Auton Agent Multi-Agent Syst (2008) 17:432–456
DOI 10.1007/s10458-008-9053-x

Artifacts in the A&A meta-model for multi-agent systems

Andrea Omicini · Alessandro Ricci · Mirko Viroli

Published online: 15 May 2008

Abstract In this article we focus on the notion of *artifact* for agents in multi-agent systems (MAS) as a basis for a new meta-model promoting the modelling and engineering of agent societies and MAS environment as first-class entities. Its conceptual foundations lay upon theories and results coming from computational sciences as well as from organisational and cognitive sciences, psychology, computer supported cooperative work (CSCW), anthropology and ethology. In the resulting agents & artifacts (A&A) meta-model, agents are the (pro-)active entities in charge of the goals/tasks that altogether build up the whole MAS behaviour, whereas artifacts are the reactive entities providing the services and functions that make individual agents work together in a MAS, and that shape agent environment according to the MAS needs. After presenting the scientific background, we define the notions of artifact in the A&A meta-model, discuss how it affects the notion of intelligence in MAS, and show its application to a number of agent-related research fields.

Keywords Artifact · A&A meta-model · MAS environment · Activity theory · Distributed cognition · Coordination · Agent intelligence · Agent-oriented software engineering (AOSE) · MAS infrastructure

1 Introduction

Agents are not alone in multi-agent systems (MAS). They interact with other agents, and with the surrounding environment as well: a shared view exists in the literature that *agent*, *society*, and *environment* can be taken as the three basic categories to interpret and model

A. Omicini · A. Ricci · M. Viroli (✉)
Alma Mater Studiorum – Università di Bologna, Cesena, Italy
e-mail: mirko.viroli@unibo.it

A. Omicini
e-mail: andrea.omicini@unibo.it

A. Ricci
e-mail: a.ricci@unibo.it

the structure and dynamics of non-trivial MAS. However, whereas the study of the direct intercourse between agents appears as quite a well-explored subject in MAS literature, grounded on solid philosophical foundations like Searle's philosophy of human language [61], the study of interaction within agent societies and of agent interaction with and through the environment apparently still lacks a well-grounded conceptual foundation.

Human activities within complex organisations cannot be accounted for by simply modelling them in terms of utterances. Instead, an articulated landscape of objects, instruments and tools both enables and constrains our course of actions within human societies. Such spatially-distributed entities are either the means or the targets of our practical reasoning, and deeply affect our cognitive processes. In the same way, agents live in complex environments populated by non-agent entities to be used for the agent's purposes: knowledge sources like databases or web pages, resources modelled as web services, physical tools like sensors or actuators, and so on. Nonetheless, no high-level abstractions capturing the whole articulation of MAS environment at the suitable level of generality are usually considered when laying the foundations of agent-based systems.

When passing the strict boundaries of computational sciences, however, general notions like tool and artifact are instead ubiquitous means to enable the understanding of social activities, the representation and construction of structured environments, and the modelling of non-trivial cognitive processes. Liberally taking from the vast literature on the subject, in this article we define a notion of *artifact* for MAS, and explore its consequences in the context of agent-based research and technologies. While its conceptual foundations lay upon theories and results coming from research in computational sciences, the notion of artifact defined here draws most of its inspiration from heterogeneous sources such as organisational sciences, cognitive sciences, psychology, CSCW, anthropology, and ethology.

According to this new perspective, MAS are modelled and engineered based on two fundamental computational abstractions, *agents* and *artifacts*. Agents are the autonomous, (pro-)active entities that encapsulate control, and are in charge of the goals/tasks that altogether define and determine the whole MAS behaviour. Artifacts are instead the passive, reactive entities in charge of the services and functions that make individual agents work together in a MAS, and shape agent environment according to the MAS needs. The resulting agents & artifacts (A&A) meta-model promotes the modelling and engineering of agent societies and MAS environment as first-class entities.

The article is structured as follows. First of all, Sect. 2 presents the multi-disciplinary background for the A&A meta-model, by showing how the notion of artifact emerges independently within many heterogeneous areas of research, from Activity Theory to Distributed Cognition, from sociology to CSCW, from anthropology to ethology. Section 3 introduces the A&A meta-model, by defining its basic abstractions and discussing the main features for agents and artifacts. Since artifacts change the way in which agents act and interact within a MAS, they also potentially change the way in which agents reason about actions: accordingly, Sect. 4 discusses how the notion of artifact affects the notion of agent intelligence in MAS, and introduces a more articulated notion of artifact—namely, *cognitional artifact*—aimed at promoting cognitive use of artifacts by agents. In order to demonstrate the effectiveness of the notion of artifact and of the A&A meta-model as well, Sect. 5 describes how artifacts impact on many key areas of agent-based research and technology, ranging from agent-oriented software engineering to agent-based technologies, from agent-based simulation to self-organisation, also including fields such as electronic institutions, collaboration support systems, and MAS argumentation.

2 Artifacts: multi-disciplinary background

The notion of complex system crosses the boundaries between many different scientific disciplines, ranging from physics to biology, from economics to sociology and organisation sciences—and obviously affects computational systems, too. The multi-disciplinary interest about such a notion has led to recognise that there exist some "Laws of Complexity" that characterise any complex system, independently of its specific nature [23]. No matter if we are modelling the behaviour of a human organisation, the life of an intricate ecosystem, or the dynamics of a huge market-place: we can expect to find some repeated patterns, some shared schema, some common laws that make all these systems look similar when observed at the right level of abstraction.

Accordingly, when focussing on MAS, adopting a multi-disciplinary view in order to understand their complexity is almost mandatory rather than merely useful: MAS, in fact, potentially exhibit all the typical features of complex systems, such as openness, situatedness, locality in control and interaction. In this perspective, when trying to re-define the conceptual basis of the MAS paradigm, it seems proper to first look at some of the many scientific research fields dealing with complex systems that are in any way relevant to MAS, then possibly draw some lessons to be brought to the MAS field. In the remainder of this section, we observe the pervasiveness of the notion of artifact through a number of heterogeneous research areas, and gather their results so as to bring some instructive lessons to the MAS field.

2.1 Artifacts in activity theory

Activity Theory (AT) is a social psychological theory born in the context of Soviet Psychology [67], nowadays widely applied also in the context of fields related to computer science, such as Computer Supported Cooperative Work (CSCW) and Human Computer Interaction (HCI)—see [30] for a survey. Broadly speaking, AT is a very general framework for conceptualising human activities—how people learn, how society evolves—based on the concept of human *activity* as the fundamental unit of analysis.

According to AT, any activity carried on by one or more participants of an organisation cannot be understood without considering the tools or *artifacts* that enable actions and mediate interactions of the components. On the one side, artifacts mediate the interaction among individuals, as well as between individuals and their *environment*; on the other side, artifacts embody the part of the environment that can be designed and controlled to support participants' activities.

So, the notion of artifact is central to AT, as a mediator for any sort of interaction in human activities. Artifacts can be either physical, such as shelves, doors, phones, and whiteboards, or cognitive, such as operating procedures, heuristics, scripts, individual and collective experiences, and languages—or, they can embody a twofold nature of both physical and cognitive artifacts, as operating manuals or computers. As mediating tools, they have both an *enabling* and a *constraining* function: on the one hand, artifacts expand out possibilities to manipulate and transform different objects, but on the other hand such objects are perceived and manipulated not 'as such', but within the limitations set by the tool.

AT focuses on social (collaborative) activities, and models them according to a three-layered structure: they can be *co-ordinated*, *co-operative*, and *co-constructive* [3,17].

- The *co-ordinated* aspect of work captures the normal and routine flow of interaction within a system, where both the means and the goals of collaborative activities are given and stable;

- The *co-operative* aspect of work concerns the mode of interactions in which actors focus on a common object and thus share the objective of the activity: in this case the object of the activity is stable and agreed upon, whereas the means for realising the activity is not yet defined;
- The *co-constructive* aspect of work concerns interactions in which actors focus on re-conceptualising their own organisation and interaction in relation to their shared objects: neither the object of work, nor the means are stable, and must be (collectively) constructed, i.e., *co*-constructed.

2.1.1 Activity theory: lessons learned

Adopting AT as a conceptual framework for MAS social activities leads to the fundamental recognition that agents are not the only basic abstractions to think and build MAS [49]. Artifacts, too, are necessary to enable and constrain agent actions, to mediate agent interactions with other agents and with the environment, to model and shape MAS *environment*, and more generally to improve agent ability to achieve their individual and social goals [54]. The three levels identified by AT for social activities can be re-interpreted in the MAS context in terms of the relationship between agents and artifacts, in particular:

co-construction—agents understand and reason about the (social) objectives (goals) of the MAS, and build up a model of the social tasks required to achieve them. This also involves identifying interdependencies and interactions to be faced and managed.
co-operation—agents design and build the artifacts which are useful to carry on the social tasks and to manage the interdependencies and interactions devised out at the previous (co-construction) stage.
co-ordination—agents use the artifacts: then, the activities meant at managing interdependencies and interactions—either designed a-priori or planned at the co-operation stage—are enforced/automated.

As developed by Ricci et al. [49] then, activity theory directly promotes the notion of *coordination artifact* to identify artifacts that are used in the context of collaborative activities in particular, mediating the interaction among actors involved in the same social context. Along this line, coordination artifacts represent a straightforward generalisation of the notion of *coordination medium* as coming from fields like coordination models and languages and distributed AI—including abstractions like tuple spaces, channels, blackboards and the alike. More general examples range from coordination abstractions such as tuple centres [38], to pheromone infrastructure [46] in the context of stigmergy coordination, to the Institution abstraction in e-institution approaches [32], to cite some.

2.2 Artifacts in distributed cognition

A vision of the same sort is promoted by Distributed Cognition [24], a branch of cognitive sciences which proposes that human cognition and knowledge representation, rather than solely confined within individuals, are distributed across individuals, tools and artifacts in the environment.

According to Distributed Cognition, intelligent processes in human activity go beyond the boundaries of individual actors, and knowledge is not confined within human minds—there is cognition that transcends individual cognition. In particular, it is recognised that knowledge representation does not pertain individual humans only: instead, representation is distributed, partially in the mental spaces of humans, partially as external representations of memories, facts, and information of any sort distributed on the objects, tools and instruments that constitute the environment.

So, the analysis of Distributed Cognition focuses on distributed cognitive systems, where people interact with external *cognitive artifacts*. There, human intelligent behaviour results from the distributed interactions with other humans and with cognitive artifacts that in the overall determines the context where human activities are situated—the physical, cultural and social context that also guides, contrains and partially determines intelligent activities. According to Norman [33], cognitive artifacts are "those artificial devices that maintain, display, or operate upon information in order to serve a representational function and that affect human cognitive performance".

Cognitive artifacts are then a product of human design and work, aimed at aiding or enhancing our cognitive abilities—like post-its, calendars, agendas, computers, etc. However, cognitive artifacts do not merely amplify our cognitive abilities: they also modify the nature of the tasks to be performed. For instance, using an address book artificially amplifies my natural ability to correctly remember my friends' phone numbers, but it also changes the way in which I store and retrieve such data.

Along this line, two ways to look at distributed cognitive systems are possible. According to the *system view*, individuals and artifacts can be taken altogether as units of observation to understand activities and actions: e.g., one may observe myself plus my address book altogether to understand my activity "call a friend". According to the *personal view*, individuals can be taken as units of observation to understand how artifacts affect them—their practical reasoning, the way in which they represent actions, plans, expectations, intentions, etc.: e.g., one may observe how the use of an address book affects my way to reason and act about calling friends.

The focus on "external" cognition obviously stresses the key role of environment in distributed cognitive systems. According to Kirsh [24], the nature of the environment—which depends on the artifacts and tools that shape it—determines the effectiveness of the work and activities of the actors that are immersed in it. *Work environments* in human organisations are a complex superposition of social, cultural, cognitive and physical constraints, and their analysis and synthesis are non-trivial issues indeed. On the one hand, in fact, how to model a work environment and define its boundaries and structure mostly depends on the tasks that have to be carried on inside it. On the other hand, designing a work environment requires a deep understanding of the individual/social tasks that have to be carried on, of the individuals in charge of the activities and of their corresponding cognitive processes, of the social interactions, and so on.

It should be noticed that the purpose of an activity is not merely to change the environment in a way that (presumably) leads to goal satisfaction, as typically assumed in the literature. Many actions in fact do not make sense under this trivial assumption: when I take note of my friend's phone number on my address book, I am not trying to immediately satisfy some goal of mine. In general, people undertake actions to save attention, memory and computation; people recruit external elements to reduce their own cognitive effort by distributing computational load; and so on. This only makes sense if people and their activities are conceived as *situated*, that is, as strongly coupled with their environment. As a result, environment design should not merely be aimed at helping people to achieve their goals, but also to make other actions easy—such as epistemic, complementary, coordinative actions.

2.2.1 Distributed cognition: lessons learned

Re-casting results from Distributed Cognition to the MAS field leads first of all to recognise that cognition and knowledge representation do not belong to agents only: cognitive processes exist in MAS that do not belong to individual agents. The MAS environment may participate to cognitive processes, by enabling and mediating individual agent actions as well as social agent interaction, through the knowledge it embeds either implicitly or explicitly. So, cognition and knowledge representation are generally distributed in MAS environment, and affect any cognitive process within a MAS. Hence, artifacts are essential parts of the cognitive processes within a MAS: most specifically, cognitive artifacts (like databases, ontologies, repositories, etc.) encapsulate knowledge as explicitly represented, and made available to cognitive agents.

The personal vs. system view of Distributed Cognition suggests a twofold way to look at the relationship between agents and artifacts. The *agent view* is the individual agent view over artifacts (and actions) in a MAS: once artifacts are exploited, they change the way in which agents act and reason about action. The *MAS view* is the social/global view about agent action and interaction in a MAS: in order to understand and possibly evaluate agent (social) action within a MAS, one should consider agents and artifacts altogether, as functional subsystems.

According to Distributed Cognition, MAS environment is structured: (i) artifacts shape MAS environment, (ii) knowledge is distributed in the environment and is encapsulated within cognitive artifacts, and (iii) the structure of the environment, and the knowledge it contains, affect the activities of agents within MAS. Moreover, artifacts should be designed not only so as to help agent actions to achieve their goals, but also to make epistemic, complementary, coordinative agent actions easier/effective.

2.3 Artifacts in sociology

By considering the conceptual framework described by Conte and Castelfranchi [10], individuals in a social system can be generally conceived as *goal-governed* or *goal-oriented* (sub)systems. Goal-governed systems exhibit forms of cognitive capabilities, explicitly representing their goals, and driving the selection of agent actions. Goal-oriented systems are directly designed and programmed to achieve some goal, which is not explicitly represented.

In both goal-governed and goal-oriented systems, goals are *internal*. *External goals* refer instead to goals that typically belong to the social context or environment where the agents are situated. External goals are sorts of regulatory states that condition agent behaviour: a goal-governed system follows external goals by adjusting internal ones.

The main point here is that not all the entities involved in (social) action have internal goals, or should be modelled as such. This is the case of passive systems—often called *objects* by Conte and Castelfranchi [10]—which are typically characterised by the concept of *use*: that is, they have no internal goals, but can in turn be exploited by goal-driven entities to achieve their own goals. Whenever changed/built by humans, objects are explicitly modified/designed to provide a certain *function* that guides their use. The function of an object is an *external* goal that influences by design its structure and behaviour, depending on the intended use(s) the designer has envisioned for the object itself.

The concept of *destination* is related but not identical to the concept of function: destination is again an external goal, attached to an object by a user in the act of using it—rather than by the designer changing/creating the object—and driving the actual usage of the object itself. Then, an object could be used according to a destination which is different from its function. Different sorts of external goals are associated by a user to an object:

– The *use-value* goal, according to which the object has the potential to allow a user to achieve its objective, and which drives the object's *selection*
– The *use* goal, which directly corresponds to the user's internal goal, and guides the actual *usage* of the object

Therefore, an object is first adapted or created with some function in mind, thus destined to a future use. Then, an object is selected for future use by a user when it has a use-value goal that somehow matches the user's internal goal. When it is finally used, the object's destination may or may not match its pre-designed function, however it has a use goal that essentially matches the user's internal goal.

2.3.1 Sociology: *lessons learned*

Drawing from the literature in sociology, the notion of goal and its diverse acceptations can be fruitfully used to characterise artifacts in MAS, too. In this perspective, MAS are built out of two sorts of entities: *agents*, as goal-oriented/goal-governed systems, and *artifacts*, as systems with no internal goals. Agents have goals, whose representation defines the notion of agency: *strong* agents are goal-governed entities with explicitly-represented goals, whereas *weak* agents are goal-oriented entities with implicitly-represented/encoded goals.

Artifacts have no internal goals, but are instead characterised by a *function* which is determined at design time, depending on the possible uses of the artifacts envisioned by the artifact designer. When used by an agent, an artifact is associated with another external goal—its destination.

The relationship between agents and artifacts can then take three distinct forms: *use*, *selection*, and *construction*. When using an artifact, an agent is driven by the artifact use goal, and actually associates an artifact to a destination that matches one of the agent's internal goals, and may be different from the artifact's intended function. When selecting an artifact for future use, an agent has to envision its possible destinations by considering the artifact use-value. Finally, when constructing an artifact—either by modifying an existing one, or by creating a new one—an agent should envision its possible future uses, and design its structure and behaviour accordingly.

So, agents associate different sorts of external goals to an artifact: the use-value, which drives the agent's selection of the artifact, and the use goal, which guides the actual usage of the artifact by the agent. Construction is instead the obvious rational consequence of a failure in the artifact selection process, or in the use of a selected artifact: then, a new artifact should be created, or obtained by manipulation of an existing one, so as to allow the agent to achieve its own internal goals.

2.4 Artifacts in CSCW

CSCW is concerned with the way in which computational technologies and artifacts enable and promote people collaborative activities. According to Schmidt and Simone [59], "CSCW seems to be pursuing two diverging strategies", which lead to two distinct trends in CSCW research: the first, which roughly includes approaches coming from Workflow Management Systems (WfMS), tends to privilege automatisation of coordination; the second, which mostly refers to classical contributions from CSCW, takes flexibility of interaction as its main goal. In other terms, the former approach stresses the role of computational entities prescribing the rules of collaboration (like workflow engines), the latter mostly accounts for the intelligent coordination capabilities of collaborative entities (like humans, or intelligent agents).

So, the gap is both between two strategies, and between two research lines—requiring a convergence, as the authors clearly acknowledge. Schmidt and Simone [59] point out two key issues in the solution to the problem: *mutual awareness* and *coordinative artifacts*.

Mutual awareness means that the actors of a collaboration activity affect and mutually perceive the other actor's activities through the *common field of work*—the shared workspace—which can reveal/conceal portions of the collaboration activities to the participants. Mutual awareness is then the basis for opportunistic, *ad hoc* alignment and improvisation, which ensure flexibility to collaborative activities.

Coordinative artifacts are instead the rulers of collaboration, acting more as *constrainers* rather than as *commanders*. By giving structure to the common field of work, coordinative artifacts encapsulate those portions of the coordination responsibilities that are better to be automatised in order to achieve efficiency in cooperation. So, on the one hand coordinative artifacts define and govern the space of the admissible articulation of activities (they work then as constrainers), on the other hand they do not impose a pre-defined course of actions that could cause unnecessary rigidity and reduce the required flexibility (they do not work as commanders).

2.4.1 *CSCW*: *lessons learned*

CSCW provides us with guidelines on how artifacts can be fruitfully exploited in MAS, promoting *automation* and *flexibility* of collaboration activities.

Automation should be delegated to coordinative artifacts—*coordination artifacts*, in the typical MAS terminology [44]. Coordinative artifacts rule MAS collaboration, and should work more as constrainers rather than as commanders. So, coordinative artifacts frame and give structure to the MAS common field of work, as specialised abstractions automatising collaboration, and making it efficient. As constrainers, then, coordinative artifacts define and govern the space of the admissible articulation of MAS collaboration activities. On the other hand, they do not impose a pre-defined course of actions over agents of a MAS, so do not prevent flexibility of intelligent agent coordination, and respect of agent autonomy.

Flexibility is instead promoted by mutual awareness. Shared MAS environment should be structured as the MAS *common field of work* in order to allow agents to mutually perceive each other's activities: MAS common field of work can reveal/conceal portions of MAS collaboration activities to the agents. The ability to observe MAS collaboration enable opportunistic alignment and improvisation of agent activities, and ensure flexibility to MAS collaboration activities.

2.5 Artifacts in anthropology & ethology

Western anthropology has long dwelt with the idea that the use of symbolic language was the main sign of human intelligence. According to Hewes [22], this has produced a deep *logocentric philosophical bias* on the part of Western scholars, which has led to largely neglect the relation between use of tools and intelligence—in particular, it is argued that the tool-making and tool-using faculty has been taken for granted as an expectable attribute, and therefore requiring less scientific examination. Hewes observes how, after many decades of activities by researchers from many diverse but related areas (such as biological and social anthropology, archaeology, linguistics, psychology, neurology, and ethology), only in very recent times the issue of the relation between language, tools and the evolution of human cognitive abilities has been faced as a single, coherent problem—see for instance Gibson and Ingold [20]. Nowadays, it is largely acknowledged that human capacity of developing and using tools is a fundamental sign of intelligence, at least as much as the use of language. The first characterisation of *Homo Abilis* is in fact its ability to use and forge tools. This is a clear sign of intelligence: and evidence of co-evolution of language and tools usage along with human intelligence is now overwhelming in modern anthropological studies [20].

Tools are not an exclusive feature of humans: beavers build dams, bees construct perfect hexagonal cells, many birds live in self-made nests. However, what is often taken as a distinguishing feature of human tools with respect to other animals' ones is the cognitive level at which the tools are conceived, designed, and used: apparently, tools are not part of human "embedded" behaviour, as in bees or birds, but they are rather the result of the rational elaboration about the relationship between the human being and his habitat—his living environment. Also, systematic and social design and use of tools is seemingly typical of the human species, and is often taken as a measure of human against animal intelligence.

More generally, our understanding of the strict relationship between tools and intelligence (human and not human) is such that we typically interpret tool-using and tool-making faculty as a fundamental revealing symptom of intelligence. For instance, ethologists commonly measure intelligence of animals by making them face problems that require the use of tools to be solved—see for instance Povinelli [47]. Even more interestingly, a sort of tool-equivalent of the Turing test has been proposed by philosopher Ronald Endicott, which was aimed at evaluating intelligence in terms of the ability to exploit tools—the so-called "Tooling Test for Intelligence" [70].

A tool, according to Martelet [26], reveals the user awareness of self or/and of the world, whenever it is built with a goal, it is stored for further/repeated use, it is used for building new tools. Tools are at the same time the first and most distinctive expression of human intelligence, along with language; and also, the most powerful amplifiers of the (both individual and social) human ability to affect the environment—to survive environment change, first, and to change the environment for the human purposes, then.

2.5.1 Anthropology & ethology: lessons learned

The hype toward language we observed in Anthropology and Ethology is today still quite evident in the MAS field. Apart from the overwhelming number of agent papers on communication, speech acts, and other language-related issues published in the last years by agent-related conferences and journals, a striking evidence comes for instance from the work by the only agent standardisation body, FIPA—for many years mostly dealing with agent *communication* actions, in spite of its being the Foundation for Intelligent *Physical* Agents. In general, when looking at agent systems from the viewpoints of anthropology and ethology, and at agent intelligence in particular, it turns to be awkwardly strange that most of the work till now has elaborated on linguistic concepts and acts—as in the agent-pervasive theory of speech acts—and has almost ignored, at least explicitly, the matter of agent tools.

So, in order to avoid (further) pernicious effects of the same "logocentric bias" in the MAS field, some shared conceptual view is required where agent actions could be framed in their most general acceptation, actually accounting for both communicative and physical actions. A notion of agent *tool*, or artifact, is then required, which could allow a theory of agent physical action to be developed at the same level of refinement as the theory of agent communicative actions. Such a theory should finally allow for a broader definition of agent intelligence, including not only the use of symbolic languages, but also the use of tools. To this end, an agent-oriented "Tooling Test for Intelligence" could be defined, aimed at measuring agent intelligence in terms of the agent's ability to exploit tools [70].

3 Agents & artifacts: basic concepts

Based on the conceptual foundations discussed in the previous section, the A&A meta-model is characterised in terms of three basic abstractions:

- *Agents*, to represent pro-active components of the systems, encapsulating the autonomous execution of some kind of activities inside some sort of environment;
- *Artifacts*, to represent passive components of the systems such as resources and media that are intentionally constructed, shared, manipulated and used by agents to support their activities, either cooperatively or competitively;
- *Workspaces*, as conceptual containers of agents and artifacts, useful for defining the topology for the environment and providing a way to define a notion of locality.

In this article, we mostly focus on the notion of artifact—the notion of workspace is at an early stage of investigation, and is not strictly necessary to provide a foundation for the artifact notion. So, in the remainder of this section we define the notions of agent and artifact as they are conceived in the A&A meta-model.

Before we proceed, it should be noted that the following are just *stipulative* definitions, rather than *lexical* ones: that is, they do not aim in any way at re-writing the notions of agent or artifact as they are generally understood in the common and in the technical language as well, but rather at providing the precise acceptations of the terms in the context of the A&A meta-model and, more generally, of our research. According to that, in the remainder of this section we define the notions of agent (Sect. 3.1), artifact (Sect. 3.2) and MAS (Sect. 3.3) as they are conceived in the A&A meta-model. In particular, the characterisation of agent that we provide should not be taken as yet another definition overcoming/adding to the many existing ones: instead, our aim here is just to point out the essential features of agency that are required in order to ground the A&A meta-model, especially as far as the relationship between agents and artifacts is concerned.

As a last preliminary annotation, in the following we adopt the classical approach where scientific definitions are expressed in terms of Genus and Differentia [11,71]. In short, a definition by Genus and Differentia should clearly delimit the domain of discourse (*genus*) and allow what is in and what is out to be clearly determined (*differentia*), while following some basic rules, such as: a definition should be essential, contain no circularity, be neither too wide nor too narrow, contain no obscurity, as well as no unnecessary "negation". While the approach might appear slightly too structured for the simple definitions in the remainder of this section, the need for clarity and simplicity was actually the main motivation behind this choice.

3.1 On the notion of agent in the A&A meta-model

Firstly and mostly, the A&A meta-model recognises *autonomy* as the fundamental defining feature for agents.

From a computational viewpoint, autonomy means that agents encapsulate (the thread of) *control*. So, control does not flow through agent boundaries: agents never give up control, nor are they controlled by anything—unless they deliberate to do so, of course. Correspondingly, agents provide no interfaces for being used, and cannot be invoked or controlled—agents can say "no", according to Odell [34]. Only data (information, knowledge) crosses agent boundaries. As a result, MAS are to be viewed as a multiplicity of distinct *loci* of control, interacting with each other by exchanging information.

Furthermore, this means that agents are the main forces governing and pushing computation in a MAS: so, an agent should also encapsulate the *criterion* for regulating the thread of control—which is indeed the very root of the word "autonomy": that is, self-government, self-regulation, self-determination. So, while this does not directly imply that agents must have a goal, or a task, to be such—literally, an agent just needs to be self-driven—goals and tasks precisely play the role of the criteria for driving control inside each agent.

From the very notion of autonomy also *agent sociality* stems. In fact, from a philosophical viewpoint, autonomy only makes sense when an individual is immersed in a society—no individual alone could be properly said to be autonomous, since the term makes no sense for an individual in isolation. Interestingly enough, this also straightforwardly explains the well-known distinction between a simple program in any sequential language and an agent—possibly, more directly than Franklin and Graesser [18], and in line with Odell [34].

Literally, the etymology of the word "agent"—from Latin "agens"—means "the one who acts". So, autonomous agents are intrinsically active entities: more precisely, they are *pro-active*, since they encapsulate control, and self-govern their own course of action—where pro-activity simply means "making something happen", rather than waiting for something to happen.

As a further consequence, the agent concept should by definition come equipped with a notion of agent *action*—that is, a conceptual model for agent actions is needed to provide a coherent notion of agent. Whatever the model, any notion of action is intrinsically connected to the notion of *change*: an agent acts in order to change something. In the context of a MAS, an agent action has two potential targets: either another agent, or the world around the agents of the MAS—the *environment*, as it is usually called [65,68]. In other words, an agent action could aim at "changing" either another agent or the MAS environment—that is, the conceptual space where agents of a MAS live and interact. Since agents are autonomous, and only data flows among them, the only way to directly affect another agent state is to provide it with some information—so, this is what is usually denoted as *speech act*, or *communication action*. Instead, change to the MAS environment is to be more easily thought as the result of *physical actions*—even though their physical nature in virtual environments could be questionable, indeed.

Finally, any "ground" model of action is strictly coupled with the context where the action takes place: so, the model of action depends on the context where agents act. In this sense, autonomous agents are essentially *situated* entities, since any agent is strictly coupled with the environment where it lives and (inter)acts.

Other "fundamental" features have often been attributed to agents, like intelligence, mobility, or the ability to learn. It is quite easy to see that a mobile agent has a wider space of admissible actions than a non-mobile one, that an intelligent agent is likely to have a more articulated and effective practical reasoning than a non-intelligent one, and that a learning agent may well improve its deliberation and planning capabilities. However, despite being (often) desirable ones, properties of that kind are not essential to agents in A&A, where they are not considered as defining features for agents.

In the end, an agent is simply defined as follows in the A&A meta-model:

Definition 3.1 (*A&A Agent*) An A&A agent is an *autonomous computational entity*.

(**genus**) Agents are computational entities
(**differentia**) Agents are autonomous, in that they encapsulate control along with a criterion to govern it

3.2 On the notion of artifact in the A&A meta-model

Given the above definition for the notion of agent, and the discussion of the pervasive notion of artifact presented in Sect. 2, the definition of the notion of artifact in the A&A meta-model is quite straightforward:

Definition 3.2 (*A&A Artifact*) An A&A artifact is a *computational entity* aimed at the *use* by A&A agents.

(**genus**) Artifacts are computational entities
(**differentia**) Artifacts are aimed to be used by agents

So, as their first characterisation, artifacts are *to be used* by agents. From use, many other features stem—which are either essential or desirable, but need not be used as definitory ones.

First of all, artifacts are *not* autonomous—unlike agents. Since they are designed to *serve* some agent's purpose, artifacts are not to follow their own course of action. An artifact is a tool in the "hands" of agents, and does not need to be self-governed. Instead, artifacts are governed by agent's use: as such, artifacts are (computationally) reactive, that is, they are reactive in terms of control. Artifacts behave in response to agent use, and their behaviour just needs to emerge when they are used by an agent.

Then, every artifact has a *function*. Artifacts are designed for use: being aimed at the agent's use, artifacts are designed to serve some purpose, and built as such. At the time of their design, they are then associated to their *function* by the artifact designer. Artifact function does not necessarily determine the actual use of the artifact by an agent: however, it incorporates the *aim* of the artifact designer, envisioning the artifact as potentially serving agent's purposes.

Since they are aimed at being used, artifacts are the primary target/means of agent's action. Also, their function is expressed in terms of change to the environment—that is, what the artifact actually does when used by an agent. So, the artifact's model, structure and behaviour are expressed in terms of agent's actions, and their effects on their environment—which makes artifacts intrinsically *situated*. Finally, since they are situated, and also structurally reactive in computational terms, artifacts are easy to be thought as reactive to changes in the environment.

In order to be used, artifacts should make *operations* available to agents. Operations change an artifact's state, make it behave and produce the desired effects on the environment. So, either explicitly or implicitly, an artifact exhibits its *interface*, as the collection of the operations they made available to agents for use.

Finally, in order to promote their use by intelligent agents, artifact function should be available to agents, and understandable by them. Also, artifact behaviour should be predictable, so that agents could effectively exploit artifacts in their practical reasoning, and actually improve their ability to act successfully. So, transparency and predictability of behaviour are indeed desirable features of A&A artifacts. Hence, Sect. 4 further elaborates on the desirable features of artifacts in the A&A meta-model, focussing on the cognitive use of artifacts by intelligent agents.

3.3 On the notion of MAS in the A&A meta-model

It is now straightforward to define the notion of MAS in the A&A meta-model in a constructive way, based on the previous definitions.

Definition 3.3 (*A&A MAS*) An A&A MAS is a *computational system* made of agents and artifacts

(**genus**) MAS are computational systems
(**differentia**) MAS basic components are agents and artifacts

This definition is not fully complete, since it lacks the essential concept of workspace: nonetheless, it is as complete as needed by this article, where our focus is on the notion of artifact in MAS, and on the agent-artifact relation as well. In the overall, the behaviour of A&A MAS results from the interaction of autonomous, self-governing entities (agents) and reactive, functional entities (artifacts). Since both agents & artifacts are situated computational entities, A&A MAS are *situated computational systems*: a MAS is always immersed within an environment, and cannot be conceived/modelled/designed in a separate way with respect to its environment.

The space of admissible interactions within a MAS in the A&A meta-model is generated by the four sorts of interaction potentially occurring between the two fundamental entities composing MAS:

communication agents *speak* with agents
operation agents *use* artifacts
composition artifacts *link* with artifacts
presentation artifacts *manifest* to agents

As a result, interaction in the A&A meta-model amounts to the four sorts above, plus the interaction with MAS environment—which, depending on the desired/required level of abstraction, we may attribute to either individual agents/artifacts, or to the MAS as a whole.

4 Cognitive use of artifacts

Once the notion of *use* has been introduced as the fundamental one in the definition of artifact and in the relation between agents and artifacts, the issue of *cognitive use* is possibly the most relevant one to be addressed. In short, the question to be answered is whether the simplest notion of artifact, as defined above, is enough to support and promote intelligent use by agents capable of cognitive processes—or, instead, a richer and more articulated definition would be required.

As discussed in Sect. 2.2 and 2.3, artifacts play a fundamental role in the way cognitive agents execute their tasks and achieve their goals. An agent equipped with cognitive (intelligent, rational) abilities can evaluate the available artifacts, select the artifact that mostly fits its needs, then use it so as to achieve its own goals more efficiently and effectively.

When artifacts enter the picture of MAS, agent's practical reasoning is deeply affected in all of its phases—deliberation, means-end reasoning, and execution. In the deliberation phase, the accessibility of a number of artifacts available to the agent's use typically extends the compass of the states-of-affairs that are actually reachable by the agent's activity. Then, the outcome of the means-ends reasoning phase clearly depends on the way in which the artifacts work, and how they can be used by agents. Finally, actual agent interactions with the artifacts, along with the structure of artifacts' operations, obviously impact on the execution phase—for instance, in the verification of the successful completion of agent's actions.

As a result, even though the notion of agent does not require cognition for agents to use artifacts—as discussed in Sect. 3.1—it is by analysing the consequences of the notion of artifact upon the agent cognitive process, and by elaborating on the opportunities that artifacts present to cognitive agents, that the true role of artifacts in MAS engineering can be well unfolded.

4.1 Cognitional artifacts

In order to be able to effectively reason about the possible outcomes of using an artifact, an agent should be somehow aware of the artifact's structure and behaviour, as well as on the way and the effects of using the artifact. To this end, knowledge about the artifact design, structure and behaviour should be injected in the artifact itself, so as to make it available to the agent inspection. Since the definition of artifact in the A&A meta-model as provided in Sect. 3.2—being targeted to the use by both cognitive and non-cognitive agents—does not include such knowledge, a richer notion of artifact is required, whose additional features could raise artifacts up to the cognitive level of agents.

Accordingly, in the following we introduce the notion of *cognitional artifact*, which specifically targets the construction of MAS where the interaction between agents and artifacts is driven by cognitive abilities.

Definition 4.1 (*A&A Cognitional Artifact*) An A&A *cognitional artifact* is an artifact aimed at the *cognitive use* by agents

(**genus**) Cognitional artifacts are artifacts
(**differentia**) Cognitional artifacts are aimed to be used in a cognitive way by A&A agents

By "cognitive use" we mean the usage of an artifact that is driven by the agent's cognition, that is, (i) by rationality—the agent acts in a compliant way with respect to the tasks it has to perform or the goals it should pursue—and (ii) by the awareness of (a) the existence and accessibility of the artifact, (b) the expected benefits of exploiting the artifact, and finally (c) the structure and behaviour of the artifact. To this end, the following properties should be defined for a cognitional artifact in the A&A meta-model:

Function description—the expected outcomes when using the artifact—namely, the function the artifact provides to agents, according to the artifact's designer intentions;
Operating instructions—the behaviour that should characterise the agent interactions with artifact—namely, the admissible procedures for using an artifact for a given purpose;
Usage interface—the details of the operations provided by the artifact—namely, the external, observable structure of the artifact.

4.2 Properties of cognitional artifacts

Function description, operating instructions, and usage interface can actually be understood in terms of three distinct levels of description of artifacts, orderly exploited by the agent: first to *select* the proper artifact, then to *plan* the exploitation of the selected artifact for a given use, finally to suitably *interact* with the artifact to execute an operation. In the remainder of this section, we discuss the three subsequent phases along with the relevant properties of cognitional artifacts.

4.2.1 Selecting an artifact by function description

In an A&A MAS, an agent lives in an environment populated by other agents and by artifacts: it is by acting within and on such an environment that tasks can be executed and goals can be achieved. In particular, since artifacts may serve different purposes, a cognitive agent should be able to fully understand all the opportunities offered by every artifact available, so as to property select the one(s) to be used. As discussed in Sect. 2.3, each artifact comes with a pre-designed *function*, which could be understood as the external goal that originally motivated its construction. This determines the so-called *use-value* of the artifact, which represents the main reason why an agent should select one artifact among a set of accessible artifacts.

Accordingly, in the A&A meta-model a cognitional artifact is equipped with a *function description* that defines the intended function that the artifact makes available to the agent's use. Essentially, function description contains information about the possible uses that the designer envisioned for the artifacts. The ways in which function description of a cognitional artifact could be actually provided to agents may vary, for instance in the syntax and semantics of the language adopted—the same obviously holds for operating instructions and usage interface. This is because the choice of the most suitable way to represent function description could be influenced by many factors, such as the given application scenario, the existence of a common ontology for all the agents, the degree of openness of the MAS, and the level of cognition of agents—possibly ranging from limited symbolic elaboration abilities to full-featured intelligent machinery. In fact, the same MAS could possibly host a number of artifacts with different forms (models, languages, technologies) for function description—for instance, whenever artifacts are re-used across different MAS.

For instance, the function description of a cognitional artifact could be provided in terms of either a list of goals that could be achieved by using the artifact, or a list of tasks that could be executed by means of the artifact. Also, such information might be coupled with pre-conditions of some sort on the agent mental state, which could be used to a-priori filter those agents that have no reasons to use the artifact. As an example, Rubino et al. [57] show how OWL-S can be used to inject the required features for a cognitional artifact, in particular, how it can be used to provide an artifact with its function description.

4.2.2 Exploiting an artifact by operating instructions

Once it has selected *which* artifact should be used, a cognitive agent should be aware of *how* it has to be exploited for the selected purpose. Generalising from our earlier work on coordination artifacts [44,66], we extend the notion of *operating instructions* to cognitional artifacts.

Similar to a manual for a physical device, operating instructions describe the behaviour that an agent should follow to meaningfully interact with the artifact—namely, to use it effectively according to the destination chosen by the agent, in other terms the selected use goal (see Sect. 2.3). Operating instructions are hence to be written in a language the agent can understand: the semantics of such a language precisely defines the admissible/required sequences of agent actions over the artifact, or equivalently, a set of synthetic plans for using the artifact in an effective way. The goal of operating instructions is then to assist agents when planning their course of actions. In short, once the artifacts to be used have been selected, their operating instructions are inspected and interiorised by the agent, and used for its planning activity. The outcome is a plan that is likely to incorporate some of the operations's sequences defined as admissible by the artifact's operating instructions.

Again, different languages can be used to this end. Languages for operating instructions could be fully operational, so aimed at describing the step-by-step behaviour, and typically indicating pre-conditions and effects of artifact operations. For instance, Viroli et al. [66] develop and exemplify an approach based on process algebras, and show how step-by-step instructions—as provided by operating instructions—could be helpful also for agents with limited intelligence. On the other hand, operating instructions could describe admissible behaviours through a more declarative approach, as discussed for instance by Rubino et al. [57]. Operating instructions could also be expressed by adopting a logic-based approach, for instance through formulas of a modal logic (such as deontic logic or dynamic logic)—this might be a more appealing solution for agents with intrinsic inferential capabilities.

4.2.3 Acting on an artifact by usage interface

Given the function the agent is interested in, operating instructions describe the admissible/relevant behaviours, expressed in terms of sequences of artifact's operations to be executed—actions over the artifact, and perceptions of relevant events. To this end, each artifact is equipped with a description of its *usage interface*, that is, the set of all operations made available by the artifact to the agent use.

For each operation, the usage interface describes how it should be invoked, and how its outcome could be perceived by the agent. Each operation is then typically described in terms of (i) its name, (ii) its arguments (their names and types), and (iii) its outcome. An agent executes an action over the artifact by specifying the operation name and by providing the suitable arguments—namely, this is the data that flows outside the agent. The operation outcome describes the events generated by the artifact while serving the operation, which the agent can perceive by sensing—namely, the data that flows inside the agent. Such events may signal (i) when the operation is actually triggered, (ii) whether it successfully completed, and what is the computed result, (iii) whether and why it failed, and (iv) whether the artifact reached a state the agent might be interested in.

The usage interface could be either static or dynamic: the set and shape of the admissible operations may be stable, or it may vary depending on the artifact inner state. Moreover, it could organise operations in distinct classes as sort of specialised interfaces, such as those used to directly control the artifact inner state, those that administer the artifact life-cycle (powering-on, changing behaviour, stopping, shutting-down), and so on [56].

4.3 Features of cognitional artifacts and intelligence

Actually, the three distinctive features of cognitional artifacts are not all mandatory for the cognitive use of an artifact. While the availability of function description, operating instructions, and usage interface altogether would fully support cognitive agents in rationally selecting an artifact, planning its use, and interacting with it, in several scenarios a subset of them could still allow for some desirable level of cognitive use.

As an example, function description might not be available for agent's direct inspection: however, an agent could still be able to induce the function of an artifact through a trial-and-error process, or by observing the actions of other agents in the MAS—and the same holds for operating instructions and usage interface. In fact, all the distinctive properties of cognitional artifacts might be implicitly defined in a non-cognitional artifact, and agents can be made aware of them by learning, or because agents have been programmed to do so.

What really characterises a cognitional artifact is that such properties are conceived when the artifact is designed, explicitly represented in it, and then made available for inspection by cognitive agents. As a result, cognitional artifacts support the construction of truly-open MAS, where cognitive agents of any sort can enter the MAS workspace, inspect available artifacts and deliberate accordingly, obtain function descriptions and select some artifacts for future use, inspect operating instructions and plan their course of action, finally obtain usage interfaces and interact efficiently and effectively.

Obviously, this scenario is quite different from typical MAS from current literature that mostly rely on speech-act communications. The existence of artifacts, and even more cognitional artifacts, and the opportunities they provide to agents, call for a new model of "agent intelligence", where the intelligence of an agent is measured both in terms of its ability to speak and understand a high-level symbolic language, and in terms of its ability to use artifacts—and possibly to forge new ones according to its needs. The *Agens Faber* approach goes along this way, based on the idea that agent intelligence should not be considered as separated by the agent ability to perceive and affect MAS environment—and so, that agent intelligence is strictly related to the artifacts that enable, mediate and govern any agent (intelligent) activity [42].

5 Artifacts for MAS: applications

The notion of artifact has a vast potential, and has already been successfully applied to a number of diverse MAS-related research fields. Classifying features of artifacts, such as *inspectability*, *malleability*, *predictability*, *linkability* and *distribution*, have already been shown to generally impact on the way in which MAS are conceived and built [43]. In the remainder of this section, we shortly recall some of the main areas where the A&A meta-model and the artifact abstraction have found their most fruitful application. In particular, we discuss how adopting artifact as a foundational abstraction for MAS impacts on AOSE (agent-oriented software engineering, Sect. 5.1), MAS languages and infrastructures (Sect. 5.2), MABS (multi-agent based simulation, Sect. 5.3), agent-based SOS (self-organising systems, Sect. 5.4) and other MAS-related research fields (Sect. 5.5).

5.1 Artifacts & agent-oriented software engineering

The first natural area of application for a MAS meta-model is AOSE [5]. Generally speaking, a new meta-model for computational systems directly impacts on the way in which computer scientists and engineers conceive and build such systems. More specifically, artifacts (and the A&A meta-model in general) can be exploited as a basis for new methodological approaches, impacting in particular on the way in which MAS societies and environment are engineered.

While early AOSE methodologies were mostly devoted to the engineering of individual agents, later developments put rather the focus on system issues like organisation [72], interaction [36], self-organisation [4], environment [68], and so on. Along the same line, the availability of artifacts as first-class abstractions promotes a balanced approach to the engineering of MAS, where active entities (the agents) are in charge of pro-actively pursuing global system goals, while reactive entities (the artifacts) are meant to provide the required functions and services. In particular, artifacts are to be used for automating the management of social issues (coordination, organisation, security), as well as for engineering MAS environment.

Introduced to handle the issues of interaction and environment in MAS engineering [36], the SODA methodology was later re-casted and extended according to the A&A meta-model, also with the aim of testing the effectiveness of the meta-model in the MAS engineering process [27]. As a result, SODA is a task-oriented methodology that separates activity and function all along the design process. In short, *tasks* in the Analysis step are mapped upon *roles* that execute *actions* in the Architectural Design step, which are then assigned to *agents* in the Detailed Design step. Correspondingly, *functions* are mapped upon *resources* that exhibit *operations*, which are then assigned to *artifacts*. *Dependencies* between tasks and functions are mapped upon *interactions* bounding roles' and resources' behaviour, then mapped upon *social artifacts*.

Social artifacts represent one of the three categories of artifacts in SODA, which also include individual and environment artifacts. On the one hand, each individual artifact handles the interaction of a single agent within a MAS, and essentially works as a mediator between the agent and the MAS itself. Since they can be used to shape the space of admissible interactions of individual agents in a MAS, individual artifacts play an essential role in engineering both organisational and security concerns in MAS. On the other hand, each environmental artifact brings an external resource within a MAS, by mediating agent actions and perceptions over resources. As such, environmental artifacts play an essential role in enabling, disciplining and governing the interaction between agents and MAS environment [52]. In turn, a social artifact rules social interaction within a MAS—even though indirectly, since it technically mediates interaction between individual, environmental, and possibly other social artifacts. Social artifacts in SODA play the role of the coordination artifacts that embody rules around which societies of agents can be built [43].

Quite obviously, the role of artifacts as a core abstraction for engineering MAS societies and environment is not limited to SODA: for instance, Molesini et al. [28] shows how artifacts could be used to extend a generic AOSE methodology so as to make it capable to handle MAS environment as a first-class abstraction.

5.2 Artifacts & MAS languages and infrastructures

MAS programming languages and infrastructures are deeply affected by a radical change in their foundational meta-model [7]. So, on the one hand, taking the standpoint of artifacts first leads to re-interpret existing agent programming languages and MAS infrastructures, by observing the occurrence of concepts and technological constructs that can be easily re-casted in terms of artifacts. On the other hand, a new meta-model for MAS paves the way for the design of new classes of programming languages, as well as for the development of new infrastructures complying with the new conceptual framework. Accordingly, in the following we first recall how some existing agent languages and MAS infrastructures have been re-interpreted and extended following the A&A meta-model, then we discuss some of the new A&A languages and infrastructures currently under development.

Since artifacts mostly affect the space of MAS interaction, the first application field for A&A is the one of MAS coordination. The notion of coordination medium—as the space where (objective) coordination [39,60] is encapsulated—has been re-interpreted and generalised in terms of *coordination artifact* [44]. Also, artifacts have impacted on both classes of coordination models, control- and data-driven [45]. So, on the one hand Dastani et al. [13] exploit a weak notion of coordination artifact so as to allow for the composition of multiple coordinated MAS based on the control-driven model Reo [2]. On the other hand, a more radical approach has led to re-interpreting the data-driven ReSpecT coordination language according to the A&A meta-model [37]. Also, the notion of coordination artifact has been used to extend the Cougaar agent middleware so as to deal with issues like survivability [73], scalability and quality of service [62].

In general, artifacts constitute a usable tool to model the interaction between agents and web services, which can be seen as standardised sorts of web-based artifacts. For instance, Rubino et al. [57] re-intepret OWL-S as a language for representing artifacts. More generally, artifacts constitute a suitable abstraction for modelling the interaction between agents and generic infrastructure components. Acay et al. [1] introduce OWL-T as an ontology for enabling agents to discover and use artifacts placed in MAS environment—there called *tools*—with no need of any a priori knowledge about them.

Besides allowing for new viewpoints on known results, a new meta-model stimulates and promotes the development of original languages and technologies that embody and support the meta-model itself. As two prominent examples, the simpA language [53] and the CArtAgO infrastructure [55] embody by their very origin all the principles of the A&A meta-models, and elaborate on the notion of artifact from a technological standpoint.

On the one hand, simpA is a library-based extension of Java which aims at providing a high-level coarse-grained support to complex multithreaded/concurrent application programming. Based on the A&A meta-model, simpA provides Java programmers with an agent-oriented abstraction layer on top of the basic OO layer, which they can use so as to organise and structure Java-based applications in terms of agents and artifacts. Among the many uses for such a language, simpA has developed into the simpA-WS technology for WS-I compliant SOA/WS applications, implementing the SA&A (Service Agents and Artifacts-based Architecture) programming model for SOA and Web Services [48].

On the other hand, CArtAgO provides MAS engineers with an infrastructure fully embodying the A&A meta-model—where agents and artifacts come along with workspaces, and define the spaces of the available API for MAS programmers [56]. Given the impact of artifacts on the cognitive process of agents, an essential test for the effectiveness of an artifact-based infrastructure is to experiment with diverse cognitive architectures for agents, differing from a conceptual as well as from a technological standpoint. To this end, CArtAgO has been integrated with *Jason* [8] and 2APL [14]—playing the role of the BDI-like agent-oriented development frameworks—and with simpA—playing the role of the activity-based, non-cognitive agent development framework—thus demonstrating how the notion of artifact shared by agents within working environments promotes a coherent and well-founded integration of heterogeneous agents of any sort [51].

5.3 Artifacts & multi-agent based simulation

Simulation is one of the hottest areas of application for complex computational systems. So, it comes not as a surprise that MABS is one of the key-area in today agent-oriented computing, and that MAS-based models shape some of the most used simulation frameworks. In the context of MABS, artifacts can be used as basic abstractions to model and simulate those parts of a complex system that are better framed in terms of passive function-oriented entities than in terms of pro-active, objective-oriented ones. This enhances expressiveness in modelling complex systems, promotes a more straightforward simulation design, and allow for a more effective simulation execution.

Once the A&A meta-model is adopted as the reference model, a complex system can be generally described at the domain level [16] in terms of a multiplicity of agents—carrying on their activities in an autonomous and asynchronous way—and artifacts—working as the basic building blocks to model the agent environment. So, besides “real agents”, also “real artifacts” can be observed and analysed in the target system, and represented in the domain model. As a simple example, a social simulation modelling the behaviour of a crowd in a tube station would map humans upon agents, and use artifacts to model the behaviour of critical objects such as sliding doors, traffic signs, barriers, and so on. Instead, the agent abstraction would not be suited for modelling traffic signs or sliding doors, since they have neither internal goal nor autonomous activity of any sort. More generally, artifacts can be used to directly model real-world artifacts whose design and behaviour can have an impact on the overall behaviour of target systems—and which would be improperly represented as agents.

As a case study for artifacts in MABS, Montagna et al. [29] reports an example of simulation adopting the A&A meta-model, which models a biological system–namely, the glycolysis metabolic pathway. There, agents in the domain model (there referenced as *bio-agents*) model those parts of the biological system that exhibit some kinds of autonomous behaviour, whereas artifacts (*bio-artifacts*) model the bio-chemical environment that contains a set of molecules, and that enables and mediates bio-agents actions and interactions. Some other examples of possible bio-agents at the intra-cellular level are given by macro-molecular components with a complex structure and behaviour, such as proteins; or, at the inter-cellular level, by cells with a state and a specific internal and interactive behaviour. Examples of bio-artifacts at the intra-cellular level are cell compartments; at inter-cellular level, cell micro-environments.

5.4 Artifacts & self-organising systems

The complexity of the applications scenarios envisioned for forthcoming MAS calls for key autonomic properties like self-diagnosis, self-repairing, self-organisation, and the like. Also, autonomy of agents and MAS copes well with self-* issues, so that self-organisation has been one of the hottest areas of development in MAS research [15]. Given the role of environment in SOS, and the relationship between artifacts and MAS environment, artifacts potentially play a key role in engineering self-organising MAS. In the following, we illustrate some of the main ideas for multi-agent SOS based on artifacts, and recall the notion of Cognitive Stigmergy as a general mechanism for promoting self-organisation in MAS.

MAS are generally considered as a good paradigm to model SOS (self-organising systems): the agent abstraction makes it possible to suitably encapsulate the micro-level behaviours that produce self-organisation patterns at the macro-level [15]. However, there are behaviours that are essential to SOS which are inherently distributed, and are not goal-oriented—so they are not naturally captured by the agent abstraction, but rather depend on SOS environment. In general, the analysis of both natural [6,9] and artificial SOS [25,46] points out the crucial role of the environment in the global dynamics of SOS. Since artifacts can be taken as the basic bricks for engineering MAS environment, they have the potential to work as the key abstractions to inject SOS mechanisms into MAS.

A key example is the case of *stigmergy* [21], used as the source of simple yet effective coordination metaphors and mechanisms to be exploited for building robust and reliable SOS in unpredictable settings. Stigmergy is generally meant as the social mechanism of coordination based on interaction through local modifications to a shared environment—as in the cases of the deposition of pheromone by ants, and of the movement of wooden chips by termites [9]—so, as a relevant case of environment-based coordination. The fact that artifacts could be effectively exploited as the basis for stigmergic coordination should not be surprising: for instance, *coordination artifacts* have been shown to provide general support for implicit communication and environment-based coordination [44]. What should be noted instead is that the notion of artifact paves the way toward a more general concept of stigmergy.

Research on stigmergy in MAS has led to a number of very interesting approaches—see Parunak et al. [46] and Valckenears et al. [64] among the many others—however it has frequently induced two main biases: (i) the agent model is very simple, and (ii) the environment model is often quite elementary [19]. By contrast, stigmergy has been shown to work as a relevant coordination mechanism also in the context of human societies and organisations, where (i) agents are possibly cognitive entities, and (ii) the environment is typically articulated and composed of suitably engineered artifacts, possibly supporting agent's cognitive processes [24,63].

Based on the adoption of artifacts as the basis for modelling MAS environment, Ricci et al. [50] defines the notion of *cognitive stigmergy* as the generalisation of stigmergy as a coordination mechanism for societies of cognitive agents. Cognitive stigmergy is based on the use of artifacts as tools populating and structuring the agent working environment, and which agents perceive, share and rationally use for their individual goals. At the *domain level*, artifacts represent the target of the agent work. At the *tool level*, artifacts represent the working tools helping agents in their activity, meant to be used both to improve agent knowledge about the *practises* in using the artifacts at the domain level, and to support *social construction* and *evolution/adaptation* of such artifacts.

5.5 Artifacts & other applications fields

Given their conceptual pervasiveness within complex agent-based environments, artifacts have of course an impact over many other areas than the ones discussed above.

In the field of electronic institutions, *computational institutions* have been defined for modelling norm-regulated MAS based on coordination artifacts as the abstractions encapsulating norms as coordination policies [58]. In the distance learning field, Nardini et al. [31] discusses how technologies based on the A&A meta-model could be exploited to build a framework for collaborative learning that would promote cognitive decentralisation and overcome typical reasoning bias of unilateral knowledge transmission. Also, Oliva et al. [35] explore the use of artifacts as abstractions aimed at providing social support for argumentation in MAS. In particular, the notion of *Co-Argumentation Artifact* is introduced as an artifact specialised in managing arguments and providing a coordination service for argumentation process in a MAS.

While many other areas of application exist, the ones discussed above in this section should clearly point out the benefits of adopting the artifact abstraction in MAS. Future work will be devoted in further developing the potential of the notion of artifact, by exploring its applicability and effectiveness within many other areas of application.

6 Conclusions

Agents in MAS are surrounded by a number of tools, objects, instruments, and services that affect their course of actions by shaping their surrounding environment. In order to provide a coherent view of MAS, a unifying abstraction is needed that could be used to modelling and engineering the world around agents of a MAS.

In this article we discussed the notion of *artifact* as an abstraction of such a sort, in the context of the A&A (agents & artifacts) meta-model for MAS. After discussing the main sources of inspiration for artifacts in heterogeneous fields such as cognitive sciences and anthropology, we precisely defined the concept of artifact in computational systems, then discussed how it could affect agent cognition in MAS. Finally, we presented some of the many agent-related research fields where the application of artifacts as a foundational notion already demonstrated its effectiveness.

Future work will be devoted to refine the notion of workspace so as to complete the formulation of the A&A meta-model, on the theoretical side, and to further advance technologies and methodologies for MAS design and development based on the A&A meta-model, on the practical side.

Acknowledgements Apart from the work of the authors, this article also benefits from the efforts of many people in the last years. First of all, the AgentLink III Technical Fora (AL3TF) hosted many intense and fruitful discussions that led to elaborate most of the results presented here. The material presented here and in many of the papers cited here was often discussed, debated, criticised, and developed during several meetings held in the context of AgentLink III, and within many of the fora in particular. As a result, many of the attendants to the AL3TF meetings have indirectly contributed to the evolution of the material presented in this article. Then, the organisers and chairs of many of the AL3TF have actively encouraged us to put together all the material discussed in this article in a structured and organised form: some of them had a substantial role in the preliminary definition of the scope and goals of article, and in pointing out some of the most relevant issues to be addressed. Among the many others, we would like to thank Monique Calisti, Cristiano Castelfranchi, Massimo Cossentino, Mehdi Dastani, Giovanna Di Marzo Serugendo, Rino Falcone, Marie-Pierre Gleizes, Michael Luck, Emanuela Merelli, Paolo Petta, and Danny Weyns. Also, we are deeply indebted with the reviewers of this article, whose remarks and criticisms resulted in largely improving the quality of the article in structure and content. Many people of our research group have brought contributions in the last years, which have been in some form incorporated in the body of this article. Among the many others, we would like to thank Antonio Natali, Enrico Denti, Ambra Molesini, Luca Gardelli, Enrico Oliva, Rossella Rubino, Matteo Casadei, Sara Montagna, and Elena Nardini. The AgentLink project, in its three subsequent incarnations funded by the European Community, supported our efforts along the years. In particular, we are grateful to the scientific crew of AgentLink III, who helped creating such a fertile environment for the development of our research. Finally, we would like to specially thank Peter McBurney, who overcame our initial objections in making this article part of this special issue. As an editor of this special issue, Peter convinced us that this article was a fit conclusion for some of the many efforts that the AgentLink community has profused in the last years.

References

1. Acay, L. D., Pasquier, P., & Sonenberg, L. (2007) Extrospection: Agents reasoning about the environment. In *3rd IET international conference on intelligent environments* (*IE 07*) (pp. 220–227).
2. Arbab, F. (2004). Reo: A channel-based coordination model for component composition. *Mathematical Structures in Computer Science, 14*(3), 329–366.
3. Bardram, J. (1998). Designing for the dynamics of cooperative work activities. In *1998 ACM conference on computer supported cooperative work* (*CSCW'98*) (pp. 89–98). ACM Press.
4. Bernon, C., Camps, V., Gleizes, M.-P., & Picard, G. (2005a). Engineering adaptive multi-agent systems: The ADELFE methodology. In B. Henderson-Sellers & P. Giorgini (Eds.), *Agent oriented methodologies* (pp. 172–202). Hershey, PA, USA: Idea Group Publishing, Chapt. VII.
5. Bernon, C., Cossentino, M., & Pavón, J. (2005b). An overview of current trends in european AOSE research. In (Omicini et al., 2005) (pp. 379–390).
6. Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm intelligence*: *From natural to artificial systems*. Santa Fe Institute Studies in the Sciences of Complexity. New York, NY, USA: Oxford University Press.
7. Bordini, R. H., Braubach, L., Dastani, M., Seghrouchni, A. E. F., Gomez-Sanz, J., Leite, J. A., O'Hare, G., Pokahr, A., & Ricci, A. (2006). A survey of programming languages and platforms for multi-agent systems. In (Omicini et al. 2006a) (pp. 33–44).
8. Bordini, R. H., & Hübner, J. F. (2006). BDI agent programming in agentspeak using *Jason* (Tutorial Paper). In F. Toni & P. Torroni (Eds.), *Computational logic in multi-agent systems*, Vol. 3900 of *Lecture notes in computer science* (pp. 143–164). Springer, 6th International Workshop, CLIMA VI, London, UK, June 27–29, 2005, Revised Selected and Invited Papers.
9. Camazine, S., Deneubourg, J.-L., Franks, N. R., Sneyd, J., Theraulaz, G., & Bonabeau, E. (2001). *Self-organization in biological systems*, Princeton studies in complexity. Princeton, NJ, USA: Princeton University Press.
10. Conte, R., & Castelfranchi, C. (Eds.), (1995). *Cognitive and social action*. University College London.
11. Copi, I. M., & Cohen, C. (1994). *Introduction to logic* (9th ed.) New York: Macmillan.
12. Cossentino, M., Fortino, G., & Russo, W. (2008). Special issue on multi-agent systems and simulation. *International Journal of Agent-Oriented Software Engineering, 2*(2).
13. Dastani, M., Arbab, F., & de Boer, F. S. (2005). Coordination and composition in multi-agent systems. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. P. Singh, & M. J. Wooldridge (Eds.), *4th International joint conference on autonomous agents and multiagent systems* (*AAMAS 2005*) (pp. 439–446). Utrecht, The Netherlands: ACM.
14. Dastani, M., Hobo, D., & Meyer, J.-J. C. (2007). Practical extensions in agent programming languages. In E. Durfee, M. Yokoo, M. N. Huhns, & O. Shehory (Eds.), *6th International joint conference on*

autonomous agents and multiagent systems (*AAMAS'07*) (pp. 923–925). Honolulu, Hawai'i, USA: ACM Press.

15. Di Marzo Serugendo, G., Gleizes, M.-P., & Karageorgos, A. (2006). Self-organisation and emergence in MAS: An overview. In (Omicini et al. 2006a) (pp. 45–54).
16. Drogoul, A., Vanbergue, D., & Meurisse, T. (2003). Multi-agent based simulation: Where are the agents?. In J. S. Sichman, F. Bousquet, & P. Davidsson (Eds.), *Multi-agent-based simulation II*, Vol. 2581 of *LNCS* (pp. 43–49). Springer, 3rd International Workshop (MABS 2002), Bologna, Italy, 15–16 July 2002. Revised Papers.
17. Engeström, Y., Brown, K., Christopher, L. C., & Gregory, J. (1997). Coordination, cooperation & communication in the courts: Expansive transitions in legal Work. In M. Cole, Y. Engeström, & O. Vasquez (Eds.), *Mind, culture, and activity*. Cambridge University Press, Chapt. 28.
18. Franklin, S., & Graesser, A. (1996). Is it an agent, or just a program?: A taxonomy for autonomous agents. In J. P. Müller, M. J. Wooldridge, & N. R. Jennings (Eds.), *Intelligent agents III. Agent theories, architectures, and languages*, Vol. 1193 of *LNCS* (pp. 21–35). Springer, . ECAI'96 Workshop (ATAL'96) Budapest, Hungary, 12–13 Aug. 1996. Proceedings.
19. Gardelli, L., Viroli, M., Casadei, M., & Omicini, A. (2008). Designing self-organising environments with agents and artifacts: A simulation-driven approach. In (Cassentino et al. 2008) (pp. 171–195).
20. Gibson, K. R., & Ingold, T. (Eds.), (1993). *Tools, language & cognition in human evolution*. Cambridge University Press.
21. Grassé, P.-P. (1959). La reconstruction du nid et les coordinations interindividuelles chez Bellicositermes natalensis et Cubitermes sp. la théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs. *Insectes Sociaux, 6*(1), 41–80.
22. Hewes, G. W. (1993). A history of speculation on the relation between tools and languages. In (Gibson and Ingold, 1993) (pp. 20–31).
23. Kauffman, S. A. (2001). *Investigations*. Oxford University Press.
24. Kirsh, D. (1999). Distributed cognition, coordination and environment design. In *European Conference on Cognitive Science* (pp. 1–11).
25. Mamei, M., & Zambonelli, F. (2005). Programming stigmergic coordination with the TOTA middleware. In *4th International joint conference on autonomous agents and multiagent systems* (*AAMAS'05*) (pp. 415–422) New York, NY, USA: ACM Press.
26. Martelet, G. (1998). *Évolution et création, tome 1*. Paris: Editions du Cerf.
27. Molesini, A., Omicini, A., Denti, E., & Ricci, A. (2006). SODA: A roadmap to artefacts. In O. Dikenelli, M.-P. Gleizes, & A. Ricci (Eds.), *Engineering societies in the agents world VI*, Vol. 3963 of *LNAI* (pp. 49–62). Springer, 6th International Workshop (ESAW 2005), Kuşadası, Aydın, Turkey, 26–28 Oct. 2005. Revised, Selected & Invited Papers.
28. Molesini, A., Omicini, A., & Viroli, M. (2008). Environment in agent-oriented software engineering methodologies. *Multiagent and Grid Systems, 4*. Special Issue on Environment Engineering for MAS.
29. Montagna, S., Ricci, A., & Omicini, A. (2008). A&A for modelling and engineering simulations in systems biology. In (Cossentino et al. 2008) (pp. 222–245).
30. Nardi, B. A. (Ed.), (1996). *Context and consciousness*: *Activity theory and human–computer interaction*. MIT Press.
31. Nardini, E., Omicini, A., & Matteucci, M. C. (2008). Toward a framework for collaborative learning based on agent-based technologies. In *International education, technology and development conference* (*INTED 2008*). Valencia, Spain, EU: IATED.
32. Noriega, P., & Sierra, C. (2002). Electronic institutions: Future trends and challenges. In M. Klusch, S. Ossowski, & O. Shehory (Eds.), *Cooperative information agents VI*, Vol. 2446 of *LNAI*. Springer.
33. Norman, D. A. (1991). Cognitive artifacts. In J. M. Carroll (Ed.), *Designing interaction*: *Psychology at the human–computer interface*, Cambridge Series On Human–Computer Interaction (pp. 17–38). New York: Cambridge University Press.
34. Odell, J. J. (2002). Objects and agents compared. *Journal of Object Technology, 1*(1), 41–53.
35. Oliva, E., McBurney, P., & Omicini, A. (2007). Co-argumentation artifact for agent societies. In S. Parsons, I. Rahwan, & C. Reed (Eds.), *4th International workshop argumentation in multi-agent systems* (*ArgMAS 2007*) (pp. 115–130) Honolulu, Hawai'i, USA: AAMAS 2007.
36. Omicini, A. (2001). SODA: Societies and infrastructures in the analysis and design of agent-based systems. In P. Ciancarini & M. J. Wooldridge (Eds.), *Agent-oriented software engineering*, Vol. 1957 of *LNCS* (pp. 311–326). Springer-Verlag, 1st International Workshop (AOSE 2000), Limerick, Ireland, 10 June 2000. Revised Papers.
37. Omicini A. (2007). Formal ReSpecT in the A&A perspective. *Electronic Notes in Theoretical Computer Sciences, 175*(2), 97–117. 5th International Workshop on Foundations of Coordination Languages and Software Architectures (FOCLASA'06), CONCUR'06, Bonn, Germany, 31 Aug. 2006. Post-proceedings.

38. Omicini, A., & Denti, E. (2001). From tuple spaces to tuple centres. *Science of Computer Programming, 41*(3), 277–294.
39. Omicini, A., & Ossowski, S. (2003). Objective versus subjective coordination in the engineering of agent systems. In M. Klusch, F. Bergamaschi, S., Edwards, P. & P. Petta (Eds.), *Intelligent information agents: An agentlink perspective*, Vol. 2586 of *LNAI: State-of-the-Art Survey* (pp. 179–202). Springer-Verlag.
40. Omicini, A., Petta, P., & Gams, M. (2005). Special issue: Hot topics in European agent research I. *Informatica, 29*(4), 377–443.
41. Omicini, A., Petta, P., & Gams, M. (2006a). Special issue: Hot topics in European agent research II. *Informatica, 30*(1), 1–82.
42. Omicini, A., Ricci, A., & Viroli, M. (2006b). *Agens faber*: Toward a theory of artefacts for MAS. *Electronic Notes in Theoretical Computer Sciences, 150*(3), 21–36. 1st International Workshop "Coordination and Organization" (CoOrg 2005), COORDINATION 2005, Namur, Belgium, 22 Apr, 2005. Proceedings.
43. Omicini, A., Ricci, A., & Viroli, M. (2006c). Coordination artifacts as first-class abstractions for MAS engineering: State of the research. In A. F. Garcia, R. Choren, C. Lucena, P. Giorgini, P., T. Holvoet, & A. Romanovsky (Eds.), *Software engineering for multi-agent systems IV: Research issues and practical applications*, Vol. 3914 of *LNAI* (pp. 71–90). Springer, Invited Paper.
44. Omicini, A., Ricci, A., Viroli, M., Castelfranchi, C., & Tummolini, L. (2004). Coordination artifacts: Environment-based coordination for intelligent agents. In N. R. Jennings, C. Sierra, L. Sonenberg, L. & M. Tambe (Eds.), *3rd international joint conference on autonomous agents and multiagent systems* (*AAMAS 2004*) (Vol. 1 pp. 286–293). New York, USA: ACM.
45. Papadopoulos, G. A., & Arbab, F. (1998). Coordination models and languages. In M. V. Zelkowitz (Ed.), *The engineering of large systems*, Vol. 46 of *Advances in computers* (pp. 329–400). Academic Press.
46. Parunak, H. V. D., Brueckner, S., & Sauter, J. (2002). Digital pheromone mechanisms for coordination of unmanned vehicles. In *1st International joint conference on autonomous agents and multiagent systems AAMAS'02* (pp. 449–450). ACM Press.
47. Povinelli, D. J. (2000). *Folk physics for Apes: The chimpanzee's theory of how the world works*. Oxford University Press.
48. Ricci, A., Buda, C., & Zaghini, N. (2007a). An Agent-oriented programming model for SOA & web services. In D. Dietrich, G. Hancke, & P. Palensky (Eds.), *5th IEEE international conference on industrial informatics* (*INDIN'07*) (pp. 1059–1064). Vienna, Austria, IEEE. Special Session on Agents Theory and Practice for Industry.
49. Ricci, A., Omicini, A., & Denti, E. (2003). Activity theory as a framework for MAS coordination. In P. Petta, R. Tolksdorf, & F. Zambonelli (Eds.), *Engineering societies in the agents world III*, Vol. 2577 of *LNCS* (pp. 96–110). Springer-Verlag, 3rd International Workshop (ESAW 2002), Madrid, Spain, 16–17 Sept. 2002. Revised Papers.
50. Ricci, A., Omicini, A., Viroli, M., Gardelli, L., & Oliva, E. (2007). Cognitive stigmergy: Towards a framework based on agents and artifacts. In D. Weyns, H. V. D. Parunak, & F. Michel (Eds.), *Environments for multiagent systems III*, Vol. 4389 of *LNAI* (pp. 124–140). Springer, 3rd International Workshop (E4MAS 2006), Hakodate, Japan, 8 May 2006. Selected Revised and Invited Papers.
51. Ricci, A., Piunti, M., Acay, L. D., Bordini, R., Hübner, J., & Dastani, M. (2008a). Integrating artifact-based environments with heterogeneous agent-programming platforms. In *7th International joint conference on autonomous agents and multiagent systems* (*AAMAS-08*).
52. Ricci, A., & Viroli, M. (2005). Coordination artifacts: A unifying abstraction for engineering environment-mediated coordination in MAS. In (Omicini et al. 2005) (pp. 433–443).
53. Ricci, A., & Viroli, M. (2007). simpA: An agent-oriented approach for prototyping concurrent applications on top of java. In V. Amaral (Ed.), *5th International symposium on principles and practice of programming in Java* (*PPPJ 2007*), Vol. 272 of *ACM International Conference Proceeding Series* (pp. 185–194). Lisboa, Portugal, ACM.
54. Ricci, A., Viroli, M., & Omicini, A. (2006). Programming MAS with artifacts. In R. P. Bordini, M. Dastani, J. Dix, & A. El Fallah Seghrouchni (Eds.), *Programming multi-agent systems*, Vol. 3862 of *LNAI* (pp. 206–221). Springer, 3rd International Workshop (PROMAS 2005), AAMAS 2005, Utrecht, The Netherlands, 26 July 2005. Revised and Invited Papers.
55. Ricci, A., Viroli, M., & Omicini, A. (2007). CArtAgO: A framework for prototyping artifact-based environments in MAS. In D. Weyns, H. V. D. Parunak, & F. Michel (Eds.), *Environments for multiagent systems III*, Vol. 4389 of *LNAI* (pp. 67–86). Springer, 3rd International workshop (E4MAS 2006), Hakodate, Japan, 8 May 2006. Selected Revised and Invited Papers.
56. Ricci, A., Viroli, M., & Omicini, A. (2008). The A&A programming model & technology for developing agent environments in MAS. In M. Dastani, A. El Fallah Seghrouchni, A. Ricci, & M. Winikoff (Eds.), *Programming multi-agent systems*, Vol. 4908 of *LNAI*. Springer. 5th International Workshop (PROMAS 2007), AAMAS 2007, Honolulu, Hawaii, 15 May 2007. Revised and Invited Papers.

57. Rubino, R., Molesini, A., & Denti, E. (2006). OWL-S for describing artifacts. In A. Omicini, B. Dunin-Kęplicz, & J. A. Padget (Eds.), *4th European workshop on multi-agent systems* (*EUMAS 2006*). Sun SITE Central Europe, RWTH Aachen University. Fourth European Workshop on Multi-Agent Systems Lisbon, Portugal.
58. Rubino, R., Omicini, A., & Denti, E. (2006). Computational institutions for modelling norm-regulated MAS: An approach based on coordination artifacts. In O. Boissier, J. Padget, V. Dignum, G. Lindemann, E. Matson, S. Ossowski, J. S. Sichman, & J. Vázquez-Salceda (Eds.), *Coordination, organizations, institutions, and norms in multi-agent systems*, Vol. 3913 of *LNAI* (pp. 127–141). Springer, AAMAS 2005 International Workshops "Agents, Norms and Institutions for Regulated Multi-Agent Systems" (ANI@REM 2005) & "Organizations in Multi-Agent Systems" (OOOPS 2005), Utrecht, The Netherlands, 25–26 July 2005. Revised Selected Papers.
59. Schmidt, K., & Simone, C. (2000). Mind the gap! Towards a unified view of CSCW. In R. Dieng, A. Giboin, L. Karsenty, & G. De Michelis (Eds.), *Designing cooperative systems: The use of theories and models*, Vol. 58 of *Frontiers in artificial intelligence and applications*. Sophia Antipolis, France, IOS Press. 4th International conference on the design of cooperative systems (COOP 2000), Proceedings.
60. Schumacher, M. (2001). *Objective coordination in multi-agent system engineering—design and implementation*, Vol. 2039 of *LNAI*. Springer-Verlag.
61. Searle, J. (1969). *Speech acts: An essay in the philosophy of language*. Cambridge University Press.
62. Siracuse, S., Zinky, J. A., Shapiro, R., & Wright, T. (2005). Scalable MAS-based control systems using QoS-adaptive coordination artifacts. In R. Mailler, P. Scerri, & R. Vincent (Eds.), *2nd Workshop on challenges in the coordination of large-scale multi-agent systems* (*LSMAS'05*). Utrecht Netherlands: AAMAS (2005).
63. Susi, T., & Ziemke, T. (2001). Social cognition, artefacts, and stigmergy: A comparative analysis of theoretical frameworks for the understanding of artefact-mediated collaborative activity. *Cognitive Systems Research, 2*(4), 273–290.
64. Valckenears, P., Van Brussel, H., Kollingbaum, M., & Bochmann, O. (2001). Multi-agent coordination and control using stigmergy applied to manufacturing control. In M. Luck, V. Mařík, O. Štěpánková, & R. Trappl (Eds.), *Multi-agent systems and applications*, Vol. 2086 of *Lecture notes in computer science* (pp. 317–334). Springer, 9th ECCAI Advanced Course ACAI 2001 and Agent Link's 3rd European Agent Systems Summer School (EASSS 2001), Prague, Czech Republic, 2–13 July 2001, Selected Tutorial Papers.
65. Viroli, M., Holvoet, T., Ricci, A., Schelfthout, K., & Zambonelli, F. (2007). Infrastructures for the environment of multiagent systems. In (Weyns and Parunak, 2007) (pp. 49–60).
66. Viroli, M., Ricci, A., & Omicini, A. (2006). Operating instructions for intelligent agent coordination. *The Knowledge Engineering Review, 21*(1), 49–69.
67. Vygotsky, L. S. (1978). *Mind and society*. Harvard University Press.
68. Weyns, D., Omicini, A., & Odell, J. (2007). Environment as a first-class abstraction in multi-agent systems. In (Weyns and Parunak, 2007) (pp. 5–30)
69. Weyns, D., & Parunak, H. V. D. (2007). Special issue on environments for multi-agent systems. *Autonomous Agents and Multi-Agent Systems, 14*(1), 1–116.
70. Wood, A. B., Horton, T. E., & Amant, R. S. (2005). Effective tool use in a habile agent. In E. J. Bass (Ed.), *2005 IEEE Systems and Information Engineering Design Symposium* (*SEADS 2005*) (pp. 75–81). Charlottesville, VA, USA: IEEE.
71. Zalta, E. N. (Ed.) (2003). *Stanford encyclopedia of philosophy*, Chapt. Aristotle's Metaphysics. Stanford University.
72. Zambonelli, F., Jennings, N. R., & Wooldridge, M. (2003). Developing multiagent systems: The Gaia methodology. *ACM Transactions on Software Engineering and Methodology* (*TOSEM*)*, 12*(3), 317–370.
73. Zinky, J., Shapiro, R., Siracuse, S., & Wright, T. (2005). Implementing QoS-adaptation in coordination artifacts by enhancing cougaar multi-agent middleware. In S. Barber, M. Kam, V. Subrahmanian, & B. Regli (Eds.), *2nd IEEE symposium on multi-agent security and survivability* (*MASS 2005*) (pp. 39–48). Philadelphia, PA, USA.