conceivable: (1) In a hierarchical organization there is an Mediator role and an Archivist role. (2) In a peer-to-peer organization we would have a cooperative Archivist role. (3) A replication society is not realistic because it would imply to replicate a full copy of the corporate memory for each resource agent. This is obviously not acceptable since the corporate memory, is a huge amount of information broken up and distributed over the intranet.

The current CoMMA system opted for the first type of society. The Annotation Mediator (AM) typically provides its services to other societies to solve their queries and requests the services of the resource agents to effectively solve them: (1) It breaks the queries and contacts the relevant Annotation Archivists (AAs) at each stage of the decomposition to get partial results. (2) It compiles the partial results to build the complete answer. The AA role is attached to a local annotation repository, and when it receives a query, it tries to obtain at least partial results from its local resources to enable the AM to handle results distributed over several information sources. The AM also allocates new annotations to AAs in a contract-net fashion. Bids are based on a similarity measure between the new annotation and the content of the archive of the AA.

Interconnection dedicated sub-society

Agents from this sub-society are in charge of matchmaking other agents using their respective needs and service provider descriptions. Each provider must first register itself with at least one middle agents and advertise its capabilities. Requests are then matched to these descriptions to find which agent can provide a required service.

The CoMMA system is implemented using the JADE platform [Bellifemine *et al.*, 2001] that provides an agent type called Directory Facilitator (DF). DF are federable to build a peer-to-peer society and in charge of managing Yellow Pages. According to FIPA[1] specifications, DF provide agent identifiers matching the service description and the ontology specified in a pattern. Thus DF are matchmakers identifying relevant providers and returning the selection of candidates to the requester. The result of the matchmaking can be further refined in a second stage. For instance the AM requires statistics from the AAs to know what types of annotations they have to decide when to appeal to them during the distributed solving process.

User dedicated sub-society

The agents from this sub-society are concerned with the interface, the monitoring, the assistance and the adaptation to users. They are, typically, requester agents. Because they are not related to a resource type, they cannot be studied using the typology we proposed. However, we can distinguish at least two recurrent roles in this type of sub-society: (1) the user interface management: to enable users to express their requests and to present results in a appropriate format (2) the management of user's profile: to archive and make the profiles available to be used for interface purposes, learning techniques, pro-active searches... The machine learning techniques are developed by the LIRMM[2] [Kiss and Quinqueton, 2001] and additional roles conceivable for this sub-society can be found in [Gandon *et al.*, 2000].

[1] Foundation for Intelligent Physical Agents - www.fipa.org

[2] Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier - www.lirmm.fr

3.2 Roles, Interactions and behaviors

From the architecture analysis, we now derive the characteristics of the identified roles, their interactions and we discuss the implementation of the corresponding behaviors in a set of agent types.

Agent roles

Roles represent the position of an agent in a society and the responsibilities and activities assigned to this position and expected by others to be fulfilled. In the design junction between the micro-level of agents and the macro-level of the MAS, the role analysis is a key step. The previous part identified the following roles, implemented in the first prototype of CoMMA: *Ontology Archivists* maintain and access the ontology ; *Enterprise Model Archivists* maintain and access the enterprise model ; *Annotation Archivists* maintain and access annotation repositories ; *Annotation Mediators* manage and mediate among a set of Annotation Archivists ; *Directory Facilitators* maintain and access yellow pages ; *Interface Controllers* manage and monitor user interfaces ; *User Profile Managers* update users' profile ; *User Profile Archivists* store and distribute users' profile.

The list of characteristics that are debated in the agent community gives an overview of the actual capabilities envisaged. Their diversity reveals the influence and the integration in MAS of results from a lot of research areas. Table 1 compiles some of them, found in the literature[3], and uses them to characterize the roles. Other methodologies, such as [Wooldridge *et al.*, 2000], propose a formalization of the roles but it was not deemed necessary for our first prototype.

	Ontology Archivist	Enterprise Model Archivist	Annotation Archivist	Annotation Mediator	Directory Facilitator	Interface Controller	User Profile Manager	User Profile Archivist
Reactive								
Complex Mental State								
Graceful Degradation	x	x	x	x	x	x	x	x
Temporally continuity	x	x	x	x	x		x	x
Autonomy								
Goal-oriented				x			x	
Collaborative	x	x	x	x	x	x	x	x
Flexible				x			x	
Proactive							x	
Personality								
Communication	x	x	x	x	x	x	x	x
Adaptability								
Learning							x	
Customizable						x	x	
Mobility								
Visual representation						x		
Veracity	x	x	x	x	x	x	x	x
Benevolence	x	x	x	x	x	x	x	x
Rationality	x	x	x	x	x	x	x	x

Table 1. Roles Characteristics

[3] [Etzioni and Weld, 1995] , [Franklin and Graesser, 1996] , [Nwana, 1996] and [Wooldridge and Jennings, 1995]

Social interactions

Following the sub-societies and roles identification comes the specification of the interactions. Interactions consist of more than the sending of isolated messages and the conversation patterns need to be specified with protocols. Agents must follow these protocols for the MAS to work properly. Protocols are codes of correct behavior, in a society, for agents to interact with each others. They describe a standard procedure to regulate information transmission between agents and they institutionalize patterns of communication occurring between identified roles. The definition of a protocol starts with an acquaintance graph at role level, that is a directed graph identifying communication pathways between agents playing the roles involved in an interaction scenario. From that we specify the possible sequences of messages. Figure 1 shows an extract of the protocol being for handling annotations submission ; it could be described as a nested contract-net.

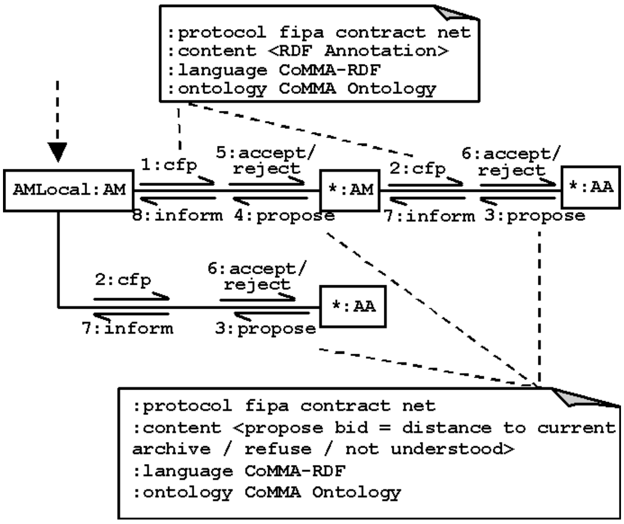

Figure 1. Interactions when allocating an annotation

The acquaintance connections among the roles and the protocols adopted derive from both the organizational analysis and the use cases dictated by the application scenarios. The acquaintance graphs and the ACL message traces are depicted using protocol diagrams, a restriction of the UML collaboration diagrams [Bergenti, and Poggi, 2000] .

The implementation of CoMMA relies on JADE [Bellifemine *et al.*, 2001], which is an open source MAS development platform compliant with the FIPA specifications. The agent communication language FIPA ACL is based, like its counterpart KQML, on the speech act theory and comes with already standardized protocols to be used or extended with the semantic of their speech acts specified in the FIPA ontology. In the first prototype, the content languages of the messages were SL for the speech acts specified by FIPA for the envelop (Figure 2-a) and the speech acts involved in the CoMMA protocols (Figure 2-b). RDF was used for the exchanged annotations and query patterns (Figure 2-c). The latest prototype uses RDF as an ACL.

```
  (QUERY-REF
a  :sender(agent-identifier :name localUPM@apollo:1099/JADE)
   :receiver(set(agent-identifier :name AM@apollo:1099/JADE))
   :content
b   ((all ?x (is-answer-for
              (query
      :pattern
c    <?xml version ="1.0"?> <rdf:RDF xml:lang="en"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:comma="http://www.inria.fr/acacia/comma#">
     <comma:Memo><comma:Designation>?</comma:Designation>
     </comma:Memo>
     </rdf:RDF>
b              ) ?x ) ) )
   :reply-with  QuerylocalUPM987683105872
   :language  CoMMA-RDF
a  :ontology  CoMMA-annotation-ontology
   :protocol  FIPA-Query
   :conversation-id  QuerylocalUPM987683105872 )

(a)FIPA ACL Envelop - (b)CoMMA SL0 Request - (c)RDF Pattern
```

Figure 2. A message asking for the title of Memos

Behavior and technical competencies

From the role and interaction descriptions we proposed and implemented agent types that fulfill one or more roles. The behavior of an agent type combines behaviors implemented by the designers to accomplish the activities corresponding to the assigned roles. For instance there is currently one agent type playing both the Ontology Archivist and Enterprise Model Archivist roles. Its behavior contains both associated behaviors which are themselves made up of sub-behaviors handling the different tasks and interactions linked to these roles. Behaviors are directly linked to the implementation choices and determine the responses, actions and reactions of the agent. The implementation of the behavior is constrained by the role but it is also subject to the toolbox of technical abilities available to the designers. As an example, an AA has a behavior handling its involvement in distributed query-solving. A task occurring in this behavior, is the projection of a sub-part of the complete query to propose partial results to the AM that will build a complete answer. Therefore the behaviors of the AA and the AM include calls to the CORESE API [Corby *et al.*, 2000] a prototype of search engine enabling inferences on RDF annotations.

4 Conclusion

In this paper we presented an approach to corporate memory management relying on the technologies of multi-agent, XML and machine learning. We described the methodology we followed do design such a system and the architecture we obtained and implemented in the first prototype of CoMMA. The full implementation of the CoMMA prototype is now finished and trials are being prepared to evaluate the complete solution. Even if groups and roles are not first citizens in the JADE platform, an organizational approach proved to be successful to specify the implementation. Several methodologies now propose formal models to support a sound organizational analysis. It would be interesting to compare how these different models would represent the CoMMA architecture and capture the design rationale. Agentization was difficult for large and complex components but this would disappear if fully finalized Agent-Oriented languages were used instead of object-oriented languages with an Agent API. Therefore from a feasibility and conceptual point of view the multi-agent paradigm proved to be extremely well-suited to design a software architecture that could be deploy in a variety of configurations to adapt to a distributed information landscape. Concerning the integration phase of the development, the agent technology proved to be extremely valuable: the different agents have been developed by distant partners having the needed experience, starting from shallow agents ; but since the agents are loosely coupled software components and that their role and interactions have been specified using a consensual ontology, the integration and setup of a first prototype was achieved in less than two days. The currently trial will enable us to discuss and evaluate this new approach with real end-users.

Acknowledgments

I thank my colleagues of ACACIA and the CoMMA consortium for our fruitful and the European Commission that funds the CoMMA project (IST-1999-12217).

References

[Bellifemine *et al.*, 2001] Bellifemine F., Poggi A., Rimassa G. Developing multi agent systems with a FIPA-compliant agent framework. *Software Practice & Experience*, 31:103-128, 2001.

[Bergenti, and Poggi, 2000] Bergenti F., Poggi A. Exploiting UML in the Design of Multi-Agent Systems. *Omicidi A., Tolksdorf R., Zambonelli F. Engineering Societies in the Agents World* - Springer, LNAI 1972:106-113, 2000.

[Berners-Lee *et al.*, 1995] Berners-Lee T., Hendler J., Lassila, O., The Semantic Web, *Scientific American*, p. 35-43, May 2001.

[Brickley and Guha, 2000] Brickley D., Guha R.V. – Resource Description Framework (RDF) Schema Specification. W3C Candidate Recommendation

[CoMMA, 2000] Corporate Memory Management through Agents, *Proc. E-Work E-Business*, 2000.

[Corby *et al.*, 2000] Corby O., Dieng R., Hébert C., A Conceptual Graph Model for W3C Resource Description Framework. *Proceedings ICCS'00* Darmstadt, LNAI 1867, 2000.

[Dieng *et al.*, 2001] Dieng R., Corby O., Gandon F., Giboin A., Golebiowska J., Matta N., Ribiere M., Méthodes et outils pour la gestion des connaissances, 2nd edn, Dunod, 2001, ISBN 2 10 006300 6

[Doyle and Hayes-Roth, 1998] Doyle P., Hayes-Roth B., Agents in Annotated Worlds, *Proc. Autonomous Agents*, ACM SIGART, p. 173-180, 1998.

[Elst and Abecker, 2001] Elst L., Abecker A., Domain Ontology Agents in Distributed Organizational Memories *Workshop on Knowledge Management and Organizational Memories*, IJCAI, 2001.

[Etzioni and Weld, 1995] Etzioni O., Weld D., Intelligent Agents on the Internet: Fact, Fiction and Forecast, *IEEE Expert/Intelligent Systems & Their Applications*, 10(4), August 1995.

[Ferber and Gutknecht, 1998] Ferber J., Gutknecht O., A meta-model for the analysis and design of organizations in multi-agent systems. *IEEE Computer Society, Proc. 3rd ICMAS*, p. 128-135, 1998.

[Franklin and Graesser, 1996] Franklin S., Graesser A., . Is it an Agent, or just a Program ? : A Taxonomy for Autonomous Agents. *Proc. 3rd International Workshop on Agent Theories, Architectures and Languages*, Springer - Verlag, 1996

[Gandon *et al.*, 2000] Gandon F., Dieng R., Corby O., Giboin A., A Multi-Agents System to Support Exploiting an XML-based Corporate Memory, *Proc. PAKM'00*, p. 10.1-10.12

[Gandon, 2001] Gandon, F., Engineering an Ontology for a Multi-Agents Corporate Memory System, *Proc. ISMICK'01*, p. 209-228, 2001.

[Kiss and Quinqueton, 2001] Kiss A., Quinqueton J. Multiagent Cooperative Learning of User Preferences, *Proc. of European Conference on Machine Learning Principles and Practice of Knowledge Discovery in Databases*, 2001.

[Lassila and Swick, 1999] Lassila O., Swick R. – Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation

[Nodine *et al.*, 1999] Nodine M., Fowler J., Ksiezyk T., Perry B., Taylor M., Unruh A.; Active Information Gathering In Infosleuth™, *Proc. Internat. Symposium on Cooperative Database Systems for Advanced Applications*, International Journal of Cooperative Information Systems, World Scientific.

[Nwana, 1996] Nwana H.S., Software agents: an overview, *The Knowledge Engineering Review*, 11(3): 205-244, 1996.

[Odubiyi *et al.*, 1996] Odubiyi J., Kocur D, Weinstein S, Wakim N. Srivastava S., Gokey C., Graham J., SAIRE A Scalable Agent-Based Information Retrieval Engine, *Proc. Autonomous Agents*, 1997.

[Panzarasa and Jennings, 2001] Panzarasa P., Jennings N.R., The organisation of sociality: A manifesto for a new science of multi-agent systems, *Proc. MAAMAW'01*, LNAI, Springer Verlag, 2001.

[Steels, 1993] Steels, L. Corporate knowledge management. *Proc. of ISMICK'93*, p. 9-30, 1993.

[Weinstein *et al.*, 1999] Weinstein P., Birmingham W., Durfee E., Agent-Based Digital Libraries: Decentralization and Coordination. *IEEE Communication Magazine*, p. 110-115, 1999.

[Wooldridge *et al.*, 2000]. Wooldridge M., Jennings N., Kinny D., The Gaia Methodology for Agent-Oriented Analysis and Design, *Autonomous Agents and Multi-Agent Systems*, 3(3):285-312, 2000.

[Wooldridge and Jennings, 1995] Wooldridge M, Jennings N., Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*, 10(2):115-152.