# MERCURIO: An Interaction-oriented Framework for Designing, Verifying and Programming Multi-Agent Systems*

Matteo Baldoni[1], Cristina Baroglio[1], Federico Bergenti[4], Antonio Boccalatte[3], Elisa Marengo[1], Maurizio Martelli[3], Viviana Mascardi[3], Luca Padovani[1], Viviana Patti[1], Alessandro Ricci[2], Gianfranco Rossi[4], and Andrea Santi[2]

[1] Università degli Studi di Torino
{baldoni,baroglio,emarengo,padovani,patti}@di.unito.it
[2] Università degli Studi di Bologna
{a.ricci,a.santi}@unibo.it
[3] Università degli Studi di Genova
{martelli,mascardi}@disi.unige.it, nino@dist.unige.it
[4] Università degli Studi di Parma
{federico.bergenti,gianfranco.rossi}@unipr.it

**Abstract.** This is a position paper reporting the motivations, the starting point and the guidelines that characterise the MERCURIO[5] project proposal, submitted to MIUR PRIN 2009[6]. The aim is to develop formal models of interactions and of the related support infrastructures, that overcome the limits of the current approaches by explicitly representing not only the agents but also the computational environment in terms of rules, conventions, resources, tools, and services that are functional to the coordination and cooperation of the agents. The models will enable the verification of interaction properties of MAS from the global point of view of the system as well as from the point of view of the single agents, due to the introduction a novel social semantic of interaction based on commitments and on an explicit account of the regulative rules.

## 1 Motivation

The growing pervasiveness of computer networks and of Internet is an important catalyst pushing towards the realization of *business-to-business* and *cross-business solutions*. Interaction and coordination, central issues to any distributed system, acquire in this context a special relevance since they allow the involved groups to integrate by interacting according to the agreed contracts, to share best practices and agreements, to cooperatively exploit resources and to facilitate the identification and the development of new products.

* Position paper
[5] Italian name of Hermes, the messenger of the gods in Greek mythology.
[6] Despite the label "2009", it is the just closed call for Italian National Projects, http://prin.miur.it/index.php?pag=2009.

The issues of interaction, coordination and communication have been receiving great attention in the area of Multi-Agent Systems (MAS). MAS are, therefore, the tools that could better meet these needs by offering the proper abstractions. Particularly relevant in the outlined application context are a shared and inspectable specification of the rules of the MAS and the verification of global properties of the interaction: in open environments, in fact, it is important to have guaranties on how interaction will take place, coping with notions like responsibility and commitment. Unfortunately, current proposals of platforms and languages for the development of MAS do not supply high level tools for directly implementing this kind of specifications. As a consequence, they do not support the necessary forms of verification, with a negative impact on the applicability of MAS to the realization of business-to-business and cross-business systems.

Let us consider, for instance, JADE [17], which is one of the best known infrastructures, sticking out for its wide adoption also in business contexts. JADE agents communicate by exchanging messages that conform to FIPA ACL [3]. According to FIPA ACL mentalistic approach, the semantics of messages is given in terms of preconditions and effects on the mental states of the involved agents, which are assumed to share a common ontology. Agent platforms based on FIPA exclusively provide syntactic checks of message structures, entrusting the semantics issues to agent developers. This hinders the applicability to open contexts, where it is necessary to coordinate autonomous and heterogeneous agents and it is not possible to assume mutual trust among them. In these contexts it is necessary to have an unambiguous semantics allowing the verification of interaction properties before the interaction takes place [54] or during the interaction [9], preserving at the same time the privacy of the implemented policies.

The mentalistic approach does not allow to satisfy all these needs [42]; it is suitable for reasoning from the *local point of view* of a single agent, but it does not allow the verification of interaction properties of a MAS from a *global point of view*. One of the reasons is that the reference model *lacks* an abstraction for the representation, by means of a public specification, of elements like (i) resources and services that are available in the environment/context in which agents interact and (ii) the rules and protocols, defining the interaction of agents through the environment/context. All these elements belong (and contribute to make) the environment of the interacting agents. Such an abstraction, if available, would be the natural means for encapsulating resources, services, and functionalities (like ontological mediators) that can support the communication and the coordination of agents [69, 68, 45], thus facilitating the verification of the properties [13]. They could also facilitate the interaction of agents implemented in different languages because it would be sufficient that each language implements the primitives for interacting with the environment [1]. One of the consequences of the lack of an explicit representation of the environment is that only forms of *direct communication* are possible. On the contrary, in the area of distributed systems and also in MAS alternative communication models, such as the generative communication based on tuple spaces [34], have been put forward. These forms of communication, which do not necessarily require a space-time coupling between agents, are not supported.

The issues that we mean to face have correspondences with issues concerning normative MAS [72] and in Artificial Institutions [33, 67]. The current proposals in this field, however, do not supply all of the solutions that we need: either they do not account for indirect forms of communication or they lack mechanisms for allowing the a priori verification of global properties of the interaction. As [33, 67] witnesses, there is, instead, an emerging need of defining a more abstract notion of action, which is not limited to direct speech acts. In this case, institutional actions are performed by executing instrumental actions that are conventionally associated to them. Currently, instrumental actions are limited to speech acts but this representation is not always natural. For instance, for voting in the human world, people often raise their hands rather than saying the name corresponding to their choice. If the environment were represented explicitly it would be possible to use a wider range of instrumental actions, that can be perceived by the other agents through the environment that acts as a medium.

Our goal is, therefore, to propose an infrastructure that overcomes such limits. The key of the proposal is the adoption of a *social approach* to communication [47, 14, 13, 12], based on a model that includes an explicit representation not only of *agents* but also of their *environment*, as a collection of virtual and physical resources, tools and services, "artifacts" as intended in the Agents & Artifacts (A&A) meta-model [45], which are shared, used and adapted by the agents, according to their goals. The introduction of environments is fundamental to the adoption of an observational (social) semantics, like the one used in commitment protocols, in that it supplies primitives that allow agents to perceive and to modify the environment itself and, therefore, to interact and to coordinate with one another in a way that satisfies the rules of the environment. On the other hand, the observational semantics is the only sufficiently general semantics to allow forms of interaction and of communication that do not rely solely on direct speech acts. As a consequence we will include models where communication is mediated by an environment, that encapsulates and applies rules and constraints aimed at coordinating agents at the organization level, and integrates *ontological mediation* functionalities. The environment will provide the *contract* that agents should respect and a context into which interpreting their actions. In this way, it will be possible to *formally verify the desired properties of the interaction, a priori and at execution time.*

## 2 Vision

The focus of our proposal is on the definition of formal models of interactions and of the related support infrastructures, which explicitly represent not only the agents but also the environment in terms of rules of interaction, conventions, resources, tools, and services that are functional to the coordination and cooperation of the agents. These models must allow both direct and indirect forms of communication, include ontological mediators, and enable the verification of interaction properties of MAS from the global point of view of the system as well as from the point of view of the single agents. The approach we plan to pursue in order to define a formal model of interaction is based on a revision in social terms of the interaction and of the protocols controlling it, along the lines of [14, 13, 12]. Furthermore, we will model the environment, in the sense introduced by the A&A meta-model [45]. This will lead to the study of communication forms mediated by the environment. The resulting models will be validated by the implementation of software tools and of programming languages featuring the designed abstractions. More in details, with reference to Fig. 1, the goals are:

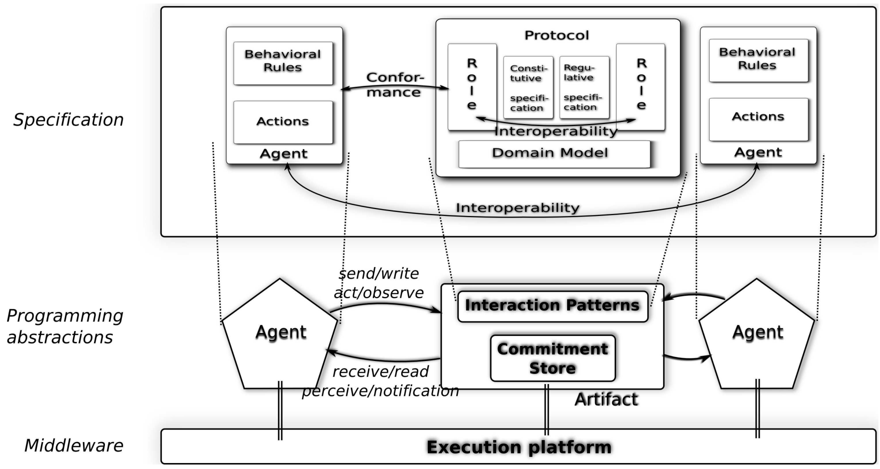


**Fig. 1.** The MERCURIO architecture.

**To introduce a formal model** for specifying and controlling the interaction. The model (top level of Fig. 1) must be equipped with an observational (commitment-based) semantics and must be able to express not only direct communicative acts but also interactions mediated by the environment. This will enable forms of verification that encompass both global interaction properties and specific agent properties such as interoperability and conformance [11]. The approach does not hinder agent autonomy, it guarantees the privacy of the policies implemented by the agents, and consequently favors the composition of heterogeneous agents. The model will be inspired by the social approach introduced in [47] and subsequently extended in [14, 13, 12].

**To define high-level environment models** supporting the forms of interactions and coordination between agents outlined above. These models must support: interaction protocols based on commitments; the definition of rules on the interaction; forms of mediated communication and coordination between agents (such as stigmergic coordination). They must also enable forms of a priori and runtime verification of the interaction. To these aims, we plan to use the A&A meta-model [60, 69, 45, 58] and the corresponding notion of programmable environment [59] (programming abstractions level of Fig. 1).

**To integrate ontologies and ontological mediators** in the definition of the defined models so as to guarantee openness and heterogeneity of MAS. Mediation will occur at two distinct levels: the one related to the vocabulary and domain of discourse and the one that characterizes the social approach where it is required to bind the semantics of the agent actions with their meaning in social terms. Ontological mediators will be realized as artifacts.

**To integrate the abstractions** defined in the above models within programming languages and frameworks. In particular, we plan to integrate the notions of agents, of environment, of direct and mediated communication, and of ontological mediators. Possible starting points are the aforementioned FIPA ACL standard and the works that focus on the integration of agent-oriented programming languages with environments [57]. The JaCa platform [59], integrating Jason and CArtAgO, will be taken as reference. This will form the execution platform of Fig. 1 and will supply the primitives for interacting with the environments.

**To develop an open-source prototype** of software infrastructure for the experimentation of the defined models. The prototype will integrate and extend existing technologies such as JADE [4] (as a FIPA-compliant framework), CArtAgO [1] (for the programming and the execution of environments), Jason (as a programming language for BDI agents), MOISE [37] (as organizational infrastructure).

**To identify applicative scenarios** for the evaluation of the developed models and prototypes. In this respect we regard the domain of Web services as particularly relevant because of the need to deploy complex interactions having those characteristics of flexibility that agents are able to guarantee. Another interesting application regards the verification of adherence of bureaucratic procedures of public administration with respect to the current normative. Specific case studies will be defined in collaboration with those companies that have stated interest towards the project.

## 3 State of Art

These novel elements, related to the formation of and the interaction within decentralized structures, find an initial support in proposals from the literature in the area of MAS. Current proposals, however, are still incomplete in that they supply solutions to single aspects. For instance, electronic institutions [30, 10, 37, 36] regulate interaction, tackle open environments and their semantics allows the verification of properties but they only tackle direct communication protocols, based on speech acts, and do not include an explicit notion of environment. Commitment protocols [47, 71], effective in open systems and allowing more general forms of communication, do not supply behavioral patterns, and for this reason it is impossible to verify properties of the interaction. Eventually, most of the models and architectures for environments prefigure simple/reactive agent models without defining semantics, that are comparable to the ones for ACL, and without explaining how such proposals could be integrated with direct communication models based on speech acts. We classify the relevant contributions in the literature according to the objectives and the methodological aspects that will be examined in-depth along the project.

### 3.1 Formal Models for Regulating the Interaction in MAS

This topic has principally been tackled by modeling interaction protocols. Most of protocol representations refer to classic models, such as Petri nets, finite state machines, process algebras, and aim at capturing the expected interaction flow. An advantage of this approach is that it supports the verification of interaction properties [54, 22, 11], such as: verifying the interoperability of the system and verifying if certain modifications of a system preserve some desired properties (a crucial issue in open domains where agents can enter/leave the system at any time). Singh and colleagues criticize the use of procedural specifications because too rigid [62, 26, 71]: agents cannot take advantage of opportunities that emerge along the interaction and that are not foreseen by their procedure. Another issue is that communication languages use a BDI semantics (FIPA ACL is an example), where each agent has own goals and beliefs but at the system level it is impossible to perform introspection of agents, which result to be black boxes. For what concerns the verification of properties this approach is suitable to allow agents to draw conclusions about their own behavior but not to verify global properties of the system [42, 66].

Both problems are solved by commitment protocols [47, 62], which rely on an observational semantics of the interaction and offer adequate flexibility to agents. Moreover, they do not require the spatio-temporal coupling of agents (as instead direct communication does). Another advantage is that, though remaining black boxes, agents agree on the meaning of actions which make the protocol and that affect the MAS social state. Since interactions are observable and their semantics is shared, each agent should be able to draw conclusions concerning the system as a whole. Unfortunately, besides some preliminary studies [63], the state of art does not contain proposals on how performing the verifications in a MAS, ruled by this kind of protocols. A relevant feature seems to be the introduction, within commitment protocols, of behavioral rules which constrain the possible evolutions of the social state [13, 12].

### 3.2 Environment Models

The notion of environment has always played a key role in the context of MAS; recently, it started to be considered as a first-class abstraction useful for the design and the engineering of MAS [69]. A&A [45] follows this perspective, being a meta-model rooted upon Activity Theory and Computer Support Cooperative Work that defines the main abstractions for modeling a MAS, and in particular for modeling the environment in which a MAS is situated. A&A promotes a vision of an endogenous environment, that is a sort of software/computational environment, part of the MAS, that encapsulates the set of tools and resources useful/required by agents during the execution of their activities. A&A introduces the notion of artifact as the fundamental abstraction used for modeling the resources and the tools that populates the MAS environment. The introduction of the environment as a new first-class abstraction requires new engineering approaches for programming the MAS environment. The CArtAgO framework [59] has been devised precisely for copying this new necessity. It provides the basis for the engineering of MAS environments on the base of: (i) a proper computational model and (ii) a programming model for the design and the development of the environments on the base of the A&A meta-model. In particular, it provides those features that are important from a software engineering point of view: *abstraction*, it preserves the agent abstraction level, since the main concepts used to define application environments, i.e. artifacts and workspaces, are first-class entities in the agents world, and the interaction with agents is built around the agent-based concepts of action and perception (use and observation); *modularity and encapsulation*, it provides an explicit way to modularise the environment, where artifacts are components representing units of functionality, encapsulating a partially-observable state and operations; *extensibility and adaptation*, it provides a direct support for environment extensibility and adaptation, since artifacts can be dynamically constructed (instantiated), disposed, replaced, and adapted by agents; *reusability*, it promotes the definition of types of artifact that can be reused as tools in different application contexts, such as in the case of coordination artifacts empowering agent interaction and coordination, such as blackboards and synchronisers. These features will be advantageous in the realization of the second goal of the project, w.r.t. approaches like [27], where commitment stores, communication constraints and the interaction mechanisms reside in the middleware, which shields them from the agents. This has two disadvantages: the first is that even though all these elements are accounted for in the high level specification, the lack of a corresponding programming abstraction makes it difficult to verify whether the system corresponds to the specification; the second is a lack of flexibility, in that it is not possible for the agents to dynamically change the rules of interaction or to adopt kinds of communication that are not already implemented in the middleware.

In the state of the art numerous applications of the endogenous environments, i.e. environments used as a computational support for the agents' activities, have been explored, including coordination artifacts [46], artifacts used for realizing argumentation by means of proper coordination mechanisms [44], artifacts used for realizing stigmergic coordination mechanisms [56, 50], organizational artifacts [36, 51, 52]. Even if CArtAgO can be considered a framework sufficiently mature for the concrete developing of software/computational MAS environments it can not be considered "complete" yet. Indeed at this moment the state of the art and in particular the CArtAgO framework are still lacking: (i) a reference standard on the environment side comparable to the existing standards in the context of the agents direct communications (FIPA ACL), (ii) the definition of a rigorous and formal semantics, in particular related to the artifact abstraction, (iii) an integration with the current communication approaches (FIPA ACL, KQML, etc.), and finally (iv) the support of semantic models and ontologies.

### 3.3 Multi-agent Organizations and Institutions

The possibility of controlling and specifying interactions is relevant also for areas like the organizational theory and electronic institutions areas [41, 72, 15]. Tendentiously, the focus is orthogonal to the one posed on interaction protocols, in that it concerns the modeling of the structure rather than of the interaction.

The abstract architecture of e-Institutions (e.g. Ameli [30]), places a middleware composed of governors and staff agents between participating agents and an agent communication infrastructure (e.g. JADE [4]). The notion of environment is dialogical: it is not something agents can sense and act upon but a conceptual one that agents, playing within the institution, can interact with by means of norms and laws, based on specific ontologies, social structures, and language conventions. Agents communicate with each other by means of speech acts and, behind the scene, the middleware mediates such communication. The extension proposed for situated e-Institutions [10] introduces the notion of "World of Interest" to model the environment, that is external to the MAS but which is relevant to the MAS application. The infrastructure of the e-Institution, in this case, mediates also the interaction of the agents in the MAS with the view of the environment that it supplies. Along this line, ORA4MAS [36] proposes the use of artifacts to enable the access of the agents in the MAS to the organisation, providing a working environment that agents can perceive, act upon and adapt. Following the A&A perspective, they are concrete bricks used to structure the agents' world: part of this world is represented by the organisational infrastructure, part by artifacts introduced by specific MAS applications, including entities/services belonging to the external environment.

According to [10] there are, however, two significant differences among artifacts and e-Institutions: (i) e-Institutions are tailored to a particular, though large, family of applications while artifacts are more generic; (ii) e-Institutions are a well established and proven technology that includes a formal foundation, and advanced engineering and tool support, while for artifacts, these features are still in a preliminary phase. One of the aims of MERCURIO is to give to artifacts both the formal foundation (in terms of commitments and interaction patterns) and the engineering tools that they are still missing. The introduction of interaction patterns with an observational nature, allowing the verification of global properties, that we aim at studying, will have an impact also on the representation of e-Institutions by means of artifacts.

### 3.4 Semantic Mediation in MAS

The problem of semantic mediation at the vocabulary and domain of discourse levels was faced for the first time by the "Ontology Service Specification" [8] issued by FIPA in 2001. According to that specification, an "Ontology Agent"

(OA, for short) should be integrated in the MAS in order to provide services such as translating expressions between different ontologies and/or different content languages and answering queries about relationships between terms or between ontologies. Although the FIPA Ontology Service Specification represents the first and only attempt to analyze in a systematic way the services that an OA should provide for ensuring semantic interoperability in an open MAS, it has many limitations. The main one is the assumption that each ontology integrated in the MAS adheres to the OKBC model [6]. Currently, in fact, the most widely accepted ontology language is OWL [7] which is quite different from OKBC and cannot be converted to it in an easy and automatic way. Also, agents are allowed to specify only one ontology as reference vocabulary for a given message, which is a strong limitation since it an agent might use terms from different ontologies in the same message, and hence it might want to refer to more than one ontology at the same time.

Maybe due to these limitations, there have been really few attempts to design and implement OAs. The first dates back to 2001 [64] and realizes an OA for the COMTEC platform that implements a subset of the services of a generic FIPA-compliant OA. In 2007 [48] integrated an OA into AgentService, a FIPA compliant framework based on .NET [65]. Ontologies in AgentService are represented in OKBC, and hence the implementation of their OA is fully compliant with the FIPA specification, although the offered services are a subset of the possible ones. The only two attempts of integrating a FIPA-compliant OA into JADE, we are aware of, are [43], and [25]. Both follow the FIPA specification but adapt it to ontologies represented in OWL. The first proposal is aimed at storing and modifying OWL ontologies: the OA agent exploits the Jena library [38] to this aim. The second proposal, instead, faces the problem of "*answering queries about relationships between terms or between ontologies*". The solution proposed by the authors exploits ontology matching techniques [31]. Apart from [25], no other existing proposal faces that problem. Among non FIPA-compliant solutions, we mention [39], which focuses on the process of mapping and integrating ontologies in a MAS thanks to a set of agents which collaborate together, and the proposal in [49], which implements the OA as a web service, in order to offer its services also over the Internet.

As far as semantic mediation at the social approach level is concerned, we are aware of no proposals in the literature. In order to take the context of count-as rules into account, we plan to face this research issue by exploiting context aware semantic matching techniques, that extend and improve those described in [40].

### 3.5 Software Infrastructures for Agents

The tools currently available to agent developers fail in supporting both semantic interoperability and goal-directed reasoning. Nowadays, the development of agents and multi-agent systems is based on two kinds of tools: agent platforms and BDI (or variations) development environments. Agent platforms, such as JADE [19, 16, 18] and FIPA-OS [2] provide only a transport layer and some basic services, but they do not provide any support for goal-directed behaviour.

Moreover, they lack support for semantic interoperability because they do not take into account the semantics of the ACL they adopt. The available BDI development environments, such as Jason [23], Jadex [24], 2APL [29] support only syntactic interoperability because they do not exploit their reasoning engines to integrate the semantics of the adopted ACL.

The research on Agent Communication Languages (ACL) is constantly headed towards semantic interoperability [35] because the most common ACLs, e.g., KQML [32] and FIPA ACL [3], provide each message with a declarative semantics that was explicitly designed to support goal-directed reasoning. Unfortunately, the research on ACLs only marginally investigated the decoupling properties of this kind of languages (see, e.g., [20, 21]). To support the practical development of software agents, several programming languages have thus been introduced to incorporate some of the concepts from agent logics. Some languages use actions as their starting point to define commitments (Agent-0, [61]), intentions (AgentSpeak(L), [55]) and goals (3APL, [28]).

## 4 Expected Results

The achievements expected from this research are of different natures: scientific result that will advance the state of the art, software products deriving from the development of implementations, and upshots in applicative settings.

The formal model developed in MERCURIO will extend commitment protocols by introducing behavioral rules. The starting point will be the work done in [14, 13, 12]. This will advance the current state of the art with respect to the specification of commitment protocols and also with respect to the verification of interaction properties (like interoperability and conformance), for which there currently exist only preliminary proposals [63]. Another advancement concerns the declarative specification of protocols and their usage by designers and software engineers. The proposals coming from MERCURIO conjugate the flexibility and openness features that are typical of MAS with the needs of modularity and compositionality that are typical of design and development methodologies. The adoption of commitment protocols makes it easier and more natural to represent (inter)actions that are not limited to communicative acts but that include interactions mediated by the environment, namely actions upon the environment and the detection of variations of the environment ruled by "contracts".

For what concerns the coordination infrastructure, a first result will be the definition of environments based on the A&A meta-model and on the CArtAgO computational framework, that implement the formal models and the interaction protocols mentioned above. A large number of the environments, described in the literature supporting communication and coordination, have been stated considering purely reactive architectures. In MERCURIO we will formulate environment models that allow goal/task-oriented agents (those that integrate pro-activities and re-activities) the participation to MAS. Among the specific results related to this, we foresee an advancement of the state of the art with respect to the definition and the exploitation of forms of stigmergic coordination [56] in the context of intelligent agent systems. A further contribution regards the flexible use of artifact-based environments by intelligent agents, and consequently the reasoning techniques that such agents may adopt to take advantage of these environments. First steps in this direction, with respect to agents with BDI architectures, have been described in [53, 50].

The MERCURIO project aims at putting forward an extension proposal for the FIPA ACL standard, where the FIPA ACL-based communication is integrated with forms of interactions, that are enabled and mediated by the environment. This will lead to an explicit representation of environments as first-class entities (in particular endogenous environments based on artifacts) and of the related model of actions/perceptions. Furthermore we will formulate an improved version of the MAS programming language/framework JaCa, where we plan to integrate the agent-oriented programming language Jason, which is based on a BDI architecture, with the CArtAgO computational framework. This result will extend the work done so far in this direction [57, 59].

In MERCURIO we will implement a prototype of the reference infrastructural model defined by the project. The prototype will be based on the development and integration of existing open-source technologies including JADE [4], the reference FIPA platform, CArtAgO [1], the reference platform and technology for the programming and execution of environments, and agent-oriented programming languages such as Jason [5] and 2APL [29]. The software platform will include implementations of the "context sensitive" ontology alignment algorithms developed in MERCURIO. The algorithms will be evaluated against standard benchmarks and also against the case studies devised in MERCURIO.

Aside from the effects on research contexts, we think that the project may give significant contributions also to industrial applicative contexts, in particular to those companies working on software development in large, distributed systems and in service-oriented architectures. Among the most interesting examples are the integration and the cooperation of e-Government applications (services) spread over the nation. For this reason, MERCURIO will involve some companies in the project, and in particular in the definition of realistic case studies against which the project's products will be validated. As regards (Web) services, some fundamental aspects promoted by the SOA model, such as autonomy and decoupling, are addressed in a natural way by the agent-oriented paradigm. Development and analysis of service-oriented systems can benefit from the increased level of abstraction offered by agents, by reducing the gap between the modelling, design, development, and implementation phases.

**Acknowledgements**

We thank S. E. Mantix for the valuable support and helpful discussions.

## References

1. CARTAGO. http://cartago.sourceforge.net.

2. FIPA OS. http://fipa-os.sourceforge.net.
3. FIPA specifications. http://www.fipa.org.
4. JADE. http://jade.tilab.com/.
5. JASON. http://jason.sourceforge.net.
6. OKBC. http://www.ai.sri.com/ okbc/.
7. OWL. http://www.w3.org/TR/owl-features/.
8. Fipa architecture board, fipa ontology service specification, 2001. http://www.fipa.org/specs/fipa00086/.
9. M. Alberti, F. Chesani, M. Gavanelli, E. Lamma, P. Mello, and P. Torroni. Verifiable agent interaction in abductive logic programming: The sciff framework. *ACM Trans. Comput. Log.*, 9(4), 2008.
10. J. L. Arcos, P. Noriega, J. A. Rodríguez-Aguilar, and C. Sierra. E4MAS Through Electronic Institutions. In Weyns et al. [70], pages 184–202.
11. M. Baldoni, C. Baroglio, A. K. Chopra, N. Desai, V. Patti, and M. P. Singh. Choice, interoperability, and conformance in interaction protocols and service choreographies. In *Proc. of the 8th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2009*, pages 843–850. IFAAMAS, 2009.
12. M. Baldoni, C. Baroglio, and E. Marengo. Behavior-oriented Commitment-based Protocols. In H. Coelho and M. Wooldridge, editors, *Proc. of 19th European Conference on Artificial Intelligence, ECAI 2010*, Lisbon, Portugal, August 2010. To appear.
13. M. Baldoni, C. Baroglio, and E. Marengo. Commitment-based Protocols with Behavioral Rules and Correctness Properties of MAS. In A. Omicini, S. Sardina, and W. Vasconcelos, editors, *Proc. of International Workshop on Declarative Agent Languages and Technologies, DALT 2010, held in conjuction with AAMAS 2010*, pages 66–83, Toronto, Canada, May 2010.
14. M. Baldoni, C. Baroglio, and E. Marengo. Constraints among Commitments: Regulative Specification of Interaction Protocols. In A. Artikis, J. Bentahar, A. Artikis, and F. Dignum, editors, *Proc. of International Workshop on Agent Communication, AC 2010, held in conjuction with AAMAS 2010*, pages 2–18, Toronto, Canada, May 2010.
15. M. Baldoni, G. Boella, and L. van der Torre. Bridging Agent Theory and Object Orientation: Agent-like Communication among Objects. In R. H. Bordini, M. Dastani, J. Dix, and A. Seghrouchni, editors, *Post-Proc. of the International Workshop on Programming Multi-Agent Systems, ProMAS 2006*, volume 4411 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 149–164. Springer, 2007.
16. F. Bellifemine, F. Bergenti, G. Caire, and A. Poggi. JADE - A Java Agent Development Framework. In R. H. Bordini, M. Dastani, J. JDix, and A. El Fallah-Seghrouchni, editors, *Multi-Agent Programming: Languages, Platforms and Applications*, volume 15 of *Multiagent Systems, Artificial Societies, and Simulated Organizations*, pages 125–147. Springer, 2005.
17. F. Bellifemine, G. Caire, A. Poggi, and G. Rimassa. JADE: A software framework for developing multi-agent applications. Lessons learned. *Information & Software Technology*, 50(1-2):10–21, 2008.
18. F. Bellifemine, G. Caire, A. Poggi, and G. Rimassa. JADE: A software framework for developing multi-agent applications. Lessons learned. *Information & Software Technology*, 50(1-2):10–21, 2008.
19. F. Bellifemine, A. Poggi, and G. Rimassa. Developing multi-agent systems with a FIPA-compliant agent framework. *Softw., Pract. Exper.*, 31(2):103–128, 2001.

20. F. Bergenti and F. Ricci. Three Approaches to the Coordination of Multiagent Systems. In *Proceedings of the 2002 ACM Symposium on Applied Computing (SAC)*, pages 367–372, Madrid, Spain, March 2002. ACM.
21. F. Bergenti, G. Rimassa, M. Somacher, and L. M. Botelho. A FIPA Compliant Goal Delegation Protocol. In M.-P. Huget, editor, *Communication in Multiagent Systems, Agent Communication Languages and Conversation Polocies*, volume 2650 of *Lecture Notes in Computer Science*, pages 223–238. Springer, 2003.
22. L. Bordeaux, G. Salaün, D. Berardi, and M. Mecella. When are Two Web Services Compatible? In M.-C. Shan, U. Dayal, and M. Hsu, editors, *Technologies for E-Services, 5th International Workshop, TES 2004*, volume 3324 of *Lecture Notes in Computer Science*, pages 15–28, Toronto, Canada, August 2004. Springer.
23. R. H. Bordini, J. F. Hübner, and R. Vieira. Jason and the Golden Fleece of Agent-Oriented Programming. In R. H. Bordini, M. Dastani, J. Dix, and A. El Fallah-Seghrouchni, editors, *Multi-Agent Programming: Languages, Platforms and Applications*, volume 15 of *Multiagent Systems, Artificial Societies, and Simulated Organizations*, pages 3–37. Springer, 2005.
24. L. Braubach, A. Pokahr, and W. Lamersdorf. Jadex: A bdi agent system combining middleware and reasoning. In *Software Agent-Based Applications, Platforms and Development Kits*. Birkhauser Book, 2005.
25. D. Briola, A. Locoro, and V. Mascardi. Ontology Agents in FIPA-compliant Platforms: a Survey and a New Proposal. In M. Baldoni, M. Cossentino, F. De Paoli, and V. Seidita, editors, *Proc. of WOA 2008: Dagli oggetti agli agenti, Evoluzione dell'agent development: metodologie, tool, piattaforme e linguaggi*. Seneca Edizioni, 2008.
26. A. K. Chopra and M. P. Singh. Nonmonotonic Commitment Machines. In F. Dignum, editor, *Advances in Agent Communication, International Workshop on Agent Communication Languages*, volume 2922 of *Lecture Notes in Computer Science*, pages 183–200, Melbourne, Australia, July 2003. Springer.
27. A. K. Chopra and M. P. Singh. An Architecture for Multiagent Systems: An Approach Based on Commitments. In *Proceedings of the AAMAS Workshop on Programming Multiagent Systems (ProMAS)*, 2009.
28. M. Dastani, B. M. van Riemsdijk, F. Dignum, and J.-J. Ch. Meyer. A Programming Language for Cognitive Agents Goal Directed 3APL. In M. Dastani, J. Dix, and A. El Fallah-Seghrouchni, editors, *Programming Multi-Agent Systems, First International Workshop, PROMAS 2003*, volume 3067 of *Lecture Notes in Computer Science*, pages 111–130, Melbourne, Australia, July 2003. Springer.
29. Mehdi Dastani. 2apl: a practical agent programming language. *Autonomous Agents and Multi-Agent Systems*, 16(3):214–248, 2008.
30. M. Esteva, B. Rosell, J. A. Rodríguez-Aguilar, and J. L. Arcos. AMELI: An Agent-Based Middleware for Electronic Institutions. In *AAMAS*, pages 236–243. IEEE Computer Society, 2004.
31. J. Euzenat and P. Shvaiko. *Ontology Matching*. Springer, 2007.
32. T. Finin, Y. Labrou, and J. Mayfield. Kqml as an agent communication language. In Bradshaw J., editor, *Software Agents*. MIT Press, Cambridge, 1997.
33. N. Fornara, F. Viganò, and M. Colombetti. Agent communication and artificial institutions. *Autonomous Agents and Multi-Agent Systems*, 14(2):121–142, 2007.
34. David Gelernter. Generative communication in linda. *ACM Trans. Program. Lang. Syst.*, 7(1):80–112, 1985.
35. S. Heiler. Sematic Interoperability. *ACM Computing Surveys*, 27(2):271–273, 1995.

36. J. F. Hubner, O. Boissier, R. Kitio, and A. Ricci. Instrumenting multi-agent organisations with organisational artifacts and agents: "Giving the organisational power back to the agents". *Autonomous Agents and Multi-Agent Systems*, 20, 2009.
37. Jomi Fred Hübner, Jaime Simão Sichman, and Olivier Boissier. Developing organised multiagent systems using the moise. *IJAOSE*, 1(3/4):370–395, 2007.
38. *Jena – A Semantic Web Framework for Java.* Online, accessed on June, 14th, 2010. http://jena.sourceforge.net/.
39. L. Li, B. Wu, and Y. Yang. Agent-based ontology integration for ontology-based application. In *Australasian Ontology Workshop, AOW 2005, Proceedings*, pages 53–59, 2005.
40. V. Mascardi, A. Locoro, and F. Larosa. Exploiting Prolog and NLP techniques for matching ontologies and for repairing correspondences. In *24th Convegno Italiano di Logica Computazionale, CILC 2009, Proceedings*, 2009.
41. C. Masolo, L. Vieu, E. Bottazzi, C. Catenacci, R. Ferrario, A. Gangemi, and N. Guarino. Social Roles and their Descriptions. In D. Dubois, C. A. Welty, and M.-A. Williams, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR2004)*, pages 267–277, Whistler, Canada, June 2004. AAAI Press.
42. P. McBurney and S. Parsons. Games That Agents Play: A Formal Framework for Dialogues between Autonomous Agents. *Journal of Logic, Language and Information*, 11(3):315–334, 2002.
43. M. Obitko and V. Snáěl. Ontology repository in multi-agent system. In M. H. Hamza, editor, *Conference on Artificial Intelligence and Applications, AIA 2004, Proceedings*, 2004.
44. E. Oliva, P. McBurney, and A. Omicini. Co-argumentation Artifact for Agent Societies. In I. Rahwan, S. Parsons, and C. Reed, editors, *Argumentation in Multi-Agent Systems, 4th International Workshop, ArgMAS 2007*, volume 4946 of *Lecture Notes in Computer Science*, pages 31–46, Honolulu, HI, USA, May 2007. Springer.
45. A. Omicini, A. Ricci, and M. Viroli. Artifacts in the A&A meta-model for multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 17(3):432–456, 2008.
46. A. Omicini, A. Ricci, M. Viroli, C. Castelfranchi, and L. Tummolini. Coordination Artifacts: Environment-Based Coordination for Intelligent Agents. In *3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2004)*, pages 286–293, New York, USA, August 2004.
47. Singh M. P. An Ontology for Commitments in Multiagent Systems. *Artif. Intell. Law*, 7(1):97–113, 1999.
48. A. Passadore, C. Vecchiola, A. Grosso, and A. Boccalatte. Designing agent interactions with Pericles. In *2nd International Workshop on Ontology, Conceptualization and Epistemology for Software and System Engineering, ONTOSE 2007, Proceedings*, 2007.
49. A. Peña, H. Sossa, and F. Gutierrez. Web-services based ontology agent. In *2nd International Conference on Distributed Frameworks for Multimedia Applications, DFMA 2006, Proceedings*, pages 1–8, 2006.
50. M. Piunti and A. Ricci. Cognitive Use of Artifacts: Exploiting Relevant Information Residing in MAS Environments. In J.-J. Ch. Meyer and J. Broersen, editors, *Knowledge Representation for Agents and Multi-Agent Systems, First International Workshop, KRAMAS 2008*, volume 5605 of *Lecture Notes in Computer Science*, pages 114–129, Sydney, Australia, September 2008. Springer.
51. M. Piunti, A. Ricci, O. Boissier, and J. F. Hübner. Embodied Organisations in MAS Environments. In L. Braubach, W. van der Hoek, P. Petta, and A. Pokahr, editors,

*Multiagent System Technologies, 7th German Conference, MATES 2009*, volume 5774 of *Lecture Notes in Computer Science*, pages 115–127, Hamburg, Germany, September 2009. Springer.
52. M. Piunti, A. Ricci, O. Boissier, and J. F Hübner. Embodying Organisations in Multi-agent Work Environments. In *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT*, pages 511–518. IEEE, 2009.
53. M. Piunti, A. Ricci, L. Braubach, and A. Pokahr. Goal-Directed Interactions in Artifact-Based MAS: Jadex Agents Playing in CARTAGO Environments. In *IAT*, pages 207–213. IEEE, 2008.
54. S. K. Rajamani and J. Rehof. Conformance Checking for Models of Asynchronous Message Passing Software. In *Computer Aided Verification CAV'02*, number 2404 in LNCS, pages 166–179. Springer, 2002.
55. A. S. Rao. AgentSpeak(L): BDI Agents Speak Out in a Logical Computable Language. In W. Van de Velde and J. W. Perram, editors, *MAAMAW*, volume 1038 of *Lecture Notes in Computer Science*, pages 42–55. Springer, 1996.
56. A. Ricci, A. Omicini, M. Viroli, L. Gardelli, and E. Oliva. Cognitive stigmergy: Towards a framework based on agents and artifacts. In Weyns et al. [70], pages 124–140.
57. A. Ricci, M. Piunti, D. L. Acay, R. H. Bordini, J. F. Hübner, and M. Dastani. Integrating heterogeneous agent programming platforms within artifact-based environments. In L. Padgham, D. C. Parkes, J. Müller, and S. Parsons, editors, *AAMAS (1)*, pages 225–232. IFAAMAS, 2008.
58. A. Ricci, M. Piunti, and M. Viroli. Environment Programming in MAS – An Artifact-Based Perspective. *Autonomous Agents and Multi-Agent Systems*. To appear.
59. A. Ricci, M. Piunti, M. Viroli, and A. Omicini. Environment Programming in CArtAgO. In *Multi-Agent Programming II: Languages, Platforms and Applications, Multiagent Systems, Artificial Societies, and Simulated Organizations*. 2009.
60. A. Ricci, M. Viroli, and A. Omicini. The A&A programming model and technology for developing agent environments in mas. In M. Mehdi Dastani, A. El Fallah-Seghrouchni, A. Ricci, and M. Winikoff, editors, *PROMAS*, volume 4908 of *Lecture Notes in Computer Science*, pages 89–106. Springer, 2007.
61. Y. Shoham. Agent-Oriented Programming. *Artif. Intell.*, 60(1):51–92, 1993.
62. M. P. Singh. A social semantics for agent communication languages. In F. Dignum and M. Greaves, editors, *Issues in Agent Communication*, volume 1916 of *Lecture Notes in Computer Science*, pages 31–45. Springer, 2000.
63. M. S. Singh and A. K. Chopra. Correctness Properties for Multiagent Systems. In M. Baldoni, J. Bentahar, M. B. van Riemsdijk, and J. Lloyd, editors, *DALT*, volume 5948 of *Lecture Notes in Computer Science*, pages 192–207. Springer, 2009.
64. H. Suguri, E. Kodama, M. Miyazaki, H. Nunokawa, and S. Noguchi. Implementation of FIPA Ontology Service. In *Workshop on Ontologies in Agent Systems, OAS'01, Proceedings*, 2001.
65. C. Vecchiola, A. Grosso, and A. Boccalatte. AgentService: a framework to develop distributed multi-agent systems. *Int. J. Agent-Oriented Software Engineering*, 2(3):290 – 323, 2008.
66. M. Venkatraman and M. P. Singh. Verifying compliance with commitment protocols. *Autonomous Agents and Multi-Agent Systems*, 2(3), 1999.
67. M. Verdicchio and M. Colombetti. Communication Languages for Multiagent Systems. *Computational Intelligence*, 25(2):136–159, 2009.

68. M. Viroli, T. Holvoet, A. Ricci, K. Schelfthout, and F. Zambonelli. Infrastructures for the environment of multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 14(1):49–60, 2007.
69. D. Weyns, A. Omicini, and J. Odell. Environment as a first class abstraction in multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 14(1):5–30, 2007.
70. D. Weyns, H. Van Dyke Parunak, and F. Michel, editors. *Environments for Multi-Agent Systems III, Third International Workshop, E4MAS 2006, Hakodate, Japan, May 8, 2006, Selected Revised and Invited Papers*, volume 4389 of *Lecture Notes in Computer Science*. Springer, 2007.
71. P. Yolum and M. P. Singh. Commitment machines. In *Proc. of ATAL*, pages 235–247, 2001.
72. F. Zambonelli, N. R. Jennings, and M. Wooldridge. Developing multiagent systems: The Gaia methodology. *ACM Trans. Softw. Eng. Methodol.*, 12(3):317–370, 2003.