TOWARDS A SELF-ADAPTIVE MULTI-AGENT APPROACH FOR ENHANCING THE QUALITY OF SERVICE PROVIDED BY OPEN INFORMATION SYSTEMS

Valérie Camps and Pierre Glize
IRIT- Université Paul Sabatier – 118 route de Narbonne 31062 Toulouse Cedex 4, France

Keywords: Web site and services personalisation, searching and browsing, emergent quality of services, adaptive multi-agent systems, self-organisation.

Abstract: Current information systems are plunged into highly dynamical environments which produce occurrences of unpredictable situations. This dynamics combined with the inherent geographical and functional distribution of such systems, make usual adaptation techniques which are global or dependent of the intended global function realised by the system, unsuitable. Our contribution concerns a partial instantiation of a local adaptation method, based on adaptive multi-agent systems, to manage the QoS of information systems. This management is done according to two points of view addressed in an integrate way: a quantitative one and a qualitative one. First obtained results, showing the benefits of cooperation to the adaptation of such systems, are then discussed.

1 INTRODUCTION

We consider open systems (Hewitt, 1982) as evolving systems composed of dynamic entities and plunged into a dynamic environment. In these conditions, how designers of open information systems (IS) can guarantee to humans who use them a given quality of service (QoS)? The usual response is to propose infrastructure, middleware, norms, protocols, Web Services. Unfortunately, this is a never-ending process in open systems: heterogeneity, incompleteness and unforeseeable situations are inescapable and have to be taken into account. We propose a quite new different approach by first agentifying all the components of an open IS and then giving them the capability to collectively converge dynamically and in real-time towards the optimal QoS. This original approach optimises the QoS of an IS by only taking into account individual characteristics of its components, instead of improving or finding new criteria to deal with its QoS as well as creating new tools to manage it. Generally speaking QoS has two constituents: *(i)* qualitative (functional) properties, defining how well the retrieved information matches the intended information such as precision, recall and noise and *(ii)* quantitative (non functional) properties, ensuring an effective flow in terms of end-to-end delay and including properties such as security, breakdown, interoperability, bandwidth.

This paper presents how the agentification process is able to tackle the above constituents of QoS in open IS from two experiments: a quantitative one, dealing with end-users and services mapping according to a need, and a qualitative one, dealing with a learning algorithm prefiguring the construction of end-users and services profiles. This work is based on the AMAS (Adaptive Multi-Agent Systems) approach allowing the design of complex systems that can be incompletely specified and for which an *a priori* known algorithmic solution does not exist. First innovative aspects of our approach are then presented. We conclude by proposing a guide to study properties of QoS in open IS tackled with an emergent problem solving approach.

Camps V. and Glize P. (2007).
TOWARDS A SELF-ADAPTIVE MULTI-AGENT APPROACH FOR ENHANCING THE QUALITY OF SERVICE PROVIDED BY OPEN INFORMATION SYSTEMS.
In *Proceedings of the Third International Conference on Web Information Systems and Technologies - Web Interfaces and Applications*, pages 295-301
DOI: 10.5220/0001286802950301

2 QUANTITATIVE QoS MANAGEMENT

Adaptive profiling of end-users/services is an inescapable approach for an IS dealing with dynamic user-service mapping, but is not sufficient. An adaptive response to problems induced by the dynamics and the heterogeneity of such systems (such as workload, failures, interoperability of the components, as well as the integration of new services...) becomes also necessary. For that, we propose a full agentification of IS components. The functional architecture we propose to tackle this problem and a cooperative protocol between the two upper agents levels are given in this paragraph. The objective of this protocol is to put in touch an entity having a task to achieve (expressed by a request) with entities able to answer (a relevant service).

2.1 Functional Architecture

This part uses two types of agents, which respectively belong to a different level of the architecture of our system (figure 1).

The first type of agent is called *representative* agent. A *representative* agent acts on behalf of the end-user or the service it represents in order to solve a submitted request. A *user* agent has to seek for *service* agents that fit as well as possible the needs expressed by its end-user (calculus for Grid Computing, service for Web Services…). Conversely, a *service* agent, during a publicity campaign for example, can have to find *user* agents likely to be interested by the service it proposes.

The second type of agent is called *site* agent. A *site* agent helps each *representative* agent it contains, locally or remotely (by communicating with other *site* agents to find new relevant *representative* agents). This agentification is required because end-users and services are numerous and geographically distributed. As it is unrealistic to gather them into a single site, we consider a distributed IS as composed of several *site* agents which respectively contain numerous *representative* agents.

Representative agents and *site* agents follow the same "cooperative" protocol (this concept is explained in §4). They interact according to representations they have on other agents' skills. This protocol can be summarized by the five following steps. Ideally, when (*i*) the received message is totally and without ambiguity understood by the agent, it processes it. When (*ii*) an agent cannot associate a meaning to the received message, it sends the message towards an agent it considers relevant for the resolution (this action is called "restricted relaxation"). Thanks to this action, the original sender agent can have the opportunity to become acquainted with a new agent. When (*iii*) only a part of the received message has a meaning for the agent, it returns to the sender a partial answer corresponding to the understood part and it sends the remainder to an agent it considers qualified (restricted relaxation). When (*iv*) the received message has several meanings for the agent, it returns the message to the sender for clarification. When (*v*) two agents want to reach a third one proposing a limited resource and their requests exceed the offer, they are faced with a situation of

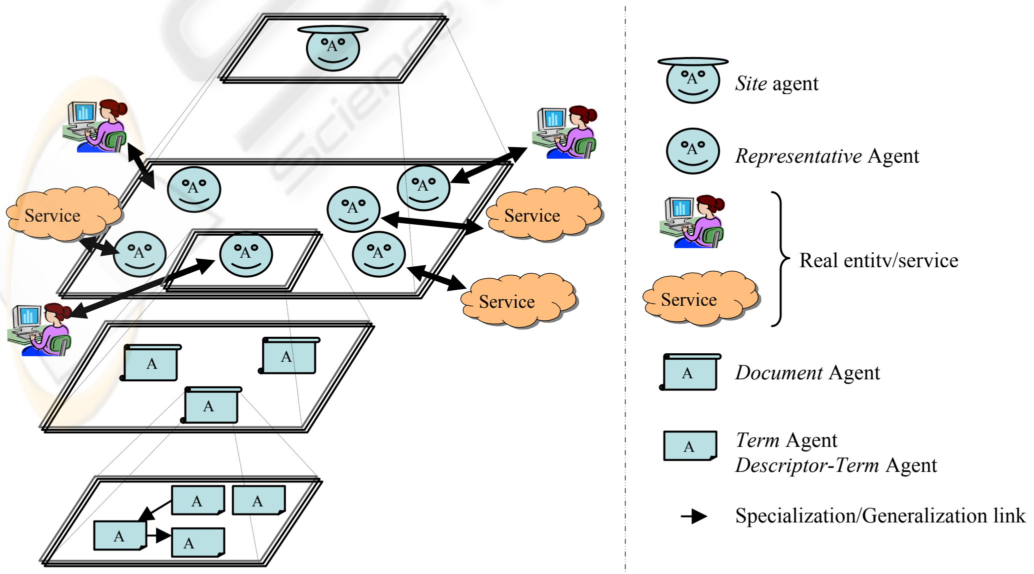

Figure 1: Functional architecture of the proposed system.

conflict. In this case, the third agent guides one of the former agents towards another having similar resource. All the agents are encapsulated by the same behaviour. An agent can thus recommend agents having similar competences when it is overloaded. This behaviour is relevant even if involved agents propose concurrent sale services because the global QoS increases and all services can potentially take benefit of it.

According to this protocol, each agent aims at promoting interactions with agents having similar centres of interest. Conversely, it tries to weaken, even to remove, links with agents having different centres of interest. Furthermore, this part has also to take it into account and self-adapts consequently to the centres of interests of the real represented actors.

2.2 Cooperation for End-Users and Services Connection

The contribution of cooperation for end-users and services mapping was already highlighted in several applications (Gleizes, 2002), (Link-Pezet, 2000). But these applications only focused on qualitative considerations: skills of other involved agents. Obtained results at the end of these projects pointed out the necessity of better taking into account the needs of end-users and proposed services. As obtained results in these applications were closely related to the chosen mode of representation of the centres of interest of involved actors, we decided to check the contribution of the cooperation in a more general context, by also taking into account quantitative properties.

We then made a simulation of a network of heterogeneous, distributed and dynamic ISs (Grid Computing, Web Services and Peer-to-Peer), implementing temporal resources, processes and requests to be solved (Cabanis, 2006). The cooperative protocol previously presented was instantiated to this context by taking into account not only the supposed skills of involved agents but also several criteria (such as CPU performance, storage capacities, standards and bandwidth). In this simulation, representations of agents are implemented by using measurements of need (standards and access rights in Web Services), of probability (for the reliability of the services) and of weighted averages for apparent performances. This simulation, developed in JavAct (see http://www.javact.org for more details), consists in 100 agents, 80% of which are devoted to Grid Computing (GC) calculus. Initially, the IS is represented by a graph of agents randomly connected. This graph evolves according to interactions between agents. 90 requests of GC calculus are submitted each second to different agents of the system. According to the previously presented protocol, each task/request can be relaxed a limited number of times (4 times in this simulation). Beyond this number, the task/request is removed and the sender *representative* agent considers its request as being without response after a given time limit (Time-out). It then adjusts consequently its representations on the *representative* agent to which it sent the request. Obtained results (see figure2) show a progressively decreasing number of relaxations and a decreasing number of Time-out (unsolved requests/tasks) during the system functioning. These results mean that gradually each agent finds its right place in the organisation in spite of unforeseeable events that can occur during the system functioning. In the second curve an asymptotic limit to 20% of time-out can be seen. It is reached when all agents devoted to GC are busy; so the system tends towards its optimality. According to the QoS, these preliminary results show well that the network, as a collective, adapts itself to the characteristics of each entity, only by local perception of criteria and treatments which are independent of any global cost function knowledge.

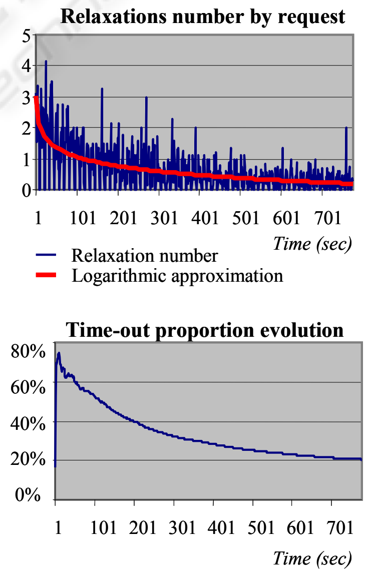

Figure 2: Cooperation contribution on QoS.

3 QUALITATIVE QoS MANAGEMENT

A *representative* agent is composed of two types of representations (also called beliefs or profile):

representations about agents already contacted during previous researches and representations about the end-user or the real service it represents. A *representative* agent is supposed described by a set of textual data (documents such as HTML pages for example). After the lemmatization of this set of documents, the objective consists in extracting from these documents a set of descriptors, i.e. a signature characterising as well as possible their "semantic" content, and therefore, the centres of interest (profile) of the represented end-user or service.

3.1 Functional Architecture

The qualitative QoS part is mainly composed of two types of agents, which respectively belong to a different level of the architecture (figure 1).

The first type of agent is called *term* agent. A *term* agent represents the lemmatized version of a word initially contained in a particular document. Each *term* agent possesses a confidence degree computed in real-time according to the evolution of the execution context and the dynamics of the environment. The more representative a term of the document is, the higher its confidence degree is. Beyond a given threshold, a *term* agent becomes a *descriptor-term* agent. The objective of a *descriptor-term* agent is to be connected/not connected to other *descriptor-term* agents semantically or contextually close/distant. Thus, a *descriptor-term* agent takes part in the construction of a terminological network which will be used to know the centres of interest of a *representative* agent. To do that, we consider each *term* agent obtained at the end of the lemmatization of the documents characterising a *representative* agent. If it is located in the neighbourhood (two *term* agents are neighbours if they are near one another in a document) of a great number of distinct *term* agents, it does not take part in the highlighting of descriptor-term. Its confidence is then reduced.

The second type of agent is called *document* agent. A *document* agent represents a particular document describing an end-user or a service. A *document* agent is initially connected to all *term* agents and *descriptor-term* agents composing it. It will then only keep links with *descriptor-term* agents which characterise it and which make up its signature. The objective of a *document* agent is to allow the highlight of semantic or contextual features which describe the centres of interest of the represented entity, while taking into account their evolution. To do that, we consider that when two *document* agents are similar, they deal with close problematic. In that case, *descriptor-term* agents common to these two documents and having a low confidence degree must change place in the organisation and try to connect themselves to *descriptor-term* agents taking part in the signature of the actual *representative* agent. The treatment is symmetrical when considered documents are dissimilar. The confrontation of documents (similarity/dissimilarity) can be realised at various levels of the mapping process: when a modification of the real entity is made, when a task or a request is submitted to the IS, when an end-user gives a feedback on the quality of the connection relation (QoS) or when an end-user explores a document (during an information retrieval).

Two types of links exist between *descriptor-term* agents. The first one is called contextual closeness link; it connects two *descriptor-term* agents having similar contextual interests. It is directed from an agent A towards an agent B where A is contextually supposed to be more specific than B. The second one is called contextual identity link: it connects two *descriptor-term* agents having similar contextual interests with a bidirectional contextual closeness link. This mean that involved *descriptor-term* agents are considered as similar in the current context.

3.2 Cooperative Profiling

We evaluated the feasibility and the relevance of our adaptive, local, and independent of semantic treatment (except the lemmatization) algorithm, on the design of profiles (Czerny, 2006). We made first experiments, based on a corpus of documents (around fifty) resulting from RFIEC platform (http://www.irit.fr/RFIEC). This corpus was composed of articles of the daily French newspaper "Le Monde" of the year 1994 (dealing with the architecture in Berlin, the drug in Holland and the French conscientious objectors) as well as a list of correspondences "Requests - Documents".

A subset of the terminological network we obtained is showed in figure 3. It was exclusively built according to the local behaviours previously presented (for a better understanding *descriptor-terms* had been translated from French into English). It presents interesting characteristics, notably the absence of meaningless terms, the existence of links between semantically/contextually close *descriptor-term* agents and a kind of semantic proximity in the neighbourhood of some *descriptor-term* agents (for example around the *descriptor-term* "narcotic"). In this example, various links between the *descriptor-term* agent "netherlands" and *descriptor-term* agents such as "narcotic", "cannabis", "drug", "methadone" and "drug addict" can be distinguished. Interests of the active *representative* agent on the Netherlands are then supposed to be related to drugs. If some links are relevant, others (such as the *descriptor-term* agent "according to" in the network and associated links) are less pertinent. A complementary work remains to be done on relations connecting two *descriptor-term* agents.

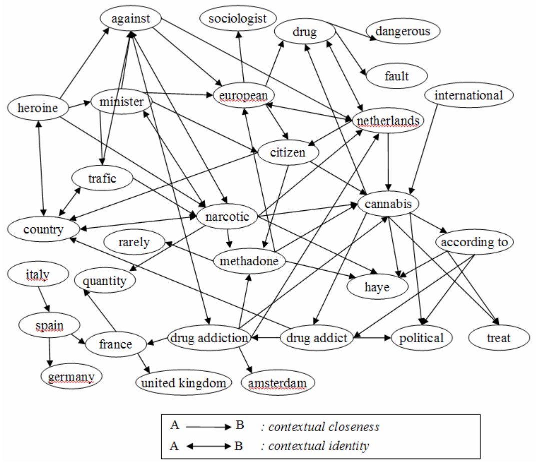

Figure 3: A subset of the obtained terminological network

4 DISCUSSION

For the last few years, user profiling has been becoming a research field of topical interest. This craze has its origins in information retrieval field. Nowadays, the number of responses provided by search engines to a user's request remains high; locating relevant information in the list of returned documents is not an easy task and needs a considerable amount of time. User modelling can contribute to several steps of the research process notably for exploring information sources and delivering only the most relevant documents to a user. (Daniels, 1986) contrasts two classes of user model: quantitative and empirical models which study the external behaviour of a user by observing his interactions with the system, and analytical and cognitive models which are interested in modelling the internal behaviour of a user and try to identify the knowledge and the cognitive processes used. Our work relates to the second point.

4.1 Qualitative QoS

User profile acquisition can be performed in an explicit way, by collecting the information provided by a user via the system interface (selection of topics, definition of attributes, explicit judgment on the relevance of document...), or in an implicit and dynamic way, by observing his behaviour when he is interacting (bookmark saving, link selection, total time spent on a page...) with the system (Lieberman, 1995), (Albayrak, 2005). Most of existing systems use vectorial representations coupled with standard weighting schemes to draw up user profiles. Semantic representations are sometimes used too. They display relations between the units of information characterising the profile by proposing a hierarchy of concepts. They are generally based on ontologies (Baziz, 2005) which confer a quite relative adaptation because they are dependent of a given domain. These approaches are sometimes coupled with techniques that take into account the evolution of the profile. Some systems employ learning algorithm adopted from neural networks or genetic algorithms (Menczer, 1997). Most of these approaches, except (Moukas, 1997), do not address their effectiveness to adapting to changing user's interests. More recent works try to take into account a temporal dimension (short/average/long terms interests) (Kilfoil, 2005) or information related to the context of the user (Bottraud, 2004). But these adaptive approaches rest on global solutions or base their reasoning on the expected result of the system, which makes them not easily applicable for simultaneously managing multiple criteria and unforeseen situations apparition.

4.2 Quantitative QoS

Researches addressing the quantitive QoS problem of applications deployed in large-scale distributed and heterogeneous environments are quite new. As (Kalogeraki, 2005) says "*The inherent ad-hoc nature of these systems makes it difficult to meet the Quality of Service (QoS) requirements of the distributed applications, thus having a direct impact on their scalability, efficiency and performance*". For example in a Peer-to-Peer network (Drougas, 2006) proposes an adaptation mechanism, which trades off service quality level with resource usage. (Cuenca-Acuena, 2004) proposes cooperative agents to gather information about the system state and services. Adaptive QoS is also required in a heterogeneous (wired and wireless) environment. In this context (Chowdhury, 2002) proposes a collaborative framework for adaptive QoS management to support interactive information sharing among distributed and heterogeneous clients, while (Angin, 1998) chooses a mobile middleware toolkit to adapt mobile services. A survey for an adaptive QoS based on middleware solutions is given in (Duran-Limon, 2004). (Chen, 2004) gives also a quite complete survey of approaches for improving the QoS in wireless sensors networks.

An important point that comes out of these works is the need of adaptive mechanisms to tackle the QoS of open and distributed systems.

4.3 Towards an Emergent Optimal QoS

We believe as well, that faced with the diversity of criteria to take into account (CPU performance, capacities storage, standards, bandwidth…), the required QoS cannot be checked and managed by an external supervision. The entities of the system should be autonomous and adapt themselves locally to environmental changes, according to what they perceive and their internal state. Thus, the learning phase is a never-ending process.

We define a system as being functionally adequate if it produces the function for which it was conceived, according to the viewpoint of an external observer knowing its finality. We consider the functional adequacy problem of an open IS as a QoS optimisation one. To reach this functional adequacy, it had been proven (Camps, 1998) that each autonomous agent composing an AMAS and following a cycle composed of three steps (perception/decision/action) must keep relations as cooperative as possible with its social (other agents) or physical environment. The definition of cooperation we use is not a conventional one (simple sharing of resources or common work). Our definition is based on three local meta-rules the designer has to instantiate according to the problem to be solved: (c_{per}) every signal perceived by an agent must be understood without ambiguity, (c_{dec}) information coming from its perceptions has to be useful to its reasoning, (c_{act}) this reasoning must lead the agent to make actions which have to be useful for other agents and the environment. If one of this meta-rule is not checked, the agent is faced to a "Non Cooperative Situations" (NCS). A NCS can be assimilated to an "exception" in traditional programming. Our approach is a proscriptive one because each agent has first of all, to anticipate, to avoid and to repair a NCS. A NCS occurs when at least one of the three previous meta-rules is not locally verified by an agent. Different generic NCSs can then be highlighted: *incomprehension* and *ambiguity* if c_{per} is not checked, *incompetence* and *unproductiveness* if c_{dec} is not obeyed and finally *uselessness*, *competition* and *conflict* when c_{act} is not checked. This approach has great methodological implications: designing an AMAS consists in defining and assigning cooperation rules to agents. In particular, the designer, according to the current problem to solve, has (i) to define the nominal behaviour of an agent then (ii) to deduce the NCSs the agent can be confronted with and (iii) finally to define the processing the agent has to perform to come back to a cooperative state.

This approach is the basis of the QoS management we propose. In the quantitative QoS management, a protocol composed of five steps had been defined. The first step is the nominal behaviour (the agent is in a cooperative state). In the fourth other steps, the agent is faced to an NCS. More precisely, in the second step the agent is faced to a total incomprehension, in the third one, it is faced to a partial incomprehension, in the fourth one to an ambiguity and in the fifth one to a conflict. In the same way, in the qualitative QoS management, two NCSs had been defined: (*i*) a uselessness NCS when a *term* agent is in the immediate neighbourhood of a great number of distinct *term* agents in a document and (*ii*) a unproductiveness NCS when two *descriptor-term* agents common to two semantically close documents have a weak confidence degree. In these two cases, involved agents do not take part in the construction of the terminological network.

Behaviours agents have to carry out when they are faced to an NCS are given for each underlined NCS. These behaviours lead to a local reorganisation of interaction links between involved agents. These agents do not have a view of the global system and do not base their reasoning on the expected collective function realised by the system.

5 CONCLUSION AND PERSPECTIVES

Because of their complexity, current ISs require new approaches to apprehend volatility, dynamics and opening problems. Traditional adaptive approaches, based on the expected function of the system are not easily applicable (even unsuited) to take into account the unforeseeable environmental constraints. Our contribution is a definition of a local adaptive approach, based on permanent cooperative interactions between entities composing the system.

We presented its partial instantiation to quantitative and qualitative QoS managements. First obtained results as well as the contribution of the cooperation on the system adaptation had been displayed. These encouraging results convinced us to study thoroughly the use of the AMAS for the QoS in open IS. Several tasks still remain to be realised: (i) to improve the learning algorithm which associates a signature to a set of documents describing a real entity and to extend its use with the simultaneous representation of several profiles (an agent must have an image of already contacted agents); (ii) to implement the interrogation of the profile built and then to integrate this learning algorithm into the general process to determine relationships; (iii) to study the use of the built terminological network and its relations to allow the expansion of requests to disambiguate a request submitted by an end-user to the system; (iv) to agentify real services/users with cooperative behaviours to obtain truly generic and adaptive networks.

All the researchers in IS consider implicitly or explicitly, that improving the QoS is a multi-criterion and dynamic optimisation problem. It is also our case, but we consider, moreover, that theoretical limitations of the usual algorithms of optimisation lead ineluctably to a reduction of this QoS progressively with the increasing complexity of such systems. New ways based on emergent problems solving can reverse this tendency.

REFERENCES

Albayrak S., Wollny S., Varone N., Lommatzsch A., Milosevic D., 2005. *Agent Technology for Personalized Information Filtering: The PIA-System*, in the 20th ACM Symposium on Applied Computing, Agents, Interactions, Mobility, and Systems (AIMS) track, Santa Fe, New Mexico, USA, pp. 54-59.

Angin O., Campbell A. T., Kounavis M. E., 1998. *The Mobiware Toolkit: Programmable Support for Adaptive Mobile Networking*, IEEE Personal Communications Magazine.

Baziz M., Boughanem M., Aussenac-Gilles N., 2005. *Semantic Networks for a Conceptual Indexing of Documents in IR*, in the 7th International Symposium on Programming and Systems (ISPS), Alger.

Bottraud J-C, 2004. *Un assistant adaptatif pour la recherche d'information : AIRA (Adaptative Information Retrieval Assistant*), Thèse de l'Université Joseph Fourier.

Cabanis V., 2006. *Etude de la dynamique auto-organisationnelle du Web fondée sur l'activité coopérative de ses composants*, Master of research report of Paul Sabatier University.

Camps V., Gleizes M.P., Glize P., 1998. *A self-organization process based on cooperation theory for adaptive artificial system*, in 1st Int. Conference on Philosophy and Computer Science "Processes of evolution in real and Virtual Systems", Krakow.

Chen D., Varshney P. K., 2004. *QoS Support in Wireless Sensor Networks: A Survey,* Proc. of the 2004 International Conference on Wireless Networks (ICWN 2004), Las Vegas, Nevada, USA, June 21-24.

Chowdhury R., Bhandarkar P., Parashar M., 2002. *Adaptive QoS Management for Collaboration in Heterogeneous Environments*, in Proc. of the 16th Int. Parallel and Distributed Processing Symposium

Cuenca-Acuna F., Nguyen T., 2004. *Self-managing federated services*, in Proc. of the 23rd IEEE Int. Symposium on Reliable Distributed Systems (SRDS 2004).

Czerny J., 2006. *Apprentissage de Représentations Locales dans un Système Multi-Agent de Recherche d'Information*, Master of research report of Paul Sabatier University.

Daniels J.P., 1986. *Cognitive models in information retrieval, an evaluation review*, Journal of documentation, 42(4), 272-304.

Drougas Y., Repantis T., Kalogeraki V., 2006. *Load Balancing Techniques for Distributed Stream Processing Applications in Overlay Environments*, in 9th IEEE Int. Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC'06).

Duran-Limon H.A., Blair G.S., Coulson G., 2004. *Adaptive Resource Management in Middleware: A Survey*, in IEEE Distributed System Online - IEEE Computer Society, Vol. 5, No. 7

Gleizes M-P, Glize P., 2002. *ABROSE: Multi Agent Systems for Adaptive Brokerage*, in 4th International Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS), Toronto, Ontario.

Hewitt C., De Jong P., 1982. *Open Systems*, AIM-691, ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-691.pdf

Kalogeraki V., Chen F, Repantis T, Zeinalipour-Yazti D, 2005. *Towards Self-Managing QoS-Enabled Peer-to-Peer Systems*, LNCS Volume 3460

Kilfoil M., Ghorbani A.A., 2005. *SWAMI: Searching the Web Using Agents with Mobility and Intelligence*, in Canadian conference on AI, pp 91-102.

Lieberman H., 1995. *Letizia: An Agent That Assists Web browsing* in Proceedings of the International Joint Conference on Artificial Intelligence, Canada.

Link-Pezet J., Gleizes M-P, Glize P., 2000. *FORSIC: a Self-Organizing Training System*, in International ICSC Symposium on Multi-Agents and Mobile Agents in Virtual Organizations and E-Commerce, MAMA'2000, Wollongong, Australia.

Menczer F., 1997. *ARACHNID: Adaptive Retrieval Agents Choosing Heuristic Neighborhoods for Information Discovery*, Proc. of the Fourteenth Int. Conference on Machine Learning.

Moukas A., 1997. *User Modeling in a MultiAgent Evolving System*, in Proc. of Workshop on Machine Learning for User Modeling, 6th Int. Conference on User Modeling, Chia Laguna, Sardinia.