An Experience on Reputation Models Interoperability based on a Functional Ontology

Laurent Vercouter
École N.S. des Mines de Saint-Étienne
158 cours Fauriel
F-42023 Saint-Étienne Cedex 02, France
Laurent.Vercouter@emse.fr

Sara J. Casare, Jaime S. Sichman, Anarosa A. F. Brandão
University of São Paulo, Brazil
Av. Prof. Luciano Gualberto, 158 - trav. 3
05508-900 São Paulo SP Brazil
{sara.casare, jaime.sichman}@poli.usp.br,
anarosabrandao@gmail.com

Abstract

Interaction between heterogeneous agents can raise some problems since agents may not use the same models and concepts. Therefore, the use of some mechanisms to achieve interoperability between models allows agents to interact. In this paper we consider the case of reputation models by describing an experience of using several existing technologies to allow agents to interoperate when they use reputation notions/values during interactions. For this purpose, we have implemented agents on the ART testbed and we make them use a functional ontology of reputation which was developed to allow the interoperability among reputation models.

1 Introduction

In open multi-agent systems, it is often a problem to achieve agents' interoperability. Agents may be heterogeneous and it is not trivial to establish interactions and cooperations between agents that use different models. We have to provide means by which agents can understand each other and translate part of their models into the model of other agents (and *vice versa*).

In this paper we will focus on the specific case of reputation models. The work presented here describes an experience of using several existing technologies to allow agents to interoperate when interacting using the notions from their reputation models. We implemented agents on the ART testbed [Fullam *et al.*, 2005] that provides an infrastructure for the experimentation of reputation models. Then, we used the functional ontology of reputation [Casare and Sichman, 2005b] to allow agents to describe their own reputation model for external purposes and a query language over OWL [Bechhofer *et al.*, 2004], called nRQL, for agent interactions.

The next section presents the background of this work and describes briefly the used tools (the ART testbed and the functional ontology). Section 3 details our proposition by explaining how an agent can use the ontology to interoperate with others. At last, section 4 shows the implementation of this work on ART.

2 Background

This section first states the current heterogeneity of reputation models and then presents the tools that were used as a background of this work: the ART testbed and the functional ontology of reputation.

2.1 Heterogeneity of Reputation Models

Multi-agent systems arised new kind of chalenges for developers due to their specific and original features as, for instance, the openness of the system, the decentralized management of tasks and the autonomy of agents. These features taken together imply that some agents, potentially implemented by different developers, can enter or leave dynamically the system, and that these agents may be involved in collective tasks and will cooperate with other agents that rely on them. Furthermore, since agents are considered as autonomous entities, we cannot assume that there is a way to control their internal behavior. These features are interesting to obtain flexible and adaptive systems but they also create new risks about the reliability and the robustness of the system. The risk that an agent does not behave well (intentionnally or not) is increased. This bad behavior can cause the failure of some tasks performed by the agent but also of collective tasks in which it is involved.

Solutions to this problem have been proposed by using trust models. The main idea of this approach is to endow agents with a model of other agents that allows them to decide if they can trust these other agents or not. Such trust decision is very important because it is an essential condition to the formation of agents' cooperation. The trust decision process has been the subject of several propositions from different researchers. Most of them use the concept of reputation as the basis of a decision. Reputation is a subject that has been studied in several works [Castelfranchi and Falcone, 1998; Sabater and Sierra, 2001; Conte and Paolucci, 2002; Abdulrahman, 2004; Muller and Vercouter, 2005] with different approaches but also with different semantics attached to the reputation concept. The word "reputation" does not have exactly the same meaning depending on the approach. Some authors consider that an agent has only one associated reputation value that is globally maintained by the system [Zacharia *et al.*, 1999], whereas others think that two agents can have a different opinion about the reputation of an agent [Sabater and Sierra, 2001; Abdulrahman, 2004;

Muller and Vercouter, 2005]. Moreover, some works consider that reputation should be relative to a given context [Castelfranchi and Falcone, 1998], to the sources used to build it [McKnight and Chervany, 2001], to the nature of its target [McKnight and Chervany, 2001], to the facet of the agent that is evaluated [Sabater and Sierra, 2001; Wang and Vassileva, 2003], and so on.

These different approaches result in a wide semantic heterogeneity of reputation models. Two different reputation models do not represent exactly the same things, and sometimes they even use the same word for different concepts. These differences lead to some incompatibility among reputation models. Agents using different reputation models cannot interact since their notions about one another reputation are not understood each other. This is a heavy constraint over the system because it imposes that every agent use the same reputation model. In order to drop this constraint we must provide a way to bring interoperability between the models.

2.2 The ART testbed

One of the consequence of the wide diversity of reputation models and of their heterogeneity is that it is very difficult to compare and to evaluate them. Each model focuses on the representation of some specific concepts that may be different from those of other models. The context and the kind of the target application is also sometimes different. These difficulties have been stated by the community during the international workshop about trust in agent societies at AAMAS'04 [tru, 2004] and it has been decided to create a working group named ART (Agent Reputation and Trust) which goal is the development of a testbed for the experimentation and comparison of trust models.

A first version of the ART testbed has been developed and used for a competition between trust models during the AAMAS'06 conference. This first version provides a simulation engine on which several agents, using different trust models, are running. The simulation consists in a game where the agents have to decide to trust or not other agents. This game is an art appraisal game in which agents are required to evaluate the value of paintings. The game can be summarized as follows:

1. The ART simulation server gives to each agent a given number of *clients*.
2. Each client gives to its agent the description of a *painting*. The client expects that the agent evaluates the painting and gives back to him an accurate value.
3. A painting is mainly characterized by an *era*. According to this era, an agent is more or less accurate in its evaluations.
4. Each agent can ask other agents to help it in its evaluation of a painting if it thinks that these other agents are more accurate than itself for the corresponding era. This exchange of information is called an *opinion transaction* and to get this evaluation the requester pays a fixed amount of money to the provider.
5. Agents also interact by the way of *reputation transactions*. The information requester gives money to the provider but this time, the information that is bought is not an opinion about a painting but the reputation of another agent (a target) for a given era. This reputation represents the estimated accuracy of the target in its evaluation of paintings of the given era.
6. At last, each agent calculates a value for the paintings of its clients and receives money for this evaluation. The game is played during several iterations. At the next iteration, the amount of clients affected at an agent at step (1) depends on the accuracy of its evaluation in the previous iteration.

More details about the ART testbed can be found in [Fullam *et al.*, 2005]. During most of this game, the incompatibility of trust models is not important because they are used locally by their agents' decision process. However, the case of reputation transaction is particular because it is the only moment where agents have to exchange information coming from their trust models. Here, interoperability is required as an agent must be able to formulate a query about reputation values that must be understood by the receiver of the message. This one must also be able to answer and to send the required values in a format that is understood by the first agent. In the current version of the ART testbed, interoperability is obtained by asking the developpers of each agent to provide a way to map their trust model into a very simple common reputation model. This simple model defines that: (i) an agent associates with each of its acquaintances a single value named "Reputation weight"; (ii) this reputation weight is a value in the domain [0:1].

2.3 The functional ontology of reputation

The diversity of domains that are interested in trust and reputation implies a wide and confusing heterogeneity of concepts and definitions. This diversity is also found in the trust models used in multi-agent systems as they are often inspired by social or economic sciences. In order to allow the interoperability of trust models, Casare and Sichman [Casare and Sichman, 2005b] proposed a functional ontology of reputation. This ontology aims at identifying and organizing the concepts and relations involved in trust models. The authors have studied the main models in multi-agent systems in order to identify the important concepts that may appear in a trust model. Then they defined a functional ontology that subsumes the main reputation models.

This ontology includes concepts for different aspects of a reputation model. It both represents reputation as a *process* (using concepts to describe the procedures, the actors, the data, ...) and as a *product* (using concepts to structure the description of a reputation product). For instance, there exists concepts to represent different reputation natures - according to the entity (agent, event, location, etc.) that is evaluated; different reputation types - according to the way it has been computed; different information sources to build reputation; etc.

An implementation of such ontology was described using the OWL language [Bechhofer *et al.*, 2004]. The authors claim that the functional ontology of reputation could be used for interoperability as it is possible to map a reputation model into concepts of the ontology. Thus, an agent can map its own reputation model into these concepts and can understand the reputation models of others if they are also mapped. The next section describes how we propose to achieve this.

3 Using the ontology for the interoperability of reputation models

Reputation building is a learning process that needs several experiences with the evaluated agents to calculate relevant values. Thus, it is often suggested that agents should interact in order to get more information and to accelerate the convergence of the learning process. The lack of interoperability between reputation models is then a problem because it prevents any interaction between agents using different models.

We propose to use the functional ontology as an intermediary between reputation models. Since the ontology subsumes most of the existing reputation models, we can assume that it is possible to define a mapping between a reputation model and part of the ontology. Interoperability is then achievable if an agent selects in its own model some information to send to another agent, maps it into ontology concepts and sends the mapped information to the other agent that can convert it according to its own model. In this section we first identify some interesting concepts of the ontology, then show how to use them for agent interaction. At last, an agent architecture based on this approach is proposed.

3.1 Concepts used to describe a reputation model

The functional ontology proposed by Casare and Sichman [Casare and Sichman, 2005b] aims at covering a broad knowledge about reputation. The concepts included in the ontology can describe several aspects about reputation but a few of them were useful for the mapping of a reputation model. For this purpose, we have identified the following concepts:

- **Reputation role** used to specify the roles fulfilled by some given agents during a reputation calculation process. The roles considered are the *target* who is evaluated, the *evaluator* or the *maintainer* that calculates a reputation value, the *transmitter* that sends a reputation value to another agent and the *recipient* that will use the reputation value.
- **Reputative entity** used to address a given entity. There are three kinds of entities: agents, non-agents (as events, locations, ...), or skills (owned by agents). Each of these entities can be the target of a reputation.
- **Reputation type** that distinguishes different reputations according to the way they where obtained. For instance, primary reputation is obtained from direct experiences with target, whereas secundary reputation is based on second hand information.
- **Reputation evaluation value** and **reputation final value** containing the value of a reputation. An evaluation value is the result of a single evaluation process whereas a final value is computed from several evaluation values and represents a global evaluation about a target.

These concepts are used to build an external description of a reputation model. This external description contains instances of these concepts.

3.2 Defining mapping between reputation models and the functional ontology

As shown in [Casare and Sichman, 2005a], it is possible to map reputation concepts between different models using the functional reputation ontology as an intermediary. The following example illustrates how it should be done using two reputation models: Cognitive Reputation Model [Conte and Paolucci, 2002] and Regret Model [Sabater and Sierra, 2001].

The Cognitive Reputation Model adopted the term “Image” in order to represent an evaluation (telling whether an agent is good or bad) formed using information acquired by agent direct experience, while in the Regret model the term “Direct Trust” refers to the evaluation of an agent behavior that is built from a direct interaction between agents.

By expressing these two concepts in terms of ontology elements, we have found that both of them are subsumed by the same concept, named “Primary Reputation”, that represents the reputation that is obtained from direct experiences with target.

So, suppose we have an agent using the Cognitive Reputation Model that is asking another agent, that is using Regret Model, some reputation information acquired by direct experience about a specific location, Rio de Janeiro. If both of them are using the ontology as an intermediary, the second one can provide its “Image” value for Rio de Janeiro while the first one can understand it as corresponding to its “Direct Trust” value.

3.3 Agent interaction about reputation

Agents interact in order to exchange information. We say that an agent interacts about reputation when the interaction purpose is to receive information about the reputation of someone or something, as shown in the above example. These interactions consist in a simple query/answer protocol, as shown by figure 1. One agent, called the *requester*, sends a query about some reputation information to another agent, called the *provider*. Then the provider answers by a message containing the required information.

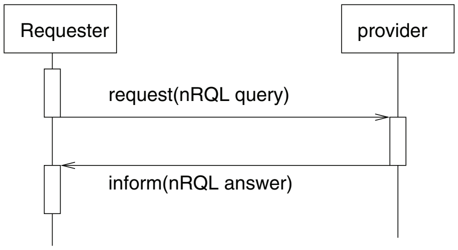


Figure 1: interaction about reputation protocol

The interesting part is not on the protocol or on the format of the messages that we kept very simple, but on the content of each message. Since the agents may use different reputation models, the content of the messages must refer to concepts of the ontology. In its current implementation, the ontology has been written using OWL [Bechhofer *et al.*, 2004]. The queries must then refer to this ontology using a query language for OWL. We decided to use nRQL [Haarslev *et al.*, 2004] that is the query language of the Racer [RacerPro, 2006] tool. This choice is motivated by the fact that Racer is actually the leading tool for reasoning about ontologies and thus nRQL is close from becoming a standard *de facto*. Then the content of a query is a string formatted as a query expressed in nRQL and the answer is a string representing the answer to this query as if it was treated by a Racer engine.

3.4 An agent architecture for interoperability

Agents must be able to formulate and understand queries and their answers in nRQL, using the concepts presented in section 3.1. Figure 2 shows an overview of a general agent architecture that allows this. The figure does not represent a full architecture but rather some elements that have to be added to an existing architecture in order to interact about reputation.

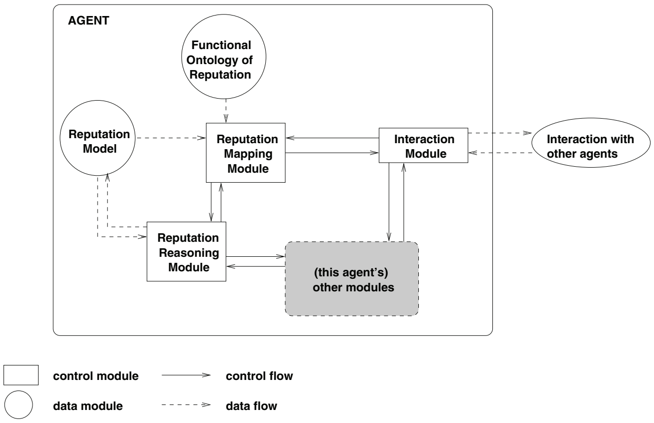

AGENT
Functional Ontology of Reputation
Reputation Model
Reputation Mapping Module
Interaction Module
Interaction with other agents
Reputation Reasoning Module
(this agent's) other modules
control module
control flow
data module
data flow

Figure 2: General agent architecture for reputation interaction

This general architecture functions as follow:

1. The interaction module receives a message. If the message is about reputation, it is transmitted to the reputation mapping module.

2. The reputation mapping module analyzes the content of the message and transforms it to comply with the reputation model of the agent. To do so, the reputation mapping module uses concepts of the functional ontology and the structure of the reputation model. It must also know how to map information from the ontology into concepts of the agent's reputation model.

3. The translated message is forwarded to the reputation reasoning module. This one can now understand it and handle it as it should. Reputation reasoning is out of the scope of our work and it is not studied further here.

4. If the reputation reasoning module needs to interact with other agents, it formulates a message according to its own reputation model and sends it to the reputation mapping module.

5. In this case, the reputation mapping module translates the content of the message into concepts of the ontology and forwards the resulting query to the interaction module that sends it to other agents.

During translation, the reputation mapping module uses especially the concepts described in section 3.1. However, it can refer to other concepts of the ontology to formulate specific relations between concepts, for instance inside a query. An example of this case is given in the next section.

4 Experimentation in the ART testbed

We have implemented agents using the ontology for interactions about reputation. For experimental validation of our approach, we used the ART testbed as a deployment infrastructure for our agents. Firstly, we implemented agents that have the ability of interact about reputation using the ontology. Secondly, we modified the ART testbed to use the reputation mapping facilities for monitoring purposes to facilitate the use of ART as an experimentation tool.

4.1 Agents' interactions in the ART testbed

In the first version of the ART testbed, the protocol used for reputation transaction rely on a common reputation model. This common model is used by agents to represent externally their own reputation model. Since the internal implementation of reputation models is free, this common model was defined to solve the interoperability problem. Thus, the implementation of each agent must provide a mapping of the internal model into the common model. In order to remain general enough, the common model is very simple: it associates with each couple *(agent, painting era)* a numerical value between 0 and 1. Then, reputation transactions in ART consist of queries about a given couple *(agent, painting era)* and an answer containing the numerical value associated with the provider's model.

The problem with this common model is that it is too simple. The mapping of complex internal reputation models into a simplistic one results in a big loss of expressiveness and details. It is thus impossible to perform finer agent interactions about reputation.

Our proposition to use the functional ontology of reputation as a kind of shared reputation model for interaction has been implemented in ART. We have modified the ART engine to allow the exchange of messages for reputation transaction and those contents are rough strings (instead of couples *(agent, painting era)* or numerical value). It is expected that these strings were queries and answers written in nRQL. The use of nRQL queries about the functional ontology allows an agent to formulate a wide range of queries about reputation models. To validate this new version of queries, we implemented agents that use them in order to formulate the same kind of queries than the ones defined in the first version of ART. This implementation was mainly a test of the relevance of the functional ontology to express such queries. The formalization of a query in Racer which result is the direct reputation of an agent called *John* for the evaluation of paintings of the *Impressionism* era is the following:

```
(retrieve (?val)
    (and (?val ReputationWeight)
         (?rep DirectReputation)
         (?rep ?val hasReputationValue)
         (?skill Impressionism)
         (?skill ?target hasReputativeRole)
         (?target ?rep hasReputation)
         (John ?skill hasSkill)))
```

This query uses the relations between concepts to find a given reputation value. The followed path is graphically represented in the figure 3. As explained before, more concepts than those identified in section 3.1 are used but only to represent relations. They are not used to request their values.

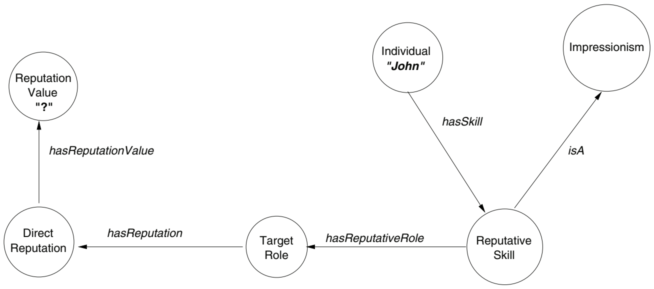


Figure 3: Query path using the ontology concepts and relations

This query means that the requester asks an instance of the *ReputationValue* class that is related to an instance of the class *DirectReputation*, related to an instance of the *Target-Role class*, related to an instance of *ReputativeSkill*. This instance of *ReputativeSkill* should be a subclass of *Impressionism* (the era is represented by the way of an ”isA” relation) and must be related to the instance of the *Individual* class that is called *John*.

With this query formulation, we show that our modifications on the ART testbed still allow to express at least the same interactions as in the first version. Furthermore, it is now possible to use an elaborated query language to perform several other requests. The receiver of a query also uses the same language to answer.

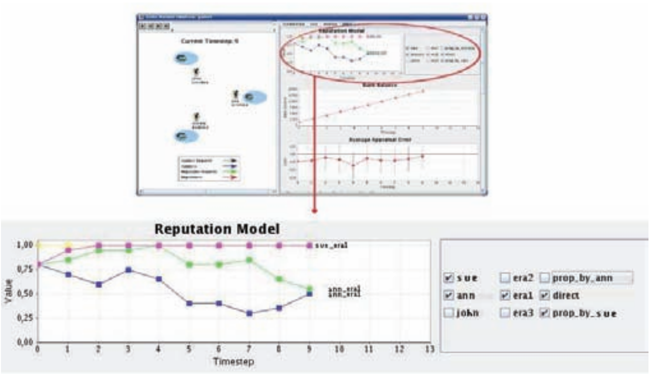
The reputation mapping module of an agent links the query to the agents' internal reputation model. As shown in section 3.2, an agent using the Cognitive Reputation Model [Conte and Paolucci, 2002] will interpret the semantics of this message as a query about the “Direct Trust” of John. Then, it is able to answer this query with the corresponding value. If the receiver is using a different model, for example the Regret model [Sabater and Sierra, 2001], the answer can still be understood by the reputation mapping module. In this case, the term “Image” matches with the required concept of the ontology and the received value will be considered as a direct trust value about John. It is also possible to formulate answers that express that the required concept is not known because it does not exist in the agent's internal reputation model. The use of nRQL allows this kind of “empty” answers.

4.2 Extending the monitoring facilities

The ART testbed has been designed to organize competitions between agents using different reputation models in order to compare them, but it is also a tool used for experimentation of a single reputation model. A researcher can thus implement its own reputation model and evaluate it using the ART testbed. But the first version of the testbed only allows the user of the testbed to state the global performance of its agents (in terms of monetary earnings). It is not possible to follow precisely the behaviour and the evolution of a reputation model.

Our idea is to use the functional ontology, and the mapping implemented for interaction, to improve the monitoring facilities of ART. When we use the ontology for agent-agent interaction, it is a way to describe the values of reputation concepts at a given time. We think that this approach can also be used for agent-human interaction. The ontology, written in OWL, is a step towards understandability by an human user. Then, if we restrain to the concepts identified in section 3.1 to avoid flooding the user with too much information, it can be used to describe the state of a reputation model.

In order to display information according to the concepts of the ontology, we modified two parts of the ART testbed: (i) The database used to store information about played games has been modified to store the values of concepts of the ontology; (ii) The graphical interfaces that read the database to display information have been adapted to display these concepts. A screenshot of the modified ART interfaces is shown in figure 4.



☑ sue ☐ era2 ☐ prop_by_ann
☑ ann ☑ era1 ☑ direct
☐ john ☐ era3 ☑ prop_by_sue

Figure 4: External display of the reputation model

If the developer of the agent has implemented a correct mapping of its reputation model into the concepts of the ontology, it will now be possible to follow precisely its evolution during the game as it can be seen in figure 4. This figure shows the reputation model of an agent called *John*. The curves indicate the value of some reputation concepts at given time steps. Here, the reputation displayed concerned two other agents named ”Ann” and ”Sue” about their evaluation of paintings of the era ”era1”. The types of reputation that are displayed are the direct reputation of *John* (the one built from its own experience with the targets) and the reputation propagated by Sue. The checkboxes at the right of the figure permit a fine selection of the concepts to display. In this example, we can see that both propagated and direct reputations of Sue are better for ”era1” than the reputations of Ann.

5 Conclusions

The work presented in this paper aims at reducing the incompatibility between reputation models. We propose to use the functional ontology of reputation defined by Casare and Sichman [Casare and Sichman, 2005b] to facilitate interoperability. Concepts from this ontology are used to build a shared reputation model. Each agent can map its own reputation model into the shared model in order to interact with other agents using different reputation model. The work described here does not propose new models or technologies but rather present a first experience in the use of existing technologies to allow agents using different reputation models to interoperate.

Such agents have been implemented on the ART testbed. We have slightly modified the first version of ART to extend it with the functional ontology. This extension widens the possibility of interactions between agents. The second contribution of this extension is about monitoring as it now permits to users to follow precisely the states and the evolution of an agent’s reputation model. The extended version of ART may provide a basis for the testbed that will be used during the next competition in 2007. As our future work, we will use the agents implementation from the ART Competition at AAMAS 2006 and make them to use the ontology in order to validate our approach with all the reputation models that were used.

Acknowledgements

This work has been developed in the project “Integration of the Functional Ontology of Reputation into the Agent Reputation Testbed (ART)” supported by FAPESP, Brazil, grant number 2005/02902-5. Jaime Sichman is partially supported by CNPq, Brazil, grant numbers 304605/2004-2, 482019/2004-2 and 506881/2004-0. Anarosa A. F. Brandão is supported by CNPq, Brazil, grant number 310087/2006-6.

References

[Abdulrahman, 2004] A. Abdulrahman. *A framework for decentralized trust reasoning*. PhD thesis, University of London, december 2004.

[Bechhofer *et al.*, 2004] Sean Bechhofer, Franck van Harmelen, Jim Hendler, Ian Horrocks, Deborah McGuinness, Peter Patel-Schneider, and Lynn Andrea Stein. *OWL Web Ontology Language Reference*, February 2004. http://www.w3.org/TR/owl-ref/.

[Casare and Sichman, 2005a] Sara Casare and Jaime Sichman. Using a functional ontology of reputation to interoperate different agent reputation models. *Journal of the Brazilian Computer Society*, 11(2):79–94, november 2005.

[Casare and Sichman, 2005b] Sara Casare and Jaime Simão Sichman. Towards a functional ontology of reputation. In *AAMAS*, pages 505–511, 2005.

[Castelfranchi and Falcone, 1998] Cristiano Castelfranchi and Rino Falcone. Principles of trust in mas: cognitive anatomy, social importance and quantification. In *ICMAS’98*, pages 72–79, Paris, 1998.

[Conte and Paolucci, 2002] Rosaria Conte and Mario Paolucci. *Reputation in Artificial Societies. Social Beliefs for Social Order*, volume 6. Springer, 2002.

[Fullam *et al.*, 2005] Karen Fullam, Tomas Klos, Guillaume Muller, Jordi Sabater, Andreas Schlosser, Zvi Topol, Suzanne Barber, Jeffrey Rosenschein, Laurent Vercouter, and Marco Voss. A specification of the agent reputation and trust (art) testbed: Experimentation and competition for trust in agent societies. In *Fourth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS’05)*, pages pp. 512–518, Utrecht, Netherlands, july 2005.

[Haarslev *et al.*, 2004] Volker Haarslev, Ralf Moller, and Michael Wessel. Querying the semantic web with racer + nrql, 2004.

[McKnight and Chervany, 2001] D.H. McKnight and N.L. Chervany. Trust and distrust definitions: One bite at a time. In *Proceedings of the AAMAS’01 Trust in Cybersocieties workshop*, pages 27–54. Springler-Verlag Berlin Heidelberg, 2001.

[Muller and Vercouter, 2005] Guillaume Muller and Laurent Vercouter. Decentralized monitoring of agent communication with a reputation model. *Trusting Agents for trusting Electronic Societies, Lecture Notes in Computer Science*, (3577):144–161, 2005.

[RacerPro, 2006] RacerPro. *Racer*, 2006. http://www.racer-systems.com/.

[Sabater and Sierra, 2001] Jordi Sabater and Carles Sierra. REGRET: reputation in gregarious societies. In Jörg P. Müller, Elisabeth Andre, Sandip Sen, and Claude Frasson, editors, *Proceedings of the Fifth International Conference on Autonomous Agents*, pages 194–195, Montreal, Canada, 2001. ACM Press.

[tru, 2004] *Seventh International Workshop on Trust in Agent Societies*, New York, USA, July 2004.

[Wang and Vassileva, 2003] Y. Wang and J. Vassileva. Bayesian network-based trust model in peer-to-peer netwoeks. In *Proceedings of the Workshop on Deception, Fraud and Trust in Agent Societies*, pages 57–68, 2003.

[Zacharia *et al.*, 1999] G. Zacharia, A. Moukas, and P. Maes. Collaborative reputation mechanisms in electronic market places. In *Proceedings of the 32th Hawaii International Conference on System Sciences*, 1999.