$\mathcal{M}$OISE^{Inst}: An Organizational Model for Specifying Rights and Duties of Autonomous Agents

Benjamin Gâteau, Djamel Khadraoui, Eric Dubois

CITI – Centre de Recherche Public Henri Tudor
29, Av. John F. Kennedy – L-1855 Luxembourg, LUXEMBOURG
{*benjamin.gateau,djamel.khadraoui,eric.dubois*}*@tudor.lu*

Olivier Boissier

SMA/G2I/ENSM Saint-Etienne
158, Cours Fauriel – F-42023 Saint Etienne Cedex 02, FRANCE
olivier.boissier@emse.fr

Abstract

In interactive multimedia applications, the objects composing the scene are increasingly considered as autonomous agents. Although autonomy brings flexibility and realism in the animation, these objects needs to be controlled. In this paper we present an organizational model $\mathcal{M}$OISE^{Inst} aiming at specifying the rights and duties of agents in society according to four points of view: structural, functional, contextual and normative. We show how this model is used within an application of interactive TV game show where avatars are represented as agents. This work was supported by an European project called Jules Verne under ITEA program.

Key words: Autonomous multimedia objects, Common rules, Electronic Institution, Norms.

1 Introduction

For a long time, interactive multimedia animation domain have specified objects' behaviours in a rigid way so that they could not behave in a non-expected way [18], [24]. Recently, in order to obtain more flexible scene, objects start to be considered as autonomous agents allowing the definition of scenes in which objects would act by adapting themselves to the context [19]. Multi-Agent Systems offer the possibility to bring more adaptability by viewing objects as agents. The adaptability of objects in the scene results from their ability to modify their behaviours according to their own goals, to the other objects or to the environment in which they are evolving. However agents' autonomy has to be controlled and limited in order to give preference to a global and coherent behaviour in the system. To this aim, we must be able to specify the rights and duties of objects in the context of the scene in which they are plugged.

In this paper we present $\mathcal{M}$OISE^{Inst}, a model for specifying rights and duties of agents within an organization. We illustrate its use in an iTV game issued from the European Jules Verne Project [1]. We show how this model may be used to specify both the organization of the game by itself and the organization of the control system aiming at supervising and controlling the multimedia game.

This paper is organized as follows. Section 2 presents an overview of the multimedia game requirements and of the general framework in which the $\mathcal{M}$OISE^{Inst} model has been developed. The following sections go into the details of this model. Finally, before concluding, section 5 compares our work to other organizational models.

2 Motivations

The actual research work deals with the iTV domain and concerns the creation of multimedia contents with which a user (a televiewer in the context of iTV) interacts. Multimedia objects involved in this creation are considered as autonomous agents able to adapt and modify their behaviours to the modification of the environment and its related scenario. The main contribution concerns the development of an organizational model [9] [5] and of an electronic Institution infrastructure [17] [4] dedicated to the interactive games management. Before presenting the general architecture of this Institution, we present the main scenario adopted in the analysis and development of the $\mathcal{M}$OISE^{Inst} specification.

2.1 Interactive game

Let's consider, a team of televiewers, each being equipped with hardware (remote control and set-top-box) and software developed within the Jules Verne project. The game consists in a "questions–answers" TV game. Being at home, each televiewer is represented in the TV Game by an Avatar (cf. Fig. 1) which is opposed to a real players' team. Avatars are directly controlled by their correspondent televiewers. The quizmaster is also supported by a virtual assistant. His role is to regulate the game and to ensure that participants follow the rules of the game. These ones define the game goals (for instance, *score 30 points to win the game*), as well as more specific rules (for example, *only the player playing the "History" role is authorized to answer question with this label in round 2*).

[1] Jules Verne is an ITEA European project.

As in all collective games, this application aim is to constrain players to adopt a team behaviour and to respect rules. The complexity comes from that teleplayers don't know each other and don't, a priori, intend to play collectively. For example in the second round the televiewer which plays the "History" role has to respect the rules expressed above and answer only questions with same label. However nothing prevents him from answering via his Avatar in spite of that. This explains on one hand why we aim at permitting multimedia objects (Avatars) to evolve autonomously regarding the game rules involved in the scene. Let's notice that in our context autonomy is not considered as autonomy with respect to the users since the Avatars are driven by televiewers. Actually only the televiewer is able to decide to answer whereas it is not his turn and to take the risk to be punished via his Avatar. The autonomy is mainly considered here with respect to the game rules since avatars, hence the user, have the liberty to follow or not those rules. On the other hand, considering the avatars as autonomous agents, we have to define a set of rules representing the game rules coupled with explicit sanctions as for instance: if the player answers while he is not authorized, his good answer brings less points than he could and a bad answer makes him lose point. Let us notice that rules change according to active rounds. Nevertheless the televiewer has to respect them otherwise he will be punished. Finally, given that framework, we must define an arbitration system to control and penalize the culprit and the team he belongs to.

2.2 Electronic Institution of interactive games regulation

In order to define rights and duties of autonomous and generic agents by means of unambiguous specifications, we think that the use of Electronic Institutions is in concordance with our needs.

These last years the Electronic Institution concept has been introduced in Multi-Agent System domain and in electronic commerce in particular. The purpose is to introduce trust among agents during their transactions [4] through an external confident. In human societies an institution defines the *game rules* [17]. These rules enclose all kinds of informal or formal constraints that human beings use to interact. The current multi-agent methods propose the modelling of these rules through normative systems [15]. These ones define an institution as a set of agents which behave according to some norms taking into account their possible violation.

In the same way we define an Electronic Institution for Interactive Games (see Fig. 1) as an autonomous agents' *organization* in which their behaviours are ruled by *norms* and controlled by an *arbitration system*. The role of this arbitration system consists in rewarding or punishing agents when they respect or not their agreements.

The interactive game is thus composed of two layers: (i) the multi-agent interactive game in which avatars as autonomous agents evolve, (ii) an institu-

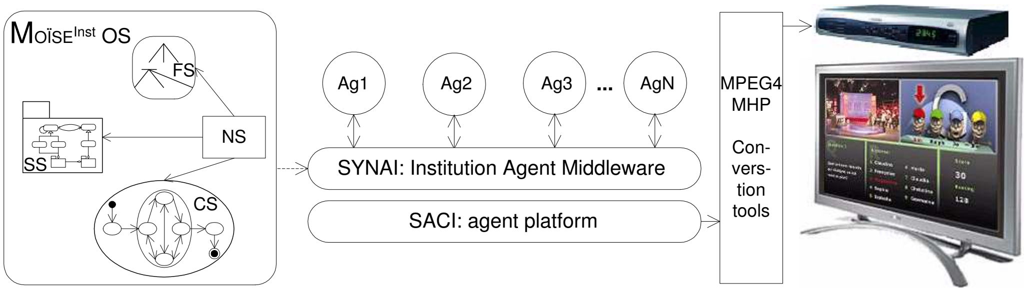

Fig. 1. Global view of MoiseInst Organizational Structure in a TV content production process composed of Structural (SS), Functional (FS), Contextual (CS) and Normative (NS) specifications, taken into account by Agents through Institution Agent Middleware SYNAI

tional multi-agent middleware called SYNAI (**SY**stem of **N**ormative **A**gents for **I**nstitution) dedicated to the management of the organization and to the arbitration. Both layers use a normative organizational model described with the $\mathcal{M}$OISE^{Inst} meta-model which is an extension to $\mathcal{M}$OISE^{+} [13]. This way, agents are able to reason on the specification described with $\mathcal{M}$OISE^{Inst}. They have the possibility to decide to take it into account or not. The institutional middleware reads this specification in order to supervise and control the agents. Both layers rely on the agents execution platform, SACI [12]. In this paper we mainly focus on the presentation of $\mathcal{M}$OISE^{Inst}.

2.3 General view of $\mathcal{M}$OISE^{Inst}

$\mathcal{M}$OISE^{Inst} is founded on the $\mathcal{M}$OISE^{+} organizational model (**M**odel of **O**rganization for mult**I**agent **S**yst**E**m) [2] [13]. $\mathcal{M}$OISE^{+} allows to specify the global expected functioning (functional specification) of an agents organization as well as the structure of this organization in terms of roles, groups and links (structural specification). A deontic specification expresses permissions, obligations and prohibitions of missions referring to the functional specification with respect to the structural specification roles.

To take into account the requirements presented in the scenario such as, for instance, the need to structure the rules according to the game rounds, we have extended the three existing specifications of $\mathcal{M}$OISE^{+} and have added a specification to describe the a-priori dynamic of the system. $\mathcal{M}$OISE^{Inst} is thus composed of (cf. Fig. 1) :

- A structural specification (SS) defining roles that agents will play and relations between these roles as well as an additional structural level named group to which roles belong and in which interaction take place ;
- A functional specification (FS) defining all goals that have to be achieved in the system ;

[2] http://www.lti.pcs.usp.br/moise/

- A contextual specification (CS) defining the different contexts influencing the dynamic of the organization as well as transitions between contexts ;
- A normative specification (NS) extending the $\mathcal{M}$OISE$^+$ deontic specification and defining clearly rights and duties of roles and groups on a mission (set of goals) and in a specific context.

These four specifications form the Organizational Specification (OS). The Organizational Entity (OE) is then built from the set of agents that have adopted a role according to the SS of the OS. From this time, the SINAY middleware manages and controls the functioning of this OE by the way of different events corresponding to the entry/exit of agents of the OE, adoption/leaving roles or groups, change of context, commitment to missions, achievement of goals, etc.

Focus is made to the main contributions of $\mathcal{M}$OISE^{Inst} that consist in CS and NS. We will first rapidly describe the structural and functional specifications that define the general framework where CS and NS take place. Since $\mathcal{M}$OISE^{Inst} aims at being used both by the agents and by the institution middleware, the specifications built with $\mathcal{M}$OISE^{Inst} concern the interactive game on one hand, called hereafter "domain", and the control of the interactive game on the other hand, called hereafter "institution".

3 Structural and Functional Specifications

Structural and functional specifications of $\mathcal{M}$OISE^{Inst} come from $\mathcal{M}$OISE$^+$. Due to lack of space we will not go into details here. The interested reader may refer to [13]. However, in order to figure out a global view of both specifications, we describe the OS built for the scenario described in section 2.1.

3.1 Structural Specification

The $\mathcal{M}$OISE^{Inst} structural specification (SS) allows to describe an organization in terms of *roles*, *groups* and *links* between roles and groups. A set of constraints expresses inter-roles compatibilities, scope of links, cardinality of roles and groups.

The SS dedicated to the domain (cf. Fig. 2) defines the structure of a team with a "Team" group. In this team, Avatars could play the following roles: "History", "Geo", "Sport", "Science" and "Chief". These roles can be adopted by only one Avatar at the same time (cardinality '1..1' on the composition link). These roles inherit from "BasicPlayer" or "Player" roles that are abstract, i.e. roles which are not adoptable by agents. A compatibility link between "BasicPlayer" and "Chief", allows the same agent to play two roles, specialization of those roles. Thus one agent may have the possibility to play at most two of those five roles. In order to avoid that an agent playing the "Chief" role could play several roles of kind "BasicPlayer", we have expressed a cardinality '4..4' for the group "Team", stating that any well formed instance

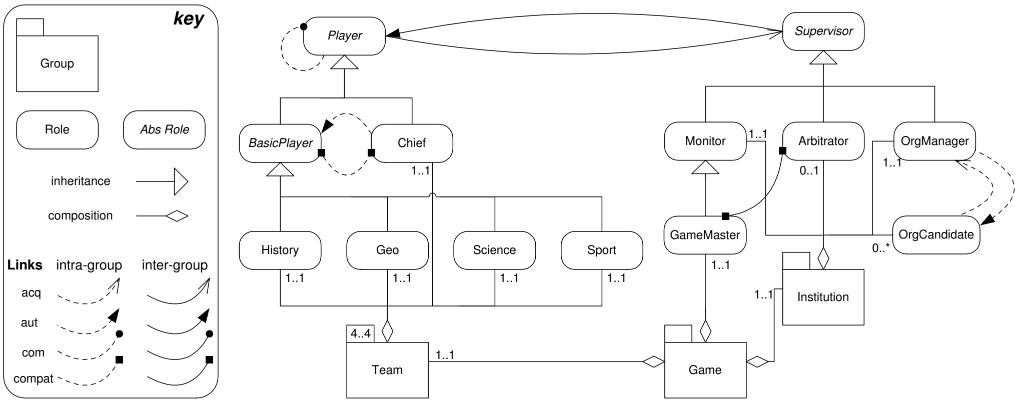

Fig. 2. Avatars scenario Structural Specification

of this group may contain four and only four agents.

In this SS, we can isolate a part, corresponding to the "Institution" group, which is dedicated to the specification of the arbitration system. This part will be considered by the institutional middleware and by the agents themselves to understand how they are controlled. As we can see, the abstract role "Supervisor" is linked with the abstract role "Player" by an inter-group *authority link*. It means that an agent playing the source role is linked to all agents playing the destination role despite the groups these agents belong to. Using the inheritance relation, our SS ensures thus that all roles inheriting the "Player" role will be under the authority of roles inheriting of "Supervisor" role. In order to allow the quizmaster virtual assistant to ask questions as "GameMaster" and/or to punish the agents as "Arbitrator", we draw an intra-group compatibility link between roles "GameMaster" and "Arbitrator" insuring that the agent playing both roles must belong to the same "Institution" group. As we can see, the "Arbitrator" role has a '0..1' cardinality meaning that having an agent playing that role is optional.

"Monitor", "Arbitrator" and "OrgManager" roles inherit from the "Supervisor" role. This means that all these roles will be played by agents belonging to the institutional middleware. The "GameMaster" role inherits from "Monitor" and is part of "Game". So this role can be played by a domain agent. "OrgCandidate" is the first role played by every agents coming in the Institution. This role is able to communicate with the "OrgManager" role in order to allow the agent to know if it can play a role in the "Team" group. As "Supervisor", the "OrgManager" role has authority on "OrgCandidate".

3.2 Functional Specification

As in $\mathcal{M}$OISE^{+} the $\mathcal{M}$OISE^{Inst} functional specification expresses the global functioning of the system as a set of social schemes. A social scheme is composed of plans binding together collective goals. Missions express the a priori distribution of those goals for achievement by the agents. Social Schemes may be reused within other social schemes. For instance, in Fig. 3 the scheme "Score Scheme" which is dedicated to the management of the scoring during the game, is referenced in the "Penalty Scheme".

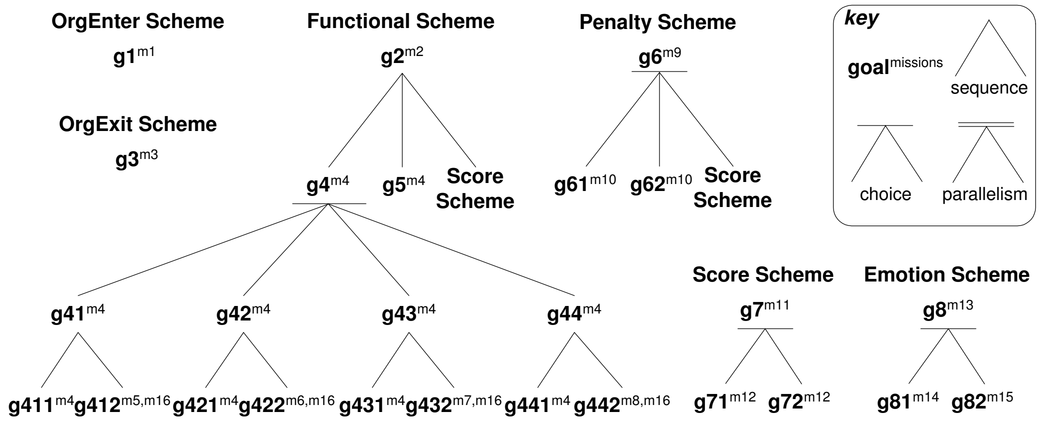

Fig. 3. Avatars scenario Functional Specification

g1: Team joined
g2: X pts scored
  g4: Topic handled
  g5: Answer evaluated
g3: Team quit

g41: "History" topic handled
  g411: "History" question asked
  g412: "History" question answered
g42: "Geo" topic handled
  g421: "Geo" question asked
  g422: "Geo" question answered

g43: "Sport" topic handled
  g431: "Sport" question asked
  g432: "Sport" question answered
g44: "Science" topic handled
  g441: "Science" question asked
  g442: "Science" question answered

g6: Sanction applied
  g61: Player ejected
  g62: Team disqualified
g7: Score changed
  g71: Score increased
  g72: Score decreased
g8: Emotion shown
  g81: Be happy
  g82: Be sad

Fig. 4. Avatars scenario goals definition

As in the SS we decompose the global functioning expressed in the FS into an institution FS which is in relation with the arbitration system, and a domain FS, the one which is in relation with the game itself. The domain global functions correspond to the scoring of the needed points to win the game, to the updating of the score and to the management of emotions. The institution functions are related to the management of the organization (entering/exiting a team) and to the arbitration (applying sanctions).

In the "Functional Scheme", the global goal "g2" corresponds to the scoring of X points ("X pts scored") (cf. Fig. 3 and Fig. 4). It is decomposed into three sub-goals that have to be achieved in sequence: "Topic handled", "Answer evaluated" and "Score changed". This latter corresponds to the root goal of the "Score Scheme". Goal "Topic handled" is achieved by the asking of a question with a topic and the answering to this question. We decompose it into four sub-goals corresponding to each label ({"History", "Geo","Sport","Science"}). The "Score Scheme" consists in choosing between increasing (resp. decreasing) the score "Score increased" (resp. "Score decreased"). The "Emotion Scheme" consists in choosing to show either an happy Avatar "Be happy" or a sad one "Be sad".

The different schemes related to the functioning of the institution, are "Penalty Scheme", "OrgEnter Scheme", "OrgExit Scheme". The "Penalty

Scheme" describes how the agents supervising the organization have to control the respect of rules and to apply the sanctions. As we can see the root goal of this scheme consisting in applying a sanction "Sanction applied" is split into "Player ejected" sub-goal to exclude a player, "Team disqualified" sub-goal to make the other team win and "Score Scheme" to change the score (a decrease most of time). The "OrgEnter Scheme" (resp. "OrgExit Scheme") defines the principal behaviours for entering/leaving an organisation. The root goals describe the goals that the agents playing "OrgCandidate" and "Player" roles have to achieve to enter or quit the "Team" group. These schemes are managed by the "OrgMamager" institutional role.

The FS define the missions consisting in the grouping of goals belonging to the different schemes into sets according to the way the designer wants the global plan to be achieved by the different agents. The link between those sets of goals and the agents will be realized through the Normative Specification that will bind roles or groups to missions. For instance, as shown in Fig. 3, mission "m2" consists only in the goal "g2" i.e. to score X points to win the game, whereas mission "m4" is composed of goals "g4", "g5", "g41", "g42", ...

4 Contextual and Normative Specifications

Given the SS and FS, we are able now to describe and specify the global architecture and the global functioning of an organization. However as shown by several works in multi-agent domain, multi-agent applications are often involved in evolving environment. Depending of the kind of evolution, the designer may be able to express at design-time some constraints on the evolution of the organization. This is the case, for instance, in our application where different rounds structure the execution of the game, imposing to change the rules. We will present the Contextual Specification that we defined to capture this requirement. After that, we will present the Normative specification of $\mathcal{M}$OISE^{Inst} used to bind all specifications in a coherent and normative organization.

4.1 Contextual Specification

The contextual specification CS of an OS describes a priori a set of contexts occupied by the corresponding OE with the transitions governing the change of context. The CS is defined as follows:

⟨CS⟩	::=	'(CS' :context ⟨contextDesc⟩* :transition ⟨transition⟩*
		[:initialCtxt ⟨contextId⟩ :finalCtxt ⟨contextId⟩]')'
⟨contextDesc⟩	::=	'(':id ⟨contextId⟩ [:subcontext ⟨CS⟩*]')'
⟨transition⟩	::=	'(':source ⟨contextId⟩ :target ⟨contextId⟩ [:event ⟨eventId⟩]')'

- ⟨contextDesc⟩ defines the description of a global state occupied by the OE. The context referenced with ⟨contextId⟩ are defined outside the OS with goals, events and actions. We define two special contexts labelled *Start* and *Stop* which are used to define initial and final contexts (represented by the black circles on Fig. 5). As we can see, also on this figure, context "Game" is composed of two concurrent sub-contexts : "Round1", "Round2", "Round3" on one hand and "MyTurn", "NotMyTurn" on the other hand. According to the different events produced in the game, both sub-contexts may evolve in parallel.
- ⟨transition⟩ defines a one way transition from a source context to a target context. The trigger of the transition is done by the production in the OE of an event ⟨event⟩. The events depend on the application. For the interactive game, events are: *beginG* and *endG* corresponding to the start and the end of the game, *chgR* corresponding to a new round, *chgT* produced by a change of turn of team to answer and *avT* if the game start with a question for Avatars (teleplayers) or *hmT* for Humans players.

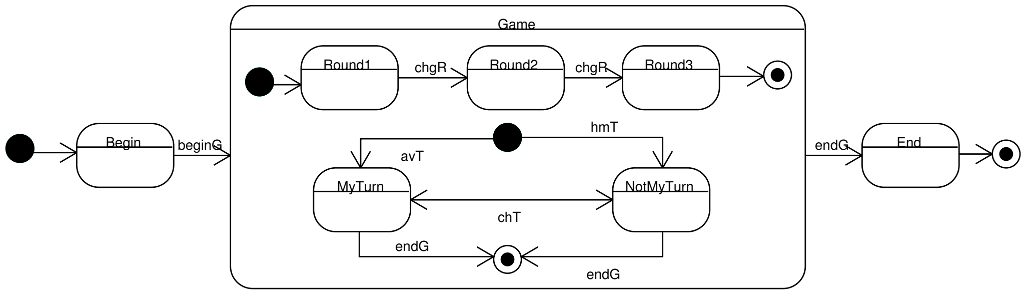

Fig. 5. Avatars scenario Contextual Specification viewed as Statechart Diagram

Fig. 5 shows the CS of our scenario. This specification starts with a synchronous state "Begin" which allows the televiewer to connect to the system. As we will see below in the Normative Specification, the Avatars are in this context authorized to join their team. Moreover it is forbidden to join the team out of this context.

A macro-context "Game" is decomposed into three rounds sub-contexts. This global context will be used to define the basic game rules while the three round sub-contexts will be used to define the corresponding specific rules. The "Game" context is also decomposed in two sub-contexts defining the turn of the players. A round sub-context and a turn sub-context can be active at the same time. Let's notice that the macro-context is active in all its sub-contexts. This way, the rules defined in the "Game" context are inherited in sub-contexts and are thus still valid.

Finally the last state is the context in which Avatars quit their team. As stated before this specification permits to clearly define contexts in which rights and duties of Avatars could be totally different. This is what we outline in the next section.

4.2 Normative Specification

Thanks to the SS, Avatars may be structured into roles and groups. In order to influence their behaviour as specified by the FS, we will define a normative specification NS which is composed of norms. As illustrated in Fig. 1 the NS is the glue between SS, FS and CS.

The concept of norm is used in a lots of domains. The most general definition is: "A standard, model, or pattern regarded as typical" [3] . In Multi-Agent Systems domain norms are defined differently according to their use (constraints, obligations, goals). We will consider here a norm as considered in legal domain. It represents an expression of rights and duties (obligations and permissions) of a role which is played by a person within a society [23]. In $\mathcal{M}$OISE^{Inst}, a norm will define a right or a duty for a role or a group to execute a mission in a particular context and during a given time supervised by an issuer which can apply a sanction if the norm is not respected.
We formally define a norm as follows:

```
⟨norm⟩  ::=  '(Norm ':id ⟨normId⟩ :weight ⟨int⟩ '::' [:conditions ⟨condition⟩]
             :operator ⟨deonticRel⟩ :bearer ⟨sentityId⟩ :issuer ⟨sentityId⟩
             :context ⟨contextId⟩ :action ⟨deonticAct⟩
             [:relation ⟨relation⟩ :deadline ⟨date⟩][:sanction ⟨normId⟩]')'
```

In this specification, different fields express the binding of the different specifications with each other:

- The :bearer field refers either to a role or a group of the SS on which the norm is applied. When the :bearer is a group, all roles belonging to this group in the SS, are the :bearer of this norm. For instance, the interdiction for the "Team" group to answer a question when it is not its turn, is inherited by all roles belonging to this group.
  A norm is also defined toward either a role or a group expressed in the field :issuer. The :issuer defines the entity which has the responsibility to control if the norm is not violated. If the :issuer is a group then all roles that belong to the group have the responsibility to detect a violation and to trigger a sanction.
- the field :context refers to the particular context of the CS in which the norm must be considered.
- the field :action connects missions of the FS to the :bearer of the norm. A :deadline may specify when the norm is valid: before ('<'), when/while ('=') or after ('>') a date expressed by the :relation field.

```
⟨deonticAct⟩  ::=  'do''('⟨missionId⟩')' | ⟨actionId⟩
⟨relation⟩    ::=  '>' | '<' | '='
```

[3] (www.dictionary.com)

- Finally, the field :sanction refers to a norm in the NS itself. It expresses a "sanction" to apply in case of norm violation.

In addition a norm is specified with:

- A :condition expressing the particular state of the OE in which the norm may be applied. Tests on the state of the OE are expressed by functions ⟨function⟩ referring to global variables that are accessible by the :issuer and :bearer of the norm. For instance, we have defined the predicates to test if a norm is violated (*violated*) or not (*respected*). We can know how much agents are part of a group (*number*) and what is the maximum of agents that a group accepts (*cardinalityMax*).

⟨condition⟩ ::= '('(⟨condition⟩ 'AND' ⟨condition⟩) | (⟨condition⟩ 'OR' ⟨condition⟩) | (⟨function⟩ ['!'] '=' ⟨value⟩)')'

⟨function⟩ ::= 'violated('⟨normId⟩')' | 'respected('⟨normId⟩')' | 'number('⟨groupId⟩')' | 'cardinalityMax('⟨groupId⟩')'

- The field :operator defines if the norm is an obligation (O), a permission (P) or a prohibition (F).
- Finally, the :weight field defines a priority level used for solving conflicts between norms, when for instance an agent could be constrained by two contradictory norms. In case of two norms, with the same :weight then the agent has to solve the conflict by taking into account its internal priorities that it can base on the :sanction attached to the norms.

As in SS and FS, in a NS, norms may be separated into those that are relevant to the institution and those that are relevant to domain. A norm which is relevant to the institution is a norm issued from a role belonging to the institution part of the SS. Such a norm is used for controlling the respect and sanctions applied to the "domain" agents.

Given this definition, we are able to express the different rules of the game as norms. Norms can be gathered according to the role which will supervise them. So we have two groups relating to the management of the organisation and the functioning of the system.

For the management of the organisation, we want an institutional norm defining the *Obligation for the "OrgCandidate" role to join a team in order to play a role in the game*:

(Norm :id N1 :weight 1 :condition (number(Team)! =cardinalityMax(Team)) :operator O :issuer OrgManager :bearer OrgCandidate :context Begin :action do(m1))

We define in this norm a condition which tests if there is still role to play in the "Team" group. The issuer is the "OrgManager" role, i.e. it controls this rule and is able to detect if the norm is not respected. With the :context field we specify that this norm is valid only if the "OrgCandidate" role is in this context. So we do not need to use the field :deadline. We define also the *Interdiction for an agent playing "OrgCandidate" role to join a team during*

the game:
(Norm :id N6 :weight 1 :operator F :issuer OrgManager :bearer OrgCandidate :context Game :action do(m1))

Functioning domain norms allow to define global rules of the game :

- *Obligation for the "Team" group to score X points to win the game* represented by:
  (Norm :id N3 :weight 1 :operator O :issuer Arbitrator :bearer Team :context Game :action do(m2))
- or *Obligation for the "GameMaster" role to ask questions and verify answer* represented by:
  (Norm :id N4 :weight 1 :operator O :issuer GameMaster :bearer GameMaster :context Game :action do(m4))
- or *Prohibition for "Player" role to answer a question during the game* represented by:
  (Norm :id N7 :weight 2 :operator F :issuer GameMaster :bearer Player :context Game :action do(m16))

We define this last prohibition ('F' operator) to authorize concerned roles during rounds to answer questions. For the first and third rounds we oblige the "Player" and the "Chief" roles to answer all questions:
(Norm :id N8 :weight 1 :operator O :issuer GameMaster :bearer Player :context Round1 :action do(m16) :relation < :deadline *answer_delay*)
(Norm :id N9 :weight 1 :operator O :issuer GameMaster :bearer Chief :context Round3 :action do(m16) :relation < :deadline *answer_delay*)

We define four norms for each role in the second round to allow concerned roles to answer question. For instance *Obligation for "History" role to answer history-labelled question* is specified by:
(Norm :id N10 :weight 1 :operator O :issuer GameMaster :bearer History :context Round2 :action do(m5) :relation < :deadline *answer_delay*)

Finally we define sanctions as institutional norms with norm violation as condition. For instance *if N6 is not respected the "Arbitrator" ejects the "OrgCandidate" role*:
(Norm :id N17 :weight 1 :condition (violated(N6)) :operator O :issuer GameMaster :bearer Arbitrator :context Game :action do(m9))

5 Related works

In the different works on organizations [9] [5] agents can be constrained to play roles and to belong to groups. Sometimes we can influence the agents behaviour by defining social contracts from an organization. Contracts can concern either two agents or an agent and the society in which it evolves [20] [3]. Norms may also be viewed as a social contracts alternative to constrain agents [2] [1] [11].

In this paper, the norm definition that we use, is derived from several works. The deontic logic is used to differentiate a right (permission) of a duties (obligation) which define the limits for the agents behaviour like in [4]. Inspired by [20] we completed the constraint resulting from the norm with a deadline [20] and an activation condition. We also added a norm issuer. At last [8] helps us to place the norm validation in a scene which we called here context.

In this paper we introduced $\mathcal{M}$OISE^{Inst} that could be compared with others Electronic Institution specification models coming from the MAS domain. In IDE-eli (*the Integrated Development Environment for Electronic Institutions*) [21] contains an Institution Definition Language (IDL) named ISLANDER [7]. This formalism defines a graphical notation that allows to obtain visual representations of scenes and protocols in an Electronic Institution. Compared to $\mathcal{M}$OISE^{Inst} the specification of role hierarchy is minimal in the sense that we can only define roles and inheritance and compatibility between roles. This model is more focused on the specification of interactions and protocols that take part to the definition of scenes. The agents have to follow the protocols to evolve in a scene. There is no sanctions defined. For this reason we think that the flexibility and autonomy of agents is limited in comparison with what permits $\mathcal{M}$OISE^{Inst}. Although it is not an institution specification but an organization specification, the TOP (*Team Oriented Programming*) framework [22] defines the abstraction level of an agents team. It is used to coordinate some heterogeneous agents with known capacities because they are described within an open architecture. The platform allows services wrapped by agents to coordinate their actions. There is no defined hierarchy or rules. However the description of services is interesting to associate methods of domain components with functions defined in a functional scheme or plan for instance. As we don't use ontologies for the moment to bind the functional specification with the skills of each heterogeneous agents taking part into our institution, this work could inspire us for our future works (see next section). The OMNI platform [6] defines on one hand an organizational dimension and on the other hand a normative dimension. The purpose is to define in a complex manner the context in which agents interact. Thus roles, groups, scenes and interactions are seen as norms. No system function, plan or executing scheme is defined here. So we can wonder how the dynamicity, the activity of the society or the agents can be defined. Furthermore if no goals are defined, norms will be held on what?

To conclude, contrary to $\mathcal{M}$OISE^{Inst}, none of these models take into consideration the whole specification points of view (structural, functional, contextual and normative). For us all the specifications are essential and even more essential than an ontological specification (like OMNI) or an interactions specification (like ISLANDER). Nevertheless these specifications are perspectives for our model.

6 Conclusion and perspectives

We have proposed in this paper the $\mathcal{M}$OISE^{Inst} model which is an extension of $\mathcal{M}$OISE^{+}. This model is considered as an institution organization specification especially through the rights and duties description of each society roles. The four elaborated specifications triggers more flexibility in the global model. This flexibility permits to integrate new functionalities (such as role, rule, goals) to the actual model. Norms are seen as relations between roles or groups and missions in a given context. Further work could specify additional contexts like exceptions as in [14,16] which permits that exceptional contexts norms could be invalid.

Two kinds of agents will evolve in our Electronic Institution: the domain agents and the institutional agents. There was no intention to impose a unique domain agents definition due to the objective of heterogeneity. However we can specify the functionalities of the middleware agents (SYNAI). With $\mathcal{M}$OISE^{Inst} we express authority roles that the agents will play, as well as the missions related to their ability to detect norms violations and to punish culprit agents. The services allowing agents to adopt roles and to commit on missions are currently part of SACI communication level. The next evolution is to transfer these services to the institution agent middleware (SYNAI).

We also are validating the current model in relation with an electronic commerce application. The objective is to show the feasibility of our approach in multiple domain areas [10].

7 Acknowledgement

These research works are funded by a grant from the Great Duchy of Luxembourg Ministry of Culture, Higher Education and Research through the LIASIT project (www.liasit.lu).

References

[1] Boella, G. and L. Lesmo, *Deliberate normative agents*, in: *Proceedings of the Workshop on Norms and Institutions in Multi-Agent Systems, 4th International Conference on Multi-Agent Systems (Agents-2000)*, Barcelona, Spain, 2000.

[2] Castelfranchi, C., F. Dignum, C. M. Jonker and J. Treur, *Deliberative normative agents: Principles and architecture*, in: N. R. Jennings and Y. Lespérance, editors, *Intelligent Agents VI – Proceedings of the Sixth International Workshop on Agent Theories, Architectures, and Languages (ATAL-99)*, Lecture Notes in Artificial Intelligence (2000), pp. 364–378, ISBN 3-540-67200-1.

[3] Dellarocas, C., *Contractual agent societies: Negotiated shared context and social control in open multi-agent systems*, in: *Proceedings of the Workshop on Norms and Institutions in Multi-Agent Systems, 4th International Conference on Multi-Agent Systems (Agents-2000)*, Barcelona, Spain, 2000.

[4] Dignum, F., *Agents, markets, institutions, and protocols*, in: F. Dignum and C. Sierra, editors, *Agent-Mediated Electronic Commerce - The European AgentLink Perspective*, Lecture Notes in Artificial Intelligence **1991** (2001), ISBN 3-540-41671-4.

[5] Dignum, V., J. Meyer, H. Weigand and F. Dignum, *An organizational-oriented model for agent societies*, in: *International Workshop on Regulated Agent-Based Social Systems: Theories and Applications (RASTA'02) at AAMAS*, Bologna, Italy, 2002.

[6] Dignum, V., J. Vazquez-Salceda and F. Dignum, *Omni: Introducing social structure, norms and ontologies into agent organizations*, in: R. H. Bordini, M. Dastani, J. Dix and A. E. Fallah-Seghrouchni, editors, *Second International Workshop on Programming Multi-Agent Systems: Languages and Tools (ProMAS 2004)*, New York, USA, 2004.

[7] Esteva, M., D. de la Cruz and C. Sierra, *Islander: an electronic institutions editor*, , **3** (2002), pp. 1045–1052, ISBN 1-58113-480-0.

[8] Esteva, M., J.-A. Rodriguez-Aguilar, C. Sierra, P. Garcia and J. L. Arcos, *On the formal specification of electronic institutions*, in: F. Dignum and C. Sierra, editors, *Agent-Mediated Electronic Commerce - The European AgentLink Perspective*, Lecture Notes in Artificial Intelligence **1991** (2001), ISBN 3-540-41671-4.

[9] Ferber, J. and O. Gutknecht, *A meta-model for the analysis and design of organizations in multi-agent systems*, in: *Third International Conference on Multi-Agent Systems (ICMAS98)*, Paris, France, 1998, pp. 128–135.

[10] Gateau, B., D. Mathevon, D. Khadraoui, O. Boissier and E. Dubois, *Transforming a secure ecommerce application into a multi-agent based solution for e-contracting*, in: M. Mohammadian, editor, *International Conference on Intelligent Agents, Web Technology and Internet Commerce - IAWTIC'2004*, Gold Coast - Australia, 2004, pp. 418–429, ISBN 1740881893.

[11] Hexmoor, H. and X. Zhang, *Norms, roles and simulated robocup*, in: *Proceedings of the Second Workshop on Norms and Institutions in MAS, 5th International Conference on Autonomous Agents*, Montreal, Canada, 2001.

[12] Hubner, J. F. and J. S. Sichman, *Saci: Uma ferramenta para implementaão e monitoração da comunicação entre agentes*, in: M. C. Monard and J. S. Sichman, editors, *International Joint Conference, 7th Ibero-American Conference on AI, 15th Brazilian Symposium on AI (Open Discussion Track)* (2000), pp. 47–56.

[13] Hubner, J. F., J. S. Sichman and O. Boissier, *Spécification structurelle, fonctionnelle et déontique d'organisations dans les systèmes multi-agents*, in: P. Mathieu and J. Müller, editors, *10èmes Journées Francophones Intelligence Artificielle Distribuée & Systèmes Multi-Agents (JFIADSMA'02)* (2002), pp. 205–216.

[14] Inverno, M. d., M. Luck and F. López y López, "Understanding Agent Systems," Springer Series on Agent Technology, Springer-Verlag, 2004, second edition edition .

[15] Jones, A. and J. Carmo, "Handbook of Philosophical Logic," Kluwer, 2001 pp. 203–279.

[16] López y López, F., M. Luck and M. d'Inverno, *Normative agent reasoning in dynamic societies*, , **2** (2004), pp. 732–739, ISBN 1-58113-864-4.

[17] North, D. C., "Institutions, Institutional Change and Economic Performance," Political Economy of Institutions and Decisions, Cambridge University Press, 1990, 1st edition edition, ISBN 0521397340.

[18] Oechslein, C., F. Klgl, R. Herrler and F. Puppe, *Uml for behavior-oriented multi-agent simulations*, in: B. Dunin-Keplicz and E. Nawarecki, editors, *Proceedings of the Second International Workshop of Central and Eastern Europe on Multi-Agent Systems (CEEMAS 2001)*, 2001.

[19] Renault, S., F. Meinkohn, D. Khadraoui and P. Blandin, *Reactive and adaptive multimedia object approach for interactive and immersive applications*, in: *Proceedings of the International Conference on Information & Communication Technologies: From Theory to Applications*, Damascus, Syria, 2004.

[20] Salle, M., *Electronic contract framework for contractual agents*, in: *Proceedings of the 15th Conference of the Canadian Society for Computational Studies of Intelligence on Advances in Artificial Intelligence* (2002), pp. 349–353, ISBN 3-540-43724-X.

[21] Sierra, C., J. A. Rodriguez-Aguilar, P. N. Blanco-Vigil, J.-L. Arcos-Rosell, and M. Esteva-Vivancos, *Engineering multi-agent systems as electronic institutions*, UPGRADE - The European Journal for the Informatics Professional **V** (2004), pp. 33–39, ISSN 1684-5285.

[22] Tambe, M., D. V. Pynadath, N. Chauvat, A. Das and G. A. Kaminka, *Adaptive agent integration architectures for heterogeneous team members*, in: *Fourth International Conference on Multi-Agent Systems (ICMAS-2000)*, 2000, pp. 301–308.

[23] Valente, A., "Legal Knowledge Engineering - A Modelling Approach," Frontiers in Artificial Intelligence and Applications **30**, IOS Press, 1995, ISBN 90-5199-230-0.

[24] Willemsen, P. J., "Behavior and scenario modeling for real-time virtual environments," Computer science, Graduate College of The university of Iowa (2000), supervisor-Joseph K. Kearney.