Developing Normative Open Multi-Agent Systems

Carolina Felicíssimo
DI–PUC-Rio / LIP6–PARIS6
R. M. de S. Vicente 225
Rio de Janeiro/RJ – Brazil
(+55) 21 2540 69 15 r.103
cfelicissimo@inf.puc-rio.br

Caroline Chopinaud
LIP6–PARIS6
104 Avenue du Président Kennedy
75016 – Paris – France
(+33) 1 44 27 87 53
caroline.chopinaud@lip6.fr

Amal El Fallah Seghrouchni
LIP6–PARIS6
104 Avenue du Président Kennedy
75016 – Paris – France
(+33) 1 44 27 87 53
amal.elfallah@lip6.fr

ABSTRACT
Open MASs can be extremely dynamic due to heterogeneous agents that migrate among them to obtain resources or services not found locally. In order to prevent malicious actions and to ensure agent trust, open MAS should be enhanced with normative mechanisms. However, it is not reasonable to expect that foreign agents know in advance all the norms of the MAS in which they will execute. Thus, this paper presents our DynaCROM approach for continuously supporting norm-aware agents with updated contextual norm information in MASs. Not with standing, the ultimate goal of a regulated MAS is to have an enforcement mechanism, we also present in the paper DynaCROM integrated with SCAAR, its solution for enforcing contextual norms.

Categories and Subject Descriptors
I.2.11 [Distributed Artificial Intelligence]: Multiagent systems.

General Terms
Management, Design, Reliability.

Keywords
Normative agents, open mas, contextual information, ontology.

1.INTRODUCTION
Openness has led to software systems that have no centralized control and that are formed of autonomous entities 9. Key characteristics of such systems are heterogeneity, conflicting individual goals and limited trust 9. Open systems also can be extremely dynamic. In this work, we assume that a multi-agent system (MAS) is an open system that puts together sets of heterogeneous, self-interested agents whose actions may deviate from the expected behavior in a context.

Norms can be used in an open MAS to regulate agent execution so that the system does not reach an undesirable state. Norms prescribe what should be done in order to fulfill a generalized expectation of behavior. In this sense, a normative MAS is a system that conforms to or is based on norms 9. Actually, norms can also be viewed as event-driven rules that trigger under appropriate conditions of events happening in a regulated system and, by doing so, create, update or cancel commitments affecting a predefined set of agents 9. Normative agents should be able to take into account the existence of social norms in their decisions (either to follow or violate a norm) and to react to violations of the norms by other agents 9.

In order to prevent malicious actions and to ensure agent trust in open MASs, these systems should be enhanced with normative mechanisms. Governance in open MASs is not straightforward since heterogeneity and autonomy rule out any assumption concerning the way third-party agents are implemented and behave 9. Furthermore, agents' internal structures are normally inaccessible suggesting that norm verification should be based on social concepts, which are externally observable. Thus, it should be possible to provide a decentralized normative mechanism, which is not hard coded inside agents and in which norms can be dynamically updated for continuously regulate agents' actions.

This paper presents how developers can implement dynamic normative open MASs, in which norms can be updated at system run-time, and also how heterogeneous norm-aware agents can execute in open MASs supported with updated contextual norm information, both by using our DynaCROM approach 9, 9, 9 (meaning dynamic contextual regulation information provision in open MASs). Not with standing, the ultimate goal of a regulated MAS is to have an enforcement mechanism, thus, we also present in the paper DynaCROM integrated with SCAAR (meaning Self-Controlled Autonomous Agents geneRator) 9. SCAAR is in charge of enforcing DynaCROM contextual norms.

The remainder of this paper is organized as follows. Section 2 presents the DynaCROM solution, including how to classify, represent and compose contextual norms. Section 3 presents the SCAAR norm enforcement mechanism. Section 4 describes a running example for explaining how DynaCROM effectively works. Section 5 points out a related work in the field and compares it with DynaCROM. Finally, we draw our conclusions and outline future work in section 6.

2.CONTEXTUAL NORM INFORMATION PROVISION IN OPEN MASs

DynaCROM aims to support norm-aware agents with updated contextual norm information in open MASs. For this, developers should classify, represent and compose their norms according to the DynaCROM approach in order to create a dynamic normative open MAS called a *DynaCROM MAS*.

2.1Contextual Norm Classification

Basically, an MAS is constituted of environments, organizations and agents playing roles and interacting 9. As environments, organizations, roles and agent interactions are important concepts for the understating of the text, we would like to characterize the meaning in which they are used in the paper.

Environments 9 are discrete computational locations, similar to places in the physical world, which provide conditions for agents to inhabit it. Environments can have refinement levels, such as a specialization relationship (e.g., country, state, etc.), but there cannot be overlaps (e.g., there cannot be two countries in the same place). An environment also can have many organizations. Organizations 9 are social locations in which groups of agents play roles. An organization can embody many sub-organizations, but each organization belongs to only one environment 9. Agents can execute in different organizations and they can also migrate among environments and organizations in order to obtain resources or services not found locally. Roles 9 are abstractions that prescribe a set of related tasks, which agents must perform in order to achieve their designed goals. Roles are defined by organizations independently of agents' individual identities. An agent can interact with any other agent in an MAS by exchanging messages.

Environments, organizations, roles and interactions suggest different contexts for regulation in open MASs.. Contexts are implicit situational information that can be used to characterize situations of agents and to provide relevant information and/or services to them, where relevancy depends on agent tasks 9. Modular context refinements allow a more flexible system for developers while they are maintaining and evolving norm information and, consequently, managing system regulation.

DynaCROM follows directions taken by research into context-aware applications that suggest top-down architectures for classifying contextual information 9,9. In DynaCROM, norm information should be classified in at least the *Environment*, *Organization*, *Role* and *Interaction* contexts. We call these contexts *regulatory contexts* and they are differentiated by the boundaries of their data (i.e. norms). More precisely, *Environment Norms* are applied to all agents in a regulated environment; *Organization Norms* are applied to all agents in a regulated organization; *Role Norms* are applied to all agents playing a regulated role; and *Interaction Norms* are applied to all agents involved in a regulated interaction. This set should be improved with additions of regulatory contexts for representing particular domain norms.

SAC'08, March 16-20, 2008, Fortaleza, Ceará, Brazil.

2.2Contextual Norm Representation

DynaCROM uses *contextual normative ontologies* to explicitly represent its data, having the Norm concept as a central asset. An *ontology* is a conceptual model that embodies shared conceptualizations of a given domain 9; and a *contextual ontology* is an ontology that represents localized domain information 9. The use of ontologies in open MASs supports heterogeneous agents with a common understanding about well-defined system regulation relating abstract concepts, in which contextual norms are formulated, to their concrete application domain.

The DynaCROM ontology defines five related concepts, as illustrated in Fig. 1[1] by multi-lines linked boxes. In each concept, the first line contains the concept's name/identification and the others lines correspond to the concept's attributes. Each attribute's line is divided in three parts. The first part has the attribute's name/identification. The second part contains the attribute's cardinality (i.e., *Instance* for a unique value and *Instance** for n-vary values) of an object property, which links the concept to the another one identified in the third part. For instance, the first line of the Role concept has "Role" as the concept's name; the second line has the multi-value object property "hasNorm", which links the "Role" and "Norm" concepts; and the third line has the object property "isPlayedIn", which links the "Role" and "Organization" concepts.

In the DynaCROM ontology, the *Role* concept encompasses the instances of all regulated roles representing the system's role regulatory context. Each role instance has associations with its norms and organization. The *Organization* concept encompasses the instances of all regulated organizations representing the system's organization regulatory context. Each organization instance has associations with its norms, main organization and environment. The *Environment* concept encompasses the instances of all regulated environments representing the system's environment regulatory context. Each environment instance has associations with its norms and owner environment (the environment it belongs to). The *Norm* concept encompasses the instances of all regulated actions' norms and it can be specialized into the *Permission*, *Obligation* and *Prohibition* sub-concepts. The *Action* concept encompasses the instances of all regulated actions from a DynaCROM MAS.

[1] For readability purposes, ontology data is presented graphically by using the Ontoviz graph plug-in 9 instead of presenting their correspondent OWL code.

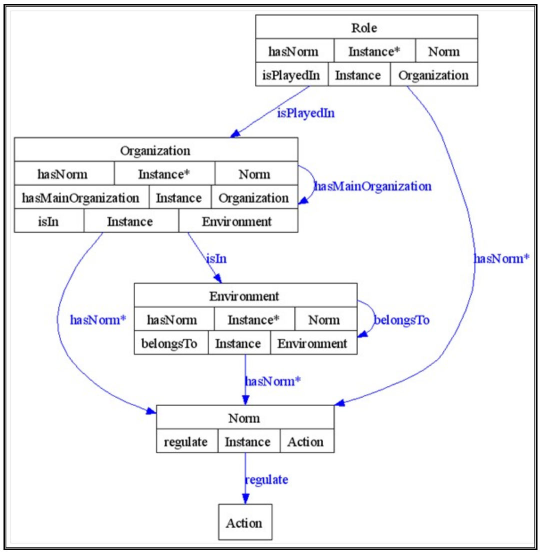

Fig. 1. The DynaCROM ontology.

The interaction regulatory context should be described in the DynaCROM ontology by using a new Norm concept linking two Role concepts. This solution follows the representation pattern presented in 9 which defines that the relation object itself must be represented by a created concept linking the other concepts from the relation (i.e. reification of relationship). For instance, suppose an obligation norm for regulating payments when deals are done between sellers and customers. This norm can be represented by a new Obligation sub-concept, called for example "*ObligationToPay*", which links the seller and customer *Role* sub-concepts.

2.3Contextual Norm Composition

After classifying and representing norms in precise levels of abstractions, contextual norms can be composed at system execution since, at any given moment an agent may be related to norms defined at one or more regulatory contexts. Compositions of related contextual norms result in sets of independent norms, in which the semantic of one norm can influence the semantic of the others. Updating the domain ontology instance of a regulated MAS and customizing different compositions of related contextual norms, both at run-time, provide the dynamism and flexibility necessary for regulation regarding social changes characteristic of open MASs.

DynaCROM uses rules to compose its contextual norms. DynaCROM rules are ontology-driven rules, i.e. they are created according to the ontology structure and they are limited by the number of related concepts to which each concept is linked. All DynaCROM predefined rules are presented in Code 1. Inputs for these rules are domain instances of the Environment, Organization and Role concepts and outputs are compositions of their related contextual norms. For instance, Rule1 (line 1 to 4) states that a given environment will have its norms composed with the norms of its owner environment. More precisely, the following process is executed: in (4), the owner environment "?OEnv" of a given environment "?Env" is discovered; in (3), the norms "?OEnvNorms" of the owner environment "?OEnv" are discovered; and in (2), these norms are composed with the norms of the given environment.

Following the same composition process, Rule2 (line 5 to 8) states that a given organization will have its norms composed with the norms of its main organization; Rule3 (line 9 to 12) states that a given organization will have its norms composed with the norms of its environment; and Rule4 (line 13 to 16) states that a given role will have its norms composed with the norms of its organization.

Code 1. Rules to hierarch DynaCROM contextual norms

```
(1)  Rule1- [ruleForEnvWithOEnvNorms:
(2)          hasNorm(?Env,?OEnvNorms)
(3)           <- hasNorm(?OEnv,?OEnvNorms),
(4)              belongsTo(?Env,?OEnv)]

(5)  Rule2- [ruleForOrgWithMOrgNorms:
(6)          hasNorm(?Org,?MOrgNorms)
(7)           <- hasNorm(?MOrg,?MOrgNorms),
(8)            hasMainOrganization(?Org,?MOrg)]

(9)  Rule3- [ruleForOrgWithEnvNorms:
(10)         hasNorm(?Org,?OrgEnvNorms)
(11)          <- hasNorm(?OrgEnv,?OrgEnvNorms),
(12)             isIn(?Org,?OrgEnv)]

(13) Rule4- [ruleForRoleWithOrgNorms:
(14)         hasNorm(?Role,?OrgNorms)
(15)          <- hasNorm(?Org,?OrgNorms),
(16)             isPlayedIn(?Role,?Org)]
```

For continuously supporting agents with updated information, DynaCROM has an inference rule engine that executes the following tasks: (i) read an ontology instance to get data (i.e., concept instances and relationships), (ii) read active rules to get how concepts must be composed, and (iii) infer an ontology instance based on the previous readings. DynaCROM was implemented as an active JADE 9 behavior, so, this process continuously executes resulting in updated norm information. Once the ontology instance or active rules are changed, this information is automatically forward to agents in the next DynaCROM execution. An overview of the norm composition process is illustrated in Fig. 2.

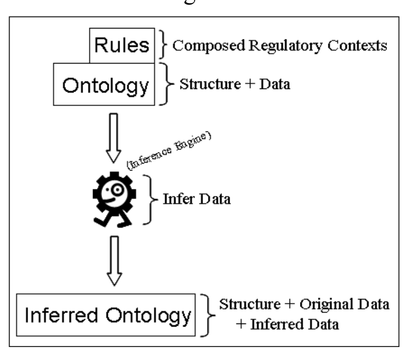

Fig. 2. The DynaCROM norm composition process.

3.CONTEXTUAL NORM ENFORCEMENT

DynaCROM is an approach for implementing dynamic normative open MASs, in which norms can be updated at system run-time, and also for continuously supporting norm-aware agents with precise information. Not with standing, the ultimate goal of a regulated MAS is to have an enforcement mechanism for continuously verify if a performed action is legal or illegal based on its defined norms. Thus, DynaCROM was enhanced with SCAAR for enforcing its contextual norms.

SCAAR is a norm enforcement mechanism that enhances agents with a self-monitoring capability for avoiding norm violation. SCAAR adds both control hooks and an enforcement core in agent codes. These additions are completely transparent to agents when they incorporate the DynaCROM behavior, which is already integrated with SCAAR.

For avoiding norm violation when an agent is executing, its control hooks automatically keep informing its enforcement core about the execution of regulated actions. Then, its enforcement core automatically keeps verifying if each action is executing according to its norms. Norms are represented by synchronized Petri nets 9 for permitting the verification of norm compliance, and inhibitor arcs are used for permitting the norm enforcement. When a regulated action occurs, its specific agent's control hook activates the Petri nets that represents the norms of the action. Each inhibitor arc of the active Petri nets is analysed for verifying if a token stands in all previous arcs (SCAAR Petri nets are formed by one-valuated arcs). If not, an inhibitor arc from the Petri net analyzed forward an exception to the agent's enforcement core. The pseudo algorithm for the SCAAR norm enforcement mechanism is presented in Code 2.

Code 2. SCAAR pseudo algorithm for norm enforcement

(1) Let I be the information received about an agent behavior;
(2) Let $\{t_1, ..., t_n\}$ be the set of transitions associated with I;
(3) Let $\{P_1, ..., P_m\}$ be the set of Petri nets associated with I;
(4) Let $\{Pact_1, ..., Pact_p\}$ be the set of activated Petri nets associated with I;
(5) Let t_{ij} be the transition i of a Petri net j;
(6) **for all** $P_k \in \{P_1, ..., P_m\}$ with $t_{1k} \in \{t_1, ..., t_n\}$ **do**
(7) $Pact_{(p+1)} \leftarrow$ create an instance of P_k if it does not already exist;
(8) add $Pact_{(p+1)}$ in $\{Pact_1, ..., Pact_p\}$
(9) **end for**
(10) Let $\{Pact_1, ..., Pact_l\}$ be the set of the activated Petri nets including $t_{ij} \in \{t_1, ..., t_n\}$;
(11) **for all** $Pact_j \in \{Pact_1, ..., Pact_l\}$ **do**
(11) **for all** $t_{ij} \in \{t_1, ..., t_n\}$ **do**
(12) activates the transition t_{ij} from $Pact_j$;
(13) **if** (t_{ij} is fireable) **then**
(14) fire the transition t_{ij};
(15) remove t_{ij} from $\{t_1, ..., t_n\}$;
(16) **else**
(17) throw an exception
(18) **end for**
(19) remove $Pact_j$ from $\{Pact_1, ..., Pact_l\}$;
(20) **end for**

4.DynaCROM at WORK

4.1Setting the Stage

The FIPA-Contract-Net interaction protocol 9 and the TAC-SCM competition 9 were considered in a simplification of a realistic example in order to illustrate the use of the DynaCROM approach. In the example, agents can play a manufacturer or a supplier role according to the following motivating scenario:

1. An American manufacturer wants to build a computer;
2. He issues a call for proposal (CFP) to suppliers;
3. Suppliers answer the CFP with their proposed prices;
4. The American manufacturer chooses a proposal and informs his decision to the chosen supplier.

To build a computer, the following four component types are necessary: CPU, motherboard, memory and hard disk. There are at least two suppliers for each component type with the base prices of their products predefined, as illustrated in 4.1.

Table 1.Computer components' information

Description	Base price (USD)	Supplier
Pintel CPU	750	Pintel
IMD CPU	650	IMD
Pintel Motherboard	350	Macrostar
IMD Motherboard	300	Basus
Memory 2 GB	150	Macrostar
Memory 2 GB	100	Basus
Hard disk 500 GB	200	Macrostar
Hard disk 500 GB	150	Basus

The four suppliers from the example (Pintel, IMD, Macrostar and Basus) were spread through different environments (i.e., countries and states), as presented in 4.1, for illustrating DynaCROM contextual norms. Basus and Macrostar also are multinational organizations. A multinational organization is an enterprise that manages production branches located in at least two countries, which can also be across multiple continents. Corporate governance includes regulation of all possible relationships among the many players involved. The domain of multinational organizations was chosen because it well illustrates important implicit contextual information that can be found in MASs.

Table 2.Multinational supplier organizations

Organization	Country	State
Pintel	USA	Missouri
IMD	USA	Virginia
Basus	Japan	Osaka
BasusUSA	USA	California
Macrostar	China	Shanghai
MacrostarJapan	Japan	Hiroshima

4.2Classifying Domain Contextual Norms

Usually, organizations do not make their norms public, thus, we created contextual norms for the multinational organizations' domain and organized them in the contexts in which they apply.

4.2.1Environment Norms

Environment Norm for Payments: In all countries, negotiations are obliged to be paid in their national currency. Negotiations outside a country are obliged to have their values converted from its national currency to the national currency of the seller's country. *Contextual Environment Norms for Payments*: (a) In *USA*, all negotiations are obliged to be paid in *American dollars*

(USD); (b) In *Japan*, all negotiations are obliged to be paid in *Yen*; and (c) In *China*, all negotiations are obliged to be paid in *Chinese Yuan (CNY)*.

Environment Norm for Calculating Prices: In North America, a finished good from every organization is obliged to have its price increased by a fixed percentage (dependent of the seller location) as taxes, for immediate delivery or if the deliver address is in North America. *Contextual Environment Norms for Calculating Prices*: (a) In *California*, a state corporate income tax rate of *8.84* is obliged to be imposed on all sales; (b) In *Virginia*, a state corporate income tax rate of *6.00* is obliged to be imposed on all sales; (c) In *Missouri*, a state corporate income tax rate of *6.25* is obliged to be imposed on all sales; and (d) In *Missouri*, a *three day sales tax holiday* occurs, every year, from the first Friday in August until midnight on the Sunday following. Orders of computers and computers' components, with the maximum cost of $3,500, are eligible for tax free during the holiday season.

4.2.2Organization Norms

Organization Norm for Providing Warranty: Organizations are obliged to give a limited lifetime warranty. *Contextual Organization Norms for Providing Warranty*: (a) *Basus organizations* are obliged to give *one year* limited lifetime warranty; (b) *Macrostar organizations* are obliged to give *six months* limited lifetime warranty; and (c) *MacrostarJapan organizations* are permitted to make an offer of *two years* limited lifetime warranty if a plus tax of 5% is accepted to be paid.

Organization Norm for Deliveries: Organizations are prohibited from delivering orders during holidays to their final destinations. *Contextual Organization Norm for Deliveries*: (a) *BasusUSA organizations* are prohibited from delivering orders during holidays to their final destinations.

4.2.3Role Norms

Role Norm for Providing Discounts: Suppliers are permitted to give up to a limited percentage of discounts. *Contextual Role Norm for Providing Discounts*: (a) *IMD suppliers* are permitted to give up to *10% discount* on orders paid in cash.

Role Norm for Accepting Placed Orders: Suppliers are obliged to request a down payment for accepting placed orders. *Contextual Role Norm for Accepting Placed Orders*: (a) *IMD suppliers* are obliged to request a down payment of *10%* for accepting placed orders.

4.2.4Interaction Norms

Interaction Norm for Providing Discounts: Suppliers are permitted to give up to a limited percentage of discounts if their products are bought in bundles. *Contextual Interaction Norm for Providing Discounts*: (a) *Pintel* and *Macrostar* suppliers are permitted to offer 15% discount if their products are bought in bundles.

4.3Representing and Composing Domain Contextual Norms

DynaCROM explicitly represents its domain contextual norms in an ontology instance and uses rules to compose them. For instance, Fig. 3. illustrates part of the DynaCROM domain ontology extended and instantiated to represent the contextual role norm for accepting placed orders of our example. This norm is represented by the "ObligationToRequestADownPayment" norm instance, which regulates the "AcceptAPlacedOrder" action instance.

The "ObligationToRequestADownPayment" role norm is composed with the environment norms "ObligationToImposeAStateCorporateIncomeTax" (inherited from Virginia) and "ObligationToPayWithNationalCurrency" (inherited from "USA"), according to Rule1, Rule3 and Rule4 from Code 1. More precisely, the following process is executed: according to Rule1, in (4), USA is discovered as the owner environment of Virginia; in (3), the "ObligationToPayWithNationalCurrency" environment norm of USA is discovered; and in (2), this norm is composed with the "ObligationToImposeAStateCorporateIncomeTax" environment norm of Virginia. According to Rule3, in (12), Virginia is discovered as the environment of IMD; in (11), the environment norms of Virginia are discovered; and in (10), these norms are added as IMD norms. According to Rule4, in (16), IMD is discovered as the organization of the "AIMDSupplier" role; in (15), the norms of IMD are discovered; and in (14), these norms are composed with the "ObligationToRequestADownPayment" role norm of the "AIMDSuppier".

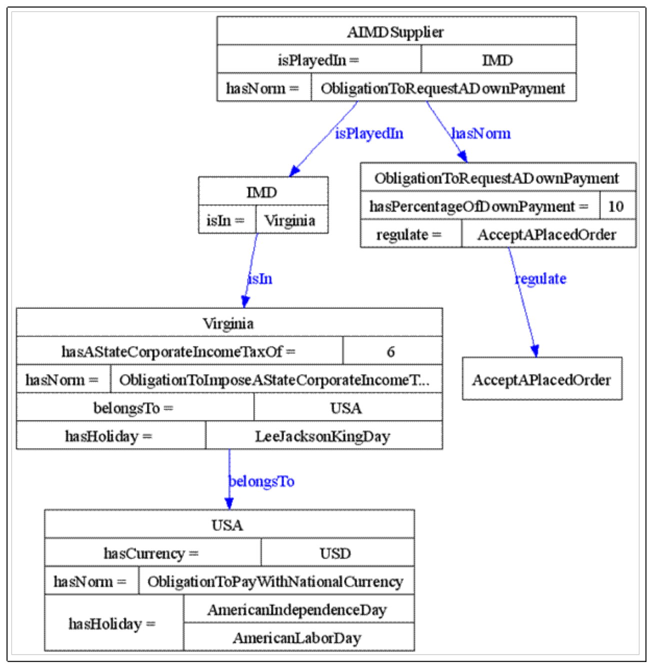

Fig. 3. A contextual role norm for accepting placed orders.

4.4Reasoning in a DynaCROM MAS

In a DynaCROM MAS, norm-aware agents are continuously supported with updated contextual norm information. Thus, they can better adapt themselves for execute correctly according to the enforcement of the system current norms. In our motivating example scenario, manufacturer agents basically can choose to buy with Pintel and Macrostar suppliers, or with IMD and Basus suppliers. Norm-aware agents are more likely to make better choices according to specific criteria, because they are concerned with more precise information. For instance, if the American manufacturer's purchase criteria is to minimize costs, then he should choose to buy the package IMD CPU and Basus components in IMD. There, the final price will be USD 1,144.80 according to the following calculation (all in USD): 650.00 (IMD

CPU) + 300.00 (IMD Motherboard) + 100.00 (Memory) + 150.00 (Hard Disk) = 1,200.00 – 10% (IMD suppliers' discount for orders paid in cash) = 1,080.00 + 6.00% (Virginia's corporate income tax). If he decides to buy in BasusUSA, then the corporate income tax of 8,84% from California should be applied instead of the 6.00% from Virginia and the final price will be USD 1,175.47. However, there the down payment of 10% required by IMD suppliers for accepting placed orders are not anymore necessary and orders will not be delivered during holidays in their final destinations. If the American manufacturer decide to buy during the sales tax holiday season, then it is better for him to buy the bundle Pintel/Macrostar in Pintel. There, the final price will be USD 1,090.00 according to the following calculation (all in USD): 750.00 (Pintel CPU) + 350.00 (Pintel Motherboard) + 150.00 (Memory) + 200.00 (Hard Disk) = 1,450.00 – 15% (Pintel/Macrostar bundle discount). Furthermore, he also can pay a plus tax of 5% in orders to profit from the warranty extension of MacrostarJapan and still will have the price of 1,144.50 (1,090.00 + 5% (for the Macrostar warranty extension)) lower than the final price of USD 1,144.80 from IMD.

--- Insert a part for illustrating the effect of norm updates ---

4.5 Enforcing Domain Contextual Norms

SCAAR norms (structure and data) are written automatically and dynamically by DynaCROM, while agents are executing, and they are based on a DynaCROM domain ontology instance. For instance, Code 3 illustrates the respective SCAAR contextual norms created for representing the norms of the ontology instance illustrated in Fig. 3. In the SCAAR norms, the "environment" variable is instantiated with the "Virginia" value and the "role" variable is instantiated with the "AIMDSupplier" value.

Code 3. DynaCROM domain contextual norms in SCAAR.

```
SCAARNorm1- [(agt: aGenericAgent)
  OBLIGED(agt DO PayWithNationalCurrency
    WITH environment.hasCurrency = "USD")
     IF (agt BE in Environment WITH
       ((environment = "USA") OR
        (environment.belongsTo = "USA"))]

SCAARNorm2- [(agt: aGenericAgent)
  OBLIGED(agt DO ImposeAStateCorporateTax
    WITH  environment.hasAStateCorporateIn-
                                comeTaxOf = "6")
      IF(agt BE in Environment WITH
         (environment = "Virginia")]

SCAARNorm3- [(agt: aGenericAgent)
  OBLIGED(agt DO RequestADownPayment
   WITH  norm.hasPercentageOfDownPayment =

                                          "10")
   AFTER agt DO AcceptAPlacedOrder
      IF(agt BE in Role WITH
                   (role = "AIMDSupplier")]
```

SCAAR is based on DynaCROM outputs – domain contextual norms – for regulating agent actions (e.g., "PayWithNationalCurrency", "ImposeAStateCorporateTax", "RequestADownPayment" and "AcceptAPlacedOrder"). For instance, in order to detect a possible norm violation when a down payment is not given to "AIMDSupplier" when accepting placed orders, SCAAR executes the algorithm from Code 2 as follows: the control hook for the action "RequestADownPayment" of the "AIMDSupplier" agent informs its enforcement core that the action is being performed, and then, the enforcement core creates an instance of a Petri net for representing the norm related to the action (i.e. `SCAARNorm3` from Code 3). The Petri net created for representing the `SCAAR-Norm3` is presented below:

$$PN_1 <P, T, Pre, Post>: \; (\; (\mathbf{p_1}, \mathbf{p_2}, \mathbf{p_3}), \; (t_{role}, \mathbf{t_{RequestADownPayment}}, \mathbf{t_{AcceptAPlacedOrder}}), \; (\mathbf{Pre}(\mathbf{p_1}, \mathbf{t}_{role = \text{"AIMDSupplier"}})), \; (\mathbf{Post}(\mathbf{p_3}, \mathbf{t}_{norm.hasPercentageOfDownPayment = \text{"10"}})) \;)$$

The enforcement core of "AIMDSupplier" waits for information about t_{role} and $t_{norm.hasPercentageOfDownPayment}$ from the control hook responsible for the action "RequestADownPayment". When the agent plays the "AIMDSupplier" role, the enforcement core receives the information about $t_{role = \text{"AIMDSupplier"}}$. Then, the place $\mathbf{p_1}$ in PN_1 is activated and the transition is fired by putting the Petri net token in the next places (i.e., from $\mathbf{p_1}$ to $\mathbf{p_2}$ and, then, from $\mathbf{p_2}$ to $\mathbf{p_3}$). When the agent tries to perform the action "AcceptA-PlacedOrder" with $t_{norm.hasPercentageOfDownPayment}$ = "0", the place $\mathbf{p_3}$ in PN_1 generates an exception and forwards it to the agent's enforcement core, which blocks the execution of the action.

5. RELATED WORK

Normative MASs as an area of research has become a major issue in the MAS field and it can be defined as the intersection of normative systems and MASs. A normative MAS is an MAS where agents can decide whether to follow the explicitly represented norms without any presumption about the internal workings of an agent or the way norms find their expression in an agent's behavior [6]. Important works concerning regulations in the domain of MASs, such as [4, 51, 44, 11, 25, 26 and 27], have been proposed recently. However, these solutions seem to lack the necessary flexibility and dynamics for norm evolution in open systems. The use of contexts permits a more precise mechanism to deal with norm evolution. Moreover, the use of ontologies and ontology-driven rules provides a reasonable dynamics for norm evolution, at system run-time, and also a meaningful way for heterogeneous agents to interpret precise contextual norm information in open MASs.

5.1 MOSES

MOSES 9 is the middleware that supports LGI (meaning *law-governed interaction*) 9, which is a decentralized coordination and control mechanism for distributed systems. LGI enables a distributed group of software actors – which may be heterogeneous, open and large – to engage in a mode of interaction governed by an explicitly specified policy called the "*law*" of this group.

Although being a well-known solution for law enforcement in distributed systems, LGI has two main limitations while enforcing norms in open MASs. The first is that LGI does not offer the support to directly enforce contextual norms; it only supports to directly enforce interaction laws. The second limitation is that LGI lacks dynamics while evolving law information. This is because, a LGI community is formed by (*LGI*) agents operating under a unique ***static*** law that must be already created when agents join it.

In order to enforce DynaCROM contextual norms by using LGI, it is necessary to decouple norm information from different levels of abstractions to the interaction level. We regard here that communicative acts, established in the interaction level, also can be viewed as organization acts, i.e. actions performed within an organization modify a fragment of social reality 9. To give the necessary dynamics for norm enforcement, DynaCROM use its output (agents' updated contextual norms) for activating LGI predefined norms, acting as a trigger mechanism. Only norms sent by DynaCROM are enforced by LGI, even if those norms are already defined in a LGI law. This is a dynamic solution because DynaCROM outputs are based on a domain ontology instance, which normally evolves according to social changes characteristic of open MASs.

The integration between DynaCROM and MOSES is illustrated in Fig. 4 and Fig. 5 by using an example of a contextual norm enforcement for regulating payments in California. The norm enforcement in a DynaCROM MAS is guaranteed by LGI police agents that are automatically created when regulated agents incorporate the DynaCROM behavior. Police agents are in charge of filtering messages exchanged between agents in order to ensure that the actions performed in the system are in conformance with their correspondent norms. Four agents play in the example. The "ACalifornianManufacturer" agent plays a manufacturer role and it has the "ACalifornianManufacturerPolice" agent for enforcing its norms. The "AMacrostarSupplier" agent plays a supplier role from a "Macrostar" organization and it has the "AMacrostarSupplierPolice" agent for enforcing its norms.

The scenario starts when the "ACalifornianManufacturer" sends a CFP message to the "AMacrostarSupplier" (see 1 from Fig. 4). The "ACalifornianManufacturerPolice" only forwards this message (see 2 from Fig. 4). When the message arrives in the "AMacrostarSupplierPolice" (see 3 from Fig. 4), he adds the information about the expected currency for answering the CFP (see 4 from Fig. 4) and delivers the message to the "AMacrostarSupplier" (see 5 from Fig. 4).

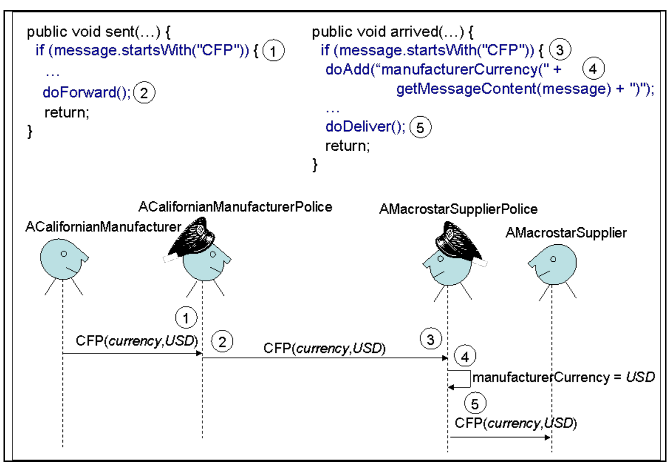

Fig. 4. Adding the information about the currency expected.

When the "AMacrostarSupplier" receives the CFP message, he answers it with a PROPOSE message to the "ACalifornianManufacturer". This message has its proposed currency and price (see 1 from Fig. 5). When the PROPOSE message arrives in the "AMacrostarSupplierPolice", the DynaCROM "ObligationToPayWithNationalCurrency" contextual norm is effectively enforced by the execution of the MOSES "checkCurrency" obligation norm (see 2 and 3 from Fig. 5). Because the proposed and expected currencies are the same, the "AMacrostarSupplierPolice" forwards the PROPOSE message (see 4 from Fig. 5) and informs a "Ok!" message to the "AMacrostarSupplier" (see 5 from Fig. 5). When the message arrives in "ACalifornianManufacturerPolice", he only delivers it to the "ACalifornianManufacturer" (see 6 from Fig. 5).

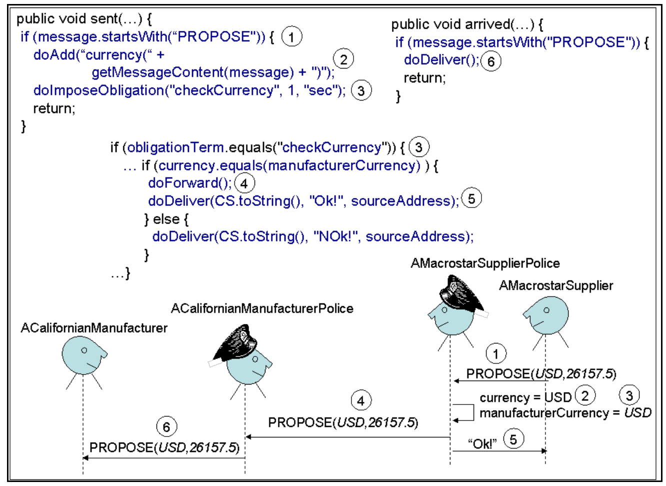

Fig. 5. Enforcing a contextual norm for regulating payments.

5.2OMNI

Bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla

5.3eInstitutions

Bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla-bla

6.CONCLUSION

The motivating question of our research is how to easily implement regulation in open MASs permitting heterogeneous agents to perform efficiently and coherently. The thesis we held here is that the complexity of norm management in open MASs can be decreased by decoupling information in contexts.

Our ongoing work, named DynaCROM, intends to be a straightforward method for smoothly applying and managing regulations in open MASs as well as for enforcing precise contextual norms. DynaCROM is still a work in progress, but we agree that it already has important contributions for the domain of regulation in open MASs. DynaCROM's main contributions are: (i) a definition of a top-down classification for contextual norms, which facilitates the tasks of elucidation, organization and management of norm information; (ii) a contextual normative ontology to explicitly represent the semantic of classified norms in a meaningful way (i.e., with a common understanding) for heterogeneous agents; (iii) a definition of a norm composition process, based on ontology-driven rules, that makes it easy to update the system regulation by both evolving norms in a unique resource (an ontology) and by activating particular rules for acquiring customized compositions of contextual norms; and (iv) a solution for enforcing contextual norms.

DynaCROM is not tightly coupled with a particular enforcement mechanism. In this paper, we present the first results of DynaCROM integrated with SCAAR and MOSES. For future work, we are currently implementing a solution for enabling SCAAR (implemented in SICStus Prolog 9) to be the fully norm enforcement mechanism of DynaCROM (implemented in JAVA). We believe that SCAAR performs better than MOSES while enforcing DynaCROM contextual norms because, in SCAAR, norms can be directly enforced in any contextual level without the need to decompose them into the interaction level.

7.ACKNOWLEDGMENTS

To ESSMA (CNPq 552068/2002-0) and EMACA (CAPES/Cofecub 482/05/07) projects, and to CNPq individual grants.

8.ADDITIONAL AUTHORS

Jean-Pierre Briot (LIP6–PARIS6, jean-pierre.briot@lip6.fr) and Carlos Lucena (DI–PUC-Rio, lucena@inf.puc-rio.br)

9.REFERENCES

[1] Artikis, A., Pitt, J., Sergot, M.: Animated specifications of computational societies, In AAMAS, 2002, 1053–1061

[2] Boella, G.; Torre, L.; Verhagen, H.: Introduction to normative multiagent systems, In the "Journal of Comput. Math. Organ. Theory", 12(2-3), 2006, 71–79

[3] Bouquet, P., Giunchiglia, F., Harmelen, F.v., Serafini, L., Stuckenschmidt, H.: C-OWL: Contextualizing Ontologies, In Proc. of the ISWC-03, LNCS 2870, Springer, 2003, 164–179

[4] Castelfranchi, C., Dignum, F., Jonker, C.M., Treur, J.: Deliberative Normative Agents: Principles and Architecture, 1999, In Proc. of the ATAL-99, 1999, 364–378

[5] Chopinaud, C., Seghrouchni, A.E.F., Taillibert, P.: Prevention of harmful behaviors within cognitive and autonomous agents, In Proc. of the ECAI'06, 205–209

[6] Collins, H.; Arunachalam, R.; Sadeh, N.; Eriksson, J.; Finne, N. and Janson, S.: The Supply Chain Management Game for the 2007 Trading Agent Competition. CMU-ISRI-07-100. 2007. Link: http://www.sics.se/tac/tac07scmspec.pdf

[7] Dey, A.: Understanding and using context, In "Personal and Ubiquitous Computing", 5(1), 2001, 4–7, ISSN: 1617-4909

[8] ***: An Approach for Contextual Regulations in Open MAS, In the AOIS@AAMAS-2006, 2006a, Japan

[9] ***: Supporting Regulatory Dynamics in Open MAS. In the COIN@AAMAS-2006, 2006b, Japan

[10] ***: Dynamic Contextual Regulations in Open Multi-agent Systems. In Proc. of the ISWC, 06, LNCS 4273, 2006d, 974–975, ISSN: 0302-9743

[11] ***: Providing Contextual Norm Information in Open Multi-Agent Systems. Extended version of 9, invited for the AOIS 06 post proceedings to be published in a LNAI Springer volume, Submitted, 2007a

[12] ***: Informing Regulatory Dynamics in Open MASs. Extended version of 9, invited for the COIN 06 post proceedings to be published in a LNAI Springer volume, Submitted, 2007b

[13] Ferber, J., Gutknecht, O., Michael, F.: From Agents to Organizations: an Organizational View of Multi-agent Systems, In the Proc. of the "Agent-Oriented Software Engineering IV" (AOSE, 2003), 2003, LNCS 2935, 214–230

[14] FIPA Contract Net Interaction Protocol Specification, 2003, Link: http://www.fipa.org/specs/fipa00029/SC00029H.html

[15] Fornara, N., Viganò, F., Colombetti, M.: Agent communication and artificial institutions, In the "Journal of Agents and Multi-Agent Systems" (JAAMAS, 07), 2007, 14:121-142

[16] Gaertner, D., Clark, K., Sergot, M.: Ballroom etiquette: a case study for norm-governed multi-agent systems. In the COIN@AAMAS-2006, 2006, Japan

[17] Grizard, A., Vercouter, L., Stratulat, T., Muller, G.: A peer-to-peer normative system to achieve social order. In the COIN@AAMAS-2006, 2006, Japan

[18] Gruber, T. R.: A translation approach to portable ontology specifications. In "Knowledge Acquisition", 5 (2), 1993, 199–220, ISSN: 1042-8143

[19] Henricksen, K., Indulska, J.: Developing context-aware pervasive computing applications: models and approach. Pervasive and Mobile Computing, In Press, Elsevier, 2005

[20] Hewitt, C.: Open Information Systems Semantics for Distributed Artificial Intelligence, In "Artificial Intelligence", 47, Issue 1-3, 1991, 79–106, ISSN: 0004-3702

[21] JADE, 2007, Link: http://jade.tilab.com

[22] Jennings, N. R.: On Agent-Based Software Engineering, In "Artificial Intelligence", 117 (2) 277-296, 2000

[23] Khedr, M, Karmouch, A.: ACAI: Agent-Based Context-aware Infrastructure for Spontaneous Applications, In the "Journal of Network & Computer Applications", 28(1), 1995, 19–44

[24] Minsky, N.: Law Governed Interaction (LGI), Link: http://www.cs.rutgers. edu/~minsky/papers/manual.pdf

[25] Minsky, N.: MOSES, 2007, http://www.moses.rutgers.edu/

[26] Murata, T.: Petri nets: Properties, analysis and applications. In IEEE 77(4). (1989) 541–580

[27] Noy, N; Rector, A. (Eds.).: Defining N-ary Relations on the Semantic Web: Use with Individuals, 2007, Link: http://www.w3.org/TR/swbp-n-aryRelations

[28] Ontoviz. Link: http://protege.cim3.net/cgi-bin/wiki.pl?OntoViz

[29] Searle, R.: The construction of social reality. NY Free Press

[30] SICStus Prolog, 2007, Link: http://www.sics.se/isl/sicstuswww/site/

[31] Silva, V.T.da.: From a conceptual framework for agents and objects to a multi-agent system modeling language, In the JAAMAS, 04, 9(1-2), 2004, 145–189, ISSN 1387-2532

[32] Thomas, G., Williams, A.B.: Roles in the Context of Multiagent Task Relationships. In Proc. of the AAAI Fall Symposium TR FS-05-08, ISBN 978-1-57735-254-9, 2005

[33] Weyns, D.; Omicini, A.; Odell, J.: Environment as a first class abstraction in multiagent systems, In the JAAMAS- 07, vol. 14, number 1, 2007, 5–30, ISSN 1387-2532