Artifacting the Organizational Mechanisms: Adding Functionality in MAS Environments

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. We propose to model organizational mechanisms by means of artifacts, which 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.**

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

I. Introduction

Organizational mechanisms [1] can be a valid method to provide coordination into organizations. They are 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. 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. So 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.

Two different types of organizational mechanisms have been defined [1]: informative and regulative. *Informative mechanisms* are 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. *Regulative mechanisms* focus on introducing changes into the environment in order to keep agents from undesired behaviors that drive the system to non-profitable states. This mechanism is in charge of producing changes in the system to reach states that improve the system's global utility. 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. Hence, two types of regulative mechanisms have been defined: (i) 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; and (ii) a *coercive mechanism* 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. For a detailed description of these organizational mechanisms, see [1].

The objective of this work is to model organizational mechanisms as artifacts, in order to facilitate system designers its usage and implementation. Agents & Artifacts (A&A) conceptual framework [2] 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.

Artifacts [2] 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.

In this paper, a description of how each organizational mechanism can be modelled by an artifact is given, so then MAS developers might be able to create the most effective artifact for their system. Thus, Section II models the organizational mechanisms as artifacts. Section III compares some of the existing artifacts with our proposal. Finally, section IV gives our conclusions on this proposal.

II. Artifacting the organizational mechanisms

This section describes how organizational mechanisms can be modeled as artifacts. Artifacts allow an easy merging of the organizational mechanisms into the environment of a MAS. Firstly, we formalize an artifact as follows:

Definition 1: 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 of 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

The informative mechanism has been modeled as an artifact, named *informative artifact*, being a passive entity used by agents to help them in their deliberative process.

Definition 2: 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 space.
  - $\mathcal{I}$ 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 (Σ), 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.

To exemplify how this type of artifacts work, we define an artifact that publicly provides norms currently active in the system, as follows. Let $Ar^{norms}_{inf} = \langle \emptyset, \{requestNorms\}, \{linkInformation\} \rangle$ 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 non-active norms, etc. The $linkInformation$ operation may be used by other artifacts to gather information related to norms that could improve their usage. Thus, this artifact encapsulates functionality for agents that request information for their personal purposes, and for other artifacts that could also be interested in some information that the artifact manages about norms. In this example there is not 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.

Notice that this artifact is not a mere repository of norms, since it allows being 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

Incentive mechanisms can also be 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 3: 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:

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

- Φ 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.

To exemplify the incentive artifacts, we employ 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.

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 4: 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.

An example of coercive artifacts is given in an organizational environment related to norms: Let $Ar_{coe}^{norms} = \langle \emptyset \;,\; \{updateActionSpace\} \;,\; \emptyset\rangle$ be a coercive artifact that aims to update agents' action spaces through time. 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. Similarly, the artifact may be employed to add actions to the agent's action space, 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.

Some examples of mechanisms that could be designed as incentive or regulative artifacts are: *normative manager*, 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.).

III. Related work

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* [3], 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. A correspondence between the operations of the *Role Evolution Coordination Artifact* and the operation of an informative artifact can be established. For example, the *getBestRolesForInteraction* operation function can be described as: $\mathcal{S}' = Serv$, where $Serv$ is a service type interaction; $St = R$, where R is the complete set of roles of the MAS; and $\mathcal{I} = \mathcal{P}(R)$, where $\mathcal{P}(R)$ are the most specialized roles for S.

Another example of an artifact that can be considered as an informative artifact is the *Co-Argumentation Artifact* (CAA) [4] 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 [4] provides two observable properties (*Social Behavior*, *Social Acceptability*) and one operation (*writeArguments*), which allows agents to store their arguments in the artifact. 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*. The *writeArguments* operation can be employed for establishing an incentive mechanism, and 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.

Other types of artifacts, such as *Coordination artifacts* [5], *Organizational artifacts* [6] or *Reputation Artifacts* [7], can also be described with features of organizational mechanisms. For example, since coordination artifacts encapsulate a coordination service, this 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). Organizational artifacts 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 one that helps informing or managing norms, which, as it has been previously explained along Section II, 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 consult reputation by using artifacts' observable properties.

IV. 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) [8]. The addition of these artifacts will enhance the metamodel with new features that will improve the organizational capabilities of the agents populating the system.

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] 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.

[2] 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.

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

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

[5] 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.

[6] 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.

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

[8] 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.