THE CASCOM ABSTRACT ARCHITECTURE FOR SEMANTIC SERVICE DISCOVERY AND COORDINATION IN IP2P ENVIRONMENTS

Cesar Caceres, Alberto Fernandez, Sascha Ossowski, Matteo Vasirani
Artificial Intelligence Group, University Rey Juan Carlos, Madrid, Spain

Keywords: Service coordination, intelligent agents, peer-to-peer computing, mobile computing.

Abstract: Intelligent agent-based peer-to-peer (IP2P) environments provide a means for pervasively providing and flexibly co-ordinating ubiquitous business application services to the mobile users and workers in the dynamically changing contexts of open, large-scale, and pervasive settings. In this paper, we present an abstract architecture for service delivery and coordination in IP2P environments that has been developed within the CASCOM project. Furthermore, we outline the potential benefits of a role-based interaction modelling approach for a concrete application of this abstract architecture based on a real-world scenario for emergency assistance in the healthcare domain.

1 INTRODUCTION

The ever-growing number of services on the WWW provides enormous business opportunities. In particular, there is a huge potential for creating added value through service coordination. For this to happen, technology must be developed capable of pervasively providing and flexibly co-ordinating ubiquitous business application services to mobile users and workers in the dynamically changing contexts of open, large-scale and pervasive application domains.

One step toward the realization of this vision is the development of an intelligent agent-based peer-to-peer (IP2P) environment. IP2P environments are extensions to conventional P2P architectures with components for mobile and ad hoc computing, wireless communications, and a broad range of pervasive devices. A major challenge in IP2P environments is to guarantee a secure spread of (personal) service requests across multiple transmission infrastructures and ensure the trustworthiness of services that may involve a broad variety of providers.

In this paper, we present an abstract architecture for service delivery and coordination in IP2P environments that has been developed within the CASCOM project (CASCOM 2005) and discuss the benefits of role based interaction modelling for its instantiation to a particular application domain.

This architecture aims at providing support for business services for mobile workers and users across mobile and fixed networks. For *end users*, the architecture provides easy and seamless access to semantic Web services anytime, anywhere, and using any device. For *network operators*, it aims towards a vision of seamless service experience providing better customer satisfaction, which in turn helps to retain current customer relations as well as attract new customers. To *service providers* the architecture offers an innovative platform for various mobile business application services.

2 CASCOM ABSTRACT ARCHITECTURE

The CASCOM abstract architecture is based on a layered approach that combines innovative network aspects, such as peer-to-peer and mobile networks, with modern software technologies, like semantic web services and intelligent agents (see Figure 1). The four main components of this architecture act as a mediator between the applications and the underlying networks.

Caceres C., Fernandez A., Ossowski S. and Vasirani M. (2006).
THE CASCOM ABSTRACT ARCHITECTURE FOR SEMANTIC SERVICE DISCOVERY AND COORDINATION IN IP2P ENVIRONMENTS.
In *Proceedings of the Eighth International Conference on Enterprise Information Systems - SAIC*, pages 205-208
DOI: 10.5220/0002459402050208

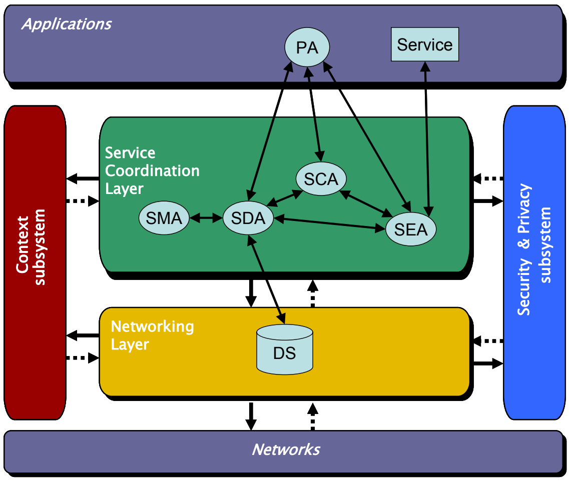

Figure 1: CASCOM abstract architecture and its elements.

The Networking Layer is located just over the networks. Its main functionalities are enabling an efficient and reliable agent message transport communication over wireless (and wireline) communication paths, providing network-related context information (e.g. QoS of a connection, network availability, etc.), to the context subsystem, and providing an agent execution environment for resource-constrained mobile devices. Requesters search dynamically for services published by different providers in a directory service (DS), which can be centralized or distributed. In the latter case, a federation of DSs could be built, where each DS registers itself in other DSs as a service. Thus, a DS can be found by querying entries in a DS. Such an approach was used in the Agentcities project (Constantinescu, 2003), where the directories were federated accordingly to "application domains".

The Service Coordination Layer is located between the Networking Layer and the Application Layer, and uses the services offered by both the Context and the Security & Privacy Subsystems. This layer has two main functionalities, the semantic service discovery (service discovery + semantic matchmaking) and the service coordination (service composition + execution monitoring and replanning). In the CASCOM abstract architecture, the semantic service discovery functionality is realised by two different types of agents: *Service Discovery Agents* (SDA) and *Service Matchmaking Agents* (SMA). In much the same way, the service coordination functionality is realised by *Service Composition Agents* (SCA) and *Service Execution Agents* (SEA). Usually, SDAs receive service specifications from the users' *Personal Agents* (PA) and acquire relevant contextual information from the context subsystem. With that information, they use the service discovery functionality of the networking layer and the semantic matching functionality of SMAs, to determine services that fulfil the received service discovery request. SDAs return then a set of descriptions/providers and their correspondent service process model and/or grounding.

SCAs are capable of creating value-added composite services that match specific service specifications. Once SCAs receive service specifications, they contact SDAs to discover existing services in a given domain constrained to the current context, and plan a composite value-added service matching the received service specification.

SEAs manage the execution of composite value-added service descriptions generated by SCAs. Since the received compound service description relies on simpler services, the execution will also coordinate the execution of these simpler services. Whenever necessary, SEAs will use SDAs to discover appropriate available service providers for each of the simpler services invoked from the compound service description.

The Context Subsystem (Lopes, 2005) is accessed by both the Network Layer and the Service Coordination Layer. Its main functionality is to discover, acquire and store useful context information (e.g. the geographical position or the user preferences). This kind of information can be accessed by both layers either by explicitly querying the context subsystem ("pulling") or by subscribing a listener that is in charge of notifying changes and events occurring in the environment ("pushing").

The Security and Privacy Subsystem is also orthogonal to the first two layers. It is responsible for ensuring security and privacy of information throughout the different components of the CASCOM infrastructure. Every node of the network should keep its data confidentiality, integrity, and availability (CIA) -- see SecPedia website. But not only data is a security concern, also the software that deals with it, especially for network-centric systems, where the misuse, theft, and unauthorized usage of computing resources is well studied. The security and privacy requirements identified are: identification, authentication, authorisation, single sign-on, and local and network security. We also need to guarantee the integrity of transmitted data, non-repudiability, traceability, privacy, delegation, and nationalization.

3 ROLE-BASED INTERACTION MODELLING

In order to improve both the efficiency and the usability of the CASCOM architecture, we claim that it is necessary to account for the *types* of interaction that certain semantic services can be used in. In this section we outline how we will apply and extend our role-based and interaction-centric modelling framework (Serrano et al. 2002) within the CASCOM project.

The abstract architecture conceives services to be delivered essentially by agents. In such an approach the agents usually act as mere wrappers for web services. The difference between a web service and a service provided by an agent boils down to a matter of interface: an agent can provide an implemented web service by a process of wrapping the service within an ACL interface in such a way that any agent can invoke its execution by sending the adequate (cg. request) message.

However, agents are not only able to execute the service but also they can engage in different communicative interactions around that service. That means the provider must be able to engage in several different interactions during the provision of a service

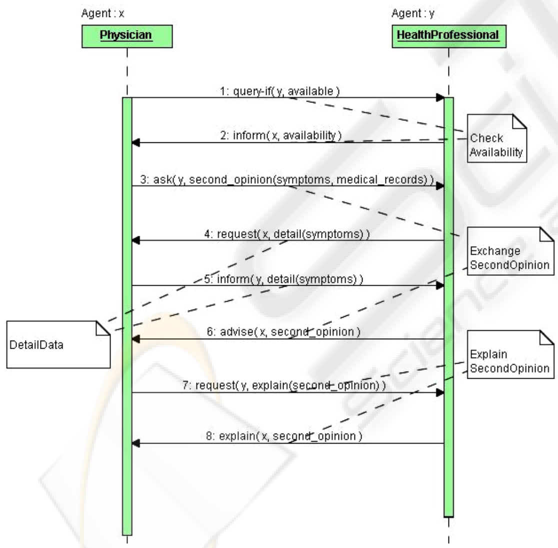

Figure 2: Second Opinion conversation.

To better explain our role-based approach, let's analyze a use case story of a healthcare scenario. In this use case a physician of a local emergency centre (or the patient) can ask a health professional for a diagnosis on the basis of the symptoms and the medical records of the patient, like exams and past diseases.

The "conversation" between the physician of the local emergency centre and the health professional can be modelled with a sequence of speech acts between the two agents involved, as depicted in Figure 2. The physician *requests* to the health professional the status of his availability, that is if he can provide a second opinion over a certain matter. If so, he *asks* him for an opinion, providing the symptoms and the medical records. If there is no sufficient information, the health professional *requests* (possibly more times) additional information and finally gives his *advisement*. If the provided diagnosis is doubtful or not clear, the physician can also solicit an *explanation*.

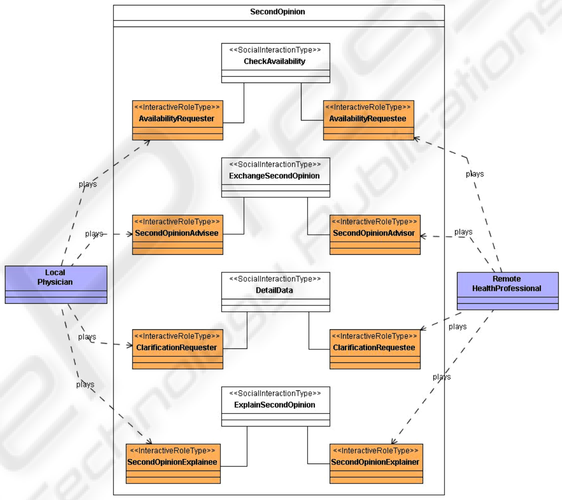

Figure 3: Second Opinion interactions.

Starting from this conversation we can isolate four different interactions: (1) the first is checking the availability of the health professional, then (2) the real second opinion exchange starts. It can comprise (3) a detailed information exchange. When the second opinion exchange finishes, an explanation (4) could occur.

Having individuated the basic interactions, now we can model the second opinion use case in a more precise way, like depicted in Figure 3, and furthermore define the roles involved in the

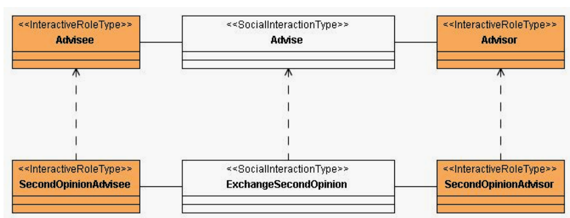

Figure 4: Roles and interactions taxonomy.

interactions. The product of this analysis should be a basic ontology of roles (Figure 4), which is possible to refine and make more general. For example, the role *SecondOpinionAdvisor* could be generalized in an *Advisor* role, as well the *ExchangeSecondOpinion* interaction could be generalized in an *Advisement* interaction, in which the Advisor informs the Advisee about his beliefs with the aim of persuading the Advisee of the goodness of these beliefs.

Role and organizational concepts are abstraction mechanisms that have been used for many years in object oriented methodologies and are usually present in most agent-oriented methodologies like GAIA (Wooldridge, 2000), MASE (DeLoach, 2001), etc. However, at least at the moment, there are no approaches that use them to describe what the services do. Furthermore, most current OWL-S matchmakers (Paolucci, 2002) (Li, 2003) only consider service inputs and outputs in their matching algorithm. We are exploring means of taking into account the roles that agents can play and the interactions in which they can engage in the publication of services in the service directory in the CASCOM project. Roles and interactions will be defined in an ontology that is being constructed.

The efficiency of the matchmaking process can be improved by previously filtering those services that are compatible in the terms of roles and interactions they can participate in. The precision of the matchmaking process can also be improved by including information regarding the roles and interactions. For instance, a service may have the symptoms and medical records as inputs and the report as output. However, the service can consist of (i) actually generating the report, (ii) retrieving a previously done or (iii) a brokering service to contact other (external) healthcare experts. The inputs and outputs are the same but the role the service plays is different (advisor, informer or broker, respectively). The idea in our approach is that, in the previous second opinion service example, we only make a detailed search for service provider agents that are able to play the *advisor* role.

4 CONCLUSIONS

In this paper, we have described the CASCOM abstract architecture that smoothly integrates intelligent agent technology, semantic Web services, peer-to-peer, and mobile computing, for intelligent peer-to-peer mobile service environments. The potential benefits of a role-based interaction modelling approach have been illustrated based on a real-world use case scenario for emergency assistance in the healthcare domain.

The adequacy of the above CASCOM architecture has been analysed for three different scenarios in the healthcare, patient telemonitoring & e-inclusion and shopping mall assistance domain.

We are currently initiating the implementation of the presented architecture. In particular, the interest of the authors will focus on the instrumentation of role-based mechanisms for service discovery and coordination. At the end of the project, we will come up with a fully fledge demonstrator for a concrete real-world business application scenario.

ACKNOWLEDGEMENTS

This work has been supported in part by the European Commission under the project grant FP6-IST-511632-CASCOM. The authors would like to thanks all project partners for their contributions.

REFERENCES

Cáceres, C., Fernández, A., Ossowski, S. CASCOM – Context-aware Health-Care Service Coordination in Mobile Computing Environments. *ERCIM News 60*, pp. 77-78

CASCOM, 2005. http://www.ist-cascom.org

DeLoach, S.A., Wood, M. F., and Sparkman, C. H., (2001), "Multiagent systems engineering". *International Journal of Software Engineering and Knowledge Engineering, 11(3)*:231-258

Li, L., and Horrock, I. (2003). "A software framework for matchmaking based on semantic web technology". In *Proc. 12th Int World Wide Web Conference Workshop on E-Services and the Semantic Web (ESSW 2003)*.

Lopes, A., and Botelho, L. (2005) "SEA: a Semantic Web Services Contextaware Execution Agent". *Proceedings 1st International AAAI Fall Symposium on Agents and the Semantic Web*, Arlington VA, USA

Paolucci, M., Kawamura, T., Payne, T., and Sycara, K., (2002). "Semantic matching of web services capabilities". *In Proceedings of the First International Semantic Web Conference on The Semantic Web*, pages 333-347. Springer-Verlag

Serrano, J.M.; Ossowski, S.; Fernández, A. (2002), "The Pragmatics of Software Agents – Analysis and Design of Agent Communication Languages", Intelligent Information Agents – The European AgentLink (Klusch et al. ed.), Springer, pp. 234–274

Wooldridge, M., Jennings, N. R., and Kinny, D. (2000), "The Gaia Methodology for Agent-Oriented Analysis and Design". *Journal of Autonomous Agents and Multi-Agent Systems* 3(3) 285-312.