

# Role Suggestion for Agents by Overhearing

Giacomo CABRI, Luca FERRARI, Letizia LEONARDI and Raffaele QUITADAMO

*Abstract-* Software agents represent an interesting paradigm to develop intelligent and distributed systems, because of their autonomy, proactiveness and reactivity; in addition, their sociality enables the distribution of the application logic in different agents that can interact together and with the host environment. In such scenario interactions must be carefully designed and managed at run-time. The concept of role has been adopted in different (agent) approaches to flexibly manage interactions; roles represent stereotypes of behavior, which are useful not only in the modeling of systems and applications, but also in their design and implementation. Overhearing is a technique that enables the observation of agents' behavior by "hearing" the exchanged messages. In general, overhearing can be useful to "label" observed agents and manage agent organizations. In this paper we explore the adoption of overhearing in conjunction with agent roles in order to provide more features to agents themselves. In particular the approach presented here can support and help the agent deciding the role to assume and then how to use it, after having observed the agent behavior.

*Index Terms*— agent interactions, roles, overhearing

## 1. INTRODUCTION

Distributed intelligent systems can rely on software agents, which are autonomous software entities with interesting features. First, they are autonomous and adaptable problem-solving entities, able to execute in open and dynamic environments and to carry out their task(s) without requiring a continue user involvement [17]. In fact, because of their autonomy, agents can play on behalf of their owner, resulting as a digital counterpart of the latter in a digital world. Moreover, due to their adaptability, agents can face dynamic and exceptional situations, representing a robust and reliable approach to build complex software systems [24]. Another important characteristic of agents is mobility, which enables agents to move themselves across the hosts of a network; these agents are called *mobile agents*. Mobile agents are suitable to play on behalf of their user since, as the user do in the real world, they can move in the digital world searching for and reaching required data, visit other sites, meet other agents and exchange information with them.

Applications based on agents often involve more than one agent, even mobile, in order to divide complex tasks into smaller ones; such kind of applications are called

MAS – Multi Agent Systems. In MAS applications, interaction between agents must be carefully modeled, since they are fundamental for the whole system, in terms of both "distribution" and "intelligence". In fact, the distribution of the applications can occur by spreading agents, which can carry out the applications' task(s) by interacting each other; these interactions can exhibit a given degree of intelligence also with the support of the underlying environment.

So far, several approaches have been proposed to manage and face agent interactions and organizations, including Tuple-Spaces [11], Group Computation [15], Activity Theory [20], Roles [12] and Overhearing [3, 4, 16]. Our work takes into consideration the last two (Roles and Overhearing), and this paper proposes a few considerations to merge the two approaches in order to enrich the Role-based one. The Role theory proposes to manage agent collaborations and interactions through the use of stereotypes of common behaviors, resulting in a powerful and easy to understand way to model interactions. The Overhearing theory proposes to manage agent collaborations through a multicast listening to all exchanged messages. Both the former theories propose to manage interactions between agents, but providing different approaches that can be combined to enhance and better customize the interaction management.

The key idea of this paper is that it is possible to overhear an agent (i.e., to observe its behavior), in order to suggest that it assumes a role and uses it to carry out its tasks. To do this, it is not enough to apply the overhearing theory as it is, to a role-based agent context: overhearing must be adapted in order to achieve a smart monitoring system, able to observe the behavior of either single agents or group of agents.

The paper is organized as follows: Section 2 glances at background of Roles and Overhearing theories and motivates a combined approach; Section 3 presents a few drawbacks of pure overhearing, while Section 4 explains how our approach works; Section 5 provides a few application examples; Section 6 proposes some implementation directions, and finally section 6 concludes the paper.

## 2. BACKGROUND & MOTIVATIONS

This section introduces both the theory of Roles and Overhearing, giving a brief background of both, and showing how enhancing roles with overhearing can result in a support system for agent decisions and proactiveness.

### 2.1 Roles

The Role theory [2] has been applied to several computer science fields, and in fact there are several

Manuscript received October 16, 2006; revised January 19, 2007  
This work was supported by the EU under the CASCADAS project and by the Italian MUR under the "Agent oriented methodologies: engineering of interactions and relationship with the infrastructures" project. This paper is extended from "Collaboration-Driven Role Suggestion for Agents" published at *IEEE Workshop on Distributed Intelligent Systems Collective Intelligence and its Applications*, Prague, CZ, June 2006.

G. Cabri, L. Ferrari, L. Leonardi and R. Quitadamo are with Dept. of Information Engineering, University of Modena and Reggio Emilia, via Vignolese, 905, 41100 Modena, Italy (e-mail: giacomo.cabri@unimore.it, luca.ferrari@unimore.it, letizia.leonardi@unimore.it, and raffaele.quitadamo@unimore.it).

definitions of the concept of *role*, depending on the considered scenario. In particular, it has been noted that, during the Object-Oriented design phase, there are classes that are not really classes, but roles [22]. Roles represent a cross-cut view of the object space, and thus can be adopted to model dynamic and open environments. Roles are typically related to associations among software entities, as emphasized in the UML language, where roles are applied to associations between objects [23].

The interesting feature of roles is that they can be used as a paradigm to smartly model the view of a complex system [13]. This is of primary importance when roles are applied to the agent world; in fact, recalling that agents can be thought as human counterparts, roles represent a conceptual tool to model the digital world in a way similar to the real one, where digital human counterpart can perform their task(s) in ways similar to those adopted by humans in the real world organizations and collaboration [25].

More in detail, roles can be applied to agents in order to both enhance their capabilities, granting a better adaptability, and to model interactions and coordination in MAS systems [12]. For example, consider an agent in charge of writing a few records into a database. By placing the code in charge of interacting with the database into a specific role, let it be the *writer*, the agent playing such role is free to discard any detail regarding the database system and its interface. In this way, just by assuming and playing the writer role, the agent (see Figure 1) can perform the record insertion without knowing any detail about the underlying database system. The use of the writer role enhances the agent capabilities, since, without that role, it is unable to insert the records in the database, and, at the same time, the writer role grants to the agent a stronger adaptability. In fact, if the database changes, it suffices to change only the role, without applying any change to the agents that will play it.

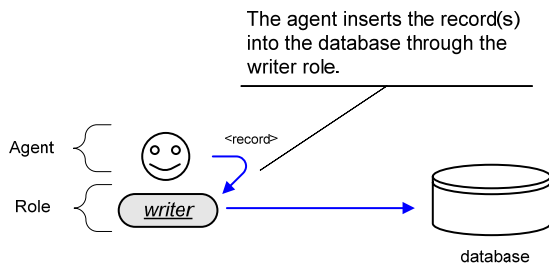


Figure 1. An agent that exploits a role to achieve its task.

As shown in Figure 2, roles can be also used to manage interactions between agents. In fact, roles can be developed with regard to each other, thus the *bidder* and *seller* roles of Figure 2 can implement a common protocol in order to let the related agents playing them make an offer and sell a good.

It is important to note that roles are tied to the local execution environment, thus they represent context-

dependent views of entities running in that environment [1], granting adaptability. With regard to the above example, this means that the bidder and seller roles belong to the local interaction context the agents are running in, and then if the agents move to another interaction context (e.g., another host), local roles can have a different implementation. It is for this reason that we claim that roles grant portability and generality: since they are tied to each interaction context, they hide context details to agents, which are free to discard those “low-level” details.

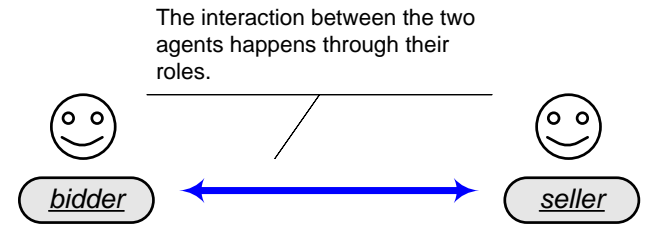


Figure 2. An interaction through the use of roles.

In order to play a role, an agent must *assume* it. In other words, an agent must choose a specific role, that means that the role assumption is considered an *active* process; but there exist some approaches where the role assumption is a *passive* approach, such as *overhearing*.

## 2.2 Overhearing

Overhearing [16] theory can handle group of collaborating agents that auto-coordinate themselves in order to reach a common aim. The key idea is that it is possible to classify, assigning a specific role, each agent simply listening to it, that is observing its message exchange with other agents [4]. Capturing and understanding the messages that agents are exchanging, it is possible to infer which role the one is playing with regard to the other; thus it is possible, for example, to understand which agent is playing the role of seller and which is playing the role of bidder in the example of Figure 2.

Typically, overhearing is used to manage groups of agents that agree on a common aim. These groups, often called *implicit organizations* [5], are made by agents that, thanks to the overhearing process, understand that they have a common task or aim; therefore, they aggregate to perform such task and to synchronize. In each group, a leader agent (called *oracle*), emerges and drives the group to achieve the final aim.

The main difference between the role concepts used in both the above theories is the following: while, according to the role theory, a role is an entity that must be actively assumed, in the overhearing theory it is a passive classifier, used only to recognize the agent behavior, instead of granting a specific behavior, as in the role theory. In other words, while, in the overhearing approach, roles are mainly used to passively get a bird’s eye view of the agent scenario, in the role theory they are actively used by agents to enhance their capabilities.









