# Enhancing MAS Environments with Organizational Mechanisms

Sergio Esparcia, Estefanía Argente
*Grupo de Tecnología Informática - Inteligencia Artificial*
*Departamento de Sistemas Informáticos y Computación*
*Universidad Politécnica de Valencia*
*Camino de Vera, s/n - 46022 Valencia, Spain*
{*sesparcia, eargente*}*@dsic.upv.es*

Roberto Centeno, Ramón Hermoso
*Centre for Intelligent Information Technologies (CETINIA)*
*University Rey Juan Carlos*
*C/ Tulipán, s/n - 28933 Móstoles (Madrid), Spain*
{*roberto.centeno, ramon.hermoso*}*@urjc.es*

***Abstract*—Organizational mechanisms can be introduced in a multi-agent system with the aim of influencing the behavior of agents to achieve their objectives in a proper way. Thus, they are designed to achieve a better coordination between agents. In this paper, we propose to model organizational mechanisms by means of artifacts, which were presented within the Agents & Artifacts approach, and that present good advantages for coordinating agents environments. We claim that artifacts, as reactive entities located into the environment of a Multi-agent System, can help agents to reach their goals, seem to be a suitable abstraction for modeling organizational mechanisms. We also give some examples of possible uses.**

***Keywords*-Artifacts; Organizational mechanisms; Environment; Agent-Oriented Software Engineering;**

## I. Introduction

Since Multi-Agent Systems (MAS) designers have changed their perspective from closed, agent-centered systems to open, organization-oriented systems [1], the environment surrounding MAS has been mainly considered as heterogeneous, unpredictable, distributed and dynamic. Being such a complex environment, it must include mechanisms and tools that help managing and controlling it.

Nowadays, the environment of a MAS [2] is being modeled as a first class abstraction of the system. Different approaches presented new concepts that help developers to model the environment. One of the most recognized is the Agents & Artifacts (A&A) conceptual framework [3], which is based on the human cooperative elements and it is characterized by three types of abstractions: (i) agents, the proactive elements of the system; (ii) artifacts, the entities that must be used by the agents; and (iii) workspaces, a portion of the environment that contains agents and artifacts and defines the topology of the system. Additionally, the workspace is the space where agents and artifacts are able to develop their functionality.

Organizational mechanisms [4] can be a valid method to provide coordination into organizations. They are mechanisms introduced in a MAS with the aim of influencing the agents' behavior towards more effectiveness with regard to some goals from both a macro and a micro perspective. Hence, these mechanisms can provide additional information to agents which may persuade them to behave in a certain way; or they can produce changes in the environment that may impose certain behaviors to agents. Thus, it is very useful to use these mechanisms in an open system where external agents are located, so then being able to promote coordination.

Seeing that artifacts are located into the MAS environment and they can also improve coordination between agents, the objective of this work is to model organizational mechanisms as artifacts, in order to facilitate system designers its usage and implementation. A generic idea of every mechanism will be given in order to allow MAS developers to create the most effective artifact for their system. We will only define the minimum features, properties and operations, that the artifacts must provide to be considered as artifacts for organizational mechanisms.

The rest of this paper is structured as follows: Section II describes the background of this work, by defining organizational mechanisms and artifacts. Section III models the organizational mechanisms as artifacts. Section IV compares some of the existing artifacts with our proposal. Finally, section V gives our conclusions on this proposal.

## II. Background

This section describes the two paradigms on which our proposal is based. On the one hand, the organizational mechanisms will be described to give an overview on how they can improve the behavior of the agents in a MAS. On the other hand, the artifacts will be depicted to show that they are a tool that facilitates the interaction between agents and their environment.

### A. Organizational Mechanisms

Organizational Mechanisms [4] are mechanisms introduced in a multi-agent system with the aim of influencing the agents' behavior towards more effectiveness with regard to the global purpose of the system. They rely on the assumption that agents participating in the system are rational, i.e. try to maximize their utility with any action they perform. In order to clarify formalization we first give a definition of

MAS on which the remainder formulae rely. We adhere the definition in [4].

*Definition 1:* A Multi-Agent System (MAS) is a tuple $\langle \mathcal{A}g, \mathcal{A}, \mathcal{X}, \Phi, x_0, \varphi \rangle$ where:

- $\mathcal{A}g$ is a set of agents; $|\mathcal{A}g|$ denotes the number of agents in the system;
- $\mathcal{A}$ is a finite action space that includes all possible actions that can be performed in the system. $\mathcal{A}$ includes an action $a_{skip}$, the action of doing nothing;
- $\mathcal{X}$ is the environmental state space;
- $\Phi : \mathcal{X} \times \mathcal{A}^{|\mathcal{A}g|} \times \mathcal{X} \to [0..1]$ is the MAS transition probability distribution, describing how the environment evolves as a result of agents' actions;
- $x_0 \in \mathcal{X}$ stands for the initial state of the MAS;
- $\varphi : \mathcal{A}g \times \mathcal{X} \times \mathcal{A} \to \{0,1\}$ is the agents' capability function describing the actions agents are able to perform in a given state of the environment. $\varphi(a, x, ac) = 1$ ($\varphi(a, x, ac) = 0$) means that agent $a$ is able (not able) to perform action $ac$ in the state $x$.

Upon this definition of MAS, two different types of organizational mechanisms may be defined: *informative* and *regulative*.

**Informative organizational mechanisms** [4] (from now on informative mechanisms) are defined as a function that given a partial description of an internal state of an agent and taking into account the partial view that the mechanism has of the current environmental state, it provides information, which may consist of a set of actions an agent can take but it is possibly not aware of, a recommendation of a particular action which is eventually a "good action" for the agent, or information about the consequences that a given action may have. Formally it is defined as follows:

$$\Gamma : \mathcal{S}' \times \mathcal{X}' \to \mathcal{I}$$

where:

- $\mathcal{S}'$ represents the set of possible partial descriptions of agents' internal states;
- $\mathcal{X}'$ is the set of partial views of environmental states;
- $\mathcal{I}$ represents an information space.

All informative mechanisms have in common that their usage is not imposed. Agents are free to use such mechanisms at their own discretion. To use an informative mechanism, an agent should provide it with part of his internal state. In fact, when rationality of agents is assumed [5], agents must use a given informative mechanism if and only if they expect that the usage of the mechanism will be advantageous for them. Informative mechanisms may improve the performance of individual agents and may have effects on the global performance of an organized MAS with respect to a global utility function. The information provided by this kind of mechanisms will improve the knowledge of an agent, since the latter includes some extra information for reasoning and thus making him to better choose his future actions.

**Regulative organizational mechanisms** [4] (from now on regulative mechanisms) share the same objective as informative mechanisms, but they focus on introducing changes into the environment in order to keep agents from undesired behaviors that drive the system to non-profitable states, that is, these mechanisms are in charge of producing changes in the system so as to reach states that improve the system's global utility. The rationale behind introducing changes in the environment is that agents perceive those changes, so possibly altering their reasoning to decide which action perform next. Such type of mechanisms rely on the existence of a system designer, that defines the preference relation over system states represented through the global utility function, and that has sufficient authority to impose certain changes in the system.

Two types of possible changes in the environment are considered: (i) introduction of incentives in order to make agents follow a desired behavior, and (ii) changes in the agents' action space. Accordingly, two types of regulative mechanisms have been defined:

- An **incentive mechanism** is a function that given a partial description of an environmental state of MAS produces changes in the transition probability distribution of MAS. Formally:
  $$\Upsilon_{inc} : \mathcal{X}' \to [\mathcal{X} \times \mathcal{A}^{|\mathcal{A}g|} \times \mathcal{X} \to [0..1]]$$
- A **coercive mechanism**, $\Upsilon_{coe}$, for $MAS$ is a function that given a possibly partial description of an environmental state of $MAS$ produces changes in the agents' capability function of $MAS$, thus adding or deleting actions from an agent's action space.
  $$\Upsilon_{coe} : \mathcal{X}' \to [\mathcal{A}g \times \mathcal{X} \times \mathcal{A} \to \{0,1\}]$$
  where $\mathcal{X}'$ represents the set of possible partial descriptions of the environmental states of $MAS$.
- Any regulative mechanism is either incentive or coercive.

Incentive mechanisms may produce changes in the consequences of agents' actions by introducing rewards and penalties. Obviously, rewards and penalties may produce variations in the expected utility of an agent's actions and, hence, rational agents would change their decisions accordingly (if they know about such incentives). Therefore, agents must be informed about the norms governing the system. In the case of coercive mechanisms the changes in the system are produced through a modification of agents' action space. New actions may be added or existing actions may be eliminated.

Both types of mechanisms emerge as an important contribution to MAS. Since nowadays MAS systems have become open and heterogeneous, and it is possible that non-collaborative agents populate a system, it is necessary to be provided with mechanisms that help the system's administrators to keep the MAS under control. Both informative and regulative mechanisms are very useful to afford this task. Aside from this, organizational mechanisms need to be implemented into the environment of a MAS. The information provided by the organizational mechanisms can be supplied to the agents by using artifacts, being an important contribution to improve the MAS environment.

### B. Artifacts

Artifacts [3] are non-proactive, but reactive entities that agents employ to achieve their goals. As artifacts do not have assigned goals, they are associated to the goals of the agent that uses the artifact. To accomplish these goals, artifacts provide a *functionality*, which is partitioned into some *operations* that agents can execute when interacting with them. These operations are part of the *usage interface* of the artifact, that is completed with the *observable properties* that agents can check without invoking any operation in it. Artifacts provide a second group of operations, called *link operations* (accessible through a link interface) that enables composition of artifacts and load distribution, since different artifacts may be located at the same or different workspaces[1] [3]. Every workspace contains a set of artifacts; and the set of workspaces composing the environment is used to define its topology. Finally, artifacts are enhanced with a *function description* (which acts as a manual) and a set of *operating instructions*, an essential feature when dealing with open systems, since external agents can discover artifacts and evaluate whether they could be useful to reach their goals.

Since artifacts are very malleable components from the environment of a MAS, designers can develop new types of artifacts according to actual system needs. The Agent Oriented Software Engineering (AOSE) community has already developed different types of artifacts [6]: (i) *basic artifacts*, which comprises artifacts that give information of very general world features (for example, clocks, calendars and timetables); (ii) *coordination artifacts* [7], that improve the coordination between agents in a MAS; (iii) *reputation artifacts* [8], that manage reputation values of agents in an organization; (iv) *cognitive stigmergy artifacts* [9], which provide information about an agent or a society of agents that can be useful to other agents or groups; (v) *organizational artifacts* [10], which are used to manage an organization; and (vi) *argumentation artifacts* [11], that manage arguments between agents. Moreover, it is possible to use the *CArtAgO* framework [12] to implement artifacts, which is engineered upon the principles of the A&A conceptual framework [3].

## III. ARTIFACTING THE ORGANIZATIONAL MECHANISMS

This section describes how both types of organizational mechanisms, informative and regulative, can be modeled as artifacts. Artifacts allow an easy merging of the organizational mechanisms into the environment of a MAS.

We formalize an artifact as follows:

*Definition 2:* An Artifact is a tuple $\langle PR, OP, LO, St \rangle$ where:

- $PR$ are the observable properties of the artifact that agents can directly check without operation invoking;
- $OP$ is the set of operations that agents can execute when interacting with it;
- $LO$ stands for link operations, which can be called by other artifacts. This type of operations enables artifact composition and functionality distribution by linking artifacts. In some cases, these operations may be used to help the initialization of another artifact;
- $St$ is the internal state of an artifact, which is not accessible by the agents populating the system.

The result of this modeling is a set of three types of artifacts. The *informative artifacts* are based on the informative mechanisms; the *incentive artifacts* are based on the incentive mechanisms; while the *coercive artifacts* are based on the coercive mechanisms.

### A. Informative artifacts

As stated previously, informative mechanisms return information about actions to an agent, given a partial description of his internal state and taking into account the partial view of the environment that the mechanism has. The informative mechanism has been modeled as an artifact, named *informative artifact*, being a passive entity that is used by agents in order to help them in their deliberative process.

*Definition 3:* An Informative Artifact is defined as an artifact $Ar_{inf} = \langle PR, OP, LO, St \rangle$ where:

- $PR \subseteq \{St \cup \emptyset\}$ are the observable properties of the informative artifact, which are a subset of the information contained into the artifact or an empty set.
- $OP : \mathcal{S}' \rightarrow \mathcal{I}$ are the operations of the artifact, where:
  - $\mathcal{S}'$ represents a partial description of an agent's internal state.
  - $\mathcal{I}$ represents the information returned by the artifact, based on the internal state of the artifact and the partial description of the agent's internal state (semantically, $\mathcal{S}' \times St \rightarrow \mathcal{I}$).
- $LO : \Theta \rightarrow \mathcal{I}$ is a link operation that is used by an artifact $Ar_1$ to obtain information from the $Ar_{inf}$ artifact, where:
  - $\Theta \subseteq (\Sigma \cup \mathcal{S}')$ is the information sent by $Ar_1$ to $Ar_{inf}$;
  - $\Sigma \subseteq \{St_1 \cup \emptyset\}$ is a partial state of $Ar_1$, being $St_1$ the internal information of $Ar_1$;
  - $\mathcal{S}'$ represents a partial description of the internal state of the agent that is requesting information to the artifact $Ar_1$;
  - $\mathcal{I}$ represents the information returned by the artifact $Ar_{inf}$ to the artifact $Ar_1$ (previously requested), based on the partial description of $Ar_1$

[1] A workspace is the portion of environment that is perceived by an agent, who is able to interact with.

($\Sigma$), the partial description of the agent's internal state who is requesting $Ar_1$ ($\mathcal{S}'$) and the internal state of the artifact $Ar_{inf}$ ($St$). Semantically: $(\Sigma \cup \mathcal{S}') \times St \rightarrow \mathcal{I}$.

- $St$ represents the internal state of the artifact, i.e. the information contained into the artifact, which is not directly accessible by agents or other artifacts.

Informative artifacts are not required to provide with link operations, so they might be only accessible by agents in its same workspace. When they offer a link operation, artifacts located in its same workspace or in other connected workspaces can obtain relevant information from this informative artifact by means of its link operations.

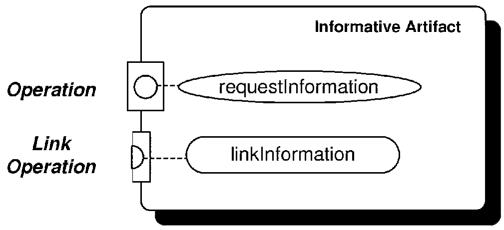


Figure 1. *Informative artifact for an informative mechanism*

Figure 1 shows a graphic representation of an informative artifact. As explained before, this type of artifact at least needs one operation: `requestInformation`($St$). Giving a partial description of the internal state of the agent, that can contain the roles, believes, facts and other features associated with the agent and his environment, this operation returns a package of information that contains: (i) the type of the information, that can be a recommendation or an advice about actions, or information about the consequences of executing an action; (ii) the description of this information, and (iii) a set of actions that are related to this information (they could be services that an agent could take, recommended services, services that have consequences to the agent, etc.). Moreover, as explained before, an informative artifact could need to make requests to other artifacts in order to obtain information or update its internal state. For this reason, the informative artifacts are enhanced with a link operation, `linkInformation`($S', St$), that is executed as a consequence of an operation ($OP$) invoked in another artifact. The requester artifact, $Ar_1$, sends the partial state of the agent (agent's internal state) that requested information, as well as a subset of its internal information. As a result, the requested artifact, $Ar_{inf}$, will return some piece of information based also on its own internal state.

To exemplify how this type of artifacts work, we define an artifact that publicly provides norms currently active in the system. This artifact would contain operations shown in Figure 2.

Let $Ar_{inf}^{norms} = \langle \emptyset, \{requestNorms\}, \{linkInformation\} \rangle$ be an artifact that aims to provide agents (on demand) with information about norms, such as the specification of norms that rule a role, active or deactivated norms, etc. The operation $linkInformation$ may be used by other artifacts in order to gather information related to norms that could improve their usage. Thus, the artifact encapsulates functionality for both, agents that request information for their personal purposes, and other artifacts that could also be interested in some information that the artifact manages about norms.

We can observe that there does not exist any observable property, since norms cannot be directly accessed by agents, but they may be requested by using the operation $requestNorms$. Consider that, since it is an informative artifact, the agent requesting for norms must send a part of his mental state in order to allow the artifact to give him back some useful information.

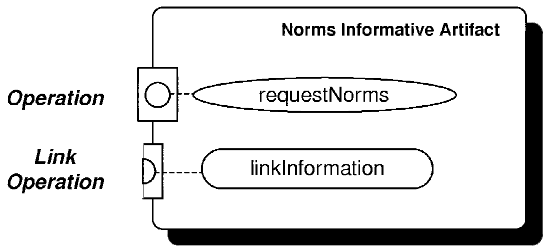


Figure 2. *Informative artifact for informing about norms*

Notice that this artifact is not a mere repository of norms, since allows to be tuned to distinguish among different types of information that should be provided to agents. Thus, the mechanism designer probably does not want that any agent could know all the norms at any time, but it could probably prefer to give the precise information the agent is interested in, in such a way that it does not disclose any sensitive information.

A typical scenario would consist on an agent requesting for the set of norms that rule a specific role that the agent wants to play. Responsibilities, duties and rights that roles specify for its enactment should make the artifact to provide suitable information on demand.

### B. Incentive Artifacts

As explained before, incentive mechanisms are mechanisms that are able to produce changes in the system environment from a global view, modifying the rewards and penalties that are active in the system. These mechanisms rely on the belief that a possibly little change in the incentive system (that can be affecting only to a small number of agents) affects the entire system. In this subsection, the incentive mechanisms are modeled as artifacts, named *incentive artifacts*. These artifacts will execute organizational changes, which bring the possibility of implementing an adaptive system, by varying elements from the system (e.g. adding or deleting norms). After a change in the incentive system of the MAS is produced, transition probabilities between different states of the system are affected. In order to carry out these changes, it is necessary to have an agent or a human playing a special role that we call 'system adapter', which is able to manage organizational changes when necessary to promote the adaptiveness of the MAS. The system adapter is the only agent that has privileges to execute the operations of an incentive artifact.

*Definition 4:* An Incentive Artifact is defined as an artifact $Ar_{inc} = \langle PR, OP, LO, St \rangle$ where:

- $PR \subseteq \{St \cup \emptyset\}$ are its observable properties;
- $OP$ : $\Delta$ is the operation that allows the *system adapter* to introduce or remove incentives in the system;
- $LO = \emptyset$, since this type of artifacts has no predefined link operations;
- $St$ represents the internal state of the artifact.

The operation of the artifact ($OP$) modifies the transition probability between different states of the system. This operation is defined as:

$$\Phi = St \rightarrow [\mathcal{X} \times \mathcal{A}^{|\mathcal{A}g|} \times \mathcal{X} \rightarrow [0..1]], \text{ where:}$$

- $\Phi$ is the MAS transition probability distribution, describing how the environment evolves as a result of agents' actions.
- $\mathcal{X}$ is the environmental state space.
- $\mathcal{A}^{|\mathcal{A}g|}$ is the set of actions executed by agents between two states of the MAS.

This operation works as follows: the agent provides some piece of information to the artifact, which might change its internal state ($St$). Given this new internal state, the transition probability between two states of the system is modified, so the behavior of the MAS changes in a global perspective.

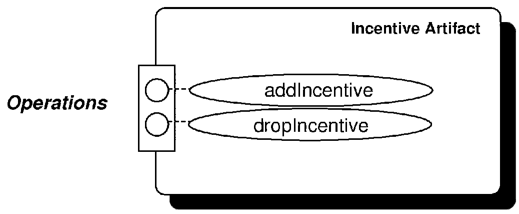


Figure 3. *Incentive artifact*

Figure 3 shows how an incentive artifact is modeled, including its minimum required features. The $OP$ set contains two different operations: `addIncentive`($\{ty, ro, ac\}$), (with $\{ty, ro, ac\} \in St$), which adds an incentive to the incentive system of the MAS; and `dropIncentive`($\{ty, ro, ac\}$) (with $\{ty, ro, ac\} \in St$), which drops an incentive from the MAS. In these functions *ty* stands for the type of incentive (reward or penalty), *ro* refers to the role or set of roles affected by this incentive and *ac* represents a set of actions that are related with this incentive.

Sometimes, it could be useful to send information about changes in the incentive system to the agents populating the MAS. In order to execute this task it is necessary to provide the environment with an informative artifact, modeled as explained in the previous subsection.

To exemplify the incentive artifacts, we have chosen an organizational environment related to norms again. Let $Ar_{inc}^{norms} = \langle \emptyset \ , \ \{addNormIncentive, dropNormIncentive\} \ , \ \emptyset \rangle$ be an incentive artifact that allows introducing positive incentives (*rewards*) and negative incentives (*penalties*) into an organization. These incentives consist of a set of possible consequences that norm fulfilment or violation, respectively, may entail. As aforementioned, the incentive mechanisms aim to improve the system performance by introducing changes in the environment that somehow influence the agents' reasoning. For this example we consider that the artifact does not contain any observable property and that it does not offer any minimum link operation to be requested by other artifacts. The usage interface ($OP$) should not be available for every agent participating in the system. That is, this kind of artifacts does not provide information, but changes the environment, so only agents with sufficient permissions to do it should use operations in $OP$, depending on the domain. In our case agents capable of playing role 'system adapter' can employ $addNormIncentive$ operation, so then attaching a penalty to a norm in case of violation; or introducing rewards for norm fulfilments. Incentives may also be updated through the time, by using $dropIncentive$ operation to remove the former and then updating with the new one by using $addNormIncentive$ operation.

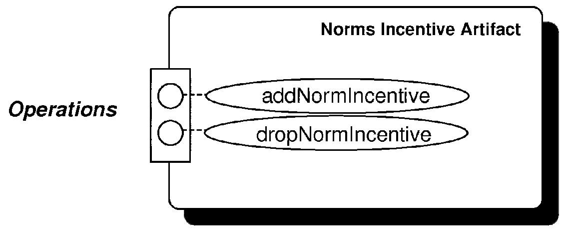


Figure 4. *Incentive artifact for norms*

### C. Coercive artifacts

As explained before, coercive mechanisms are aimed to produce changes in the environment of the system by producing changes in the agents' capability functions, given a possibly partial description of MAS. As it occurs with incentive mechanisms, coercive mechanisms are also relying on the existence of the 'system adapter' role, which is able to promote organizational changes.

Formally, a coercive artifact is defined as:

*Definition 5:* A Coercive Artifact is an artifact $Ar_{coe} = \langle PR, OP, LO, St \rangle$ where:

- $PR \subseteq \{St \cup \emptyset\}$ are its observable properties;
- $OP : St \rightarrow [\mathcal{A}g \times \mathcal{X} \times \mathcal{A} \rightarrow \{0,1\}]$ is the operation carried out by the coercive artifact, where:
  - $\mathcal{A}g$ is an agent of the MAS;
  - $\mathcal{A}$ is the action space that includes all possible actions that can be performed in the system.
- $LO = \emptyset$, since this type of artifacts has no predefined link operations;
- $St$ represents the internal state of the artifact.

The operation $St \rightarrow [\mathcal{A}g \times \mathcal{X} \times \mathcal{A} \rightarrow \{0,1\}]$, given the artifact's internal state, returns the capability for executing an action or not, 1 and 0 respectively. Internally, this operation works as follows: the artifact needs its internal state ($St$) as well as the information provided by the system adapter ($\mathcal{A}g$ and $\mathcal{A}$) in order to execute this operation. After compiling all this information, the artifact calculates the new action space of the agent. This change can be seen as a local change but, since agents are related between them, changes in a single agent might produce changes in a set of agents, i.e. in the global state of the MAS.

Similarly to the informative and incentive artifacts, the coercive artifacts do not have mandatory observable properties. In case of having them, they are a subset of the internal knowledge of the artifact. The number of the available observable properties will depend on the purpose of each artifact. Additionally, they are not required to have link operations.

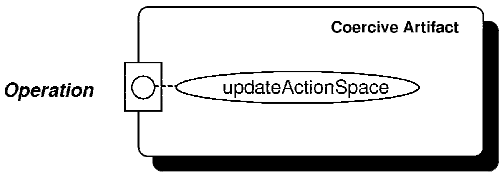


Figure 5. *Coercive artifact*

Figure 5 shows how a coercive artifact is modeled, including their minimum requirements. The only operation defined in this artifact is `updateActionSpace`($\{ag, a\}$) (where $\{ag, a\} \in St$ and $ag \in \mathcal{A}g, a \in \mathcal{A}$), which receives an agent and and action from the system adapter and returns the capability for the given agent to perform this action.

As done with incentive artifacts, to show an example of coercive artifacts an organizational environment related to norms is taken:

Let $Ar_{coe}^{norms} = \langle \emptyset \ , \ \{updateActionSpace\} \ , \ \emptyset \rangle$ be a coercive artifact that aims to update agents' action spaces through time. As we stated in Section II-A, coercive mechanisms directly modify agents' action spaces to keep the former from undesirable behaviors. Thus, this artifact will be in charge of modifying those action spaces on demand of some special agents that have the permission to introduce these changes in the environment. Therefore, if one of the agents with sufficient permissions (i.e. 'system adapter') observes that, for instance, the violation of a norm occurred, he could take the decision of banning some actions to the agent that did not fulfil that norm, trying to avoid that behavior in the future. In the same way that the artifact may remove actions from an agents' action space, it might also add actions to it, if agent's behavior is being acceptable. For instance, the system could test participants with a trial period to ensure that they behave accordingly to system's objectives, allowing them to perform more and more actions progressively.

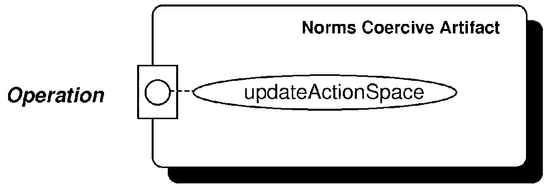


Figure 6. *Coercive artifact for norms*

Some examples of mechanisms that could be designed as incentive or regulative artifacts are: *normative manager*, that is, encapsulating dynamic consequences that fulfilment or violation of norms may entail; or *traffic sanctions manager*, where different sanctions may be applied about driving rules, even introducing constraints in the environment (roads can be closed, driver licenses could be taken away, etc.).

## IV. RELATED WORK

This section is aimed to compare our proposal with other available artifacts presented by different authors, in order to determine whether existing artifacts have the features of informative, incentive or coercive artifacts.

Informative artifacts provide a common functionality for MAS. Some of the artifacts presented by the community of researchers provide information to agents after receiving information about a partial view of their internal state, so they could be seen as informative artifacts. For example, the *Role Evolution Coordination Artifact* [13], that is aimed to build and evolve a role specialization taxonomy, which consists on a set of roles with a concrete order, over time; and make this information available to the agents. This artifact contains three operations: (i) *getBestRolesForInteraction*, that provides the most specialized roles for a given service type interaction; (ii) *getAgentsForRoles*, which provides the set of agents that play at least one of the roles in a given set of roles; and (iii) *getRolesForAgent*, that provides the set of roles that a given agent plays in the system.

Previously, in Section III-A, we have formally defined the operation of an informative artifact as $\mathcal{S}' \times St \rightarrow \mathcal{I}$. A correspondence between the operations of the *Role Evolution Coordination Artifact* and the operation of an informative artifact can be established. In this way, the *getBestRolesForInteraction* operation function can be described as follows:

- $\mathcal{S}' = Serv$, where $Serv$ is a service type interaction.

- $St = R$, where $R$ is the complete set of roles of the MAS.
- $\mathcal{I} = \mathcal{P}(R)$, where $\mathcal{P}(R)$ are the most specialized roles for $S$.

Similarly, the function *getAgentsForRoles* has the following correspondence:

- $\mathcal{S}' = \mathcal{P}(R)$, where $\mathcal{P}(R)$ is a set of roles.
- $St = \mathcal{A}g$, where $\mathcal{A}g$ is the complete set of agents of the MAS.
- $\mathcal{I} = \mathcal{P}(\mathcal{A}g)$, where $\mathcal{P}(\mathcal{A}g)$ is the set of agents that play at least one of the roles in $\mathcal{P}(R)$.

Finally, the function *getRolesForAgent* presents the following correspondence with an operation of an informative artifact:

- $\mathcal{S}' = \mathcal{A}g$, where $\mathcal{A}g$ is an agent of the system.
- $St = R$, where $R$ is the complete set of roles of the system.
- $\mathcal{I} = \mathcal{P}(R)$, where $\mathcal{P}(R)$ is the set of roles that $\mathcal{A}g$ plays in the system.

Another example of an artifact that can be considered as an informative artifact is the *Co-Argumentation Artifact* (CAA) [11] which gives assistance to argumentation processes. The agents share their arguments (i.e. a partial view of their internal state) with the artifact. Then, the artifact evaluates the arguments provided by all the agents and calculates both the "social acceptability" (the acceptability of the arguments of a concrete agent) and the "social behavior" (the acceptability of the arguments from a global perspective).

The CAA implementation proposed in [11] provides two observable properties (*Social Behavior*, *Social Acceptability*) and one operation (*writeArguments*), which allows agents to store their arguments in the artifact.

Following the formalization of artifacts for organizational mechanisms, this CAA can be modeled as both an informative artifact and an incentive artifact. In this case, this artifact can be implemented with two different operations: *getSocialValues* and *writeArguments*.

It is possible to establish a correspondence between the operation *getSocialValues* of a Co-Argumentation Artifact and the required operation of an informative artifact. The partial description of an agent's internal state ($\mathcal{S}'$) is represented in a CAA as the argument that the agent will use during the argumentation process.

$$\mathcal{S}' = Arg_t$$

where $Arg_t$ is an argument provided by the agent $t$.

In this example, the internal state of the artifact ($St$) is the set of arguments that it has stored up to this moment.

$$St = \bigcup_{i=1}^{n} Arg_i$$

where $St$ is the compilation of $n$ arguments.

Finally, the information ($\mathcal{I}$) returned by the artifact are the values of the Social Acceptability and Social Behavior.

$$\mathcal{I} = \{SocAcc, SocBeh\}$$

where $SocAcc$ is the Social Acceptability and $SocBeh$ is the Social Behavior.

The *writeArguments* operation can be employed for establishing an incentive mechanism. In this case, this function can only be used by an agent playing the system adapter role, which can take advantage of this artifact by controlling which arguments that agents propose have to be stored inside the artifact so as to promote a concrete behavior towards a global goal of the system. In this way, only those arguments that might help to promote this expected behavior will be stored using the *writeArguments* operation.

As explained before in Section III-B, the operation of an incentive artifact implies:

$\Phi = St \rightarrow [\mathcal{X} \times \mathcal{A}^{|\mathcal{A}g|} \times \mathcal{X} \rightarrow [0..1]]$

Thus, this operation modifies the transition probability distribution of the system, taking into account the partial view of the internal state the own artifact has ($St$). Figure 7 depicts how it is performed. For example, Agent A and Agent B propose their arguments, $Arg_A$ and $Arg_B$, respectively. These arguments are received by the system adapter who decides that $Arg_A$ better helps promoting the expected social behavior, so he employs the *writeArguments* operation to store this argument in the artifact, and thus apply an incentive.

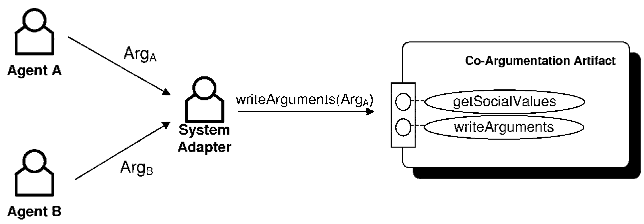


Figure 7. *A CAA working as an Incentive Artifact*

In a similar way, other types of artifacts, such as *Coordination artifacts* [7], *Organizational artifacts* [10] or *Reputation Artifacts* [8], can also be described with features of organizational mechanisms. Therefore, since coordination artifacts encapsulate a coordination service, this coordination service can be implemented by means of an informative artifact (providing useful information to the agents), an incentive artifact (modifying the transition probability between different states of the system) or a coercive artifact (allowing or banning agents from developing different actions). Regarding organizational artifacts, they are used to manage an agent organization in order to help the organization reach its goals from a global, social level. A clear example of this type of artifacts is an artifact that helps informing or managing norms, which, as it has been previously explained along Section III, it can be modeled as an informative artifact (providing norms currently active in the system), an incentive artifact (introducing positive or negative incentives into an organization) or a coercive artifact (removing actions from agent's action space or including new possible actions). Finally, reputation artifacts encapsulate the collection of norm violations of the participants in a system and then aggregate them allowing agents to request reputation by using artifacts' observable properties.

As shown in this section, relevant features of many existing artifacts can be modeled following the proposed formalization of artifacts for organizational mechanisms. In most cases, current operations offered by existing artifacts make them to be easily modeled as informative artifacts. Moreover, the proposed formalization can also be useful to extend operations of current artifacts so as to apply incentive and/or coercive mechanisms into the environment.

## V. Conclusions

Organizational Mechanisms are aimed to improve coordination between agents in a MAS, trying to change this coordination from a micro perspective (i.e., the perspective of individual agents), providing useful information to the agents (informative mechanisms); and a macro perspective (i.e., the perspective of the whole MAS), by modifying either action consequences (incentive mechanisms) or agents' capability functions (coercive mechanisms).

In this work, these mechanisms have been modeled as artifacts to facilitate developers to better deploy and implement them, as well as adding functionality in MAS environments. Three types of Artifacts for Organizational Mechanisms have been defined: (i) *Informative Artifacts*, which provide information to an agent based on the internal state of this agent and the partial view of the environment that the artifact has; (ii) *Incentive Artifacts*, that modify the global behaviour of the system by changing the incentive system of the MAS; and (iii) *Coercive Artifacts*, that update the action space of an agent. All these artifacts make use of the environment of a MAS, so they can explode all knowledge they have about the entities populating the system.

As a future work, we are working on the integration of the Artifacts for Organizational Mechanisms into a metamodel that is conceived to develop Organization Centered MAS such as the Virtual Organization Model (VOM) [14]. The addition of these artifacts will enhance the metamodel with new features that will improve the organizational capabilities of the agents populating the system. Additionally, the presented artifact will be implemented in the THOMAS framework [15], in order to evaluate their performance.

## Acknowledgment

This work is supported by TIN2009-13839-C03-01, TIN2006-14630-C03-02, PROMETEO/2008/051 projects of the Spanish government and CONSOLIDER-INGENIO 2010 under grant CSD2007-00022.

## References

[1] E. Argente, A. Giret, S. Valero, V. Julian, and V. Botti, "Survey of mas methods and platforms focusing on organizational concepts," in *Frontiers in Artificial Intelligence and Applications*, vol. 113. IOS Press, 2004, pp. 309–316.

[2] H. V. D. Parunak and D. Weyns, "Special issue on environments for multi-agent systems," *Auton. Agents Multi-Agent Syst.*, vol. 14, no. 1, pp. 1–4, February 2007.

[3] A. Ricci, M. Viroli, and A. Omicini, "Give agents their artifacts: the A&A approach for engineering working environments in MAS," in *Proceedings AAMAS*, 2007, p. 150.

[4] R. Centeno, H. Billhardt, R. Hermoso, and S. Ossowski, "Organising MAS: a formal model based on organisational mechanisms," in *Proc. SAC*, 2009, pp. 740–746.

[5] A. Rao and M. Georgeff, "Modeling rational agents within a BDI-architecture," *Readings in agents*, pp. 317–328, 1997.

[6] S. Esparcia and E. Argente, "A functional taxonomy for artifacts," in *Proceedings HAIS*, 2010.

[7] A. Omicini, A. Ricci, M. Viroli, C. Castelfranchi, and L. Tummolini, "Coordination artifacts: Environment-based coordination for intelligent agents," in *Proc. AAMAS*, 2004, pp. 286–293.

[8] J. Hubner, O. Boissier, and L. Vercouter, "Instrumenting multi-agent organisations with reputation artifacts," *COIN@ AAAI*, vol. 2008, 2008.

[9] A. Ricci, A. Omicini, M. Viroli, L. Gardelli, and E. Oliva, "Cognitive stigmergy: Towards a framework based on agents and artifacts," in *Environments for MultiAgent Systems III*, ser. LNAI. Springer, May, pp. 124–140.

[10] J. Hubner, O. Boissier, R. Kitio, and A. Ricci, "Instrumenting multi-agent organisations with organisational artifacts and agents," *Auton. Agents Multi-Agent Syst.*, pp. 1–32.

[11] E. Oliva, P. McBurney, and A. Omicini, "Co-argumentation artifact for agent societies," *LNCS*, vol. 4946, p. 31, 2008.

[12] A. Ricci, M. Viroli, and A. Omicini, "CArtAgO: An Infrastructure for Engineering Computational Environments," in *Proceedings E4MAS*, 2006, pp. 102–119.

[13] R. Hermoso, H. Billhardt, and S. Ossowski, "Role Evolution in Open Multi-Agent Systems as an Information Source for Trust," in *Proc. AAMAS*, 2010.

[14] N. Criado, E. Argente, V. Julian, and V. Botti, "Designing virtual organizations," in *Proceedings PAAMS*, ser. Advances in Soft Computing, vol. 55, 2009, pp. 440–449.

[15] A. Giret, V. Julian, M. Rebollo, E. Argente, C. Carrascosa, and V. Botti, "An open architecture for service-oriented virtual organizations," in *Seventh international Workshop on Programming Multi-Agent Systems.PROMAS 2009*, 2009, pp. 74–88.