CASMAS: An Agent-Based Support for Modulated Participation in Cooperative Applications

Federico Cabitza, Marco P. Locatelli and Marcello Sarini

University of Milano – Bicocca {cabitza, locatelli, sarini}@disco.unimib.it

***Abstract*—This paper proposes CASMAS: an agent-based model to design an environment of collaborative applications by taking into account the notion of community. Within this model, communities are characterized by declarative rules that express and shape the participative behavior of the community members. The degree of participation of each member can dynamically change according to her physical location and her position in the logical space of the applications used within the community. The paper shows how this approach can facilitate the design of collaborative applications that are community-aware, that is augmented with mechanisms by which to manage different levels of participation of the community members.**

***Index Terms*—Computer-supported cooperative work, Multi-agent systems, Pervasive Computing**

I. Background and Motivations

As widely recognized in the specialistic literature [1] the multi-agent approach makes easier to define a clear separation between the units of computation and the interactions among them in order to achieve some application goals through "separation of concern" and modularity [2]. Moreover agents can be conceived of as useful tools to describe (complex) systems from a systemic point of view. Because of the complexity of systems to design, it is impossible to predict (and design) in advance all the possible behaviors of the running system: hence agents are provided with simple behaviors and interaction capabilities and let interact within some computational environment so that the system is able to cope with unpredictable patterns of conditions by exhibiting an overall behavior that is an emerging property of the system itself [3]. The relevance of these approaches is also due to some important characteristics that they provide to designers: distributedness, openness, scalability, incremental design. In fact, agents are inherently distributed, and this makes the system more easily open, in terms of the possibility to add new elements given that they behave according to the established protocol; and robust, in terms of easy substitution of malfunctioning agents and of modification of incrementally designed agents.

More recently the characteristics of agent-based approaches have been also considered in the light of the design of applications that support collaboration among people [4]. In the area of computer supported cooperative work (CSCW) cooperative applications pose strong requirements in terms of flexibility, adaptability, openness to environment in order to reflect the complexities of real work settings, that is of environments (or workplaces) where people work distributed in space, and can freely join and leave dynamically collaboration spaces, where collaborative behaviors can change according to the context. Agent-based approaches have been proposed to support different aspects of human collaboration: some of them are focused on the management of workflows which require adaptivity and dynamicity in dealing with a flow of work (representing either tasks to be accomplished or documents) among team members (see for instance [5]). Other agent-based approaches deal with coordination issues, ranging from support to not very structured interactions among members of small groups like the ones occurring in meetings [6] to more prescriptive interactions among distributed actors mediated by appropriate coordination mechanisms [7]. One of the most critical aspects concerning human collaboration is about how people act, learn, and interact together within the so-called communities of practice. In our view, the notion of community (in the sense initially proposed, and denoted as Community of Practice, by Wenger [8] and further articulated by Andriessen [9]) is a good mean to conceptualize how people mutually recognize, gather together, interact, collaboratively access and share resources, and move around to meet other people and exploit further resources. In fact, a community is spontaneously built, grows and evolves legitimating various degrees of participation of incoming members on the basis of its internal rules, conventions and practices: this is usually called "legitimate peripheral participation". The degree of participation of an actor is proportional to its distance from the center of the community, i.e., from the locus where the (physical and/or logical) ties which link its members together are stronger.

In our view, the possibility of considering different degree of participation of community members is a crucial aspect to be taken into account so as to design applications supporting

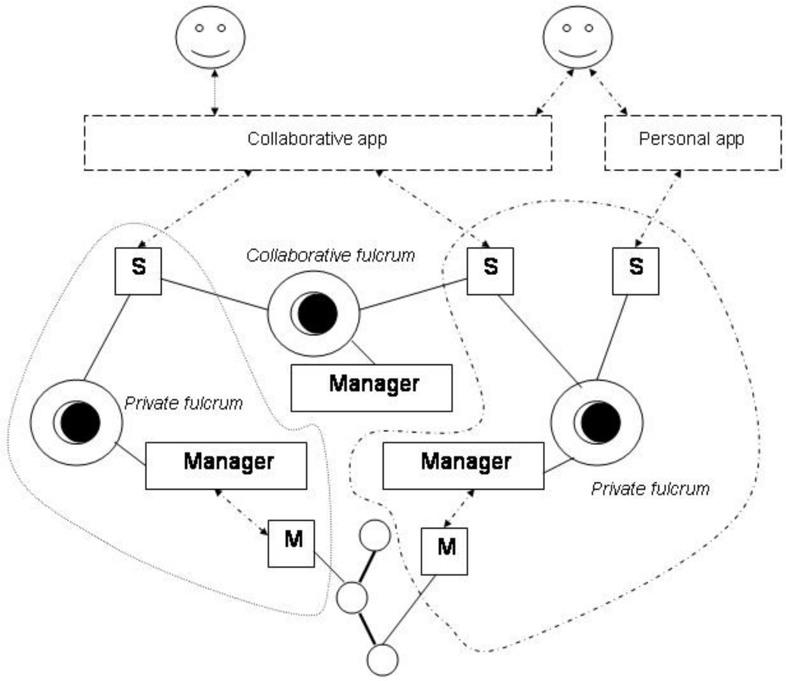

Fig. 1. The CASMAS model.

collaboration among community members or, in other words to build a new typology of applications which are more community-aware in terms of support based on their inner membership and participation mechanisms. With community-aware applications we hint to the fact that applications conceived as supportive of cooperative work can also play a significant role in supporting community life and community-oriented activities. We think this can be made possible if these applications are embedded within a network of interactions and information flows that occur between the human actors, their personal devices and the applications they use. In this common and shared information space (cf. later the concept of Fulcrum) the cooperative applications can then become aware of the way different levels of participation are managed with respect to the users' ability to get accustomed and align with the practices, conventions, artifacts, and knowledge sharing (learning) modalities of the community they belong to.

Taking into account the above considerations about agent-based approaches and the relevance of modulating community participation, we aim at defining an agent-based model, the CASMAS (Community-Aware Situated Multi-Agent Systems), which could be used to design community-aware applications. This model results from an extension of Santana, a framework for the management of distributed inference based on reactive behaviors programming [10], with the main features of MMASS (Multi-layered Multi-Agents Situated Systems), a model that has been proposed for managing and modulating awareness information in cooperative applications [11, 12]. In fact, our model should be able to recognize and support modulated participation of members of a community where modulated participation calls for a notion of metrics and the latter for a notion of topological space. For this reason, the Santana framework, which is able to model distributed computational capabilities together with the sharing of information and reactive behaviors through rules mobility, has been integrated with the MMASS model in which the topological space and the consequent modulated diffusion of information are first-class objects.

Other agent-based models like Co-Field [13] implemented by TOTA [14] could be used to modulate the different degree of participation of community members since they take into account concepts of topological space, distance and propagation of information which is modulated within the space itself. But in those cases the modulation of information is influenced according to only a topological structure usually representing the underlying network architecture rather than one or more topological structures representing also logical aspects of the domain. Indeed, CASMAS allows for the definition of general criteria by which to establish the level of membership of people in a community through the notions of topological space and of field diffusion, which can represent a combination of both physical and logical aspects that dynamically characterize the community membership. Moreover, the same notions allow for the computation and modulation of different levels of participation. Hence by our model, we provide designers with a richer semantics in defining different metrics expressing possible levels of participation since the model makes possible to combine the mutual physical position of users, as well as their logical location, in order to define how the information can be modulated through the environment and according to the relationships among members. The paper is organized as follows. The next section presents the CASMAS (Community-Aware Situated Multi-Agent Systems) model, which integrates the main features of Santana and of MMASS. Then, a high-level software architecture to implement the

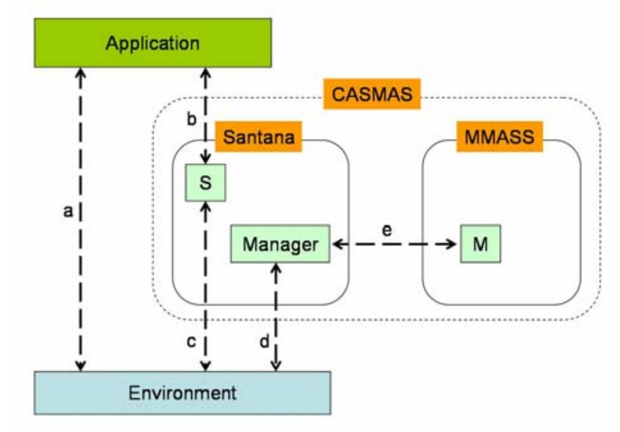

Fig. 2. The CASMAS' software architecture.

model is presented. Next the model is illustrated through a scenario. The state of its current implementation and its foreseen developments conclude the paper.

II. MMASS AND SANTANA IN A NUTSHELL

MMASS is a multi-agent model based on the *perception-reaction* paradigm. Agents are located on *sites* that constitute a *topological space* determining the agents mutual perception. In fact, agents can *directly interact* when they are located in close sites or can *remotely interact* when they are sensitive to the signals emitted by other agents. These signals within the MMASS model are called *fields* and their intensity is modulated by space according to a *diffusion function*, which takes into account the space topology. A *sensitivity function* characterizes each agent type and takes its current state as argument. The perception of a field by an agent triggers a reaction that can cause a *change of* the perceiver's *state or position or the emission of a field* by the perceiving agent. A system can be composed of several topological spaces (*multi-layers*), each characterized by its agents and their behaviors; layers communicate by means of *exported-imported fields*. Ad hoc layers that fictitiously represent applications can send information to the MMASS by means of imported fields and in so doing awareness information about those applications can be properly managed (more details on this architecture can be found in [12]).

Santana is a methodological framework conceived for the development of distributed inference systems in the Pervasive Computing application domain. The Santana framework is grounded on the *interconnection metaphor* in that any Intelligent Environment is conceived as a *web of computational sites* where devices of different computational and interactional capabilities interact. Interaction is realized (or better yet, mediated) through a *blackboard* mechanism, that is through a common space where devices share contextual information (called *facts*) as well as reactive behaviors (called *rules*), which can be acquired by or moved across the computational sites. In this way, the pervasive environment can reach an intelligent behavior as a result of synchronous inference activities exhibited by distributed computational sites. Moreover, a blackboard approach makes the computational environment quite *flexible towards dynamic situations*: new devices, new actors leaving and joining the system, interaction patterns varying according to the context can be dynamically managed by means of suitable meta-rules that act as bridges between concepts (represented by declarative facts) and rules and that hence allow for the (de)activation of behaviors on an event-driven basis (i.e., the local, as well as the "global", control flow is not completely predetermined by the programmers of the devices and applications involved in the same pervasive environment).

A. Their integration into CASMAS

Grounding on the two models outlined above, we then propose CASMAS: a model by which to conceive a "loose" integration between collaborative applications so that they can become more "community-aware". To reach such high-level goal, CASMAS combines the MMASS functionality of modulating information between agents and the Santana functionality of supporting cooperating agents in sharing information and behaviors (e.g., tasks and ways to accomplish them). The combination of these two approaches fits the requirements of a *cooperative intelligent environment* that in CASMAS is interpreted as a *constellation of dynamically defined and interacting communities*. On one hand, cooperation requires the notion of agent as entity able to perceive context and propagate information on that context, as well as the notion of modulated mutual perception (awareness) among agents, that is a first-class concept of MMASS. On the other hand, cooperation in an Intelligent Environment requires the functionality of Santana to manage disparate and scattered devices, private and common information spaces as well as agents that are aware of context and endowed with behaviors that are adaptive and reactive to context [15]. Accordingly, the rationale behind CASMAS is to model a cooperative Intelligent Environment as composed of two main parts. First, a set of common information spaces, called *fulcra*, by which information and behaviors concerning communities practices and individual actors are managed (respectively, *cooperative fulcra and private fulcra* - see Figure 1). Each fulcrum is accessed by *S-agents*, one for each

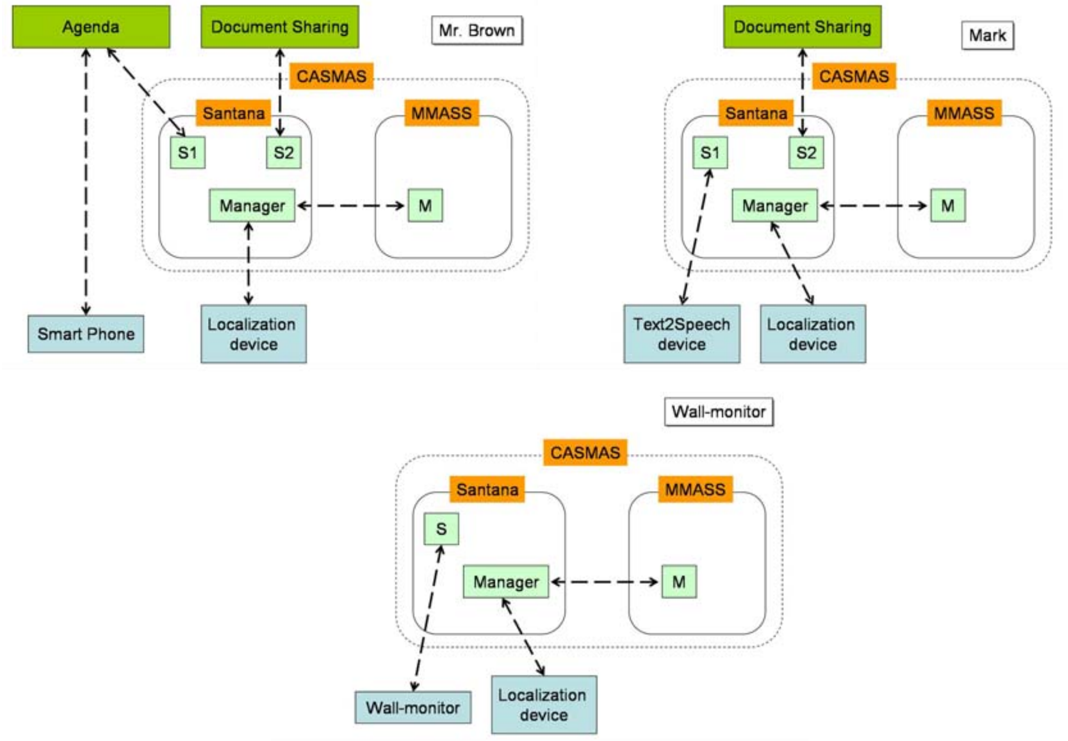

Fig. 3. Examples of configuration of the CASMAS architecture.

(human) actor involved in the (cooperative) application(s) in use by the community[1]. Through the fulcrum, S-agents can share both declarative representations of context (facts) and reactive behaviors that characterize the community in terms of conventions, practices or shared knowledge. Accordingly, these behaviors are called *community rules* since, by being *shared* and *followed* by all the community members they literally make and demarcate the community. As a community-oriented specialization of Santana, CASMAS provides the designer with two transparent mechanisms to manage community rules that are implemented through suitable CASMAS *meta-rules*. The first is called *community enforcing* and it is used to manage inhibition of community rules, as well as updating and overwriting once they have been fetched within each S-agent. By means of this mechanism, community rules can dynamically change to reflect a more context-aware alignment of the community members towards common and ever-changing cooperative goals. The second mechanism is called *community participating*. Through this mechanism, according to the number of members that activate a community rule and the number of activations of that rule, the *salience* of that rule is dynamically changed (that is, it is modified the rule attribute expressing the probability of the rule to be chosen, activated and then executed at a certain contextual condition). This *participating* mechanism allows a community to change dynamically its nature and policies, also according to the contextual response of its members to these policies. In this way a rule that has been first "injected" into a fulcrum with a low salience (and that hence can be seen as just a *suggestion*) can become a (shared) *practice* and then even a prescriptive *direction* according to its changing and growing salience.

All S-agents that stand proxy for a human actor[2] (e.g., A) in some collaborative fulcra are also connected with the private fulcrum associated to A: this allows a smooth interaction between private and cooperative tasks and information repositories, thus fulfilling a well known requirement of cooperation.

The second part of a CASMAS model encompasses a set of topological spaces that are "inhabited" by M-agents whose behavior is defined according to the MMASS model; CASMAS spaces can have a dynamic structure, a feature inherited by MMASS, but this feature is not used in the scenarios described later. Besides conveying contextual information, the role of M-agents is to compute the degree of participation of human actors in the communities that are built around the collaborative fulcra. The interplay between sensitivity to fields and fields propagation, which depends on M-agents state and position, "shapes" the M-agents mutual perception and computes how tight their mutual proximity is. This information flows towards the fulcra described above; these "react" to this flow by implementing the desired degree of participation through the adaptive behavior capability provided by Santana. This flow of information, modelled in

[1] The behavior of S-agents can either fully define the cooperative application associated to the collaborative fulcrum or, more realistically, define an interface between the cooperative application and the pervasive environment in which it is activated.

[2] In principle, when we refer to human actors also artificial ones could be considered. However, since the focus of the paper is on human cooperation, we will refer to human actors; if cooperation involves artificial actors as well, the extension of the illustrated mechanisms to them is immediate.

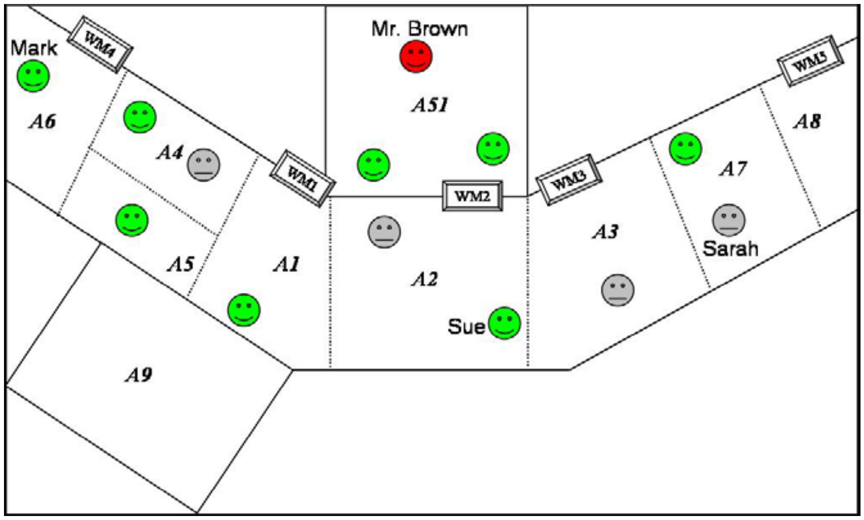

Fig. 4. The described scenario at the University. PCC workshop is located in room A51.

terms of exported fields, is the basic means by which the integration between Santana and MMASS is realized, and it makes the interaction with the external environment bi-directional. In order to realize this integration, while keeping the frameworks both fully decoupled and autonomous in their use and implementation, at each fulcrum is associated a special S-agent, called Manager. This agent is characterized by some *rendering rules*, that is rules that transform topological representations of the MMASS model into declarative representations (facts) by which the communication between the fulcrum and the M-agents populating the topological space(s) is managed. The following section illustrates the CASMAS functionalities through two scenarios: a simple scenario will show the communication patterns in some detail, while a more complex one will more clearly describe the CASMAS expressive power.

III. CASMAS SOFTWARE ARCHITECTURE

As stated in the previous sections, CASMAS is a model by which to conceive a "loose" integration between collaborative applications so that they can become more "community-aware". Accordingly the model must be open to the software applications and to the environment as well. Due to these requirements, the points of interaction between the model architecture, the applications and the devices must be identified, so to characterize the high-level software architecture of CASMAS (see Figure 2). The architecture is composed of a Santana module, which includes S-agents (S in Figure 2), a Manager, a private fulcrum and the community fulcra; and of a MMASS module, which includes M-agents (M in Figure 2) and a topological graph. In our view, the interaction between the environment and the CASMAS architecture is delegated to the S-agents in that they can interact both with the software applications and the environment (arrows *b* and *c* in Figure 2). Conversely, the Manager can only interact with the environment (arrow *d* in Figure 2) and specifically only with the localization devices in order to acquire the physical location of the person that they are associated with. Software applications, which are entities outside the CASMAS architecture, can interact directly with the environment (arrow *a* in Figure 2).

The interaction between an S-agent (or the Manager) and the devices in the environment can be bidirectional and it is mediated by a proxy fact, i.e., a fact that represents the visible state of a device and that is declared in the private fulcrum associated to the device's user. The S-agent that interacts with the device owns those rules that can be fired by changes in the proxy fact; by modifying the proxy fact, this agent is also able to modify the state of the corresponding device. In this way, the S-agent and the device are fully decoupled but the strict relationship between them is preserved by putting the rules only in the interested S-agent. This approach has several benefits: first, the device is potentially visible to all the S-agents linked to the private fulcrum; in this way, two S-agents can interact with two different non-overlapping functionalities of the same device. Secondly, the S-agent that owns the rules by which to interact with the device can delegate this interaction to another S-agent (linked to the same private fulcrum) simply by sending it the related rules; thirdly, the system is more fault tolerant in that, e.g., if the S-agent that interacts with the device stops working, another S-agent could manage the interaction with the device.

A. CASMAS at Work

To illustrate how the CASMAS model achieves its goal of supporting collaboration, we describe a scenario and the related CASMAS mechanisms.

The PerCom University is endowed with ID emitters that allow the identification of different zones of its building and with wall-monitors that show information about ongoing initiatives. Every member always carries at least one localization device (eventually embedded in something that the person carries always with her, e.g., the wrist-watch) that is able to perceive area IDs.

Today the University hosts a workshop entitled "PCC:

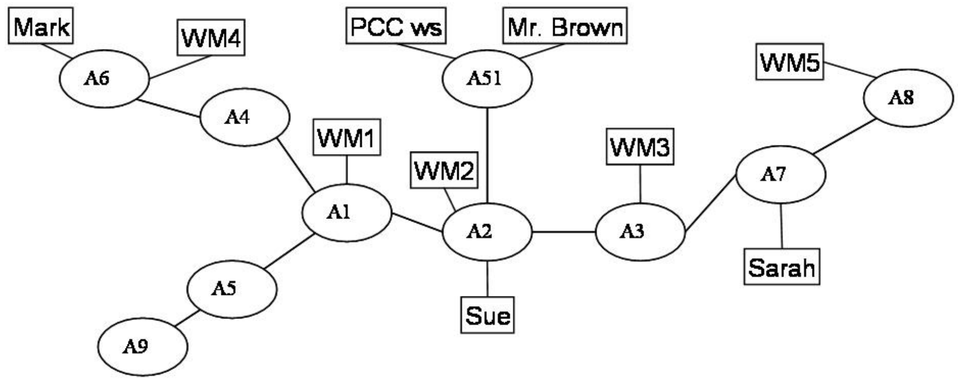

Fig. 5. Localization graph and M-agents located on it.

Pervasive Computing Challenges" that is scheduled for 10 a.m. Sue, Sarah and Mark (see configuration of the CASMAS architecture in Figure 3) are three people working at the PerCom University: Sue and Mark are interested in the PCC workshop. Currently Sue is in the corridor close the workshop room while Mark and Sarah are far from it (see Figure 4). Moreover, Mr. Brown (see configuration of the CASMAS architecture in Figure 3), the workshop speaker, is in the coffee room having a cup of good coffee.

Since Mr. Brown had previously set the commitment in his Agenda, at 9.45 a.m. his Smart Phone vibrates and shows a message reminding the scheduled event. When Mr. Brown arrives at the workshop room, his Agenda infers that the workshop is going to start and publishes this information. In order to reduce the information overload in the spirit of calm technologies [16], the information is showed on the wall-monitors (see configuration of the CASMAS architecture of a wall-monitor in Figure 3) close to the workshop room and notified only to the persons who are far from it. Therefore, Mark's personal device perceives the information "workshop PCC is starting" while Sue and Sarah's personal devices do not perceive it because of two different reasons: Sarah is not interested in the workshop, while Sue is interested but she is close the workshop room so she can see the notification on the wall-monitor.

When persons interested in the workshop approach the workshop room, they become member of the PCC workshop community (in this scenario the degree of participation to the community is limited to being or not member of the community, next we provide more information about modulated participation) and share rules and information that characterize it: for example, any "ringing device" owned by participants must be turned to silent mode. This happens to Mark and Sue when they enter in the workshop room. In addition, the PCC community states that the workshop speaker can publish his Curriculum Vitae (CV) and that members of the community can retrieve it if they like. Before the workshop begins, Mr. Brown publishes his CV to the community through his personal device; since Sue and Mark are members of the community their device either automatically retrieves the speaker's CV or asks them if they want it, according to their preferences, through the "Document Sharing" collaborative application (see Figure 3).

In order to model the illustrated scenario in CASMAS, a localization graph (see Figure 5) is needed to take into account the physical location of the different entities (people, devices, activities) and model the information modulation accordingly. The M-agent of the PCC workshop, which is an activity, emits a field on the localization graph to notify that the workshop is occurring.

Moreover, the model encompasses as many private fulcra as many human actors are using an instance of the Agenda application, and a single collaborative fulcrum that manages the workshop policies.

When Mark schedules in his Agenda that he will take part in the PCC workshop, his Agenda's S-agent asserts in his private fulcrum the fact (see Figure 6)

1) X is interested in the PCC workshop

(where X is a parameter that represents the person) so that the Agenda's Manager forwards this information to Mark's M-agent through the filtering rule

2) if X is interested in the PCC workshop then send the external field "PCC workshop fields" to the X's M-agent

The same holds for Sue.

According to the scenario, the sensitivity function of wall-monitors' M-agents let them perceive only fields about workshops happening in the areas close to them.

When Mr. Brown enters the PCC workshop room, his M-agent perceives the PCC workshop field and emits the "PCC workshop is starting" field on the graph; Figure 7 illustrates the field diffusion and the perception by M-agents which represent persons and wall-monitors in their various locations. Hence, the wall-monitors close to the workshop room show this information, because their M-agents have perceived the field. This happens also to Mark because his M-agent communicates to the Manager of his private fulcrum that the

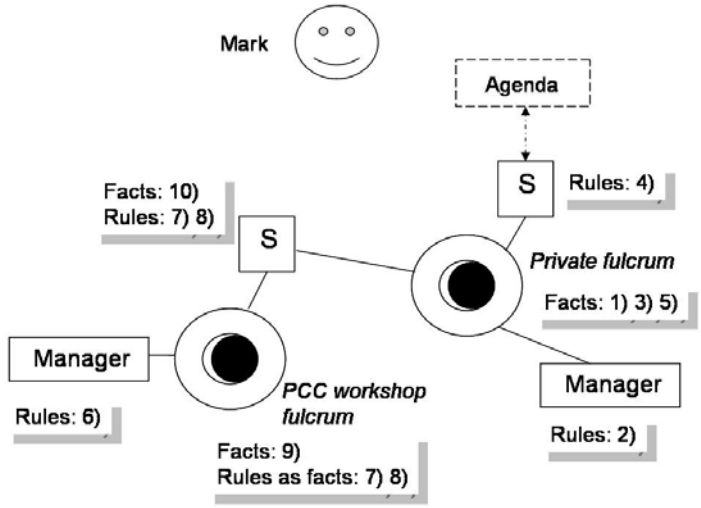

Fig. 6. Distribution of rules and facts on S-agents and fulcra.

PCC workshop is starting; the Manager declares the fact

3) the PCC workshop is starting

in the private fulcrum so the S-agent related to his Agenda can react to it, due to the rule

4) if the PCC workshop is starting and X is interested in the PCC workshop then send the message "the PCC workshop is starting" to the X's Agenda

and the Agenda notifies Mark accordingly. Instead, Sue's personal device does not inform her although her M-agent is sensible to the "workshop is starting" field, because she is near to the workshop room and the field intensity is lower than the perception threshold.

Becoming aware that the PCC workshop begins, Sue and Mark move to the workshop room. Since their M-agents perceive the workshop field with the highest intensity, each of them sends this information to the Manager of the private fulcrum; consequently, it infers that the person is participating to the workshop and asserts the fact

5) X is member of the PCC workshop community

This fact is transferred from the private fulcrum to the workshop fulcrum, through rules (provided by Santana) that allow exchanging facts between fulcra, and triggers the Manager of the PCC workshop fulcrum to add an S-agent for the new member to the workshop fulcrum, as stated by the community rule

6) if X is member of the PCC workshop community then create an S-agent for X in the PCC workshop fulcrum

As members of the workshop community, Sue and Mark's proxies are endowed with the community rules

7) if X is member of the PCC workshop community then quiet all X's "ringing devices"

8) if the speaker's CV is available and X is interested in it then retrieve it

so their "ringing devices" automatically switch to silent mode; in addition, their S-agent acquire inferential rules to perceive and retrieve the CV of the speaker. When Mr. Brown publishes his CV, i.e. his S-agent asserts the fact

9) the speaker's CV is available

in the PCC workshop fulcrum, rule 8) fires on the device of the members of the fulcrum; hence, Mr. Brown's CV is retrieved by and presented on Sue and Mark's device if they have declared an interest in it, through the fact

10) X is interested in the speaker's CV

This scenario illustrates some central aspects of our approach to support collaboration. First, the environment is proactive, i.e., it is able to sense the location of the actors, make them aware of events and activate services accordingly. Secondly, to this aim the environment manages the (interaction with existing) single-user and cooperative applications as well as a mechanism to compute different degrees of participation in them. To better describe modulated participation –a first-class concept of CASMAS- let us come back to the previous scenario, which illustrates a basic use of the field diffusion mechanism, namely the joining of members to the workshop community when they enter the room. A more sophisticated use of field diffusion would use modulation to realize a more articulated notion of participation. In fact, the different values of a diffused field can trigger the activation of different behaviors of S-agents in

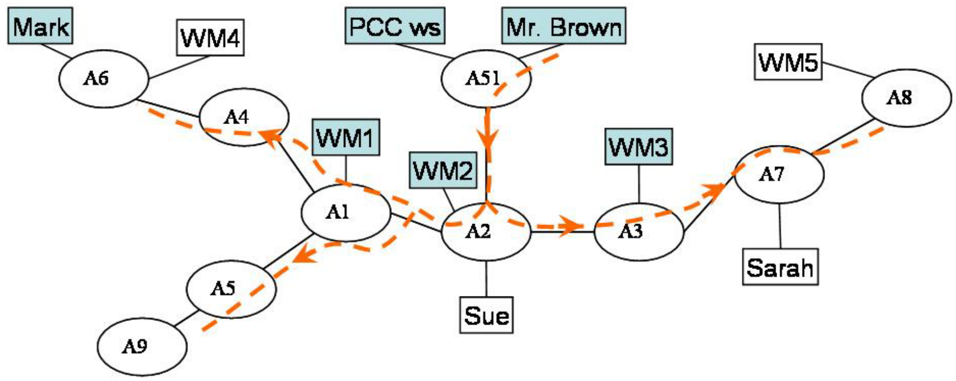

Fig. 7. Diffusion of the "PCC workshop is starting" field (hatched line) on the graph and perception by M-agents (colored agents perceive the field).

the receiving fulcra. This is realized through a mechanism that involves the M-agent linked to the graph where the field is diffused and the Manager. Manager owns rules to evaluate the degree of participation to the community based on the field perceived and exported by the M-agent; once they are executed, these rules assert into the private fulcrum facts that represent the degree of participation of the S-agent linked to the related community (e.g., rules sensible to the PCC workshop fields assert facts for the S-agent linked to the PCC workshop community). These facts are checked in the if-side of the community rules (already loaded in the corresponding S-agent); when the intensity of the field changes, new facts expressing the degree of participation are asserted; consequently different community rules can fire within the S-agent, and hence make it to participate to the community in a different way. In the scenario, the workshop field can be perceived with low intensity by agents located further away in the topological graph since the corresponding people are late and approaching the workshop room. In this case, they could be considered members of the community but with a more peripheral degree of participation: for example, they could listen on their Smart Phone to the voice of the speaker but with a limited access to the presented material or the speaker's CV; in this case, the Manager asserts the fact that represents a peripheral participation of the S-agent to the community, so that the rule that retrieves the CV is prevented from firing. The CV however will become available to them when they enter the room, because the perceived workshop field would get the highest intensity; consequently the fact that represents a full participation to the community is asserted, and then the rule to retrieve the CV can fire on it.

This example of modulated participation uses again the physical distance as a parameter. One could conceive situations in which the topology expresses logical distance between entities and modulates participation according to it. For example, suppose that the collaborative environment contains different fulcra that support a community in combination with different cooperative applications (e.g., workflow, co-authoring, shared repositories, distributed systems); moreover, S-agents that access those fulcra contain rules that capture the interactions occurring in each of them between any two participating actors. This kind of information can be transmitted and organized in a topological space (usually called *social network* [17]) where the distance between the M-agents corresponding to any pair of actors expresses the degree of interaction among them. Moreover, M-agents own a sensitivity function expressing their availability to help an actor's request to solve an unexpected problem: for example, availability can be computed in terms of work overload or single actor's preferences, by means of the same rule-based mechanism described in the workshop scenario.

The environment, which could include several fulcra, a logical space associated to many of them as well as a physical one, can be modeled and managed by applying the integrated approach of CASMAS: the approach has the obvious advantage to manage uniformly the physical and logical features characterizing the environment and to support actors in their private and collaborative interaction by means of mutual perception and modulated participation to the applications that are available within the environment.

IV. Conclusion

This paper presented CASMAS, an agent based model for the design of collaborative community-aware applications. With community-aware we intend cooperative applications that - by means of the CASMAS constructs - are augmented with mechanisms managing different levels of participation of actors as members of communities. In order to let cooperative application become community-aware, the CASMAS model combines and integrates two models that were previously proposed within the multi-agent system research: Santana and MMASS. The former has been adopted for its ability to support the design of applications for Intelligent Environments, i.e., environments encompassing distributed and heterogeneous devices whose computational power can be combined together in order to build a context-aware environment that is able to react more aptly to the users' needs. CASMAS can be seen as an extension of Santana aimed at supporting the design of cooperative applications that could be used also in those domains whose requirements are characterized within the Ambient Intelligence and Ubiquitous Computing research fields. The latter one is a multi-agent model that has been adopted for its ability to conceive of agents as entitities situated on topological spaces representing both logical and physical aspects of a domain. The propagation of information among agents is modulated by means of the concept of field propagation, which is borrowed from Physics.

In the CASMAS model, primitives provided by Santana are used to let the various collaborative applications share those information and behaviors that concern and characterize the community of their users; MMASS is used to model how the level of participation of different community members can be modulated: this modulation occurs taking into account how the domain dependent information, which is relevant to affect the level of participation, is modulated through topological spaces representing either logical or physical aspects. As a result of the integration between Santana and MMASS, CASMAS provides designers with some additional mechanisms to let information be exchanged among Santana components and MMASS agents. Moreover, our proposal aims at making possible a seamless sharing of the rules regulating the community members' behaviors according to their current level of participation.

Currently, we are involved with the implementation of the Santana framework and of the MMASS model by means of the DJess platform [18], a middleware based on declarative programming by which distributed inference systems can share facts and rules through a blackboard interaction model. We are also investigating how to integrate the CASMAS model with other agent-based models: in particular with the ABACo Multi-Agent Framework [7] so that CASMAS fulcra can become places where people are strongly supported in coordinating their activities.

REFERENCES

[1] M. Wooldridge, "Agent-based Computing," *Interoperable Communication Networks*, vol. 1, pp. 71--97, 1998.

[2] P. Ciancarini and M. J. Wooldridge, *Agent-Oriented Software Engineering*, vol. 1957 LNCS: Springer-Verlag, 2001.

[3] R. A. Brooks, "Intelligence without representation," *Artificial Intelligence*, vol. 47, pp. 139-159, 1991.

[4] Y. Yiming and E. Churchill, "Agent Supported Cooperative Work," in *Multiagent systems, articial societies, and simulated organizations*, G. Weiss, Ed.: Kluwer Academic Publishers, 2003.

[5] S. Aknine and S. Pinson, "Managing Distributed Parallel Workflow Systems Using a Multi-agent Method," in *Agent Supported Cooperative Work*, Y. Yiming and E. Churchill, Eds.: Kluwer Academic Publishers, 2003.

[6] C. Ellis, J. Wainer, and P. Barthelmess, "Agent-augmented Meetings," in *Agent Supported Cooperative Work*, Y. Yiming and E. Churchill, Eds.: Kluwer Academic Publishers, 2003.

[7] M. Divitini, M. Sarini, and C. Simone, "Reactive Agents for a systemic approach to the construction of Coordination Mechanisms," in *Agents supported Cooperative work*, E. Churchill and Y. Yiming, Eds.: Kluwer Academic Press, 2003.

[8] J. Lave and E. Wenger, *Situated Learning: Legitimate Peripheral Participation*: Cambridge University Press, 1991.

[9] J. H. E. Andriessen, "Archetypes of Knowledge Communities," Second Communities & Technologies Conference (C&T2005), Milan, Italy, 2005.

[10] F. Cabitza, B. Dal Seno, M. Sarini, and C. Simone, "Being at One with Things: The Interconnection Metaphor for Intelligent Environments," The IEE International Workshop on Intelligent Environments (IE05), University of Essex, Colchester, UK, 2005.

[11] S. Bandini, S. Manzoni, and C. Simone, "Heterogeneous Agents Situated in Heterogeneous Spaces," *Applied Artificial Intelligence*, vol. 16, pp. 831-852, 2002.

[12] C. Simone and S. Bandini, "Integrating Awareness in Cooperative Applications through the Reaction-Diffusion Metaphor," *Computer Supported Cooperative Work, The Journal of Collaborative Computing*, vol. 11, pp. 495-530, 2002.

[13] M. Mamei, F. Zambonelli, and L. Leonardi, "Distributed Motion Coordination with Co-Fields: A Case Study in Urban Traffic Management," 6th IEEE Symposium on Autonomous Decentralized Systems (ISADS 2003), Pisa(I), 2003.

[14] M. Mamei, F. Zambonelli, and L. Leonardi, "Tuples On The Air: A middleware for context-aware computing in dynamic networks," 23rd International Conference on Distributed Computing Systems (ICDCSW '03), 2003.

[15] G. D. Abowd and A. K. Dey, "Towards a Better Understanding of Context and Context-Awareness," Workshop on The What, Who, Where, When, and How of Context-Awareness - Conference on Human Factors in Computing Systems (CHI 2000), The Hague, The Netherlands, 2000.

[16] M. Weiser and J. S. Brown, "Designing calm technology," *PowerGrid Journal*, vol. 1, 1996.

[17] B. Wellman and S. D. Berkowitz (eds.), "Social structures: A network approach." Cambridge: Cambridge University Press, 1988.

[18] F. Cabitza, M. Sarini, and B. Dal Seno, "DJess - A Context-Sharing Middleware to Deploy Distributed Inference Systems in Pervasive Computation Domains," IEEE International Conference on Pervasive Services 2005 (ICPS'05), Santorini, Greece, 2005.