Informing Regulatory Dynamics in Open MASs

Carolina Felicíssimo[1], Ricardo Choren[2], Jean-Pierre Briot[1, 3], and Carlos Lucena[1]

[1] DI, PUC-Rio, Rua Marquês de São Vicente, 225, Gávea
Rio de Janeiro, RJ, 22453-900, Brasil
{cfelicissimo, lucena}@inf.puc-rio.br
[2] SE/8 – IME, Praça Gen Tibúrcio 80, Urca
Rio de Janeiro, RJ, 22290-270, Brasil
choren@de9.ime.eb.br
[3] LIP6, 8 rue du Capitaine Scott
75015 Paris, France
jean-pierre.briot@lip6.fr

Abstract. We believe that, in the near future, all multi-agent systems (MASs) will be open, permitting agents to migrate among MASs to obtain resources or services not found locally. In this scenario, open MASs should be enhanced with norms for restricting agents' actions and, consequently, avoiding unexpected behavior. In this work, we present a case study where an open MAS is enhanced with contextual norms. Agents from this MAS are continuously supported with precise norm information, according to their contexts (implicit situational information) and, thus, can make better decisions. Although the presented case study is very simple, it clearly shows how agents can be influenced by norm information while acting in regulated open MASs.

1 Introduction

Multi-agent systems (MASs) have emerged as a powerful technology for developing information systems that clearly require several goal-oriented problem-solving entities [21]. Information systems tend to be both: formed of autonomous entities and without centralized control [17]. Following this direction, we believe that, in the near future, all MASs will be open and composed of many sets of heterogeneous self-interested agents, migrating among MASs for obtaining resources or services not found locally. Because agents' actions will probably deviate from expected behavior, following individual goals, regulatory mechanisms will be a fundamental feature of open MASs.

Important works concerning regulations in open MASs, as [1], [3], [12], [13] and [14], have been proposed recently. However, in these works, it is missed a precise mechanism for regarding the different levels of norm abstractions. Consequently, it is hard to define specific norms for particular cases and to evolve norms. Thus, we are currently working on an approach for explicitly supporting regulation in open MASs. Our approach, called DynaCROM (*dynamic contextual regulation information provision in open MASs*) [6], [7], [8], [9], continuously provides precise norm information according to agents' contexts. DynaCROM is based on a top-down modeling of contextual norms, on a meta-ontology for representing norm semantics and on a rule mechanism for composing norms. Furthermore, DynaCROM implementation is summarized as an agent behavior, independent of agents' original codes. Norm-aware agents can use DynaCROM answers (updated contextual norms) to make better decisions and, thus, achieve their goals faster. Developers of regulations in open MASs can use DynaCROM as a flexible solution for updating and managing systems' norms at run-time.

This work presents a case study where DynaCROM continuously provides updated contextual norm information to agents, according to their different environments, organizations, roles and interactions. These agents use norm information while making decisions. The remainder of this paper is organized as follows: Section 2 briefly presents DynaCROM; Section 3 describes our case study; Section 4 compares DynaCROM with related works; and, finally, Section 5 concludes the work and outlines directions for future works.

2 Norm-Aware Open Multi-Agent Systems

MASs are generally made up of environments, organizations and agents [20]. Environments [28] are discrete computational locations (similar to places in the physical world) that provide conditions for agents to inhabit it. Organizations [10] are social locations where groups of agents play roles inside it. Roles are abstractions that define a set of related tasks for agents achieving their designed goals [26]. Agents interact with others, from the same or different organizations and environments.

Environments, organizations, roles and interactions suggest different contexts for agents in MASs. Contexts can be defined as pieces of information that characterize the situation of participants [1]. Context-aware systems use contexts to provide relevant information and/or services to their users, where relevancy depends on the users' tasks [1]. In our definition, regulated context-aware MASs provide updated contextual norm information to their agents.

Researches into context-aware applications suggest top-down architectures for modeling contextual information [22]. Following this direction, DynaCROM suggests to model the norms of an open MAS in *Environment*, *Organization*, *Role* and *Interaction* contexts. These *regulatory contexts* are differentiated by their boundaries. 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. *Interaction norms* are applied to all agents involved in a regulated interaction.

DynaCROM regulatory contexts and their data (norms) are explicitly represented by an ontology, which provides a meaningful understanding for heterogeneous agents. For the DynaCROM ontology, the following definitions are valid: an *ontology* is a conceptual model that embodies shared conceptualizations of a given domain [16]; a *contextual ontology* is an ontology that has contextual information [1]; and, a *contextual normative ontology* is an ontology that has contextual norm information.

Fig. 1 illustrates the DynaCROM ontology. It is made up of six related concepts. The *Action* concept encompasses all instances of regulated actions. The *Penalty* concept encompasses all instances of fines to be applied when norms are not fulfilled. The *Norm* concept encompasses all instances of norms from all regulatory contexts. The *Environment* concept encompasses all instances of regulated environments; and, each environment encompasses its associated norms and its owner environment (the environment it belongs to). The *Organization* concept encompasses all instances of regulated organizations; and, each organization encompasses its associated norms, main organization (the organization to which it is associated) and environment. The *Role* concept encompasses all instances of regulated roles; and, each role encompasses its associated norms and organization.

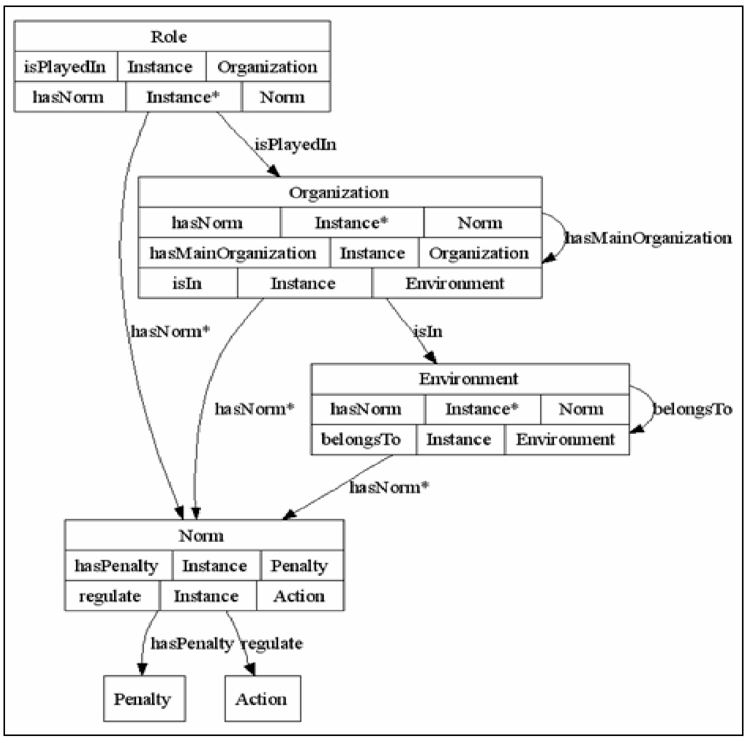

Fig. 1. The DynaCROM ontology

While regulating open MASs from different domains, the DynaCROM ontology must be instantiated with particular domain instances and it can be extended with domain concepts and interaction norms. Interaction norms should be implemented by following a representation pattern, from the Semantic Web Best Practices document [24]. This pattern defines that the relation object itself must be represented by a created concept that links the other concepts from the relation (i.e., reification of the relationship). So, in DynaCROM ontologies, an interaction norm should be represented by a new Norm sub-concept linking two Role concepts.

Norms should control environments, organizations, agent roles and agent interactions by defining which actions are *permitted*, *obliged* and *prohibited*. A *permitted norm* defines that an act is allowed to be performed; an *obliged norm* defines that an act must be performed; and a *prohibited norm* defines that an act must not be performed. Norms from related regulatory contexts should be easily composed during systems' run-time. For this, DynaCROM uses rules and a rule inference engine for both composing related contextual norms and informing them to agents.

The main idea behind using rules is to permit dynamics and flexibility while compositing related contextual norms. Instead of spread implementations of norm compositions in agents and in regulated systems, these implementations are centralized in a rule file. While executing as an active behavior, DynaCROM keeps reading both a domain ontology instance, for getting its data, and the rule file for compositions of contextual norms. Thus, all informed information is updated.

Modular context refinements allow a more flexible system, providing a better support to manage regulatory dynamics. DynaCROM offers a simple way to manage norm evolutions, without the need to stop the system execution, in two different cases. The first case is when norms need to be added, updated or deleted. For this case, simply updating the ontology instance makes the evolution done. The second case is when new compositions of contextual norms are desired. For this case, simply updating the specific rules for the new compositions concludes the evolution. DynaCROM execution can be summarized by the following tasks: read the ontology instance for getting both data and how concepts are structured; read a rule file for getting how concepts must be composed; and, finally, infer an ontology instance based on the previous readings. Fig. 2 illustrates an overview of the DynaCROM execution process.

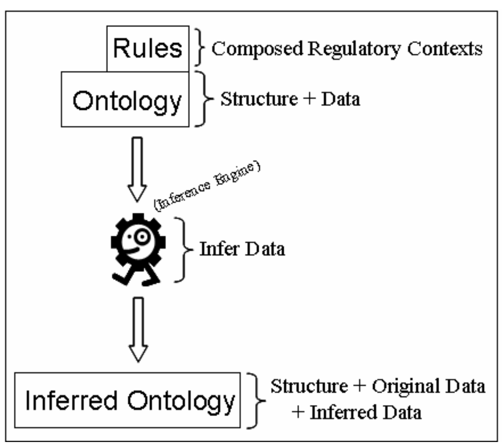

Fig. 2. The DynaCROM execution process

3 Case Study

The domain of multinational corporations is used to present our case study. This domain was chosen because it well illustrates important implicit contextual information found in MASs. In our case study, a regulated open MASs continuously provides updated contextual norm information, permitting agents to make better decisions.

The world of our case study is created as follows: Canada and the United States of America are environments located in the North America environment; Argentina, Brazil and Chile are environments located in the South America environment; Hpie Canada and Hpie Argentina organizations are branches from the Hpie main organization; Dellie Brazil and Dellie Chile organizations are branches from the Dellie main organization; Dellie organizations have the supplier, manufacturer and customer roles; and, Hpie organizations have the supplier, manufacturer, distributor, retailer and customer roles. All entities from our world are illustrated in Fig. 3.

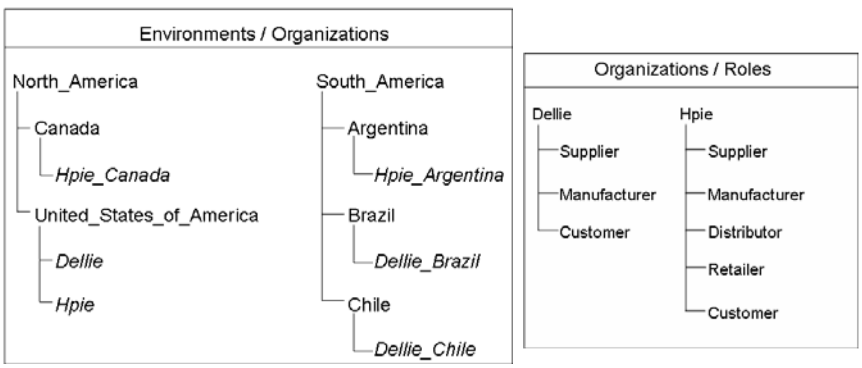

Fig. 3. The environments, organizations and roles from our case study

3.1. Examples of Environment, Organization, Role and Interaction Norms

Normally, organizations do not make their norms public because they are of strategic importance to their businesses. Because of this, based on the available information collected from several corporate Web sites, we created norms for our environments, organizations, roles and agent interactions and classified them according to our four regulatory contexts.

3.1.1. Examples of Environment Norms:

a. In North America, the price of a finished good from every organization has a percentage of its price (depending on the seller's location) added as taxes if the delivery is immediate (carry-out) or if the delivery address is in the seller's location.

b. In Canada, a finished good from every organization has 15% of the price value added as taxes if the delivery is immediate (carry-out) or if the delivery address is in Canada.

c. In the state of the Dellie headquarters (in the United States of America), a finished good from every organization has 8% of the price value added as taxes if the delivery is immediate (carry-out) or if the delivery address is in the state of the Dellie headquarters.

d. In the state of the Hpie headquarters (in the United States of America), a finished good from every organization has 5% of the price value added as taxes if the delivery is immediate (carry-out) or if the delivery address is in the state of the Hpie headquarters.

e. In South America, taxes are included in the price of every finished good.

3.1.2. Examples of Organization Norms:

a. Hpie organizations have to follow the direct sales to customer model, i.e. sales of the organization's products can only be made between: suppliers and manufacturers, or manufacturers and distributors, or distributors and retailers, or retailers and customers.

b. In Hpie Argentina, sales of the organization's products can only be made between: suppliers and manufacturers, or manufacturers and distributors, or distributors and retailers, or distributors and customers, or retailers and customers.

c. In Dellie organizations, only suppliers and manufacturers are permitted to sell organization's products to customers.

d. In Dellie Chile, sales of the organization's products can only be made between: suppliers and manufacturers, or manufacturers and customers.

3.1.3. Examples of Role Norms:

a. In Dellie, customers receive only complete orders.

b. In Hpie Canada, suppliers must ship orders on their due dates.

c. In Dellie Brazil, suppliers must ship orders until their due dates.

d. In Dellie Brazil, customers must receive orders until one day after their due dates.

e. In Hpie Argentina, customers must make a down payment of 10% for every order placed.

3.1.4. Examples of Interaction Norms:

a. In Dellie, manufacturers have permission to pay in up to 30 days after they receive their orders from suppliers.

b. In Dellie Brazil, manufacturers have a 10% discount off the total price of their orders if the payment to their suppliers is made in cash.

c. In Hpie Canada, suppliers have the permission to ship incomplete orders to manufacturers.

For our case study, the DynaCROM ontology was extended for representing both interaction norms and the role sub-concepts (supplier, manufacturer, distributor, retailer and customer). Then, the extended ontology was instantiated for representing all the domain norms written above. Fig. 4, Fig. 5, Fig. 6 and Fig. 7 illustrate different parts of the DynaCROM ontology instance created for our case study.

3.2. Applying Environment, Organization, Role and Interaction Norms

The following five subsections present different issues resulting from the application of contextual norms. Subsections 3.2.1 and 3.2.2 present scenarios where norm-aware agents make decisions based on given norm information. Subsections 3.2.3 and 3.2.4 exemplify, respectively, restriction and relaxation of contextual norms. Finally, subsection 3.2.5 exemplifies how composition of contextual norms can generate conflicts.

3.2.1. A Scenario where Customers Need their Orders in Due Dates

For exemplifying how norm-aware agents can make decisions based on given norm information, a scenario is given with a customer in North America looking for Hpie products. This customer needs his orders on the due dates. For minimizing delivery expenses, the customer will choose to buy in Hpie or in Hpie Canada (Hpie organizations in North America) depending on their current norms. Fig. 4 illustrates the current norms related to the Hpie and Hpie Canada contexts.

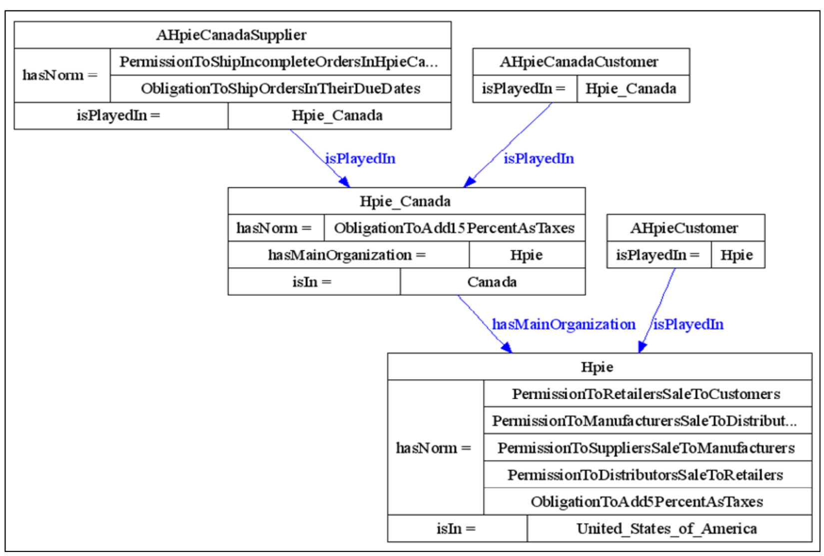

Fig. 4. Norms related to the Hpie and Hpie Canada contexts

If the customer decides to buy in Hpie (being "*AHpieCustomer*"), he is restricted to buying products only from retailers (organization norm 3.1.2a), but he pays only 5% of the price value as taxes if the delivery is immediate (carry-out) or if the delivery address is in the state of the Hpie headquarters (environment norm 3.1.1d).

If the customer decides to buy in Hpie Canada (being "*AHpieCanadaCustomer*"), he has to pay 15% of the price value as taxes if the delivery is immediately (carry-out) or if the delivery address is in Canada (environment norm 3.1.1b). In Hpie Canada, the customer can also buy direct from suppliers and, doing that, he has the guarantee that his orders will be shipped on their due dates (role norm 3.1.3b). However, if Hpie Canada is also regulated through Hpie norms (its main organization norms), the customer is restricted to buying products only from retailers (organization norm 3.1.2a), but he pays only 5% of the price value if the delivery is immediate (carry-out) or if the delivery address is in the state of the Hpie headquarters (environment norm 3.1.1d).

Because Hpie and Hpie Canada are organizations in North America, both are also regulated through the North America environment norm 3.1.1a. This norm is more general than the environment norms 3.1.1b and 3.1.1d and, thus, does not affect the current regulation.

3.2.2. A Scenario where Manufacturers Look for Good Deals with Suppliers

For another example of how norm-aware agents can make decisions based on given norm information, a scenario is given with a manufacturer in North America looking for suppliers. This manufacturer has flexibility for choosing good deals with suppliers. For minimizing delivery expenses, the customer can choose to buy with Dellie, Hpie or Hpie Canada suppliers (North America suppliers). Fig. 5 illustrates the current norms related to the Dellie context.

If the manufacturer decides to buy in Dellie with one of the Dellie suppliers (being "*ADellieManufacturer*"), he has the benefit payoff being able to pay in up to 30 days after he receives his orders (interaction norm 3.1.4a). Besides this, he pays 8% of the price value as taxes if the delivery is immediately (carry-out) or if the delivery address is in the state of the Dellie headquarters (environment norm 3.1.1c).

If the manufacturer decides to buy in Hpie Canada with one of the Hpie Canada suppliers (being "*AHpieCanadaManufacturer*"), he has the permission to receive incomplete orders before their due dates (interaction norm 3.1.4c). However, he has to pay 15% of the price value as taxes if the delivery is immediate (carry-out) or if the deliver address is in Canada (environment norm 3.1.1b).

If the manufacturer decides to buy in Hpie with one of the Hpie suppliers (being "*AHpieManufacturer*"), he pays only 5% of the price value as taxes if the delivery is immediate (carry-out) or if the delivery address is in the state of the Hpie headquarters (environment norm 3.1.1d).

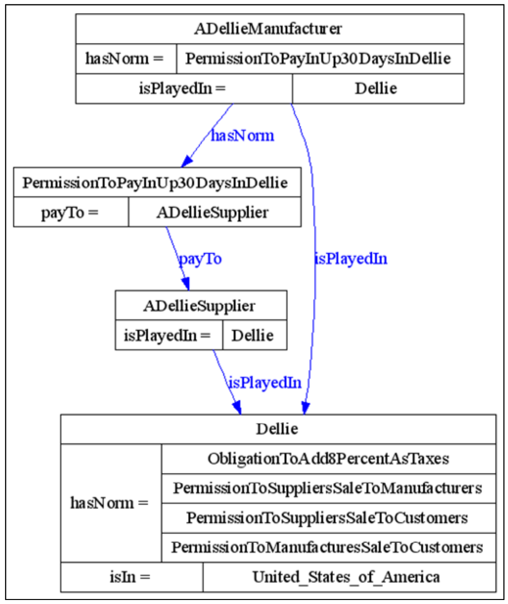

Fig. 5. Norms related to the Dellie context

3.2.3. A Scenario where Norms Are Restricted

For exemplifying restriction of contextual norms, a scenario is given with organization norms 3.1.2a and 3.1.2b. Hpie Argentina is regulated through the organization norm 3.1.2b, but as Hpie is its main organization, it is also regulated through the Hpie organization norm 3.1.2a. Thus, by the composition of contextual norms, Hpie Argentina distributors are no longer allowed to sell directly to customers. This scenario is illustrated in the left side of Fig. 6. (note that the dashed norm from the left side of Fig. 6. – "*PermissionToDistributorsSaleToCustomers*" – is not presented in Hpie).

3.2.4. A Scenario where Norms Are Relaxed

For exemplifying relaxation of contextual norms, a scenario is given with organization norms 3.1.2d and 3.1.2c. Dellie Chile is regulated through the organization norm 3.1.2d, but as Dellie is its main organization, it is also regulated through the Dellie organization norm 3.1.2c. Thus, by the composition of contextual norms, Dellie Chile suppliers are now allowed to sell direct to customers. This scenario is illustrated in the right side of Fig. 6. (note that the dashed norm from the right side of Fig. 6. – "*PermissionToSuppliersSaleToCustomers*" – is only presented in Dellie).

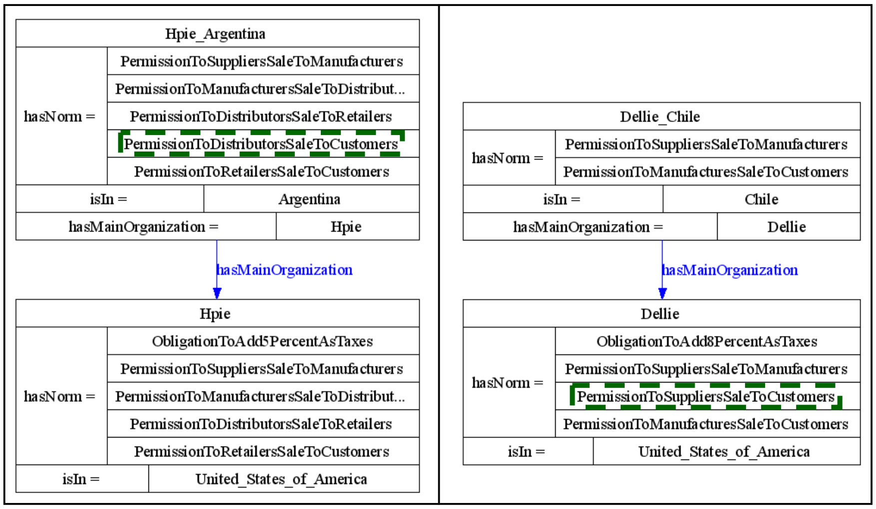

Fig. 6. Compositions of contextual norms resulting in restriction and relaxation of norms

3.2.5. A Scenario where Norms Are Conflicting

For exemplifying how composition of contextual norms can generate conflicts, a scenario is given with the role norm 3.1.3c, from Dellie Brazil suppliers, and with the role norm 3.1.3d, from Dellie Brazil customers. These norms state the same subject (deadline to ship orders) in an opposite way. The role norm 3.1.3c states that suppliers are obliged to ship orders until their due dates, but the role norm 3.1.3d states that customers can receive their orders until one day after their due dates. Fig. 7 illustrates this scenario.

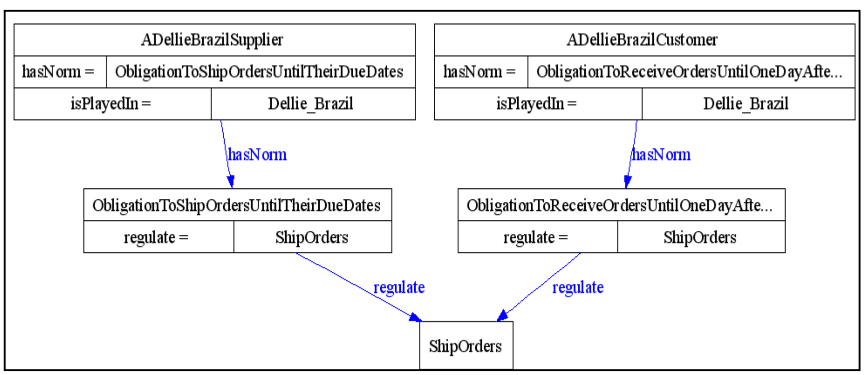

Fig. 7. Compositions of contextual norms resulting in a conflict for the action of ship orders

It is important to remark here that, in this work, we do not make any assumptions about the problem of how to resolve raised conflicts when norms state the same subject in an opposite way. However, we suggest enhancing conflicted norms with priorities, as a very simple idea to minimize the problem.

3.3. Case Study Implementation

Our case study was implemented inside the Eclipse platform [5], using the Java programming language [15] and the Jena API [19]. The Jena API was used as a programmatic environment for OWL [23] and as a rule-based inference engine (rules were written according to the Jena rule syntax [19]). The Protégé Editor [25] was used to extend and instantiate the DynaCROM ontology. Our agents were implemented in JADE [18], extending its *Agent class* with both an attribute for agents' locations and two specific behaviors. One behavior, called *Migratory*, makes agents move randomly from one location to another. The other behavior, called *Normative*, continuously informs agents about their current contextual norms. Once an agent migrates, its location attribute is updated and, consequently, the answers from the *Normative* behavior change, informing the new contextual norms to which the agent is currently bound. Fig. 8 illustrates the code responsible for adding the *Migratory* and *Normative* behaviors inside our JADE agents.

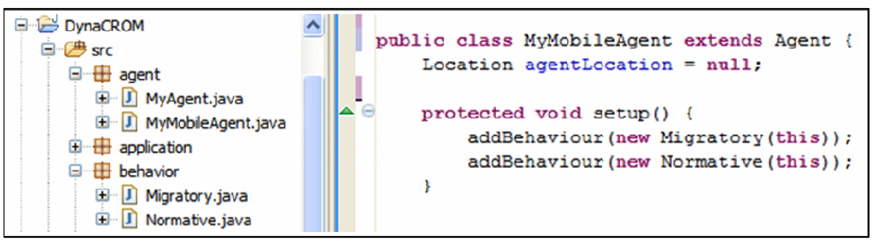

Fig. 8. Adding the *Migratory* and *Normative* behaviors inside our agents

JADE containers were used for representing the abstractions of environments and organizations. North America, South America, Canada, the United States of America, Argentina, Brazil, Chile, Dellie, Hpie, Hpie Canada, Hpie Argentina, Dellie Brazil and Dellie Chile are all the JADE containers created for our case study. These containers offer the technical support for agents with the *Migratory* behavior change locations. Fig. 9 illustrates the JADE containers for the United States of America, Canada, Brazil and Chile environments with some agents inside them. For instance, in Brazil there is an agent, called "******MobileAgent1*", with the *Migratory* and *Normative* behaviors. Once this agent migrates, its location attribute is updated. Subsequently, the *Normative* behavior gets the new agent location and, then, informs the contextual norms to which it is currently bound.

Fig. 9. Part of our system's world implemented as Jade containers

Our *Normative* behavior uses rules for compositions and retrievals of contextual norms. These rules are *ontology-driven*, i.e. they are created based on how DynaCROM regulatory concepts are linked to each other (see the structure of its ontology in Fig. 1). Rules can be activated and deactivated, at run time, for changing the current compositions of contextual norms. To activate rules, it is necessary to remove rules' comment marks; to deactivate rules, it is necessary to insert rules' comment marks, both from a rule file.

All the rules used for the scenarios described in the previous subsections are presented in Table 1. When **Rule 1** is activated, it states that a given environment will also be regulated with its owner environment norms; when **Rule 2** is activated, it states that a given organization will also be regulated with its main organization norms; when **Rule 3** is activated, it states that a given organization will also be regulated with its environment norms; when **Rules 1**, **2** and **3** are activated, they state that a given organization will also be regulated with the norms from its main organization and environment; when **Rule 4** is activated, it states that a given role will also be regulated with its organization norms. When **Rules 1**, **2**, **3** and **4** are activated, they state that a given role will also be regulated with the norms from its organization, its organization's main organization and environments.

Table 1. Rules for compositions of contextual norms

Rule 1- [ruleForEnvironmentWithOwnerEnvironmentNorm:
(?Environment *belongsTo* ?OwnerEnvironment)
(?OwnerEnvironment *hasNorm* ?OwnerEnvironmentNorm)
-> (?Environment *hasNorm* ?OwnerEnvironmentNorm)]

Rule 2- [ruleForOrganizationWithMainOrganizationNorm:
(?Organization *hasMainOrganization* ?MainOrganization)
(?MainOrganization *hasNorm* ?MainOrganizationNorm)
-> (?Organization *hasNorm* ?MainOrganizationNorm)]

Rule 3- [ruleForOrganizationWithEnvironmentNorm:
(?Organization *isIn* ?Environment)
(?Environment *hasNorm* ?EnvironmentNorm)
-> (?Organization *hasNorm* ?EnvironmentNorm)]

Rule 4- [ruleForRoleWithOrganizationNorm:
(?Role *isPlayedIn* ?Organization)
(?Organization *hasNorm* ?OrganizationNorm)
-> (?Role *hasNorm* ?OrganizationNorm)]

The *Normative* behavior represents the core of DynaCROM. This is because it is responsible for implementing its execution process, illustrated in Fig. 2. The most important part of this implementation is presented in Table 2. The DynaCROM process starts when the "*getOntModel()*" method (see line 8) retrieves both the ontology structure (related regulatory contexts) and data (norms). The defined compositions of contextual norms are defined by activations and deactivations of rules written in the "*rulesToComposeNorms.rules*" file (called in line 4). The "*reasoner*" variable (see line 5) represents the rule-based inference engine which, based on the retrieved ontology instance and active rules, automatically deduces the defined compositions of contextual norms. This result is kept in the "*inferredModel*" variable (see line 7), which will be used by DynaCROM for informing current contextual norms for agents.

Table 2. The core of the DynaCROM implementation

```
Model m = ModelFactory.createDefaultModel();                             (1)
Resource configuration =  m.createResource();                            (2)
configuration.addProperty(ReasonerVocabulary.PROPruleSet,                (3)
 ontologyDir.concat("rulesToComposeNorms.rules"));                       (4)

Reasoner reasoner =                                                      (5)
 GenericRuleReasonerFactory.theInstance().create(configuration);         (6)

InfModel inferredModel =                                                 (7)
 ModelFactory.createInfModel(reasoner, this.getOntModel());              (8)
```

4 Related Work

Our work was compared to OMNI (*Organizational Model for Normative Institutions*) [27]. OMNI is a framework for modeling agent organizations composed of three dimensions: *Normative*, *Organizational* and *Ontological*. OMNI contains the three levels of abstractions with increasing implementation detail: the *Abstract Level*, which has the statutes of the organization to be modeled, the definitions of terms that are generic for any organization and the ontology of the model itself; the *Concrete Level*, which refines the meanings defined in the previous level, in terms of norms and rules, roles, landmarks and concrete ontological concepts; and, finally, the *Implementation Level*, which has the Normative and Organizational dimensions implemented in a given multi-agent architecture with the mechanisms for role enactment and for norm enforcement.

Comparing our work with OMNI, both define a meta-ontology with a taxonomy for regulations in open MASs and use norms to recommend right and wrong behavior. The use of norms can inspire trust in regulated MASs. One difference is that, in OMNI, enforcement is carried out by any internal agents from the system while in our work it can be carried out by some trusted agents or by some system's enforcement mechanisms. A second difference, and the most important, is that in OMNI the idea of regulatory contexts is not explicit and separated in different levels of abstractions, especially for the environment and role norm contexts. Our approach is based on the environment, organization, role and interaction regulatory contexts to simplify the enforcement and evolution processes. For instance, the social structure of an organization in OMNI describes, in the same level of abstraction, norms for roles and groups of roles. Group of roles is used to specify norms that hold for all roles in the group. We use the organization regulatory context to specify organization norms that hold for all roles from an organization and use the role regulatory context to specify role norms, both regulatory contexts from different levels of abstractions.

In [13], a distributed architecture for endowing MASs with a social layer is proposed. This architecture explicitly represents and manages normative positions via rules. Every external agent from the architecture has a *dedicated governor agent* connected to it, enforcing the laws of executed events. DynaCROM also uses rules to manage normative agent positions, but executed actions, instead of executed events, are the focus of the regulation. Moreover, DynaCROM provides a more precise mechanism for regulation, while permitting the use of contextual norms. Furthermore, agents are enhanced with a normative behavior that informs them about current systems' contextual norms based on active actions instead of having many agents monitoring all events executed in the regulated system.

5 Conclusion

In this work, we focused on a case study for exemplifying different issues resulting from the application of contextual norms. It was presented two scenarios where norm-aware agents make decisions based on given norm information and three scenarios where compositions of contextual norms result in restrictions, relaxations and conflicts.

In the presented case study, we used our DynaCROM solution for continuously informing the current contextual norms of agents from an open MAS. Norm-aware agents use DynaCROM answers (updated contextual norms) to make better decisions and, thus, achieve their goals faster. Developers of regulations in open MASs use DynaCROM as a flexible solution for updating systems' norms at run-time.

For future work, we are planning to use the Jess rule engine [11] instead of the Jena engine [19], mainly addressing issues such as ease-of-use, expressiveness and reasoning. We are also planning to use Jadex instead of Jade for enhancing BDI agents with DynaCROM answers. We aim to discover how we can interfere in agents' beliefs, which are, normally, pre-defined during the agents' design phase.

Acknowledgments

This work was partially funded by the projects ESSMA (CNPq 552068/2002-0) and EMACA (CAPES/COFECUB 482/05 PP 016/04), and by CNPq individual grants.

References

1. Artikis, A.: “Executable Specification of Open Norm-Governed Computational Systems.” PhD thesis, Imperial Colllege London, 2003.
2. Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., Stuckenschmidt, H.: “C-OWL: Contextualizing Ontologies”. In Proc. of ISWC-2003, LNCS vol. 2870, p. 164-179, Springer Verlag. 2003
3. Cliffe, O.; De Vos, M.; and Padget, J.: “Specifying and Reasoning about Multiple Institutions”. In the AAMAS06 Workshop on Coordination, Organization, Institutions and Norms in agent systems (COIN-2006). Hakodate, Japan.
4. Dey, A.: “Understanding and using context”. Personal Ubi /comput 2001; 5(1)
5. “Eclipse”. Electronic available in: <http://www.eclipse.org/>. Accessed in October2006.
6. Felicíssimo, C. H., Lucena, C. J. P. de, Briot, J.-P., Choren, R.: “Regulating Open Multi-Agent Systems with DynaCROM”. In the *Second Workshop on Software Engineering for Agent-oriented Systems* (SEAS, 2006). Brazil, 2006.
7. Felicíssimo, C. H., Lucena, C. J. P. de, Briot, J.-P., Choren, R.: “An Approach for Contextual Regulations in Open MAS”. In the *Eight International Bi-Conference Workshop on Agent Oriented Information Systems* (AOIS-2006) at AAMAS, Hakodate, Japan, 2006.
8. Felicíssimo, C. H., Lucena, C. J. P. de: “An Approach to Regulate Open Multi-Agent Systems Based on a Generic Normative Ontology”. First Workshop on Software Engineering for Agent-oriented Systems (SEAS, 2005). Brazil, 2005. Electronic available in: <http://www.les.inf.puc-rio.br/seas2005/file/cFelicissimo.pdf>. Accessed in October 2006.
9. Felicíssimo, C.H., Lucena, Carlos, Carvalho, G., Paes, R..: “Normative Ontologies to Define Regulations Over Roles in Open Multi-Agent Systems”. AAAI Fall Symposium “Roles, an Interdisciplinary Perspective: Ontologies, Programming Languages, and Multiagent Systems”, Technical Report FS-05-08. USA. November, 2005. ISBN 978-1-57735-254-9.
10. Ferber, J., Gutknecht, O., Michael, F.: “From Agents to Organizations: an Organization View of Multi-Agent Systems”. In Proc. of the AOSE, 2003. Australia, 2003.
11. Friedman-Hill, E.: “Jess, the Rule Engine for the Java Platform”. At Sandia National Laboratories. Electronic available in: <http://herzberg.ca.sandia.gov/>. Accessed in October 2006.
12. Gaertner, D., Clark, K., Sergot, M.: "Ballroom etiquette: a case study for norm-governed multi-agent systems". In the AAMAS06 Workshop on Coordination, Organization, Institutions and Norms in agent systems (COIN-2006). Hakodate, Japan.
13. Garcia-Camino, A.; Rodriguez-Aguilar, J.-A.; J. A.; Sierra, C., Vasconcelos, W.: “A Distributed Architecture for Norm-Aware Agent Societies”. Procs. 3rd International Workshop on Declarative Agent Languages and Technologies (DALT'05), 25 July 2005, Utrecht, The Netherlands.
14. García-Camino, A.; Rodriguez-Aguilar, J.-A.; Sierra, C.; Vasconcelos, W.: “Norm-Oriented Programming of Electronic Institutions: A Rule-based Approach”. In the AAMAS06 Workshop on Coordination, Organization, Institutions and Norms in agent systems (COIN-2006). Hakodate, Japan.

15. Gosling, J., Joy, B., Junior, G.L.S., Bracha, G.: "The Java Language Specification". ISBN 0-201-31008-2. Electronic available in: <http://java.sun.com/>. Accessed in October 2006.
16. Gruber, T. R.: "A translation approach to portable ontology specifications". Knowledge Acquisition, Vol. 5, Issue 2. June, 1993. Pages 199-220. ISSN: 1042-8143.
17. Hewitt, C.: "Open Information Systems Semantics for Distributed Artificial Intelligence". Artificial Intelligence, V. 47, Issue 1-3. 1991. p. 79 - 106. ISSN: 0004-3702.
18. "JADE". Electronic available in: <http://jade.tilab.com/>. Accessed in October 2006.
19. "Jena". Electronic available in: <http://jena.sourceforge.net/>. Accessed in October 2006.
20. Jennings, N. R.: "On Agent-Based Software Engineering". Artificial Intelligence, 117 (2) pages 277-296. 2000.
21. Jennings, N., Sycara, K. and Wooldridge, M.: "A Roadmap of Agent Research and Development". Journal of Agents and Multi-Agent Systems, 1:7-38, 1998.
22. Khedr, M, Karmouch, A.: "ACAI: Agent-Based Context-aware Infrastructure for Spontaneous Applications". Journal of Network & Computer Applications, Vol. 28, Issue 1, p. 19-44. 1995.
23. McGuinness, D. L. and Harmelen, F. v.: "OWL Web Ontology Language Overview". Electronic available in: <http://www.w3.org/TR/owl-features/>. Accessed in October 2006.
24. Noy, N, Rector, A. (Eds.): "Defining N-ary Relations on the Semantic Web: Use with Individuals". Electronic available in: <http://www.w3.org/TR/swbp-n-aryRelations/>. Accessed in October 2006.
25. "Protégé". Electronic available in: <http://protege.stanford.edu/>. Accessed in October 2006.
26. Thomas, G. e Williams, A. B.: "Roles in the Context of Multiagent Task Relationships". AAAI Fall Symposium "Roles, an Interdisciplinary Perspective: Ontologies, Programming Languages, and Multiagent Systems", Technical Report FS-05-08. ISBN 978-1-57735-254-9. USA. November, 2005.
27. Vázquez-Salceda, J., Dignum, V., Dignum, F.: "Organizing Multiagent Systems". Journal of Autonomous Agents and Multi-Agent Systems, 11, 307-360, 2005.
28. Weyns, D., Parunak, H.V.D., Michel, F., Holvoet, T. and Ferber, J.: "Environments for Multiagent Systems State-of-the-Art and Research Challenges". In Environments for Multi-Agent Systems (E4MAS, 2004), Springer 2005, p.1-47. ISBN 3-540-24575-8.