International Journal of Information Technology & Decision Making
Vol. 16, No. 5 (2017) 1409–1439

DOI: 10.1142/S0219622015500182

Fostering Knowledge Reuse in Communities of Practice by Using a Trust Model and Agents

Juan Pablo Soto[*,‡], Aurora Vizcaíno[†,§]
and Mario Piattini[†,¶]

[*]*Universidad de Sonora*
Department of Mathematics
Hermosillo, México

[†]*Universidad de Castilla – La Mancha*
ALARCOS Research Group
Ciudad Real, Spain
[‡]*jpsoto@mat.uson.mx*
[§]*aurora.vizcaino@uclm.es*
[¶]*mario.piattini@uclm.es*

Published 13 August 2015

Currently, knowledge management is a key issue for companies as it gives them a competitive advantage. A Community of Practice (CoP) is a means to encourage employees to manage knowledge and enables them to exchange knowledge and experience. Members of these communities, however, are often geographically distributed. This hinders the development of feelings of trust between their members, which limits knowledge reuse. Our proposal seeks to minimize the effect of lack of trust between CoP members, thereby fostering the exchange of knowledge. To achieve this goal, we propose a trust model to calculate trust among CoP members, along with a multi-agent architecture to automatically manage the trust model in a CoP. The agents calculate a trust value in each situation, taking the user's profile into account. We also present a tool that recommends sources of knowledge and documents that are trustworthy.

Keywords: Knowledge reuse; communities of practice; trust model, software agents.

1. Introduction

For companies, knowledge management is crucial at present as knowledge has become the most valuable asset for organizations. Good knowledge management improves employees' learning and encourages them to share information.[1] Communities of Practice (CoP) are a means to help organizations to attain the goal of sharing knowledge.[2] CoP can be defined as groups of people who share a concern, a set of problems, or a passion about a topic, and who extend their knowledge and expertise in this area by interacting on an ongoing basis.[3]

CoP fulfill a number of functions with regard to the creation, accumulation, and diffusion of knowledge in an organization[4]:

— They act as catalysts for the exchange and interpretation of information. Since members have a shared understanding, they know what information should be communicated to others and how to present information in useful ways. Thus, a CoP which is spread throughout an organization is an ideal channel for the movement of information across organizational boundaries, such as: best practices, tips, feedback, etc.

— CoP can retain knowledge in a "living" manner, unlike a database or a manual. Even the reutilization of certain tasks and processes can be carried out in a way which responds to local circumstances and which is thus more useful to practitioners. CoP preserve the tacit aspects of knowledge that formal systems cannot capture. For this reason, they are ideal for initiating newcomers into a practice.

— They can steward competencies, maintaining the organization at the cutting edge. Members of these groups discuss novel ideas, work together on problems, and keep up with developments inside and outside the organization. When a community commits to being at the forefront of a field, members distribute responsibility for keeping up with, or pushing for, new developments. This collaborative approach makes membership valuable, because people see their skills and knowledge as being part of a dynamic, forward-looking community.

— They provide homes for identities. Unlike teams and business units, where efforts are directed to the aims of the team or unit itself, CoP are organized around what matters to their members. Thus, for a CoP identity is important. It helps their members sort out what they pay attention to, participate in, and stay away from. Having a sense of identity is a crucial aspect of learning in organizations. If companies wish to benefit from people's creativity, then they must support communities as a means to help them to develop their identities.

1.1. *Trust*

Tedjamulia and colleagues[5] examine various factors that affect community members' behavior such as a feeling of membership or trust. An individual's sense of belonging to a particular team is central to the success of the team because it provides the 'glue' that can promote desirable team cohesion for individuals working together from different locations. It fosters the building of trust in the community.[65,67] Developing community membership implies a clear role, responsibility and the fostering of trust.

Trust motivates individuals in a cohesive community to collaborate and cooperate with others in the community. Collaboration brings together the knowledge, experience and skills of the community members to achieve the collective goals of a project.[65] Moreover, many authors consider that trust facilitates problem solving, by encouraging information exchange and by strengthening the influence of team members. A member of a team should trust that his team mates are competent, knowledgeable and willing to collaborate effectively to deliver business values to customers.[68] People in real life, in general, and in companies, in particular, prefer to exchange knowledge with "trustworthy people".

Conversely, a lack of trust increases the likelihood of misunderstandings and misinterpretation; people work better with others they have confidence in; they avoid interacting with those they do not trust.[6] People with a consistently low reputation in terms of trust will eventually be isolated from the community, since others will rarely accept their justifications or arguments and will limit their interactions with them.

Trust is especially important in virtual communities, where the absence of workable rules implies reliance on the socially acceptable behavior of others.[73,74] However, the development of trust in a virtual setting may be more difficult than in co-located meetings[9] as there is no face-to-face communication. Missing face-to-face interaction and continuous communication between team members can result in poor team bonding. Consequently, people have less confidence in other members' contributions, and knowledge sharing and reuse can decrease as a result.[6] This, in turn, limits the advantages of CoP — due to lack of trust.

1.2. *Recommender systems*

At present organizations must operate in a climate of rapid market change and high information volume, which increases the necessity to create CoP that support knowledge management. It is possible to consider certain Recommender Systems for CoP, which recommend knowledge, information or documents to members of the community with the goal of reusing a company's intellectual capital.

However, these kinds of systems are not always welcomed by a company's employees, as noted by Lawton,[66] for a number of reasons:

(1) Employees often feel overloaded with work, making the introduction of new information into the system difficult.
(2) On occasions employees waste a considerable amount of time searching for information due to lack of research skills or ability to discriminate relevant information.
(3) If there is no quality control with regard to the information introduced into the system, the employees can waste time in sifting through the information and ascertaining its usefulness.
(4) Employees may introduce information of poor value into the system just for the sake of contributing, to take advantage of the incentives which some companies offer to employees who contribute to knowledge creation.

1.3. *Research goal*

The purpose of the present research was to seek to increase or reinforce the trust in a CoP. To do this, we designed a trust model focusing on evaluating the trustworthiness of a knowledge source within a CoP. The goal of this trust model is to detect the most trustworthy knowledge sources (from the people who belong to the CoP). We then considered how to calculate the trust values automatically and to provide this information to CoP members. We came to the conclusion that software agents could be very suitable for that purpose as they can act as recommenders for the members (each user could have his/her own agent) and provide the users with information about the most trustworthy knowledge sources.

Our second contribution is a multi-agent architecture that is implemented with the idea of helping CoP members to know what the most trustworthy knowledge sources are. With this information, members can decide whether to interact with the knowledge which is provided by these "trustworthy sources" or not. We believe that users will feel more confident using the knowledge of the CoP and it will also be easier to find what they need to know, as the agents recommend sources or even actual pieces of knowledge. To illustrate how the agents work, a tool has been implemented and tested.

2. Trust and Reputation: Our Proposal

There is no universal agreement on the definition of trust,[6] despite the fact that this concept forms the basis for economic activity and without it things such as credit agreements, business contracts and customer confidence would not be possible.[7] Hinds and McGrath define trust as confidence in the ability and intention of an information source to deliver correct information.[10] Wang and Vassileva see trust as being a peer's belief in another peer's capabilities, honesty and reliability based on his/her own direct experiences.[18] And Paul and McDaniel take trust to be a subjective expectation that one agent has in another's future behavior, based on the history of their encounters.[28] Others consider trust to be an implicit set of beliefs that the other party will refrain from opportunistic behavior and will not take advantage of the situation.[13,14] Researchers have studied trust from different perspectives: economic and social trust,[15] sociological and psychological foundations of trust,[16] multimedia commerce systems[48]; and interpersonal and inter-organizational trust or initial/evolved trust and contractual trust.[17]

Another concept, closely related to trust, is that of reputation. Mui *et al.* define reputation as a perception that one agent has of another's intentions and norms.[12] Barber and Kim refer to it as the amount of trust an agent has in an information source, created through interactions with information sources.[17] Wang and Vassileva define reputation as a peer's belief in another peer's capabilities, honesty and reliability, based on recommendations received from other peers.[18]

We based our work on Wang and Vassileva's definitions, considering that the difference between both concepts (trust and reputation) depends on who has previous experience. Thus if, for instance, a person has direct experience of a knowledge source we can say that this person has a trust value in that knowledge. However, if another person has had previous experience and recommends a knowledge source to us, we can say that this source has a reputation value.

Our aim is to provide a trust model based on real-world social properties of trust in CoP. There are other trust models which take into account social aspects, such as the Marsh trust model, which has strong sociological foundations.[19] However, the author introduces a large number of variables into the model, making it large and complex. Another model is that of Abdul-Rahman and Hailes, in which previous experience, either from the agent itself or from a recommender, are the only factors considered.[7] Therefore, we suggest that an effective and practical trust model for the virtual environment does not yet exist.

Since the concept of trust in CoP influences many social properties, our trust model has been defined through the consideration of various factors. Both objective and subjective factors are included as decision-making frequently takes into account both types of factors. This issue is one which creates difficulty when modeling concepts closely related to human behavior, such as trust or reputation. Factors that we considered when modeling the concept of trust include the following:

- **Position.** This refers to the place that a person has in the organization in which the CoP exists. We believe that this influences the level of trust, because employees often consider information that comes from a boss as being more reliable than that which comes from a colleague in the same (or a lower) position.[20] However, this is not a universal truth and depends on the situation. In a collaborative learning setting, for instance, collaboration is more likely to occur between people of a similar status than between a boss and his employee or between a teacher and pupils.[21] In an enterprise, this position can be established in different ways by using, for example, an organizational diagram or by classifying the employees according to the knowledge that a person has. This is illustrated in Allen's proposal,[22] which distinguishes between:

— *Technological gatekeepers*, defined as those actors who have a high level of knowledge interconnectedness with other local firms and also with sources of knowledge which are from outside the community. Basically, these act by channeling new knowledge into the community and diffusing it locally; and

— *External stars*, which are highly interconnected with external knowledge sources but have hardly any interaction with other local organizations.

Such different positions inevitably influence the way in which knowledge is acquired, diffused and eventually transformed within the local area. For this reason, as will be explained later, this factor will be calculated in our research by taking into account a weight which can strengthen this factor to a greater or to a lesser degree. This is an objective factor, since it is provided or indicated by the exterior (i.e., it may be provided by the organization or by the community itself).

Level of expertise. Expertise can be defined as the skill or knowledge of a person in a particular area. This is an important factor, since people tend to place more trust in experts than in novice employees. An "individual" level of knowledge is embedded in the skills and competencies of the researchers, experts and professionals working in the organization.[23] Wiig has proposed a model in which five degrees of experience or of knowledge internalization are considered[24]:

(1) *Novice*, a person who may be aware or unaware of the knowledge and how it can be used.
(2) *Beginner*, a person who knows that the knowledge exists and where to obtain it but cannot reason with it.
(3) *Competent*, a person who knows about the knowledge, and can use and reason with that knowledge when given external knowledge bases such as documents and people who are willing to assist him/her.
(4) *Expert*, a person who knows that the knowledge exists, holds that knowledge in his/her memory, understands where it applies, and reasons with it without any external aid.
(5) *Master*, a person who fully internalizes the knowledge, and has a deep and integrated understanding of the values, judgments and consequences of using that knowledge.

Level of experience can be seen as an objective or subjective factor, depending upon where this concept originates. For example, if it is specified by the organization, it will then be considered as objective. However, if its value is given by the opinion of another agent, it will be seen as subjective.

Previous experience. Trust in a decision depends on the "truster's" relevant prior experience and knowledge.[25,26] Experience and knowledge form the basis for trust in future familiar situations.[27] Thus, members of CoP have greater trust in those knowledge sources from which they have previously obtained more "valuable information". That being so, previous experience either increases or decreases trust, and this factor may be extremely useful in the detection of trustworthy knowledge sources in CoP. This factor is subjective because it depends on a person's opinion.

Intuition. When people do not have any previous experience they often use their "intuition" to decide whether or not they are going to trust something — some authors have called this issue "indirect reputation or prior-derived reputation".[28] Westerlund and colleagues note that shared values increase the likeliness of success in creating trust and commitment.[6] Moreover, in human societies, each of us probably has different prior beliefs about the trustworthiness of strangers we meet. Sexual or racial discrimination might be a consequence of such prior belief.[28,71] We have attempted to model intuition according to the similarity between the profiles of two agents: the greater the similarity between one agent and another, the greater the level of trust. This is, of course, a highly subjective value because it is like a 'hunch' and depends directly on the point of view of each person.

As will be explained later, it is possible to decide to give more importance to one factor or to another according to the setting in which the trust model is to be used.

For this reason, we have given each factor a weight which either emphasizes a factor or decreases its importance. An explanation and illustration of how to use this model is given in Sec. 4. Before doing this, we give a brief explanation of the multi-agent architecture that we developed for the recommendation of trustworthy sources. The agents of this architecture will use our trust model to advise members about which source is most trustworthy or most advisable for him.

3. A Multi-Agent Architecture to Support CoP

3.1. *Why agents?*

The artificial agent paradigm constitutes a metaphor for systems with purposeful interacting agents, and this abstraction is close to the way we humans think about our own activities.[29] Moreover, agents can improve the performance of individuals, as well as that of the overall system in which they are situated.[30] This foundation has led to an increased interest in social aspects such as motivation, leadership, culture or trust.[31] Agents can be useful in enabling the creation of trust between two or more actors by acting as mediators. Also, artificial agents have features that help us to emulate the members of a CoP.

Agents have the following useful properties[32]:

- *Autonomy*: Agents operate without the direct intervention of humans or others and have some kind of control over their actions and internal states. This feature is very useful as agents can help CoP members without bothering them.
- *Social ability*: Agents interact with other agents (and possibly humans) via some kind of agent communication language. This feature is also very useful in the emulation of human interactions in CoP as agents can exchange opinions among themselves. In our model agents also share their experiences to help inform other agents.
- *Pro-activeness*: Agents take initiative to achieve their own goals. Agents can exhibit flexible behavior, providing knowledge both "reactively", on user request, or "pro-actively", anticipating the user's knowledge needs. This feature is important as agents can help CoP members to discover information or knowledge sources when the user is searching for knowledge.

In addition to the above properties, the specific characteristics of intelligent agents make them promising candidates in providing a knowledge management system solution.[33] Also software agent technology can monitor and coordinate events, meetings and disseminate information,[34] by building and maintaining organizational memories.[35] Further, agents can learn from their own experiences in the same way that members of a CoP learn from their interactions with other members.

Given the above reasons, we have used agents to overcome the main shortcoming of many knowledge management systems in that the technological architecture of such systems does not match human social behavior.[36–39] We believe that artificial agent technology can help CoP members to reuse knowledge and foster their social interaction and collaboration, as they will be advised as to which are the most trustworthy source/s.

3.2. *Multi-agent architecture*

An architecture determines the mechanisms that an agent uses to react to perceptions, as well as to act, communicate, etc. Most authors consider two levels of response — reactive and deliberative — as the typical levels of a multi-agent system.[40] However, since we seek to model a social setting, we have added a social level, as some previous works have done, in an attempt to emulate social behavior.[41,42] Our architecture is therefore a multi-level one, composed of the reactive level and a second level in which the social and deliberative levels have been joined. Previous works have frequently separated the deliberative and social levels,[43] but after designing various proposals we realized that, in our case, both layers are closely related as all the actions that an agent can carry out need to be consistent with social interactions. Owing to the manner in which the inputs are obtained and the actions are generated (see Fig. 1), this architecture can be categorized as a horizontal architecture as both levels can receive input from the interpreter and generate the actions. The appropriate level will be activated depending on the type of perception.

Reactive level: This is the agent's capacity to perceive changes in its environment and to respond to those changes at the precise moment at which they occur. At this level the agent will execute the request of another agent without using any type of reasoning.

Deliberative-Social level: The agent has a type of behavior which is orientated towards objectives; that is, it takes the initiative to plan its performance, with the purpose of attaining its goals. At this level the agent decides on the best plan of action to follow to fulfil its objectives by using information that it receives from the environment as well as by using its beliefs and intuitions. At this level there are individual goals, which refer to the deliberative aspect, and social or cooperative goals, which refer to the social aspect.

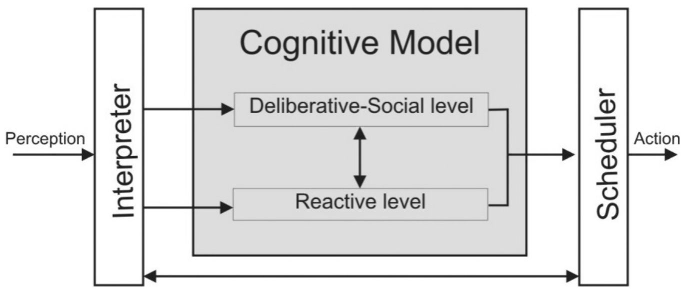

Fig. 1. General architecture.

Two further important components of our architecture are the *Interpreter* and the *Scheduler*. The *Interpreter* is used to perceive changes that take place and to decide which level must respond. The *Scheduler* indicates how the actions should be scheduled and executed.

Each of the levels of our architecture is described in the following subsections.

3.2.1. *Reactive level*

This level must respond at the precise moment at which an event has been perceived (see Fig. 2). For instance, when an agent is consulted about its position within the organization or when a user wishes to send simple answers to the system. This level is made up of the following modules:

Behavior generator: This component is necessary for the development of the architecture as it has to select the agent's behavior. This, therefore, is where the rules that trigger the agent's actions are stored. To trigger off an action, various pieces of information are taken into account, such as the internal module or the agent's interests and beliefs.

Internal model: As an agent represents a person in a CoP, this stores the user's features in order to calculate the trust value using the trust model explained previously. This module stores the following user profile information:

— Expertise. This refers to the degree of experience that the person represented by the agent has, and in what domain. This information can be consulted by other agents, as will be illustrated in Sec. 4.

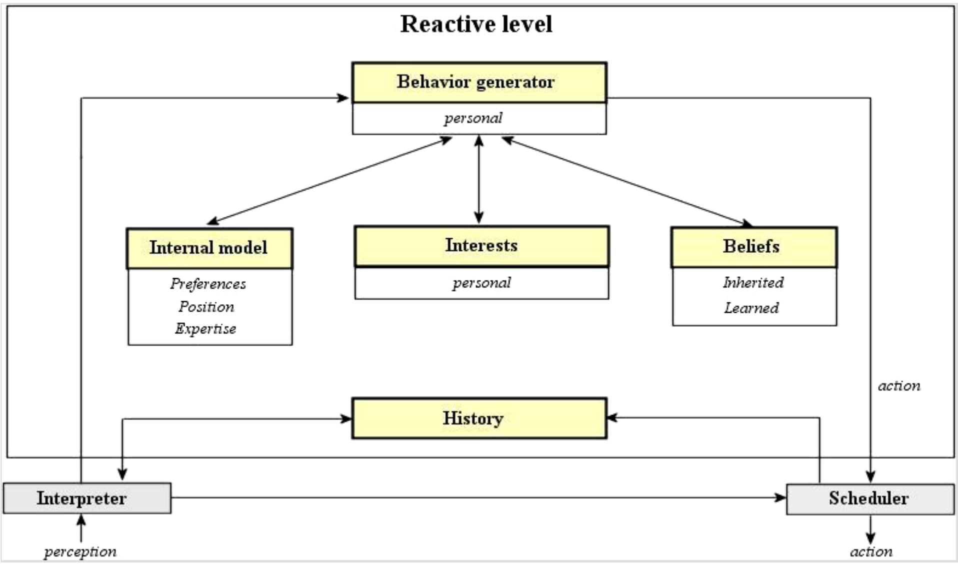

Fig. 2. Reactive architecture.

— Preferences. Here we attempt to represent user preferences. The aim is that the user agent is able to adapt to its user when, for instance, it shows him/her information. Various tests can be carried out by the user to clarify their cognitive profile. These may include, for example, the Felder–Silverman test, which tells us whether the agent is representing a visual user (one who prefers visual representations of presented information-pictures, diagrams, flowcharts, etc.), a verbal user (one who prefers written and spoken explanations) or another kind of user supported by the Felder-Silverman model.[44,45]

— Position. This information can be consulted by other agents wishing to obtain information about a given agent. That is also important for the user agent, since the opinions or beliefs of bosses often have more weight than the opinions of others. In this case the user agent must communicate that fact to other agents.

Interests: These are individual interests which represent the user's needs, such as obtaining knowledge related to a particular topic.

Beliefs: The beliefs module is composed of inherited beliefs and lessons learned from the agent itself. Inherited beliefs are the organization's beliefs as received by the agent. Examples of this might be an organizational diagram or the philosophy of the company or community. Lessons learned are those that the agent obtains during its interaction with the environment. This interaction can be used to establish parameters through which to discover what the agent can trust (agents or knowledge sources).

History: This component stores the agent's interactions with the environment.

3.2.2. *Deliberative-social level*

In this level, the agent's behavior is based on goals. The agent has several defined goals and it attempts to achieve these goals by scheduling actions. As our intention is to represent human behavior in CoP, it is necessary to bear in mind that this human behavior must benefit the whole community. That being so, the agent must consider its individual goals, but it must also act by taking community goals and the community's profit into account, and that is why we have considered a social and deliberative level. The former attempts to achieve social goals (community goals) and the latter focuses more upon achieving individual goals.

At this level, the agent obtains information about the environment and, by taking its interests and intuitions into account, decides on the best plan for reaching its goals (see Fig. 3).

The components of the Deliberative-Social level are:

Social Interests: This component represents community interests. These interests are created when the community comes into being. All communities may share certain interests such as:

- Ensuring that community members are in a constant state of collaboration.
- Identifying and maintaining experts in the community.

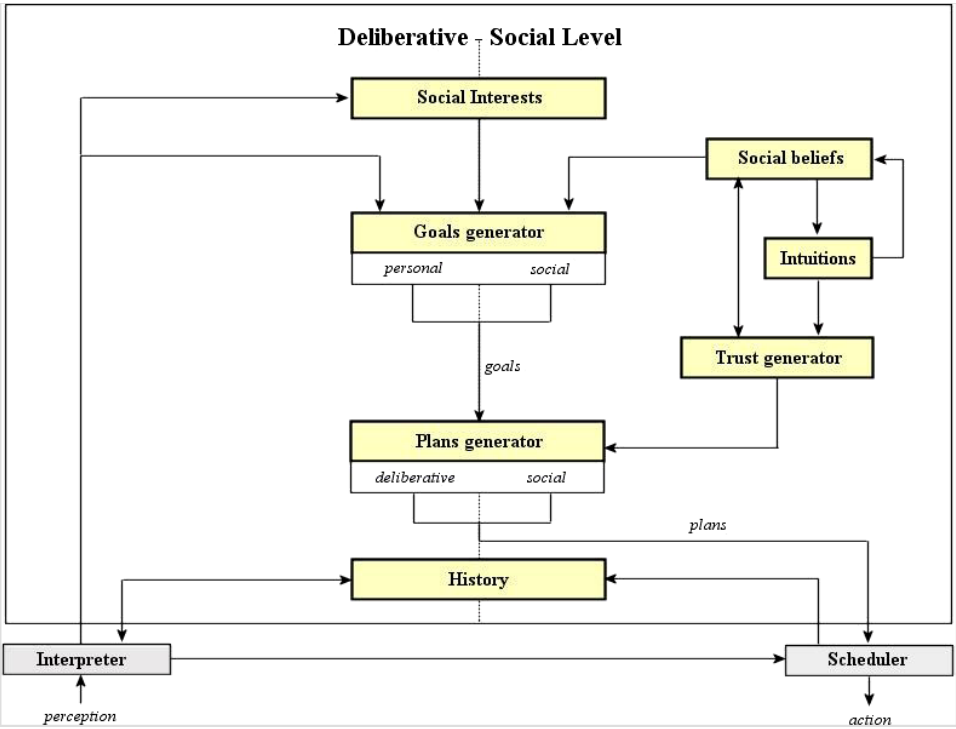

Fig. 3. Deliberative-social level.

- Keeping community knowledge updated.
- Maintaining a trustworthy environment in which community members share their knowledge.

There are also personal interests which influence the whole community, such as sharing suitable knowledge.

Social Beliefs: This module represents a view that the agent has of the environment. In our case these beliefs are composed of the idea that the agent has of the communities and their members. For instance, this module contains information about the community's topics, what areas other members are working in, etc.

Intuitions: Intuitions are beliefs that have not been verified but which an agent thinks may be true. According to Mui and colleagues, intuition has not yet been modeled by agent systems.[28] In this work, we have attempted to adapt this concept by comparing the agents' profiles (as was mentioned in Sec. 3) to obtain an initial value of intuition that can be used to form a belief about an agent when intuition is proved to be true.

Goal Generator: Depending on the state of the agent, this module must decide what the most important goal to be achieved is.

Trust Generator: This is one of the most important modules of this architecture and is in charge of calculating the trust values by using the trust model. An explanation of how the agents calculate this value will be presented in Sec. 4.

Plans Generator: This module is responsible for evaluating how a goal can be attained and which plans are most suitable if this goal is to be achieved — remembering that plans are a specification of the actions that an agent may carry out in order to attain its goals.

History: This component stores the agents' interactions with the environment.

4. A Tool to Test the Trust Model and the Multi-Agent Architecture

This section describes a prototype for a tool developed using this multi-agent architecture and where the agents use the trust model explained previously. This tool is used to test the trust model and to prove that it is possible to implement a multi-agent architecture. It does this by calculating previously how trustworthy the source who proposed that knowledge is and/or the source who evaluated it. The agents of this tool recommend documents to their members by using the trust model.

4.1. *Knowledge sources and knowledge objects*

In order to understand how the tool works, it is necessary to explain the difference between two concepts that will be used: Knowledge Source and Knowledge Object (KO). A Knowledge Source is a generator of knowledge which may be: a person, a book, etc., and various KOs can be obtained from it. Consequently, a KO is a piece of knowledge (e.g., a document) that comes from a Knowledge Source. In a CoP the main Knowledge Sources are its members, so the tool also considers people as being key Knowledge Sources. The tool represents each CoP member with an agent called the "User Agent". Each time a person uses a KO his/her user agent reminds him/her that s/he should rate the KO. In Refs. 24 and 46, the authors describe certain attributes that should be considered when analyzing knowledge, such as:

— Importance: How relevant is this KO for you? This is to discover whether a KO is related to the topic at hand.
— Useful: How useful is the KO for the CoP?
— Time of relevance: How long that knowledge will be useful, since a piece of knowledge may sometimes be relevant over a certain period of time and may later become obsolete.
— Granularity: This indicates whether the knowledge is very general or specific.

The first two values will be used by a User Agent when it needs to evaluate a Knowledge Source and is an essential input into the trust model. The third value will be used by the system to control whether a KO has become obsolete, and the last will categorize the KO.

4.2. *Types of agent*

The first type of agent within a community is the User Agent. Each User Agent can assume three types of behavior or roles similar to the tasks that a person may carry out when working with knowledge management. The User Agent will play one role or another, depending upon whether the person that it represents carries out one of the following actions:

— The person contributes a new KO to the community/ies in which s/he is registered. In this case his/her User Agent plays the role of Provider.
— The person uses a KO previously stored in the community. The User Agent will therefore be considered as a Consumer.
— The person helps other users to achieve their goals by, for example, giving an evaluation of a certain KO. In this case the role is that of Partner.

Figure 4 shows that in Community 1 there are two User Agents playing the role of Partner, one User Agent playing the role of Consumer and another playing the role of Provider.

The second type of agent within a community is called the Manager Agent (represented in black in Fig. 4), which must manage and control its community.

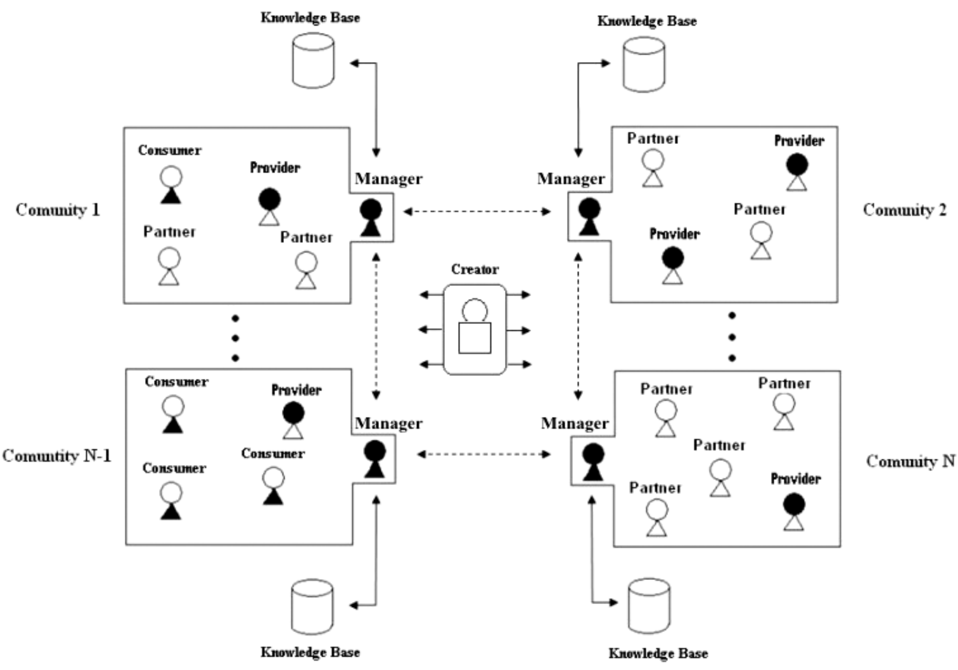

Fig. 4. Communities of agents.

4.3. *The trust model*

In order to make the search for KOs easier, the users in a community can choose a topic from those which are available in the community and the User Agent will attempt to discover a Knowledge Source related to this topic. The topics are defined by the community members, for instance, the Software Engineering Community might have topics such as: quality patterns, software testing tools, business process model, etc.

The general idea is to consider those KOs which come from trustworthy Knowledge Sources according to the user's opinion or needs. In order to discover which Knowledge Sources are trustworthy, the user agents will use the Trust Generator module (see Fig. 3), which implements the trust model as follows. The basic formula used by the trust generator is (1), in which all factors explained in Sec. 2 are represented. According to the amount of previous experience, some factors will or will not be used, as this section sets out to explain.

$$T_{ij} = w_p P_j + w_c \mathrm{LE}_j + w_i I_{ij} + \mathrm{PE}_{ij}, \tag{1}$$

where P_j is the Position of the Agent "j" in the CoP or in the organization in which the CoP exists. LE_j is the Level of Expertise that the person represented by the Agent "j" has in a particular domain. I_{ij} is the intuition that the User Agent "i" has with regard to the Agent "j" and finally PE_{ij} is the value of Previous Experience that the Agent "i" has had with the Agent "j". Finally, w_p, w_e, and w_i are weights with which the trust value can be adjusted according to the degree of knowledge that one agent has about another. So then, if an Agent "i" has had frequent interactions with another Agent "j", then Agent "i" will give a low weight (or even zero) to w_i since, in this case, PE is more important than Intuition. The same may occur with w_e, w_p. So the weights may have the value between 0 and 1, depending on the previous experience that an agent has.

In order to illustrate how this formula is used, let us imagine that an Agent "i" must evaluate how trustworthy another Agent "j" is. Agent "i" will therefore use formula (1) in which T_{ij} is the value of j's trust in the eyes of i.

4.4. *Factors used in the formula for the trust generator*

Each factor of the formula is calculated as follows:

4.4.1. *Position*

When a member joins a community, that person must indicate his/her hierarchical position within the organization. His/her software agent will then calculate the Position (P) value of that person by using the following formula:

$$P = \mathrm{UPL}/N, \tag{2}$$

where UPL is the User's Position Level. N is the Number of Levels in the organization. Therefore, if an organization has five possible position levels for instance, then

$N = 5$, and if the new member has a level of UPL $= 2$, then the value of P will be $2/5 = 0.4$. Thus, the different values of P for a community with five levels will be: 0.2, 0.4, 0.6, 0.8 and 1.0 for the UPL 1, 2, 3, 4 and 5, respectively. The P values will always be between 0 and 1.

Situations may exist in which P will not have been taken into account. This may occur, for example, in those organizations in which all the members have the same hierarchical position or whose members do not wish to consider this criterion. In these cases w_p (weight of position) will be zero and position will not be considered in the formula. A further situation exists in which w_p is equal to zero. This occurs when the value of the Previous Experience PE $> U$ (U being a threshold which is chosen when creating the community). In this case, the agent will use the following formula to calculate the w_p value:

$w_p = \text{floor } (U/\text{PE}_{ij})$ being $\text{PE}_{ij} > 0$,
$U =$ Threshold of Previous Experience.
$\text{PE}_{ij} =$ Value of Previous Experience of an agent "i" with another agent "j".

Thus, when PE_{ij} is greater than a particular threshold U, w_p will be 0, and the position factor will consequently be ignored. However, when one agent does not have enough PE of another, it may use other factors to obtain a trust value. On the other hand, when the agent has had a considerable amount of PE with this agent or with the knowledge that it has provided, then it is more appropriate to give more weight to this factor, since previous experience is the key factor in all trust models, as will be described in Sec. 4.4.4. Therefore, although an Agent "j" has a high value of position, if most of Agent i's previous experience of j has not been successful, then the position will be ignored. This thus avoids the situation of, for instance, a boss who does not contribute with valuable documents, but is considered trustworthy solely because s/he is a boss.

One reason for this incipient hierarchical differentiation is that individuals form inferences and make judgments of others' competence and power based on only seconds of observation.[70–72] Therefore, differences in task participation, which emerge within minutes of interaction, can produce hierarchical differentiation that shapes the entire group experience. Informal hierarchy also emerges from stereotype-based expectations that individuals have of others before they have had a chance to meet.[71]

4.4.2. *Level of Expertise (LE)*

As was mentioned previously, this factor is used to represent the level of knowledge and know-how that a person has in a particular domain. In this prototype of the tool this factor may change, since a person may become more expert in a topic as time goes by.

The levels of expertise considered when creating a community are also indicated, for instance: novice, beginner, competent, expert and master. Each time a new member joins a community, s/he will indicate the level of expertise that s/he considers him/herself to have. If the members of the community and their LE are known to the creator of the community, then that person can introduce them in the tool. Once the LE has been introduced, the user agent will calculate the value for this level by using the following formula:

$$\mathrm{LE} = L/\mathrm{NT} + \mathrm{AV}_j, \tag{3}$$

where L is the level of expertise that was introduced and NT is the number of levels in the community. The term AV_j is the Adjustment Value for Agent "j". This term is extremely important, since it will be used to adjust the experience of each user. This term was introduced to avoid two situations:

— That a person either deliberately or mistakenly introduces a level of expertise (L) that is not his/her level.
— That, whilst in the community, a person becomes more expert but there is no recognition of that.

Initially AV_j will be 0, and each time a member interacts with a document or information provided by j, the member will rate this document or information and send this evaluation to the Manager Agent in charge of managing the community. The Manager Agent will verify whether the evaluation is negative or positive. If it is positive, then Agent j's LE can be modified by calculating AV_j as:

$$\mathrm{AV}_j = (\mathrm{VL}_n - \mathrm{VL}_{n-1})/\mathrm{PT}(n \neq 1).$$

If it is negative, then:

$$\mathrm{AV}_j = -(\mathrm{VL}_n - \mathrm{VL}_{n-1})/\mathrm{PT}(n \neq 1),$$

where VL_n is the value of a particular Level of expertise. PT is the Promotion Threshold which is used to determine the number of positive rates needed for promotion to a superior Level of expertise.

Let us illustrate this with an example. In a community there are four levels (Beginner = 0.25, Competent = 0.50, Expert = 0.75, Master = 1.0). In this version of the tool it is assumed that at least 5 rates are necessary to change the level, so PT will be 5, and AV_j will be $0.25/5 = 0.05$. This is therefore the value that will be added when a positive rate is received or that will be subtracted when a negative rate is received. Thus, one particular experience can either increase or decrease the value of trust. With five positive rates $(5 * 0.05 = 0.25)$ there is thus a promotion to the next level. In other words, an agent whose position was, for instance, beginner, will be promoted to competent. However, the position could decrease if 5 negative rates are received.

4.4.3. *Intuition*

This term is used when the PE is low and other factors need to be used to calculate a trust value. This is one contribution of our work, since most of the earlier trust models are based solely on previous experience. The agents attempt to emulate human behavior, as people often trust more in people who are similar to themselves, have a greater expertise and/or a higher position than themselves. For instance, a person who has to choose between information from two different people will normally choose that which comes from the person who has the same background, same customs, etc. as him/her.[71] By following this pattern, the agents compare their own profiles with those of the other agents, in order to decide whether a person appears to be trustworthy or not. So, the more similar the profiles of two agents are, for instance i and j, the greater the I_{ij} value in formula (1) will be. We could say that an agent 'thinks' "I do not know whether I can trust this agent but it has similar features to me so it seems trustworthy". The agents' profiles may alter according to the community in which they are working. In our case, as the data stored in the agents' profiles are 'position' and 'expertise', both these features will be taken into account. Bearing that in mind, the factors that the tool compares are:

— Expertise Difference (ED)
— Position Difference (PD)

Thus, the Intuition value of an Agent i about j (I_{ij}) is:

$$I_{ij} = \mathrm{ED}_{ij} + \mathrm{PD}_{ij} \quad \text{where } \mathrm{ED}_{ij} = \mathrm{LE}_i - \mathrm{LE}_j \quad \text{and} \quad \mathrm{PD}_{ij} = P_i + P_j. \tag{4}$$

This formula is based on the idea that a person normally has a greater level of trust in people who have a higher level of expertise or who are in a higher position than that person him/herself. Hence, when an agent compares its profile with that of another agent with higher values, the value of intuition will be positive. Let us consider the case of Agent "i", which has values of $\mathrm{LE}_i = 0.5$ and $P_i = 0.5$. This agent wishes to know how trustworthy another Agent "j" is. In this case the agent will use formula (1) and, depending on the information that it has about j, it will or will not be necessary for it to calculate the intuition factor. In this situation we shall suppose that there is little previous experience and that this must be calculated. The values for the Agent "j" are $\mathrm{LE}_j = 0.2$ and $P_j = 0.6$.

$$I_{ij} = 0.2 \quad \text{as } \mathrm{ED}_{ij} = 0.3 \quad \text{and} \quad \mathrm{PD}_{ij} = -0.1.$$

As with position, intuition will or will not be calculated depending on the level of PE. Thus, the weight of intuition (see formula (1)) w_i will be calculated as follows:

$$w_i = \mathrm{floor}(U/\mathrm{PE}_{ij}) \quad \text{with } \mathrm{PE}_{ij} \neq 0.$$

4.4.4. *Previous experience*

This factor is the most decisive of all the factors in formula (1). In fact, all the previous factors depend on it, as an agent will decide whether or not to use the remaining factors according to the value of PE. PE is obtained through the interactions that the agent itself has, so this is direct experience. By interaction we mean that one agent uses a KO provided by another. Each time one agent interacts with another, the first agent asks its user to rate that KO, in order to discover whether the document was:

— important,
— useful,
— up-to-date,
— very general or very specific.

The agent then labels this interaction, for example, if a KO is rated very bad, the interaction value (or Current Experience Value (CE)) will be −0.3. If the KO is rated bad, the CE is −0.2; and 0.1, 0.2 or 0.3 for the cases in which the KO is rated medium, good or very good, respectively. Therefore, the CE value may modify the PE value in accordance with the following formula:

$$\mathrm{PE}_{ij}(x) = \mathrm{PE}_{ij}(x-1) + \mathrm{CE}_{ij}(x), \tag{5}$$

where $\mathrm{PE}_{ij}(x)$ is the value of Previous Experience that the Agent "i" has about another Agent "j" in an interaction x.

$\mathrm{PE}_{ij}(x-1)$ is the value of Previous Experience that the Agent "i" had about another Agent "j" before the interaction x.

$\mathrm{CE}_{ij}(x)$ is the value of the experience that i has had with j in the interaction x.

For example, if an Agent "i" has just taken part in an interaction with another Agent "j", and this is labeled as "bad" but the value of $\mathrm{PE}_{ij}(x-1)$ was 0.8, then the value of $\mathrm{PE}_{ij}(x)$ will be 0.6. This value is obtained from $(0.8 + (-0.2))$ where -0.2 is the value of $\mathrm{CE}_{ij}(x)$. Agent "i" will also send the Manager Agent the value of $\mathrm{CE}_{ij}(x)$, since, as is explained in the Level of Expertise, these values can alter the Level of Expertise indicated initially, either increasing or decreasing it.

As has been explained previously, the Position and Intuition factors depend on the PE value. When an agent has sufficient PE then Position and Intuition can be ignored, and only the PE and the Level of Expertise will be considered. The latter is also included to ensure that an agent takes advantage not only of its own previous experience but also of that of other agents, since LE is adjusted by the AV_j which comes from other agents' previous experience.

4.5. *An example*

In order to illustrate how the prototype for the tool works, let us look at an example. If a user selects a topic and wishes to search for knowledge objects related to that subject, his/her User Agent will follow this algorithm:

The input into this algorithm will be a set of KOs. Each KO may or may not have been evaluated previously, so a KO may already have a list of evaluations (along with the identity of each person who evaluated it), or it may appear without any evaluation. This aspect will be taken into account by the algorithm, which therefore marks out two groups:

Group 1: This group is formed of the KOs that have been evaluated. This is the most important group, since if there are previous evaluations about a KO the agent has more information about it, to know whether it is advisable to recommend it or not.

Group 2: These KOs have not been used previously, so the tool does not have any evaluations about them.

Let us now observe how each group is processed by the algorithm. In Group 1 the KOs will be ordered by a Recommendation Rate (RR) which is calculated for each KO. Hence, RR_k signifies the Recommendation Rate for a particular KO called k, and is obtained from:

$$\mathrm{RR}_k = w_1 \mathrm{TE}_{ik} + w_2 T_{ik}, \tag{6}$$

where TE_{ik} is the weighted mean of the evaluations determined by the trust that an Agent "i" has in each evaluator (the person who has previously evaluated that KO "k"). TE_{ik} is calculated as:

$$\mathrm{TE}_{ik} = \frac{\sum_{j=1}^{n} E_{jk} T_{ij}}{\sum_{j=1}^{n} T_{ij}}. \tag{7}$$

Therefore, T_{ij} is the trust value that the User Agent "i" has in the Knowledge Source "j", and E_{jk} is the evaluation that an Agent "j" has made about a particular KO "k".

The parameter T_{ik} used in formula (6) similarly indicates the trust that an Agent "i" has in a knowledge source "k". Both w_1 and w_2 are weights which are used to adjust the formula. The sum of w_1 and w_2 should be 1.

One advantage of formula (6) is that it permits us to change these weights in accordance with the CoP' preferences, since some CoP may prefer not to take the T_{ik} into consideration and, in this case, w_2 would be zero. Other CoP might wish to give a small weight to this factor and more weight to TE_{ik}, so w_1 could be 0.8 and w_2 0.2. These weights therefore give more importance (more weight) to the trust obtained by taking previous evaluations into account.

The algorithm would then calculate the RR of each KO related to a topic that a user is interested in, and would later show a list with the KOs ordered according to the RR. In the case of there being a high quantity of KOs, then only those with a higher RR would be shown.

Group 2 will use another formula to calculate the RR for each KO, since in this case there are no results of previous evaluations of the KOs. The formula used is therefore:

$$\mathrm{RR}_k = w_1 T_{ix} + w_2 \mathrm{Re}_x, \tag{8}$$

where T_{ix} is the Trust that the User Agent "i" has in the Knowledge Source "x" which provides the KO "k", and Re_x is the reputation that the Knowledge Source has (according to other members' opinion of the agents of other members). This Re_x value is calculated by asking those agents with a higher trust value in the eyes of Agent "i" about the Knowledge Source — this value is obtained by using formula (9).

$$\mathrm{Re}_x = \frac{\sum_{j=1}^{n} T_{jx} T_{ij}}{\sum_{j=1}^{n} T_{ij}}, \tag{9}$$

where T_{jx} is the trust that an Agent "j" has in the Knowledge Source "x" and T_{ij} is the trust value that the Agent "i" has in Agent "j". The agent's opinion about Knowledge Source "x", therefore, is adjusted by the opinion that the Agent "i" has with regard to the agent who is giving its "opinion" (trust value in the Knowledge Source "x").

Figure 5 shows the interface of the tool, displaying the results of a search sorted by the trust values and divided into two sections. In the first one, the KOs that have evaluations are shown (table on the left) and in the second one those documents that have not been evaluated yet (table on the right). The user can choose to open a KO just by clicking on the KO.

This manner of rating trust helps companies to detect a problem which is being seen increasingly in those companies or communities in which employees introduce information which is not valuable, just because they are rewarded if they contribute knowledge to the community. Thus, if a person introduces a KO that is not related to the community with the sole aim of obtaining rewards, the situation can be detected, since when another person evaluates that KO, its rating will be low and the contributor's value of previous experience will likewise become very low. The community agent is thus able to detect whether there is a "fraudulent" member in the community.

In addition, the prototype makes it easier to exchange and reuse information, since the most suitable documents are recommended.[47] For this reason, the

Fig. 5. Showing and sorting results.

prototype can also be understood as a knowledge flow enabler, which encourages knowledge reuse in companies.

5. Evaluation

In order to evaluate the efficiency of our proposal, several proof-tests have been carried out, some of them theoretical and some practical. Here we are going to describe how the trust model was tested in a theoretical way.

To carry out the evaluation, a simulator was developed, which simulates users' behavior in a community. Of course the users are simulated by using software agents. We decided to use a simulator, as apart from the fact that it was faster to obtain results, the main reason for employing it was that all the different possibilities could be emulated in it. In addition, using the tool described above in a community would have taken a long time.

The agents in the simulator exchanged information about the KO, as well as information related to the position and level of expertise that the person who they were representing supposedly has. The agents can also exchange opinions about the other agents' contributions.

This simulation enabled us to evaluate the trust model and to know the trust level that an Agent "x" (Agx) has about another Agent "y" (Agy) and how this level changes according to the evaluations that the Agx carries out about the different Agx contributions.

One of the goals of this simulation is to analyze which is better: to use all the factors proposed in our model or to use just previous experience as other previous works have done.

To do so eight cases were considered — see Table 1. For all cases the criteria were:

— The community has 30 KOs,
— The community was formed by 10 agents,
— The agent evaluator was Agx,
— The agent which was evaluated was Agy,

Table 1. Cases and factors considered.

Number of case	Factors considered
Case 1	Previous Experience
Case 2	Previous Experience and Position
Case 3	Previous Experience and Intuition
Case 4	Previous Experience, Position and Intuition
Case 5	Previous Experience and Expert Level of Expertise
Case 6	Previous Experience, Intuition, Level of Expertise
Case 7	Previous Experience, Position and Level of Expertise
Case 8	Previous Experience, Intuition, Position and Level of Expertise

— Number the evaluations carried out by the agents during the simulation: initially, 25; after that 50, then 75, after 100 and so on. It is therefore possible to see how the level of trust of Agy changes during the simulation according to its own evaluations and those of the rest of the agents.

To avoid making this paper too long, only the last case is explained here. This particular case (8) considers all the factors proposed in the trust model: previous experience, intuition, level of expertise and the agent's position within the community.

Let us imagine that the agent to be evaluated (Agy) and the agent evaluator (Agx) has the highest level of expertise and position in the community. For instance, if we establish a maximum of five levels for level of expertise and position within the community, the level of agent Agy would be 5 for both. The intuition among both agents for this experiment will therefore also be high. As illustrated in Fig. 6, the initial value of trust for this particular case is 3, attributable to the fact that when considering factors such as the position, level of expertise and intuition, the value of trust increases its value up to the maximum level for each factor. Figure 6 also illustrates the variation of the value of trust that agent Agx has on Agy during the first 100 ratings.

In addition, as shown in Fig. 6, after the first 50 ratings the value of trust reaches the highest value 5. So, Agy could be considered as a trustworthy agent in the community. That means that the KO proposed by Agy would be recommended by the agents to their users. On the other hand, when considering only the previous experience, the agent would obtain the highest level of trust when approaching 100 interactions. Therefore, a lot of interactions (more than 50) would be necessary to reach a level of trust near 3, which is a medium level. Agy, which is a trustworthy knowledge source, would not be considered like this in the case of using just previous experience. What is more, if minimum values of trust are obtained, as happened when considering only previous experience, it is a problem for Agy as other actors

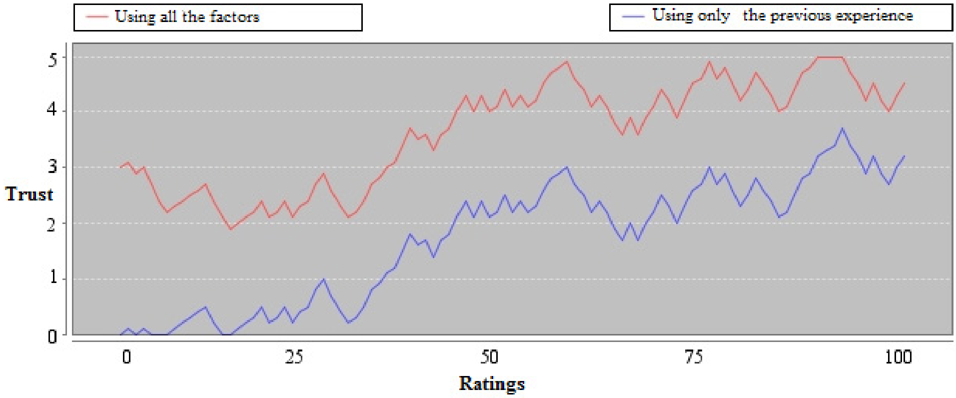

Fig. 6. Comparison between using all the trust factors and only considering previous experience.

Fig. 7. Distribution of the trust values during the first 100 ratings.

may no longer trust him and isolate him. This situation should be avoided, as the goal of a community is to take advantage of the knowledge of all users.

If we use the minimum values for the level of expert and position for the agent Agy, the variation in the trust values for the first 100 ratings would be as shown in Fig. 7.

As illustrated, the value of trust of the agent before interacting is 0.4, which corresponds to the minimum values of the factors of position (0.2) and level of expertise of the agent (0.2). These values were agreed when the case study was designed. The level of trust in Agy does not therefore start at 0 as occurs when only previous experience is taken into account. It is thus easier to integrate this agent into the community as a new agent can obtain an initial value. As is shown in Fig. 7, the level of trust can decrease as the initial contributions of this agent were evaluated negatively by Agx — so the value for trust goes down to 0. Consequently we can see that the level of trust can increase or decrease depending on the other agents' evaluations.

When considering all the trust factors the agents can behave as real users in a community would. This is because trust is measured not only on the basis of the experience gained after the interaction but also on the basis of personal factors such as the position and level of expertise and social factors such as previous experience and intuition. The latter allows an agent to deal with situations of uncertainty that require an opinion with respect to newcomers to the community of users. Finally, we can conclude that if we consider all the factors proposed it will be easier and faster to obtain reliable trust values rather than by considering only previous experience. This is due to the fact that agents obtain an initial value for trust and do not need much prior experience to get a suitable value of trust.

In order to evaluate the tool we are carrying out a performance test. The test is oriented towards the Manager Agent's performance. There is only one Manager Agent per Community and it is important to discover whether this agent is able to manage all the incoming messages from User Agents when a high number of users are connected to a Community.

The preliminary results indicate that the tool starts to become less efficient when there are more than 800 users consulting at the same time. However, as we recommend this tool to small/medium-sized companies it is expected that the number of users will be less than 800 and that the tool will work efficiently.

6. Discussion

This research can be compared with other proposals that use agents and trust in knowledge exchange, since trust and reputation are topics that are widely considered when talking about effective interaction between agents.[49] For example, Abdu-Rahman and Hailes propose a model that allows agents to decide which agents' opinions they trust most and that set out a protocol based on recommendations.[7] This model is based on a reputation or word-of-mouth mechanism. The main problem with this approach is that each agent must keep rather complex data structures that represent a kind of global knowledge about the whole network. Shulz and colleagues propose a framework for exchanging knowledge in a mobile environment.[50] They use delegate agents, which are spread out within the network of a mobile community and employ trust information to serve as the virtual presence of a mobile user. Another interesting piece of work is that of Wang and Vassileva, in which the authors describe a trust and reputation mechanism that allows peers to discover partners who meet their individual requirements through individual experience and by sharing experiences with other peers with similar preferences.[18] This work focuses on peer-to-peer environments. Other pieces of work for peer-to-peer environments include that by: Yu and colleagues where testimonies from several witnesses are combined to determine the trustworthiness of another peer[51]; Xiong and Liu, which also weighs the feedback from other individuals,[52] as we do; and Wang and Vassileva, which proposes a Bayesian network-based trust model.[53] The conditions in a peer-to-peer environment, however, are different to those in a CoP, since the first can have more malicious or unreliable peers. This is because many of them focus on eCommerce settings, where buyers are vulnerable to risks because of potential, incomplete or distorted information provided by sellers.

Barber and Kim present a multi-agent belief revision algorithm based on belief networks.[17] In their model the agent is able to evaluate incoming information, to generate a consistent knowledge base, and to avoid fraudulent information from unreliable or deceptive information sources or agents. This work has a similar goal to ours. However, the means of attaining the goal are different. In Barber and Kim's case they define reputation as a probability measure, since the information source is assigned a reputation value of between 0 and 1. Moreover, each time a source sends knowledge, that source should indicate the certainty factor that the source has of that knowledge. In our case, the focus is very different, since it is the receiver who evaluates the relevance of a piece of knowledge, rather than the provider, as in Barber and Kim's proposal.

Another similar work is that of Wang and colleagues, who use peer-to-peer technology for knowledge sharing in CoP.[54] Thus, the goal of our work could be considered to be quite similar, but the methods used are very different, since they use peer-to-peer technology and we use agents and, furthermore, their work does not take into account concepts of trust or reputation.

This research can also be compared with other proposals that use agents and trust models in knowledge exchange. A summary of some of the models studied is shown in Table 2. With regard to trust, in models such as eBay[55] and Amazon,[56] which were proposed to resolve specific situations in online commerce, the ratings are stored centrally and the reputation value is computed as the sum of those ratings over six months. Thus, reputation in these models is a single global value. These models are too simple in terms of their trust values and the way in which they are aggregated for a CoP, but they work quite well for giving guidance to their clients. Zacharia and colleagues present the Sporas model, a reputation mechanism for loosely connected online communities.[57] In this model, among other features, new users start with a minimum reputation value, the reputation value of a user never falls below the reputation of a new user, and users with very high reputation values experience much smaller rating changes after each update. The problem with this approach is that when a person has a high reputation value it is difficult to change that reputation, or the system needs a high amount of interactions to change it. Another approach presented by Zacharia and colleagues is Histos, which is a more personalized system than Sporas and is orientated towards highly connected online communities.[57] Another reputation model is called REGRET, in which the reputation values depend on time: the most recent rates are more important than previous rates.[58] In the AFRAS model, which is based on Sporas but uses fuzzy logic, a complex computing reputation mechanism is used which handles reputation as a fuzzy set, while decision making is inspired in a cognitive human-like approach.[59]

The main differences between these reputation models and our approach are that these models need an initial number of interactions to obtain a good reputation value and it is not possible to use them to discover whether or not a new user can be trusted. A further difference is that our approach is orientated towards collaboration between users in CoP. Other approaches are more orientated towards competition and most of them are tested in auctions.

Another important feature of our trust model, and that which makes it different from previous models, is that even when a user is new to the community and other agents do not have any previous experience of working with him/her, the trust model allows agents to obtain a preliminary trust value by considering other factors such as the new agent's position and level of expertise, along with the intuition that each agent has with regard to the new member.

Oza and colleagues conducted a study based on an empirical investigation of 18 mature software companies located in India.[69] The researchers describe several

Table 2. Other trust and reputation models.

Model	Authors	Trust and reputation management	Features
ebay	—	Centralized approach	Simple values obtained through interactions.
Sporas	Zacharia[57]	Centralized approach.	Reduces changes when reputation is very high. Most recent reputation values are the most important.
Histos	Zacharia[57]	Centralized approach. Pairwise ratings in the system as a directed graph.	Divides reputation into three dimensions: individual, social and ontological.
Regret	Sabater and Sierra[58]	Distributed approach.	Most recent reputation values are the most important. Presents a witness reputation component.
Afras	Carbo and Molina-Lopez[59]	Distributed approach.	Based on BDI agents. Based on Sporas model but using fuzzy logic. Compares and combines fuzzy sets.
Fire	Huynh *et al.*[60]	Distributed approach.	Four main components: interaction trust, role-based trust, witness reputation, and certified reputation.
TRSIM	Caballero *et al.*[61]	Distributed approach. Trust and reputation values are obtained as global values only associated to a peer.	This model considers trust and reputation as emergent properties of direct interactions between agents, based on multiple interactions between two parties.
Trust model for virtual organization	Hermoso *et al.*[62]	Distributed approach. In this model the confidence is a local rating based on direct interactions; reputation is a rating based on opinions of others; and trust is a rating built as a result from combining.	This model takes into account key concepts of organizational model, such as roles and interactions, as well as their aggregation in groups or organizations.
Trust model for virtual communities	Abdul-Rahman and Hailes[7]	Trust model for virtual communities based on qualitative ratings for estimating trust.	The authors focus on evaluating trust from past expertise and reputation coming from recommender agents without considering explicitly virtual organizations structures.
RepAge	Sabater *et al.*[63]	This model has been designed with an special attention to the internal representation of the elements used to build images and reputations as well as the inter-relations of these elements.	Computational model based on a cognitive theory of reputation. Based on BDI agents.
Travos	Teacy *et al.*[64]	The system is centralized and specifically designed for online communities. It works by users giving ratings to the performance of other users in the community, where ratings consist of a single value that is used to obtain positive and negative feedback values.	Trust is calculated using probability theory taking account of past interactions between agents, and when there is a lack of personal experience between agents, the model draws upon reputation information gathered from third parties.

critical success factors for achieving an initial value for trust, and eventually maintaining trust in software outsource relationships. They suggest that trust is considered to be very fragile in outsourcing relationships. This is another important feature of our trust model, and that which makes it different from previous models, that even when a user is new to the community and other agents do not have any previous experience of working with him/her, the trust model allows agents to obtain a preliminary trust value by considering other factors such as the new agent's position and level of expertise, along with the intuition that each agent has about the new member. In this way we attempt to model human features as when a person has to evaluate something and s/he has no previous experience people tend to use other aspects such as his/her intuition in order to decide whether or not to trust it.

7. Conclusions and Future Work

Communities of Practice are often geographically distributed, thus decreasing the feeling of trust between their members. Consequently there is less knowledge sharing. We realize that trust plays an important role in determining the success and failure of projects for co-located and distributed CoP.

Taking this fact into account, in this paper we propose a trust model and a multi-agent architecture to calculate a trust value which will be used by the agents to propose trustworthy sources to the users. As a consequence of these recommendations, an increase in the reuse of knowledge in CoP is expected.

One important contribution of this paper is the trust model, as it helps to detect experts in a community, since those knowledge sources with high trust values are supposed to be people who contribute with valuable knowledge. The trust model also helps to detect fraud when users contribute with nonvaluable knowledge.

In this paper, the implementation of a trust model and multi-agent architecture has been described. We have also evaluated our contributions theoretically. Here, only the theoretical part has been described. However, we suggest that the trust model and multi-agent architecture are useful in most on-line communities, even when it is a heterogeneous community, as the trust model takes the profile of each person into account and there is an agent acting on behalf of each person. That means that the trust value obtained in each case is personal to each user. This is because his/her profile is used to calculate the trust value and to obtain the intuition value. Moreover, the agents also remember the previous experience that the user has with other users by using the evaluations that the first made about the second.

A challenge for our proposal could be the presence of "lurkers". We can analyze this from two points of view: From the lurker's point of view, in the case of the tool implemented, s/he can use the KOs from the community as his/her agent will recommend the most trustworthy source and knowledge to him/her. This person is obliged to evaluate the documents consulted, however, from the point of view of the community, "lurkers" who do not contribute with any KO are not a big problem as the "lurker" does contribute nonetheless with the evaluations they make each time that they use a KO. These evaluations can help to increase or decrease the trustworthy value of a source. This means that the community might obtain a profit from this member even if s/he does not provide any KO. Of course, we are assuming all the members are honest (including "lurkers") when they make an evaluation of a KO.

One important contribution of the prototype is that it detects experts in a community, since those knowledge sources with high trust values are supposed to be people who contribute with valuable knowledge.

We are currently searching for other functionalities that could be added to these proposals, such as the detection of experts in a topic, since people who contribute with the most useful KO could, at first sight, be considered as experts in that topic.

References

1. J. Calmet and P. Maret, Toward a trust model for knowledge-based communities, in *Proc. 3rd Int. Conf. Web Intelligence, Mining and Semantics (WIMS)* (ACM, New York, 2013).
2. H. Gebert *et al.*, Knowledge-enabled customer relationship management — Integrating customer relationship management and knowledge management concepts, *J. Knowl. Manag.* **7**(5) (2003) 107–123.
3. E. Wenger, R. McDermott and W. M. Snyder, *Cultivating Communities of Practice: A Guide to Managing Knowledge* (H.B.S. Press, 2002).
4. E. Wenger, *Communities of Practice: Learning, Meaning, and Identity* (Cambridge University Press, Cambridge, UK, 1998).
5. S. J. J. Tedjamulia *et al.*, Motivating content contributions to online communities: Toward a more comprehensive theory, in *Proc. 38th Annual Hawaii Int. Conf. System Sciences* (IEEE Computer Society, 2005), p. 193b.
6. M. Westerlund *et al.*, Trust and commitment in social networking — Lessons learned from two empirical studies, *25th IMP Conf.*, Marseille, France (2009), pp. 1–10.
7. A. Abdul-Rahman and S. Hailes, Supporting trust in virtual communities, *33rd Hawaii Int. Conf. Systems Sciences (HICSS'00)* Vol. 6 (IEEE Computer Society, 2000), p. 6007.
8. B. Misztal, *Trust in Modern Societies* (Polity Press, Cambridge MA, 1996).
9. S. Jarvenpaa and D. E. Leidner, Communication and trust in global virtual teams, *Organization Science* **10**(6) (1999) 791–815.
10. P. Hinds and C. McGrath, Structures that work: Social structure, work structure and coordination ease in geographically distributed teams, *20th Anniversary Conf. Computer Supported Cooperative Work*, Banff, Alberta, Canada (2006), pp. 343–352.
11. D. H. McKnight, L. L. Cummings and N. L. Chervany, Initial trust formation in new organizational relationships, *Academy of Management Review* **23** (1998) 473–490.
12. D. L. Paul and R. R. McDaniel, A field study of the effect of interpersonal trust on virtual collaborative relationship performance, *MIS Quarterly* **28**(2) (2004) 183–227.
13. D. Gefen, Reflections on the dimensions of trust and trustworthiness among online consumers, *SIGMIS Database* **33**(3) (2002) 38–53.
14. D. Gefen and D. Straub, Managing user trust in B2C e-Services, *e-Service Journal of Applied Systems Analysis* **2**(2) (2003) 7–23.
15. M. Klemettinen *et al.*, *Privacy, Trust and Group Communications, in Enabling Technologies for Mobile Services* (John Wiley & Sons, Ltd., 2007), pp. 185–225.

16. B. McEvily, V. Perrone and A. Zaheer, Trust as an organizational principle, *Organizational Science* **14**(1) (2003) 91–103.
17. K. Barber and J. Kim, Belief revision process based on trust: Simulation experiments, *4th Workshop on Deception, Fraud and Trust in Agent Societies*, Montreal, Canada (2001), pp. 1–12.
18. Y. Wang and J. Vassileva, Trust and reputation model in peer-to-peer networks, in *Proc. IEEE Conf. P2P Computing* (2003), pp. 150–157.
19. S. Marsh, Formalising trust as a computational concept, Ph.D. Thesis, University of Stirling (1994).
20. S. Wasserman and J. Glaskiewics, *Advances in Social Networks Analysis* (Sage Publications, 1994).
21. P. Dillenbourg (ed.), Introduction: What do you mean by "collaborative learning"? in *Collaborative Learning: Cognitive and Computational Approaches* (Elsevier Science, 1999), pp. 1–19.
22. T. J. Allen, *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information within the R&D Organizations* (MIT Press, Cambridge, MA, 1984).
23. I. Nonaka and H. Takeuchi, *The Knowledge Creation Company: How Japanese Companies Create the Dynamics of Innovation* (Oxford University Press, 1995).
24. K. Wiig, *Knowledge Management Foundation* (Schema Press, 1993).
25. R. Hardin, The street level epistemology of trust, *Politics and Society* **21** (1993) 505–531.
26. A. Josang, The right type of trust for distributed systems, in *New Security Paradigms* (1996), pp. 119–131.
27. N. Luhmann, *Trust and Power* (Wiley, Chichester, 1979).
28. L. Mui, A. Halberstadt and M. Mohtashemi, Notions of reputation in multi-agents systems: A review, *Int. Conf. Autonomous Agents and Multi-Agents Systems (AAMAS'02)* (2002), pp. 280–287.
29. M. Wooldridge and P. Ciancarini, Agent-oriented software engineering: The state of the art, in *Agent Oriented Software Engineering*, eds. M. Wooldridge and P. Ciancarini, NAI, Vol. 1975 (Springer Verlag, 2001).
30. S. Sen, Reciprocity: A foundational principle for promoting cooperative behavior among self-interested agents, *Second Int. Conf. Multi-Agent Systems* (1996), pp. 322–329.
31. R. Fuentes, J. Gómez-Sanz and J. Pavón, A social framework for multi-agent systems validation and verification, *ER Workshops 2004*, eds. S. Wang *et al.*, LNCS, Vol. 3289, (Springer-verlag, 2004), pp. 458–469.
32. M. Wooldridge and N. R. Jennings, Intelligent agents: Theory and practice, *The Knowledge Engineering Review* **10**(2) (1995) 115–152.
33. S. Mercer and S. Greenberg, A multi-agent architecture for knowledge sharing, *Sixteenth European Meeting on Cybernetic and Systems Research*, Viena (2001), pp. 1–6.
34. S. Balasubramanian, R. Brennan and D. Norrie, An architecture for metamorphic control of holonic manufacturing systems, *Computers in Industry* **46**(1) (2001) 13–31.
35. A. Abecker, A. Bernardi and L. van Elst, Agent technology for distributed organizational memories: The frodo project, *5th Int. Conf. Enterprise Information Systems*, Angers, France (2003), pp. 3–10.
36. A. Vizcaíno, J. Soto, J. Portillo and M. Piattini, Developing knowledge management systems from a knowledge-based and multi-agent approach, *International Journal of Knowledge Management* **3**(4) (2007) 67–83.
37. M. Kwan and P. Balasubramanian, KnowledgeScope: Managing knowledge in context, *Decision Support Systems* **35**(4) (2003) 467–486.

38. R. Maier and U. Remus, Implementing process-oriented knowledge management strategies, *Journal of Knowledge Management* **7**(4) (2003) 62–74.
39. W. Scholl *et al.*, The future of knowledge management: An international delphi study, *Journal of Knowledge Management* **8**(2) (2004) 19–35.
40. H. Ushida, Y. Hirayama and H. Nakajima, Emotion model for life like agent and its evaluation, in *Proc. Fifteenth National Conf. Artificial Intelligence and Tenth Innovative Applications of Artificial Intelligence Conference (AAAI'98/IAAI'98)*, Madison Wisconsin, USA (1998), pp. 62–69.
41. I. Ferguson, TouringMachines: An Architecture for Dynamic, Rational, Mobile Agents, Ph.D. Thesis, Clare Hall, University of Cambridge, UK (1992).
42. R. Imbert, Una Arquitectura Cognitiva Multinivel para Agentes con Comportamiento Influido por Características Individuales y Emociones, Propias y de Otros Agentes, Ph.D. Thesis, Universidad Politécnica de Madrid (2005) (in Spanish).
43. R. Imbert and A. de Antonio, When emotion does not mean loss of control, in *Intelligent Virtual Agents*, eds. T. Panayiotopoulos, J. Gratch, R. Aylett, D. Ballin, P. Olivier and T. Rist, Lecture Notes in Computer Science (Springer-Verlag, London, 2005), pp. 152–165.
44. R. Felder and L. Silverman, Learning and teaching styles in engineering education, *Engineering Education* **78**(7) (1988) 674–681.
45. R. Felder, Matters of styles, *ASEE Prism* **6**(4) (1996) 18–23.
46. M. Rao, *Knowledge Management Tools and Techniques: Practitioners and Experts Evaluate KM Solutions* (Elsevier, Amsterdam, 2005).
47. O. Rodriguez-Elias, A. Martínez, A. Vizcaíno, J. Favela and M. Piattini, A framework to analyze information systems as knowledge flow facilitators, *Journal of Information and Software Technology* **50**(6) (2008) 481–498.
48. G. Xu *et al.*, A multi-attribute rating based trust model: Improving the personalized trust modeling framework, in *Multimedia Tools and Applications* (Springer, 2015), pp. 1–17.
49. S. D. Ramchurn, T. D. Huynh and N. Jennings, Trust in multi-agent systems, *The Knowledge Engineering Review* **19** (2004) 1–25.
50. S. Schulz, K. Herrmann, R. Kalcklosch and T. Schowotzer, Trust-based agent-mediated knowledge exchange for ubiquitous peer networks, *AMKM*, LNAI, Vol. 2926 (2003), pp. 89–106.
51. B. Yu, M. P. Singh and K. Sycara, Developing trust in large-scale peer-to-peer systems, *First IEEE Symp. Multi-Agent Security and Survivability* (2004), pp. 1–10.
52. L. Xiong and L. Liu, PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities, *IEEE Transactions on Knowledge and Data Engineering* **16**(7) (2004) 843–857.
53. Y. Wang and J. Vassileva, Bayesian network-based trust model, in *Web Intelligence* (2003), pp. 372–378.
54. C.-Y. Wang, H.-Y. Yang and S. -C. Chou, Using peer-to-peer technology for knowledge sharing in communities of practice, *Decision Support Systems* **45**(3) (2007) 528–540.
55. ebay (2007), Available at http://www.ebay.com.
56. Amazon.com (2007), Available at http://www.amazon.com.
57. G. Zacharia, A. Moukas and P. Maes, Collaborative reputation mechanisms in electronic marketplaces, *32nd Annual Hawaii Int. Conf. System Science (HICSS-32)* (1999), pp. 372–378.
58. J. Sabater and C. Sierra, Review on computational trust and reputation models, *Artificial Intelligence Review* **24**(1) (2005) 33–60.

59. J. Carbo and J. M. Molina-Lopez, An extension of a fuzzy reputation agent trust model (AFRAS) in the ART testbed, *Soft Computing* **14**(8) (2010) 821–831.
60. T. Huynh, N. Jennings and N. Shadbolt, An integrated trust and reputation model for open multi-agent systems, *Autonomous Agents and Multi-Agent Systems* **13**(2) (2006) 119–154.
61. A. Caballero, J. A. Botía and A. Gómez-Skarmeta, On the behaviour of the TRSIM model for trust and reputation, in *Proc. 5th German Conf. Multiagent System Technologies* (Springer-Verlag, Leipzig, Germany, 2007), pp. 182–193.
62. R. Hermoso, R. Centeno, H. Billhardt and S. Ossowsky, Extending virtual organizations to improve trust mechanisms, in *Proc. 7th Int. Joint Conf. Autonomous Agents and Multiagent Systems*, International Foundation for Autonomous Agents and Multiagent Systems, Estoril, Portugal, Vol. 3 (2008), pp. 1489–1492.
63. J. Sabater, M. Paolucci and R. Conte, Repage: REPutation and ImAGE among limited autonomous partners, *Artificial Societies and Social Simulation* **9**(2) (2006) 3.
64. L. Teacy, J. Patel, N. Jennings and M. Luck, Coping with inaccurate reputation sources: Experimental analysis of a probabilistic trust model, in *Proc. Fourth Int. Joint Conf. Autonomous Agents and Multiagent Systems (AAMAS'05)* (2005), pp. 997–1004.
65. G. Asproni, Motivation, teamwork, and agile development, *Agile Times* **4**(1) (2004) 8–15.
66. G. Lawton, Knowledge management: Ready for prime time? *Computer* **34**(2) (2001) 12–14.
67. C. Fiol and J. O'Connor, Identification in face-to-face, hybrid, and pure virtual teams: Untangling the contradictions, *Organization Science* **16**(1) (2005) 19–32.
68. S. Nerur, R. Mahapatra and G. Mangalaraj, Challenges of migrating to agile methodologies, *Communications of ACM* **48** (2005) 72–78.
69. N. Oza, T. Hall, A. Rainer and S. Grey, Trust in software outsourcing relationships: An empirical investigation of Indian software companies, *Information and Software Technology* **49**(5) (2006) 345–354.
70. J. Magee, A. Galinsky and D. Gruenfeld Power, propensity to negotiate, and moving first in competitive interactions, *Personality and Social Psychology Bulletin* **33**(2) (2007) 200–212.
71. A. Todorov, A. Mandisodza, A. Goren and C. Hall, Inferences of competence from faces predict election outcomes, *Science* **308**(5728) (2005) 1623–1626.
72. C. Ridgetway, E. Boyle, K. Kuipers and D. Robinson, How do status beliefs develop? The role of resources and interactional experiences, *American Sociological Review* **63**(3) (1998) 331–350.
73. R. Yaich, O. Boissier, P. Jaillon and G. Picard, An adaptative and socially-compilant trust management system for virtual communities, *Web Intelligence and Agent Systems — Web Intelligence and Communities* **11**(4) (2013) 315–338.
74. C. Ridings, D. Gefen and B. Arinze, Some antecedents and effects of trust in virtual communities, *The Journal of Strategic Information Systems* **11**(3-4) (2002) 271–295.