PUC

ISSN 0103-9741

Monografias em Ciência da Computação
n° 32/06

Inferring Contextual Laws with Rules for Regulations in Open Multi-Agent Systems

Carolina Howard Felicíssimo   Carlos José Pereira de Lucena

Jean-Pierre Briot   Karin Koogan Breitman

Marco Antonio Casanova

Ricardo Choren

Departamento de Informática

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO

RUA MARQUÊS DE SÃO VICENTE, 225 - CEP 22453-900

RIO DE JANEIRO - BRASIL

Monografias em Ciência da Computação, No. 32/06 ISSN: 0103-9741
Editor: Prof. Carlos José Pereira de Lucena August, 2006

Inferring Contextual Laws with Rules for Regulations in Open Multi-Agent Systems *

Carolina Howard Felicíssimo[1], Carlos José Pereira de Lucena[1], Jean-Pierre Briot[1,2], Karin Koogan Breitman[1], Marco Antonio Casanova, Ricardo Choren[3]

[1]Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio)

[2]Laboratoire d'informatique de Paris 6 (LIP6)

[3]Instituto Militar de Engenharia (SE/8 – IME)

{cfelicissimo, lucena, karin, casanova}@inf.puc-rio.br, jean-pierre.briot@lip6.fr and choren@de9.ime.eb.br

Abstract. Rules and Rule-based inference engines have become a powerful technology to support basic Semantic Web tasks. In this work, we focus our attention in the essential task of regulation in open domains. More precisely, the rule layer of DynaCROM – our approach for dynamic contextual regulation in open multi-agent systems (MAS) – will be detailed. The DynaCROM rule layer permits developers of system regulation to easily customize several compositions of contextual laws, by activating and deactivating (few) rules. Thus, by continuous snapshots of a regulated system, developers can analyze agents' performance and, then, manually influence both law enforcement and agents' behavior. The result is a more balanced regulation. Moreover, meta-rules can also be specified in order to automatically raise pre-defined sets of rules, when their conditions are satisfied. Therefore, the rule layer introduces flexibility and precision in the DynaCROM solution, as a straightforward method for regulation in open MAS.

Keywords: Contextual Regulation, Open Multi-Agent Systems, Rules, Inference, Ontologies.

* This work has been sponsored by the Ministério de Ciência e Tecnologia da Presidência da República Federativa do Brasil

In charge for publications:

Rosane Teles Lins Castilho
Assessoria de Biblioteca, Documentação e Informação
PUC-Rio Departamento de Informática
Rua Marquês de São Vicente, 225 - Gávea
22453-900 Rio de Janeiro RJ Brasil
Tel. +55 21 3114-1516 Fax: +55 21 3114-1530
E-mail: bib-di@inf.puc-rio.br
Web site: http://bib-di.inf.puc-rio.br/techreports/

Table of Contents

1 Introduction

Agent-based computing is rapidly emerging as a powerful technology for the development of distributed and complex software systems. In a multi-agent system (MAS) several software agents try to achieve their own goals. An open MAS is a MAS with no centralized control in which agents can unrestrainedly join or leave the system. Open systems are always subject to unanticipated interactions [21] caused by members that do not conform to recommendations of correct behavior. This risk imposes the need for regulatory mechanisms to prevent malicious actions and to inspire trust for the agents of open MAS.

Regulation in open MAS is a challenging activity. Besides being inherently complex, this kind of system presents practical difficulties since it encompasses heterogeneous agents, which have their own design and implementation. The key characteristics of open MAS are: agent heterogeneity, conflicting individual goals and limited trust [1].

Regulation in open MAS should be, for developers, easy and simple to be implemented and, for agents, they must be expressed in a well-defined model for enabling reasoning. Furthermore, regulatory dynamics should be supported. As a solution which matches these requirements, we proposed in [8] and [9] an approach for dynamic contextual regulation in open MAS, now called DynaCROM.

The DynaCROM solution evolved with significant improvements in how to automatically customize several compositions of contextual laws, using (few) rules. Moreover, formulas for the total numbers of possible compositions of laws, from distinct regulatory contexts, were discovered and presented here. These formulas reflect the relation between rules and compositions of contextual laws. All evolutions in DynaCROM, responsible for automating regulation in open MAS, are the focus of this work.

The remainder of this paper is organized as follows: Section 2 briefly presents our DynaCROM approach for dynamic contextual regulation in open MAS, by explaining what contextual regulation, a top-down modeling of laws and the DynaCROM normative meta-ontology and process are; Section 3 details the DynaCROM rule layer and introduces our proposed formulas that relates rules and the total numbers of possible compositions of contextual laws; Section 4 describes a simple case study from the domain of multinational corporations; finally, Section 5 offers our conclusions and points directions for future work.

2 Regulations in Open MAS

MAS are typically composed of environments, organizations, agents, agent roles and agent interactions [18]. In our definition, environments [31] are discrete computational locations (similar to places in the physical world) that provide conditions for agents to inhabit it; organizations [11] are social locations where group of agents play roles, seeking to achieve their goals; roles are abstractions that define a set of related tasks for agents [29]; and, finally, interactions can be done by agents from the same or from different organizations and environments.

2.1 Contextual Regulations in Open MAS

Environments, organizations, roles and interactions suggest different contexts for regulation in open MAS. Contexts are implicit situational information [7] that might be used to characterize situations of participants. Modular context refinements allow a more flexible system and provide a better support for developers, facilitating the task of information maintenance.

Context-aware systems use contexts to provide relevant information or services to their users, where relevancy depends on the users' tasks [7]. In our definition, regulated context-aware systems use pre-defined information of laws to control environments, organizations, agent roles and agent interactions. Thus, laws should define which actions are permitted (allowed to be performed), obliged (must be performed), and prohibited (must not be performed), in a regulated system. Deliberative normative agents [5] use law information to better adapt their behaviors.

Law enforcement can be carried out a priori, avoiding law violation, or a posteriori, penalizing infringing agents. A priori regulation guarantees law compliance, by enforcing the laws of a regulated action; however, a priori regulation results in an extra overload during a system's execution. A posteriori regulation does not guarantee law compliance, but inhibit infringing agents with explicit punishments.

In open MAS, a large number of heterogeneous agents enter and leave the system, without restraint, performing a large number of actions. Consequently, the effectiveness of a priori law enforcement is reduced because of the need of excessive overloads for the regulation. In order to provide a more effective regulation mechanism, we proposed in [8] an approach for a posteriori law enforcement. This approach is based on a top-down context modeling of laws and on a regulatory meta-ontology, both briefly explained in the following two sub-sections.

2.2 A Top-Down Contextual Modeling of Laws

Contexts are tacitly understood for most people, but they are generally hard to evidence. For developers of systems' regulations, organizing laws according to a top-down contextual modeling eases the elicitation task because information is better structured. However, sometimes, it is difficult to classify laws according to the defined contexts due to their subjectivity.

Researches in context-aware applications suggest top-down architectures for contextual modeling [22]. According to this direction, we address in DynaCROM the following four regulatory contexts: Environment, Organization, Role and Interaction Laws. Environment Laws are applied to all agents from the regulated environment. Organization Laws are applied to all agents from the regulated organization. Role Laws are applied to all agents playing the regulated role. Interaction Laws are applied to all agents involved in the regulated interaction.

Figure 1 illustrates the boundaries of environment, organization, role and interaction laws. For instance, one agent from the environment on the right side interacts with an agent from the environment on the left side, both regulated through an interaction law. Moreover, these agents can also be regulated through customized compositions of environment, organization and/or role laws, for a more complete regulation.

DynaCROM regulatory contexts have different levels of abstraction, but they are not hierarchical. Laws from different regulatory contexts can be freely composed for restricting or relaxing a current system's regulation, by different law enforcement.

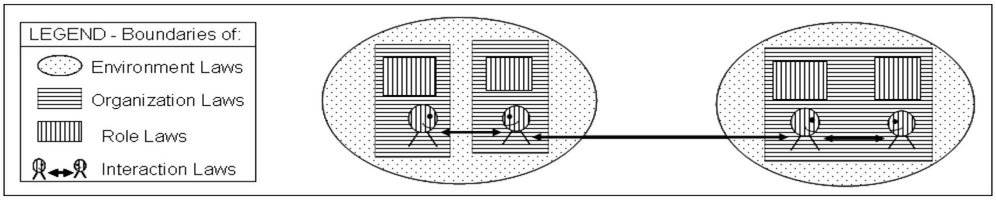

Figure 1. Interactions possibly regulated through different compositions of contextual laws

We believe that DynaCROM regulatory contexts are not targeted to a particular application domain, but they rather represent a minimum set for generic contextual regulation in open MAS. For more complex MAS, this set should be improved with additions and refinements of particular domain regulatory contexts. For instance, while regulating negotiations in an open MAS from the domain of multinational corporations, which sells distinct types of products for final users and enterprises, specific domain contexts should be created to represent particular possibilities (e.g., modes of payment, vantages for financing, etc.). Moreover, the characteristics of negotiations performed among agents from distinct organizations or, even, environments, should be preserved, by specific contexts and laws. For instance, it is expected that the negotiations made between agents from two distinct environments have to be paid with the local currency where the sell is made (i.e., the buyer agent must have to convert his currency to the currency of the seller location).

2.3 The DynaCROM Meta-Ontology

Regulation in open MAS cannot be implemented as agent code because we do not have any control over agents' development. Instead, a standard, centralized, external and well-defined model should be used. Ontologies [14] can explicitly represent information, in a meaningful way, for agents to automatically process their contents [24].

DynaCROM defines a normative meta-ontology (see Figure 2), where its four regulatory contexts are represented by six related concepts (Environment, Organization, Role, Norm, Penalty and Action); and, Environment, Organization, Role and Interaction Laws should be represented by instances of these concepts. Thus, we believe that regulation in open MAS can be done by simply instantiating six related concepts [10].

In the DynaCROM normative meta-ontology, all concepts have a minimal set of predefined attributes (defined by object properties that link concepts, or by data properties). The Action concept encompasses all instances of the regulated actions. The Penalty concept encompasses all instances of the punishments to be given in case norms are not fulfilled. The Norm concept encompasses all instances of norms from all regulatory contexts; however, each norm encompasses the permission, obligation or prohibition instance for its associated action and penalty. The Environment concept encompasses all instances of the regulated environments; however, each environment encompasses its associated norms and owner environment (the environment which it belongs to). The Organization concept encompasses all instances of the regulated organizations; however, each organization encompasses its associated norms, main organization (the organization which it is associated to) and environment. The Role concept encompasses all instances of the regulated roles; however, each role encompasses its associated norms and organization.

The DynaCROM normative meta-ontology must be extended with both new domain concepts and all modeled interaction laws. Interaction laws should be implemented by following a representation pattern, from the Semantic Web Best Practices document [25], which 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). In DynaCROM ontology, an interaction law should be represented by a new Norm sub-concept, which links two Role concepts. Furthermore, to accomplish regulation, the DynaCROM ontology (extended or not) has to be instantiated with all system laws, according to the contextual modeling done.

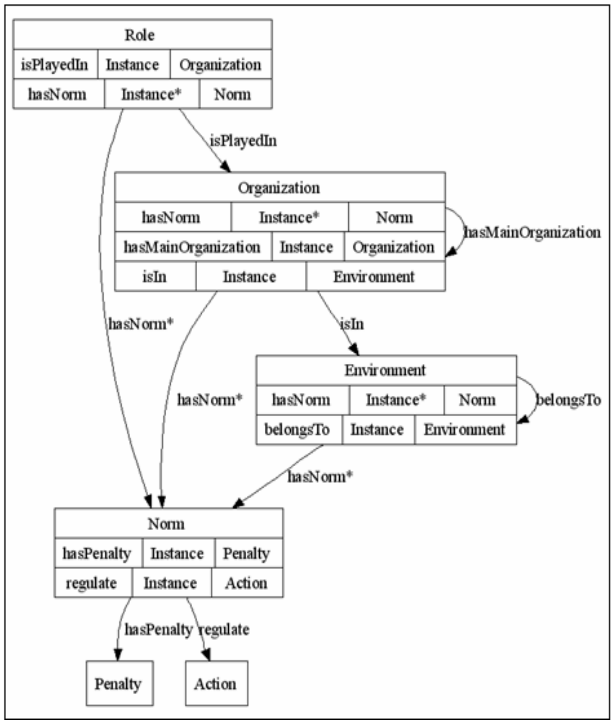

Figure 2. The DynaCROM Meta-Ontology

2.4 The DynaCROM Process

Context-aware systems can support three main categories of features: (i) presentation of information and services to users; (ii) automatic execution of services for users; and (iii) tagging of context to support later information retrieval [7]. Regarding feature (i), DynaCROM offers its normative meta-ontology for users to manually represent their modeled contextual laws. Agents will then be able to automatically process the information. Regarding features (ii) and (iii), DynaCROM offers pre-defined rules and a mechanism to dynamically change those, permitting several inferred compositions of customized contextual laws.

Working with contexts, ontologies, rules and rule-based inference engines is what we envision to be the main assets of DynaCROM. **Error! Reference source not found.** presents an overview of the DynaCROM process. The ontology explicitly represents the DynaCROM four regulatory contexts (by related concepts expressed in the ontology structure), and it also represents environment, organization, role and interaction laws (by instances expressed in the ontology data). Activation and deactivation of rules are used to manually specify customized compositions of contextual laws. A rule-based inference engine automatically deduces the composed contextual laws (according to the ontology, active rules and a given action) and expresses those laws into an inferred ontology.

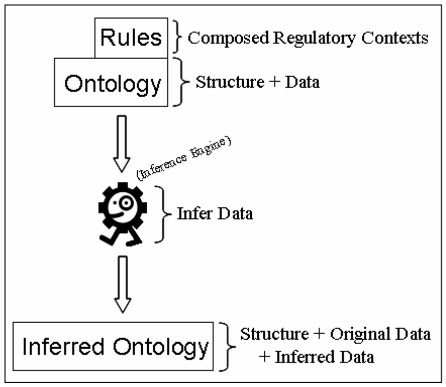

Figure 3. The DynaCROM process

3 The DynaCROM Context-Aware Rules

Classifying laws according to a top-down modeling and structuring them in a normative ontology instance, support flexibility in the system regulation with several possibilities of composed laws. Compositions are done based on related contexts, relaxing or restricting the law enforcement. However, instead of implementing the desired compositions of contextual laws inside agent codes or system classes, and have to change implemented codes every time other compositions of contextual laws are required, DynaCROM offers a better solution. Based on rules and using an inference engine support, DynaCROM offers several customized compositions of contextual laws, achieved by simple activations and deactivations of (few) pre-defined rules. DynaCROM rules are ontology-based rules, i.e. they are created according to the ontology structure (by linking only related concepts), and consequently, the numbers of rules and possible compositions of contextual law, for each regulatory context, are finite.

3.1 Rule for the Environment Context

In DynaCROM, each environment can have its laws composed with only the laws of its owner environment, totalizing **one** possible composition of contextual laws. For instance, Brazilian laws can be composed with only South American laws (in our case study from Section 4, South America is the owner environment of Brazil).

(1) Total of compositions of contextual laws for
each environment instance = $\Sigma^{1}_{=\iota} C^{1}_{\iota} = \mathbf{1}$

Table 1 presents the rule that is responsible for composing laws of a given environment with the laws of its owner environment. For instance, given an environment (e.g., Brazil), a rule-based inference engine follows the "*belongsTo*" ontology property to deduce its owner environment (e.g., South America); with the owner environment value, the rule-based inference engine follows the "*hasNorm*" ontology property to deduce the owner environment laws and, then, it composes the inferred laws with the original environment laws.

Table 1. The rule for the Environment context

```
Rule 1- [ruleForEnvWithOEnv:
        (?Env belongsTo ?OEnv)
        (?OEnv hasNorm ?OEnvNorm)
          -> (?Env hasNorm ?OEnvNorm)]
```

3.2 Rule for the Organization Context

In DynaCROM, each organization can have its laws composed differently with the laws from its five related concepts (Main Organization, Organization's Environment, Main Organization's Environment, Owner Environment of the Organization's Environment and Owner Environment of the Main Organization's Environment), totalizing thirty one possible compositions of contextual laws. For instance, Figure 4 presents all these possible compositions of contextual laws for a given organization.

(2) Total of compositions of contextual laws for each organization instance = $\Sigma^{5}_{1=\iota} C^{5}_{\iota} = \mathbf{31}$

C1	Org + MOrg.	**C16**	Org. + MOrg. + OrgEnv. + OEnvOfOrgEnv.
C2	Org. + OrgEnv.	**C17**	Org. + MOrg. + OrgEnv. + MOrgEnv.
C3	Org. + OEnvOfOrgEnv.	**C18**	Org. + MOrg. + OrgEnv. + OEnvOfMOrgEnv.
C4	Org. + MOrgEnv.	**C19**	Org. + MOrg. + OEnvOfOrgEnv. + MOrgEnv.
C5	Org. + OEnvOfMOrgEnv.	**C20**	Org. + MOrg. + OEnvOfOrgEnv. + OEnvOfMOrgEnv.
C6	Org. + MOrg. + OrgEnv.	**C21**	Org. + MOrg. + MOrgEnv. + OEnvOfMOrgEnv.
C7	Org. + MOrg. + OEnvOfOrgEnv.	**C22**	Org. + OrgEnv. + OEnvOfOrgEnv. + MOrgEnv.
C8	Org. + MOrg. + MOrgEnv.	**C23**	Org. + OrgEnv. + OEnvOfOrgEnv. + OEnvOfMOrgEnv.
C9	Org. + MOrg. + OEnvOfMOrgEnv.	**C24**	Org. + OrgEnv. + MOrgEnv. + OEnvOfMOrgEnv.
C10	Org. + OrgEnv. + OEnvOfOrgEnv.	**C25**	Org. + OEnvOfOrgEnv. + MOrgEnv. + OEnvOfMOrgEnv.
C11	Org. + OrgEnv. + MOrgEnv.	**C26**	Org. + MOrg. + OrgEnv. + OEnvOfOrgEnv. + MOrgEnv.
C12	Org. + OrgEnv. + OEnvOfMOrgEnv.	**C27**	Org. + MOrg. + OrgEnv. + OEnvOfOrgEnv. + OEnvOfMOrgEnv.
C13	Org. + OEnvOfOrgEnv. + MOrgEnv.	**C28**	Org. + MOrg. + OrgEnv. + MOrgEnv. + OEnvOfMOrgEnv.
C14	Org. + OEnvOfOrgEnv. + OEnvOfMOrgEnv.	**C29**	Org. + MOrg. + OEnvOfOrgEnv. + MOrgEnv. + OEnvOfMOrgEnv.
C15	Org. + MOrgEnv. + OEnvOfMOrgEnv.	**C30**	Org. + OrgEnv. + OEnvOfOrgEnv. + MOrgEnv. + OEnvOfMOrgEnv.
		C31	Org. + MOrg. + OrgEnv. + OEnvOfOrgEnv. + MOrgEnv. + OEnvOfMOrgEnv

Legend of Abreviattions: **Org.**: Organization; **OrgEnv.**: Organization's Environment; **OEnvOfOrgEnv.**: Owner Environment of an Organization's Environment; **MOrg.**: Main Organization; **MOrgEnv.**: Main Organization's Environment; **OEnvOfMOrgEnv.**: Owner Environment of a Main Organization's Environment

Figure 4. All possibilities of compositions of contextual laws for an organization instance

Table 2 presents all five rules responsible for composing laws of a given organization with the laws of its related concepts. For instance, when **Rule 2** is active, the **C1** composition from Figure 4 is achieved. More precisely, given an organization (e.g., Hpie Brazil), a rule-based inference engine follows the "*hasMainOrganization*" ontology property to deduce its main organization (e.g., Hpie); with the main organization value, the rule-based inference engine follows the "*hasNorm*" ontology property to deduce the main organization laws and, then, it composes the inferred laws with the original organization laws.

Table 2. The rules for the Organization context

```
Rule 2- [ruleForOrgWithMOrgNorm:
        (?Org hasMainOrganization ?MOrg)
        (?MOrg hasNorm ?MOrgNorm)
          -> (?Org hasNorm ?MOrgNorm)]

Rule 3- [ruleForOrgWithOrgEnvNorm:
        (?Org isIn ?OrgEnv)
        (?OrgEnv hasNorm ?OrgEnvNorm)
          -> (?Org hasNorm ?OrgEnvNorm)]

Rule 4- [ruleForOrgWithOEnvOfOrgEnvNorm:
        (?Org isIn ?OrgEnv)
        (?OrgEnv belongsTo ?OEnvOfOrgEnv)
        (?OEnvOfOrgEnv hasNorm ?OEnvOfOrgEnvNorm)
```

```
        -> (?Org hasNorm ?OEnvOfOrgEnvNorm)]

Rule 5- [ruleForOrgWithMOrgEnvNorm:
      (?Org hasMainOrganization ?MOrg)
      (?MOrg isIn ?MOrgEnv)
      (?MOrgEnv hasNorm ?MOrgEnvNorm)
        -> (?Org hasNorm ?MOrgEnvNorm)]

Rule 6- [ruleForOrgWithOEnvOfMOrgEnvNorm:
      (?Org hasMainOrganization ?MOrg)
      (?MOrg isIn ?MOrgEnv)
      (?MOrgEnv belongsTo ?OEnvOfMOrgEnv)
      (?OEnvOfMOrgEnv hasNorm ?OEnvOfMOrgEnvNorm)
        -> (?Org hasNorm ?OEnvOfMOrgEnvNorm)]
```

Table 3 presents how rules 1, 2, 3, 4 and 5 from Table 2 have to be combined for achieving all thirty one compositions of contextual laws (**C1** to **C31** in Figure 4) for a given organization.

Table 3. Activating and deactivating rules for regulation

C1	Rule 1
C2	Rule 2
C3	Rule 3
C4	Rule 4
C5	Rule 5
C6	Rule 1 + Rule 2
C7	Rule 1 + Rule 3
C8	Rule 1 + Rule 4
C9	Rule 1 + Rule 5
C10	Rule 2 + Rule 3
C11	Rule 2 + Rule 4
C12	Rule 2 + Rule 5
C13	Rule 3 + Rule 4
C14	Rule 3 + Rule 5
C15	Rule 4 + Rule 5
C16	Rule 1 + Rule 2 + Rule 3
C17	Rule 1 + Rule 2 + Rule 4
C18	Rule 1 + Rule 2 + Rule 5
C19	Rule 1 + Rule 3 + Rule 4
C20	Rule 1 + Rule 3 + Rule 5
C21	Rule 1 + Rule 4 + Rule 5
C22	Rule 2 + Rule 3 + Rule 4
C23	Rule 2 + Rule 3 + Rule 5
C24	Rule 2 + Rule 4 + Rule 5
C25	Rule 3 + Rule 4 + Rule 5
C26	Rule 1 + Rule 2 + Rule 3 + Rule 4
C27	Rule 1 + Rule 2 + Rule 3 + Rule 5
C28	Rule 1 + Rule 2 + Rule 4 + Rule 5
C29	Rule 1 + Rule 3 + Rule 4 + Rule 5
C30	Rule 2 + Rule 3 + Rule 4 + Rule 5
C31	Rule 1 + Rule 2 + Rule 3 + Rule 4 + Rule 5

3.3 Rule for the Role Context

In DynaCROM, each role can have its laws composed differently with the laws from its six related concepts (Organization, Main Organization, Organization's Environment, Main Organization's Environment, Owner Environment of the Organization's Environment and Owner Environment of the Main Organization's Environment), totalizing sixty three possible compositions of contextual laws.

(3) Total of compositions of contextual laws for each role instance = $\Sigma^{6}_{1=\iota}\ C^{6}_{\iota}$ = **63**

All six rules responsible for composing the laws of a given role with the laws of its related concepts follow the same pattern as presented in Tables 2 and 3 for the organization concept. For instance, Table 4 presents the rule that composes laws from both a given role (e.g., a supplier) and its organization (e.g., Hpie).

Table 4. An rule for the Role context

```
Rule 7- [ruleForRoleWithOrg:
        (?Role isPlayedIn ?Org)
        (?Org hasNorm ?OrgNorm)
          -> (?Role hasNorm ?OrgNorm)]
```

3.4 Rule for the Interaction Context

In DynaCROM, each interaction can have its laws composed differently with the laws from its seven related concepts (Role, Organization, Main Organization, Organization's Environment, Main Organization's Environment, Owner Environment of the Organization's Environment and Owner Environment of the Main Organization's Environment, for each player), totalizing two hundred and fifty four possible compositions of contextual laws.

(4) Total of compositions of contextual laws for each interaction instance = $2^* \Sigma^{7}_{1=\iota}\ C^{7}_{\iota}$ = **254**

All seven rules responsible for composing the laws of a given interaction with the laws of its related concepts follow the same pattern as presented in Tables 2 and 3 for the organization concept. For instance, Table 5 presents the rule that composes laws from both a given interaction and their roles. As an interaction is performed between agents playing different roles, Rule 13 results in compositions of laws for both roles.

Table 5. An rule for the Interaction context

```
Rule 13- [ruleForInteracWithRole:
        (?Interac isPlayedBy ?Role)
        (?Role hasNorm ?RoleNorm)
          -> (?Interac hasNorm ?RoleNorm)]
```

3.5 The Role of the DynaCROM Rules

In DynaCROM, rules can be manually changed (i.e., activated or deactivated, inserted, deleted, or updated) during a system's execution, for different compositions of contextual laws. Consequently, the current regulation is automatically and dynamically updated, by different inferred composed laws.

For the four DynaCROM regulatory meta-contexts (environment, organization, role and interaction laws), a total of three hundred and forty nine compositions of contextual laws (1 from (1) + 31 from (2) + 63 from (3) + 254 from (4)) can be achieved with only nineteen rules (1 for (1) + 5 for (2) + 6 for (3) + 7 for (4)). Furthermore, compositions of contextual laws from specific domain regulatory contexts can increase this total, if the domain contexts were created as extended concepts in the DynaCROM meta-ontology. Thus, the numbers of both extra compositions of contextual laws and rules, for each new domain context, are directly proportional to the number of concepts its new concept is related to in the DynaCROM ontology (i.e. $\Sigma^{n}_{i=1} C^{n}_{i} = 2^{n} - 1$).

The role of the DynaCROM rule layer is to better support a more dynamic and balanced regulation. By continuous system snapshots, developers can analyze performances of agents and, then, manually, they can influence both the enforcement of laws (by changing rules) and agents' behavior. For instance, perceiving that the volume of negotiations in two branches of a main organization is unbalanced, the developer of the system regulation can relax the current laws from the organization branch where the volume of negotiations is low, and he can restrict the current laws from the organization branch where the volume of negotiations is high. Doing that, agents will change organizations, looking for better deals. Moreover, meta-rules can also be specified in order to automatically raise pre-defined sets of rules, when their conditions are satisfied. For instance, a meta-rule can be specified for automatically raise a set of rules when a pre-defined volume of negations is achieved.

4 Case Study

The domain of multinational corporations was chosen to explain how rules can be used in DynaCROM for inferring customized compositions of contextual laws. A multinational corporation (organization) is an enterprise that manages production branches located in at least two countries, possibly located in different continents. Law enforcement is applied to all regulated actions performed by agents, which move randomly from one location (environment or organization) to another, bound to the compositions of contextual laws of the new location.

4.1 Examples of Contextual Laws

For this case study, Hpie is the main organization, having Hpie Cuba and Hpie Brazil as its branches. Hpie corporations have the following roles: manufacturer, supplier, distributor and customer. Hpie is in USA, which is in turn in North America; Hpie Cuba is in Cuba, which is in turn in Central America; and Hpie Brazil is in Brazil, which is in turn in South America. Corporation laws are usually private because they are strategic to the corporation businesses. Thus, we created environment, organization, role and interaction laws, based on public laws collected from several corporate Web sites. These laws were classified according to the DynaCROM top-down modeling.

1. **Examples of Environment Laws:**

   1.1. In North America, a finished good from every organization has its price increased by a percentage (dependent of the seller location) as taxes, for immediate delivery or if the deliver address is in North America.

   1.2. In USA, every organization product has its price increased by 8% as taxes, for immediate delivery or if the deliver address is in USA.

1.3. In South America, every shipped order has its price increased by 15% as taxes, for immediate delivery or if the deliver address is outside South America.

1.4. In USA, all negotiations have to be paid in American dollars (USD), the national currency. Negotiations outside USA have to have their values converted from USD to the national currency of the country where the buyer is.

1.5. In Cuba, all negotiations have to be paid in Cuban pesos (CUP), the national currency. Negotiations outside Cuba have to have their values converted from CUP to the national currency of the country where the buyer is.

1.6. In Brazil, all negotiations have to be paid in Reais (R$), the national currency. Negotiations outside Brazil have to have their values converted from R$ to the national currency of the country where the buyer is.

2. **Examples of Organization Laws:**

2.1. In Hpie, all paid orders must have detailed receipts.

2.2. In Hpie Cuba, every product has one year of warranty.

2.3. In Hpie Brazil, every placed order must have a down payment of 10%.

3. **Example of Role Laws:**

3.1. Hpie Cuba manufacturers have to provide refunds or replacements for every defective product when substantial defects cannot be fixed in four attempts.

3.2. Hpie Cuba manufacturers have to provide, with one month, refunds or replacements for every defective product, when substantial defects cannot be fixed in four attempts.

3.3. Hpie Brazil suppliers have to ship orders in their due dates.

3.4. Hpie Brazil suppliers can give 5% as discount for orders paid in cash.

4. **Example of an Interaction Law:**

4.1. Hpie Brazil suppliers have the permission to ship incomplete orders to manufacturers.

The created laws for this case study were instantiated inside the DynaCROM meta-ontology. Figure 5 illustrates part of the DynaCROM ontology instance, presenting all environments and organizations, and some roles and contextual laws.

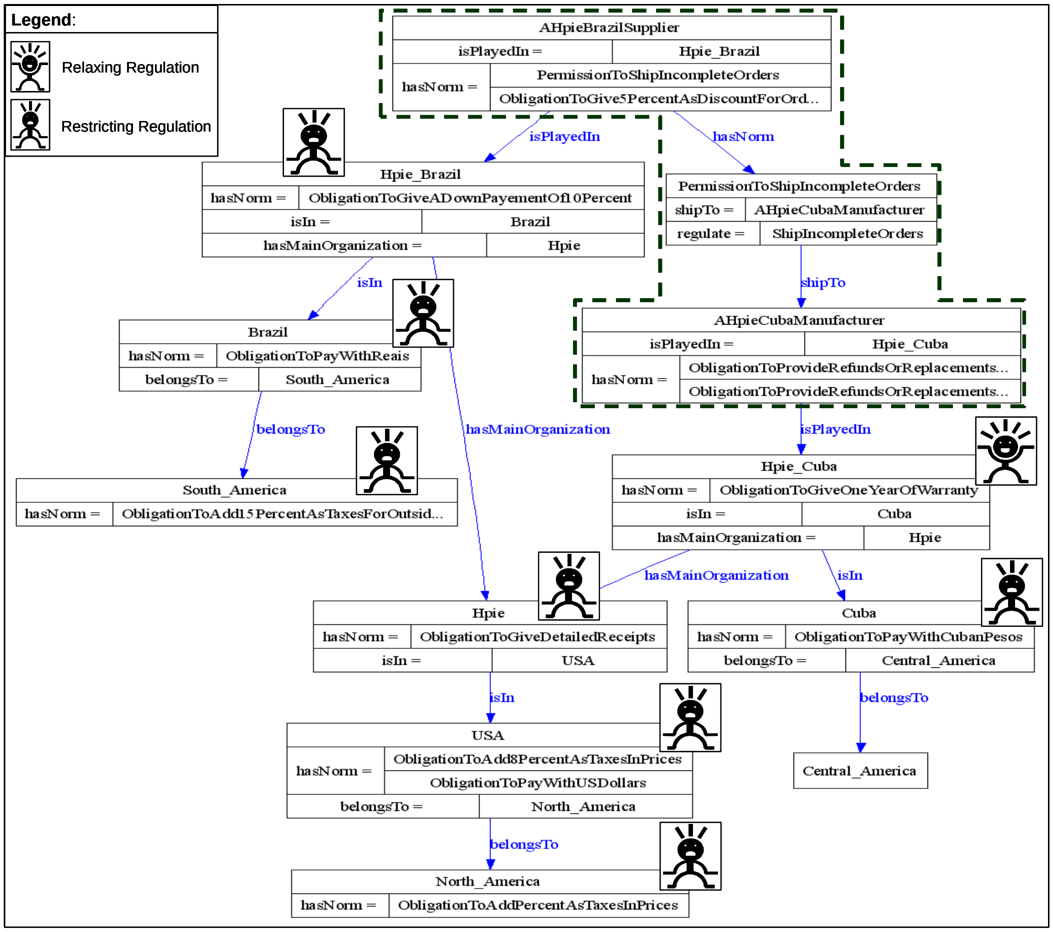

Figure 5. An example of a DynaCROM ontology instance with its four regulatory contexts

4.2 Influencing Regulation by Compositions of Contextual Laws

In our case study, rules were manually activated and deactivated during the system's execution, for new customized compositions of contextual laws. For instance, Figure 5 illustrates, with distinct icons, the interaction law 4.1 semantically restricted by seven examples, and semantically relaxed by one example, both because new compositions of laws from the *Environment*, *Organization* and *Role* contexts. The interaction between an Hpie Cuba manufacturer and an Hpie Brazil supplier happens because the Hpie Cuba manufacturer has to honor its laws 3.1 and 3.2 (i.e., he has a deadline to replace a defective product), but an important part for the product's assembly is not found in Hpie Cuba. The interaction law 4.1 states that Hpie Brazil suppliers are permitted to ship incomplete orders to the Hpie Cuba manufacturer.

While composing interaction and role laws (by activating Rule 13 from Table 5), the interaction between the Hpie Cuba manufacturer and the Hpie Brazil supplier is influenced by the Hpie Brazil supplier laws 3.3 and 3.4. With the law 3.3, the Hpie Brazil supplier is now obliged to ship the Hpie Cuba manufacturer's orders in their due dates. With the law 3.4, the Hpie Brazil supplier is now permitted to give 5% of discount, if the Hpie Cuba manufacturer pays with cash.

While composing interaction and organization laws, the interaction between the Hpie Cuba manufacturer and the Hpie Brazil supplier is influenced by the Hpie Cuba law 2.2 and by the Hpie Brazil law 2.3. With the Hpie Cuba law 2.2, the Hpie Cuba manufacturer now has to honor its laws 3.1 and 3.2 for only one year (the specified warranty period). With the Hpie Brazil law 2.3, every placed order now must have a down payment of 10%.

C1	Interaction + AHpieBrazilSupplier	**C11**	Interaction + AHpieCubaManufacturer
C2	Interaction + Hpie Brazil	**C12**	Interaction + Hpie Cuba
C3	Interaction + Brazil	**C13**	Interaction + Cuba
C4	Interaction + South America	**C14**	Interaction + Central America
C5	Interaction + Hpie	**C15**	Interaction + Hpie
C6	Interaction + USA	**C16**	Interaction + USA
C7	Interaction + North America	**C17**	Interaction + North America
C8	Interaction + AHpieBrazilSupplier + Hpie Brazil	**C18**	Interaction + AHpieCubaManufacturer + Hpie Cuba
C9	Interaction + AHpieBrazilSupplier + Brazil	**C19**	Interaction + AHpieCubaManufacturer + Cuba
C10	Interaction + AHpieBrazilSupplier + Hpie	**C20**	Interaction + AHpieCubaManufacturer + Hpie

Figure 6. An interaction influenced by compositions of contextual laws

While composing interaction and environment laws, the interaction between the Hpie Cuba manufacturer and the Hpie Brazil supplier is influenced by the Brazil law 1.6. With this law, the Hpie Cuba manufacturer has to convert his CUP (national currency from Cuba) to R$ (national currency from Brazil) for all his payments.

Several others compositions of laws can be done. According to (4), from **Section 3.4**, the interaction between the Hpie Cuba manufacturer and the Hpie Brazil supplier can be influenced by two hundred and fifty four different contextual compositions of laws. This number of compositions is found by doubling the sum of the different combinations of laws from seven contexts of each player involved in the interaction.

For instance, the seven contexts for the Hpie Brazil supplier player are: Hpie Brazil supplier, Hpie Brazil, Brazil, South America, Hpie, USA and North America. The seven contexts for the Hpie Cuba manufacturer player are: Hpie Cuba manufacturer, Hpie Cuba, Cuba, Central America, Hpie, USA and North America. Figure 6 illustrates some compositions of contextual laws for the interaction law 4.1. **C1** to **C10** represent the compositions from the Hpie Brazil supplier player and **C11** to **C20** represent the compositions from the Hpie Cuba manufacturer player. Just because Hpie is the same main organization of Hpie Brazil and Hpie Cuba, then, some combinations of their contextual laws are the same (such as **C5** and **C15**, **C6** and **C16**, **C7** and **C17**, etc.). Thus, in this example, the interaction law 4.1 will be composed with less than the total of two hundred and fifty four contextual laws.

5 Conclusion

Following the DynaCROM approach, laws are structured in regulatory contexts. Thus, developers are better assisted while they are maintaining and evolving laws. Furthermore, DynaCROM also makes possible to automatically achieve several customized compositions of contextual laws by simply activating rules.

The current implementation of our case study, from the domain of multinational corporations, was done with Java [12], the Jena API [20] and JADE [19]. The Jena API was used as a programmatic environment for OWL [24], to write the DynaCROM rules (according to the Jena rule syntax) and as a rule-based inference engine to compose contextual laws. JADE was used to implement our normative agents; JADE containers were used to represent our environments and organizations (e.g., USA, Cuba, Brazil, Hpie, Hpie Cuba and Hpie Brazil), offering possible locations for agents to go.

As for future work, instead of continuing to work with rules in Jena, we are planning to adopt a rule language specifically developed for the Semantic Web, like RuleML [3], [4], [15], SWRL [16] and TRIPLE [6], [27], [30], and their respective rule-based inference engines, like [2], [13], [23], [28] for RuleML, [13], [17] for SWRL, and [26], [30] for TRIPLE. The reason about that is because Jena is not specific to deal with rules, but rather offers generic solutions. Today, all DynaCROM nineteen rules were already rewritten in RuleML. For instance, the **Rule 1** from **Table 1**, which was originally written according to the rule syntax of Jena, was rewritten according to the rule syntax of RuleML, as presented in Table 6.

Table 6. The environment regulatory context rule in RuleML

```
<Implies>
  <head>
    <Atom>
      <Rel>hasnorm</Rel>
      <Var> Env</Var>
      <Var>OEnvNorm</Var>
    </Atom>
  </head>
  <body>
    <And>
      <Atom>
        <Rel>hasNorm</Rel>
        <Var>OEnv</Var>
        <Var>OEnvNorm</Var>
      </Atom>
      <Atom>
        <Rel>belongsTo</Rel>
        <Var>Env</Var>
        <Var>OEnv</Var>
      </Atom>
    </And>
  </body>
</Implies>
```

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.; Pitt, J. and Sergot, M.; *Animated specifications of computational societies*. In Proc. of AAMAS-2002, Part III, pages 1053-1061, Italy.

[2] Bassiliades, N.; Antoniou G.; *DR-DEVICE*. In: <http://iskp.csd.auth.gr/systems/dr-device.html>

[3] Boley, H.; Grosof, B. and Tabet, S.; (2005) *RuleML Tutorial*. In: <http://www.ruleml.org/papers/tutorial-ruleml.html>

[4] Boley, H.; Tabet, S. and Wagner, G.; *Design Rationale of RuleML: A Markup Language for Semantic Web Rules*. In: Proc. SWWS'2001, Stanford, CA, USA.

[5] Castelfranchi, C.; Dignum, F.; Jonker, C.M. and Treur, J. *Deliberative Normative Agents: Principles and Architecture*. In Proc. of ATAL-99, USA.

[6] Decker, S. et al.: (2005) *TRIPLE - an RDF Rule Language with Context and Use Cases*. In: Proc. W3C Workshop on Rule Languages for Interoperability, Washington, D.C., USA, pp 27-28.

[7] Dey, A.; *Understanding and Using Context. Personal and Ubiquitous Computing*. Springer-Verlag, London, UK; Vol. 5, No.1, (2001) 4-7.

[8] Felicíssimo, C.; Lucena, C.; Briot, J.-P. and Choren, R.; *An Approach for Contextual Regulations in Open MAS.* In Proc. of the AOIS-2006 at AAMAS06

[9] Felicíssimo, C.; Lucena, C.; Briot, J.-P. and Choren, R.; *Supporting Regulatory Dynamics in Open MAS*. In Proc. of the COIN-2006 at AAMAS06

[10] Felicíssimo, C. and Lucena, C.; *An Approach to Regulate Open Multi-Agent Systems Based On A Generic Normative Ontology*. In Proc. of SEAS-2005. Brazil. In: <http://cato.les.inf.puc-rio.br:8080/carol/Publications/seasCAROL.pdf>

[11] Ferber, J.; Gutknecht, O. and Michael, F.; *From Agents to Organizations: an Organization View of Multi-Agent Systems*. In Proc. of the AOSE, 2003. Australia, 2003.

[12] Gosling, J.; Joy, B.; Junior, G.L.S. and Bracha, G.; *The Java Language*. In: <http://java.sun.com/>.

[13] Grosof, B.; Dean, M.; Ganjugunte, S.; Tabet, S. and Neogy, C.; *SweetRules*. In: <http://sweetrules.projects.semwebcentral.org/>

[14] Gruber, T. R.: *A translation approach to portable ontology specifications*. Knowledge Acquisition, Vol. 5, Issue 2. June, 1993. Pages 199-220. ISSN: 1042-8143.

[15] Hirtle, D., Boley, H. (2005) *The Modularization of RuleML* - Version 0.9. RuleML Markup Initiative. In: <http://www.ruleml.org/modularization/#Model>

[16] Horrocks, I.; Patel-Schneider, P.F.; Boley, H., Tabet, S.; Grosof, B. and Dean, M.; (2004) *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*.

[17] Institut für Informatik, Freie Universität Berlin; *An Engine for SWRL rules in RDF graphs*. In: <http://www.ag-nbi.de/research/swrlengine/>

[18] Jennings, N. R. *On Agent-Based Software Engineering. Artificial Intelligence*. 117 (2) pages 277-296. 2000.

[19] JADE. In: <http://jade.tilab.com/>.

[20] Jena. In: <http://jena.sourceforge.net/>.

[21] Hewitt, C.; *Open Information Systems Semantics for Distributed Artificial Intelligence*. AI, 47(1-3), p. 79-106. 1991. ISSN: 0004-3702.

[22] Khedr, M; Karmouch, A.; *ACAI: Agent-Based Context-aware Infrastructure for Spontaneous Applications*. J. of Network & Computer Applications, 28(1), pp. 19-44.

[23] Mandarax. In: <http://mandarax.sourceforge.net/>

[24] McGuinness, D. L. and Harmelen, F. v.: *OWL*. In: <http://www.w3.org/TR/owl-features/>.

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

[26] Ontotext, a Semantic Technology Lab of Sirma. *TRREE, Triple Reasoning and Rule Entailment Engine*. In: <http://www.ontotext.com/trree/index.html>

[27] Sintek, M., Decker, S.; *TRIPLE---A Query, Inference, and Transformation Language for the Semantic Web*. In Proc. of ISWC-2002, Sardinia.

[28] Spencer, B.; *The Design of j-DREW*. In: <http://www.cs.unb.ca/~bspencer/dagstuhl-jdrew.pdf>

[29] Thomas, G. and Williams, A. B.; *Roles in the Context of Multiagent Task Relationships*. AAAI Fall Symposium "Roles, an Interdisciplinary Perspective: Ontologies, Programming Languages, and Multiagent Systems", FS-05-08. ISBN 978-1-57735-254-9. USA. 2005.

[30] TRIPLE; (2006). *The TRIPLE Language*. In: <http://triple.semanticweb.org/>

[31] 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 Proc. of the E4MAS, 2005, p.1-47. ISBN 3-540-24575-8.