# A Multi-Agent System for Dynamic Service Composition in Ambient Intelligence Environments

Mathieu Vallée[*] ,
Fano Ramparany[†] and Laurent Vercouter[‡]

**Abstract.** *Pervasive computing environments involve a variety of smart devices, which tend to overcharge humans with complex or irrelevant interaction. Ambient Intelligence pushes forward a vision where technology is integrated into everyday objects with the intent of making users' interaction with their surrounding environment simpler and more intuitive. In this paper, we expose how Ambient Intelligence can benefit from the coupling of a service-oriented approach and multi-agent systems towards more appropriate interactions with users. Our approach combines multi-agent techniques with semantic web services to enable dynamic, context-aware service composition, thus providing users with relevant high level services depending on their current context and activity.*

## 1. Introduction

Current progress in computer systems heads towards the pervasive computing era, when nearly every object in our environment will host computing and communication capabilities. However, the number and variety of smart devices raises concerns about the amount of time and attention users must devote to interact with them, and strongly calls for new developments aiming at "relieving the interaction cognitive load" [12]. Ambient Intelligence (AmI) turns a networked system of smart devices and sensors into an environment acting as a global interface between users and information systems.

One key requirement for AmI environments is automatic adaption of the behavior of systems to users' activity and context. Specifically, AmI seeks to favor communication among objects and to enable objects to sense with the environment, thus reducing configuration effort and irrelevant user solicitation. A key issue in designing such adaptable systems is the abstraction gap between the raw functionalities offered by a pervasive system and the higher level user needs. We believe that bridging this gap calls for reconfigurable systems that makes users' interaction easier by building integrated applications from appropriate resources dynamically [15]. Our starting PhD work contributes to this vision by investigating novel approaches towards adaptable and reconfigurable systems. In this paper, we first present some background regarding dynamic adaptation to context, use of a service-oriented approach and service composition. We then highlight how our work plans to address issues in the design of adaptable and reconfigurable systems for AmI. We finish with an overview of our working environment and expected contributions.

[*]PhD student, TECH/ONE, France Telecom R&D, Grenoble, France, email: mathieu.vallee@rd.francetelecom.com
[†]PhD advisor, TECH/ONE, France Telecom R&D, Grenoble, France, email: fano.ramparany@rd.francetelecom.com
[‡]PhD advisor, G2I/SMA, École Nationale Supérieure des Mines, St-Étienne, France, email: Laurent.Vercouter@emse.fr

### 1.1. System Adaptation Through Context-Awareness

When designing systems that adapt to users' needs and activity, taking the interaction context into account is of paramount importance [5]. The use of context as a mean of adapting the behavior of a system to better interact with users is widely investigated. However, existing works [13] [17] usually demonstrate adaptation in closed systems, where context adaptation can be handled through hard coded context rules. We now wish to expand context-awareness to open, dynamic systems, where not only context but also the system's functionalities continuously evolve. Such a shift is necessary for real large-scale AmI environments, and calls for further investigation on systems in which applications react to user activities and context as well as system dynamicity by adapting and reconfiguring themselves.

### 1.2. Service-Oriented Architecture for AmI

This need for adaptable and reconfigurable applications strongly calls for a modular design approach, which the Service-Oriented Architecture (SOA) paradigm tends to provide. In this approach, mostly illustrated by the Web service architecture, applications are built using independent, loosely-coupled pieces of software (called services) that achieve a specific, coarse-grained functionality. Each service is developed independently, and can be invoked by other services or clients using a declarative description of access points and accepted messages.

Using a service-oriented approach has been proposed for pervasive computing and ambient intelligence environments ([11], [7]). In those fields, one trend covers works which aim at delivering services (especially Web services) to mobile devices in a pervasive computing situation ([9], [8]). We focus more on using SOA as a design approach for the AmI environment itself. In this vision, each device provides services through which external clients can obtain information or control its behavior. Beyond the achieved modularity, one of the expected benefits of this approach is the ability to build systems from heterogeneous devices and services in open environments. Specific designs based on standardized programmatic interfaces (which are efficient in well defined systems [14]), will not be suitable in those cases. Ongoing work is investigating interoperability issues in such heterogeneous environments involving services from various sources.

### 1.3. The Role of Service Composition

In a service-oriented approach, we can distinguish two abstraction levels in proposed services. On the one hand, basic services are elementary functionalities, usually provided by devices in AmI environments. On the other hand, composite services aggregate a set of functionalities into higher level applications, closer to the users' actual needs. Thus, service composition enables to fill the abstraction gap between user activity and elementary service level.

Service composition has widely been addressed in the Web service field. Existing composition frameworks [1] enable expressing and enacting complex Web service compositions. However, they rely on explicitly named services, which are not discovered dynamically. On the contrary, the Semantic Web Services (SWS) approach [2] is a step toward dynamic service discovery and composition [16] [18], where intelligent systems try to build composite services from abstract user requirements without manual selection of services. SWS leverage knowledge representation techniques, with ontologies describing a domain in a formal manner, and AI planning methods to make composition systems more autonomous.

Since automated service composition is required in AmI environment where available services are not known in advance, SWS composition seems a promising approach. However, the major drawback of Web Service composition is a lack for dynamicity. As it is designed to work with a central control point, which plans composition as a workflow and subsequently enacts services following this workflow, it can hardly adapt to dynamic and decentralized environment such as envisioned for AmI.

## 2. Using a Multi-Agent System as a Dynamic Service Composition Infrastructure

As an answer to service composition issues, our proposal in this PhD work is to investigate a Multi-Agent System (MAS) as a complement to a service-oriented approach towards adaptable and reconfigurable AmI environments, in which users interactions will become simple and intuitive.

In the following, we detail salient features of MAS with regard to AmI requirements, and we elaborate on a global overview of our proposal.

### 2.1. Multi-Agent System features for Ambient Intelligence

A Multi-Agent System is a distributed system composed of autonomous entities, called agents. These agents need to interact and cooperate in order to achieve global tasks. One of the main properties of MAS is that it relies rather on the distribution of cooperation algorithms than on centralized processes. The decentralized and loosely coupled nature of the network makes it possible to design applications that are highly flexible, scalable and adaptive. The multi-agent paradigm fits well to AmI environments requirements from several perspectives.

MAS provide *decentralized control* based on distributed autonomous entities. While a centralized control system is doomed to become increasingly complex and failure prone as the range of possible services and situations extends, a multi-agent methodology appears as a natural way to design truly scalable and robust systems.

MAS support complex interactions between entities, using high level semantic languages. Such a feature seems essential in AmI environments dealing with *various, heterogeneous information* from physical sensors, services or users preferences. Integration of such data is only possible at a higher level where all kind of information (about services, context ...) is expressed semantically.

In a MAS, autonomous entities with limited capabilities coordinate in order to achieve complex tasks. *Emergent coordination and flexible organization* patterns enable groups of agents to create and reconfigure application dynamically depending on current conditions. Such patterns seem well adapted to dynamic composition of elementary functionalities in an open, dynamic AmI environment.

A MAS is a society of agents with capacities and roles. Such a metaphorical representation of a system similar to a human society is of tremendous interest when closely interacting with users, as it can facilitates their *understanding and control of the system's behavior*.

Despite these appealing features, MAS adoption in the pervasive computing and AmI fields is still limited. While some work focused on using mobile agents as a way to handle user mobility in ubiquitous computing systems [6], they did not seek to exploit the full potential of MAS. Some other work demonstrate the relevancy of MAS-based pervasive systems towards adaptable and user-centric behavior [3] [4] [10]. However, designing such systems in a full-agent point of view has some drawbacks. A major obstacle is that no agent support is currently expected from device manufacturers, thus hindering the achievement of large-scale pervasive MAS in the near future. We also notice that performance is a tremendous issue in a MAS approach: agents reasoning and communications needs are often assumed to be too much resource consuming to be suitable for a constrained pervasive environment.

Considering these aspects, we believe that designing a MAS service composition infrastructure as an intelligent control layer over an AmI service-oriented architecture is a relevant trade-off. The SOA approach gives a realistic view of pervasive environments where heterogeneous services are provided by various manufacturers, thus enabling to easily build complex and efficient applications through service composition. The MAS approach complements this modular and flexible infrastructure by providing high level adaptation to users' activity. By combining both approaches, we expect building a fully adaptive system, which takes users activities and context into account to adapt and reconfigure themselves. We also wish to address performance issues thanks to a flexible agent approach were trade-offs between computation/communication and optimality are easier to handle than in a centralized system.

### 2.2. Bringing Multi-Agent System and Service-Oriented Architecture Together

We now give an overview of our proposed architecture in which a MAS infrastructure controls an AmI SOA to adapt its behavior.

In a way similar to SWS, we use semantic service descriptions to abstractly describe services' functionalities. We enrich these descriptions with contextual information, denoting the context of use of services. Using both aspects is key in an AmI environment where the usage of a service depends not only on its functionalities but also on its context of use. We consider that descriptions can be provided by various sources (manufacturer, users) since they do not need to comply with a commonly defined standard. The approach is also flexible enough to use incomplete descriptions or to rely on default ones when none is provided. Further issues would be to learn and complete descriptions through experiments using services, or to interpret natural language descriptions into structured semantic descriptions.

Using such descriptions, service agents can dynamically discover and enhance their capabilities with existing services. Depending on their interests or their current role in an activity, agents can either discover services to answer a current need or pro-actively provision services for expected use. Apart from these service agents, various agents handle other information that will have an impact on the behavior of the system. Thus, explicit user commands, physical context or privacy settings will be taken into account by user agents, context-aware agents or privacy agents respectively.

Among this multi-agent system, a user activity involves a group of heterogeneous agents. The formation and evolution of groups emerges from users' activities, since multiple services and information are involved, but can be structured by customizable rules defining the kind of services needed in a given situation. Such rules are a way to manage incremental levels of abstraction, similarly to task decomposition rules in hierarchical planning. All along the interaction, service providing agents try to discover and propose suitable services, while the task of other agents is to influence the behavior of the group with respect to external information (context, privacy, explicit user commands ...).

Service composition is achieved dynamically through the group's interactions. Each time an action is required, the group expresses it as a service or a service workflow invocation, whose enactment is delegated to the SOA level. At the opposite of Web service approach, the service composition is not completed with the specification and execution of a workflow. The group of agents dedicated to the activity remains active to follow the user in its interaction, reacting to changes, adapting and recomposing according to available information.

## 3. A Use Case Scenario

As an illustration of the intended behavior of our system, we provide here a typical use case scenario describing how an AmI environment can adapt to a communication activity.

Alice is calling her brother Bob. She wants to arrange a dinner for her mother's birthday. In Bob's intelligent home, the call request is received by the AmI system instead of directly ringing a phone. An agent is then assigned the task to build up a suitable service to enable Bob to receive this video call. It begins to ask other agents to join this activity, especially looking for video rendering and recording services that will enable Bob to best communicate with Alice. Bob is currently in is living room, so agents will come up with services from his TV set and his web cam.

However, while agents are joining this activity group, Bob's privacy keeping agent informs the group that Bob is not alone in his living room and that his privacy settings forbid to automatically start a video call without his explicit permission. The group then changes its behavior and starts looking for a way to inform Bob that his sister is willing to reach him. Several notification services are available, such as playing a sound on the Hi-Fi set or using a more discrete ambient light notification. As Alice's call is not urgent, the system decides to use the light notification that will not disturb Bob, who is currently talking with his guests. It also advices Alice's environment to propose her a stand-by mode, that will notify her when Bob will be available.

As Bob was actually waiting for a call from his sister, he quickly notices the light modification and decides to answer her. He wants to have a private conversation, so he temporarily leaves his guests to reach his bedroom, where other rendering devices are available. While he is moving in the house, some context-aware agents notice that Bob's context conditions are changing. This information triggers a lookup for new services, so that to propose him a video call in the best conditions. As Bob actually approaches the wall screen and stands in front of it, the system activates this screen to start and display Alice's video.

## 4. Methodological Approach and Expected Contribution

### 4.1. A Multidisciplinary Approach

As AmI investigates challenging issues which involves various capacities, our interest domain spans on several fields. Pervasive computing forms the basis of our working context, and provides us with exciting possibilities but also harsh resource constraints. We are concerned by issues in context-awareness and the service oriented approach, although we rely on external work on these points. Our central interest is in multi-agent systems, as a solution to dynamical and reconfigurable systems, and we also work on semantic web and semantic services as a mean to bridge the gap between services and MAS.

### 4.2. External Contributions Usage

Our work will benefit from ongoing work on the development of a full-featured AmI service oriented infrastructure. Using this infrastructure, we will be able to easily discover and execute services provided by various devices. The infrastructure also addresses some interoperability issues that will arise when considering heterogeneous devices.

As another tremendous aspect, we will benefit from a context management system. This system is designed as a context knowledge base that will gather information from sensors and will make high level information data available. Information will be integrated in a semantic model of context and we will be able either to query or to asynchronously receive high level context information.

### 4.3. Our Goal : Building a High-Level Service Composition Infrastructure

Given those two underlying contributions, our goal is to develop the service composition infrastructure described in section 2.2.. We already achieved preliminary work on semantic and contextual services descriptions that enable agents to discover and reason about services. We now elaborate on MAS architecture and coordination techniques that will enable us to design a flexible and powerful composition infrastructure. Further work will also be carried out on services descriptions and interactions between a service-oriented infrastructure and our MAS composition system.

We finally aim at building a complete prototype demonstrating the proposed scenario and others, with the help of the underlying SOA and context management system. We will assess how this prototype validates our approach in term of flexibility and adaptation as well as performance results. Hopefully, we will also be able to provide user studies to demonstrate the interest of our system to enhance users' interactions.

## 5. Summary and Future Works

In this paper, we presented an approach to empower AmI environments with applications that can adapt and reconfigure depending on users activity, interaction context and available services. Our approach aims at combining a service-oriented approach and multi-agent systems to enable adaptable and reconfigurable agent groups to control lower level services while taking into account various and heterogeneous information. Our future work will be to design suitable solutions and to demonstrate the benefits provided by our approach.

## 6. Acknowledgments

The PhD research described in this paper started on October 2004 at FranceTelecom R&D, Grenoble France with the collaboration of G2I/SMA laboratory, St-Étienne France. This work is partially supported by the European Commission under Project IST Amigo. We would also like to thank Gilles Privat and Olivier Boissier for their contribution to the definition and orientation of this PhD work.

## References

[1] Tony Andrews, Francisco Curbera, Hitesh Dholakia, Yaron Goland, Johannes Klein, Frank Leymann, Kevin Liu, Dieter Roller, Doug Smith, Satish Thatte, Ivana Trickovic, and Sanjiva Weer-

awarana. Business process execution language for web services specification, version 1.1 dated may 5, 2003. http://www-128.ibm.com/developerworks/library/ws-bpel/.

[2] Anupriya Ankolekar, Mark Burstein, Jerry R. Hobbs, Ora Lassila, David L. Martin, Drew McDermott, Sheila A. McIlraith, Srini Narayanan, Massimo Paolucci, Terry R. Payne, and Katia Sycara. DAML-S: Web Service Description for the Semantic Web. In Ian Horrocks and James Hendler, editors, *The Semantic Web - ISWC 2002: First International Semantic Web Conference, Sardinia, Italy, Proceedings*, volume 2342 / 2002, pages 348–363. Springer-Verlag Heidelberg, June 9-12, 2002.

[3] Paolo Busetta and Massimo Zancanaro. Open social agent architecture for distributed multimedia. In *Workshop on Agents at Work: Deployed applications, 2nd International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS 2003), Melbourne, Victoria, Australia*, July 2003.

[4] Augusto Celentano, Daniela Fogli, Piero Mussio, and Fabio Pittarello. Agents for distributed context-aware interaction. In *Workshop on AI in Mobile Systems (AIMS 2002), European Conference on Artificial Intelligence (ECAI 2002), Lyon, France*, July 22, 2002.

[5] Anind K. Dey. *Providing architectural support for Building Context-Aware Applications*. PhD thesis, Georgia institue of Technology, 2000.

[6] Nataliya Hristova, Gregory M.P. O'Hare, and Terry Lowen. Agent-based ubiquitous systems: 9 lessons learnt. In *Workshop on System Support for Ubiquitous Computing (UbiSys'03), 5th International Conference on Ubiquitous Computing (UbiComp 2003), Seattle, WA, USA*, October 12, 2003.

[7] Valérie Issarny, Daniele Sacchetti, Ferda Tartanoglu, Françoise Sailhan, Rafik Chibout, Nicole Levy, and Angel Talamona. Developing ambient intelligence systems: A solution based on web services. *Automated Software Engineering*, 12(1):101–137, January 2004.

[8] Noriaki Izumi, Akio Sashima, Koichi Kurumatani, and Hideyuki Nakashima. Semantic services coordination for human and agent communities. In B. Burg, J. Dale, Tim Finin, H. Nakashima, L. Padgham, C. Sierra, and S. Willmott, editors, *2nd International Workshop on Challenges in Open Agent Environments, 2nd International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS 2003), Melbourne, Victoria, Australia*. Springer-Verlag, July 2003.

[9] Soraya Kouadri Mostéfaoui and Béat Hirsbrunner. A context-based services discovery and composition framework for wireless environments. In *Proceedings of the IASTED International Conference on Wireless and Optical Networks (WOC'03), Banff, Alberta, Canada*, pages 637–642, July 14-16, 2003.

[10] Christian Kray. *Situated Interaction on Spatial Topics*. PhD thesis, Universität des Saarlandes, Germany, 2003.

[11] Ryusuke Masuoka, Yannis Labrou, Bijan Parsia, and Evren Sirin. Ontology enabled pervasive computing applications. *IEEE Intelligent Systems*, 18(5):68–72, Sep./Oct. 2003.

[12] Gilles Privat. Des objets communicants à la communication ambiante. *Les Cahiers du Numérique*, 3(4):23–44, 2002.

[13] Anand Ranganathan, Robert E. McGrath, Roy H. Campbell, and M. Dennis Mickunas. Ontologies in a pervasive computing environment. In *Workshop on Ontologies and Distributed Systems, 18th International Joint Conference on Artificial Intelligence, Acapulco, Mexique*, August 9th 2003.

[14] Manuel Roman and Roy H. Campbell. A middleware-based application framework for active space applications. In *Proceedings of ACM/IFIP/USENIX International Middleware Conference (Middleware 2003), Rio de Janeiro, Brazil*, 2003.

[15] Umar Saif, Hubert Pham, Justin Mazzola Paluska, Jason Waterman, Chris Terman, and Steve Ward. A case for goal-oriented programming semantics. In *Workshop on System Support for Ubiquitous Computing (UbiSys'03), 5th International Conference on Ubiquitous Computing (UbiComp 2003), Seattle, WA, USA*, October 12, 2003.

[16] Katia Sycara, Massimo Paolucci, Anupriya Ankolekar, and Naveen Srinivasan. Automated discovery, interaction and composition of semantic web services. *Journal of Web Semantics*, 1(1):27–46, September 2003.

[17] Maja Vukovic and Peter Robinson. Adaptive, planning-based, web service composition for context awareness. In *Proceedings of the 2nd International Conference on Pervasive Computing (Pervasive 2004), Vienna, Austria*, April 2004.

[18] Dan Wu, Evren Sirin, Bijan Parsia, James Hendler, and Dana Nau. Automatic web services composition using SHOP2. In *Proceedings of ICAPS'03 Workshop on Planning for Web Services, Trento, Italy*, June 2003.