# A Foundational Ontology of Organizations and Roles

Guido Boella[1] and Leendert van der Torre[2]

[1]Dipartimento di Informatica - Università di Torino - Italy. email: guido@di.unito.it
[2]University of Luxembourg. e-mail: leendert@vandertorre.com

**Abstract.** In this paper we propose a foundational ontology of the social concepts of organization and role which structure institutions. We identify which axioms model social concepts like organization and role and which properties distinguish them from other categories like objects and agents: the organizational structure of institutions, the relation between roles and organizations, and the powers among the components of an organization. All social concepts depend on descriptions defining them, which are collectively accepted, and the description defining the components of organizations, including roles, are included in the description of the organizations they belong to. Thus, the relational dependence of roles means that they are defined in the organizations they belong to. Finally, powers inside organizations are defined by the fact that components of an organization can access the state of the organization whose definition they depend on and of the other components, thus violating the standard encapsulation principle of objects.

## 1 Introduction

In order to constrain the autonomy of agents and to control their emergent behavior in multiagent systems, the notion of organization has been applied [1]. According to Zambonelli *et al.* [2] "a multiagent system can be conceived in terms of an organized society of individuals in which each agent plays specific roles and interacts with other agents". For Zambonelli *et al.* "an organization is more than simply a collection of roles [...] further organization-oriented abstractions need to be devised and placed in the context of a methodology [...] As soon as the complexity increases, modularity and encapsulation principles suggest dividing the system into different sub-organizations".

There is not yet a common agreement, however, on how to model organizations and roles, and, in particular, which are the ontological assumptions behind them. For example, departments and roles are parts of an organization, but they do not exist without it. Can organizations be explained by means of agent based models? Or can they be better modelled with the object oriented paradigm?

Since the existence of institutions depends on what Searle [3] calls the construction of social reality, it is possible that institutions, organizations and roles have very different properties with respect to objects or agents. Searle argues that social reality is constructed by means of so called "constitutive rules" which state what "counts as" institutional facts in the institution. Constitutive rules define institutions: they exist only because of the collective acceptance of constitutive rules by a community.

Searle's construction of social reality does not explain all issues, in particular, the fact that some institutions have a structure in terms of sub-institutions and roles. We

will call them organizations. Thus Searle's analysis is not a sufficient starting point for a foundational ontology, that specifies which are the properties distinguishing social reality from objects and agents. We need to know the axioms which allow to distinguish them from, rather than specifying all the properties of organizations, including those in common with agents. Thus the research questions of this paper are:

- How do organizations and roles differ from objects and agents?
- How can a foundational ontology of social entities, like organizations and roles, be constructed?

In [4–6] we start studying some properties of social entities. However, these works are based on a very specific multiagent framework, which uses the so called agent metaphor, i.e., the attribution of mental attitudes to social entities to explain them.

So in this paper we analyse organizations using an axiomatic ontology and we consider additional properties. The methodology we choose is to extend the ontology of Masolo *et al.* [7]. The main properties of their framework are three. First, it allows to express the fact that social concepts are defined by means of descriptions. Second, it explains the definitional dependence of a role from another concept and the relational nature of roles. Last, it offers a temporalized classification relation, used for modelling the fact that roles are anti-rigid.

We extend Masolo *et al.* [7]'s axiomatic ontology to model institutions and their organizational structure, to explain the asymmetry in the relations defining roles, and to introduce the notion of power relations internal to the organizations. With this work we want to justify the decisions taken in our other works about normative systems, organizations and roles, showing that they all share a common denominator. Second, we want to show that current object oriented representation languages like UML can be extended using the ontology developed in this work, so to ensure a large applicability.

This paper is structured as follows. First, we consider the differences between social reality and objects and agents. In Section 3, we present Masolo *et al.* [7]'s model. In Section 4, starting from the limitations of [7] we extend it to define the foundational ontology. In Section 5, we consider the relation of this ontology with our other works. Conclusions end the paper.

## 2 The properties of organizations

The role of knowledge representation and software engineering is to provide models and techniques that make it easier to handle the complexity arising from the large number of interactions in a system [8]. Models and techniques allow expressing knowledge and supporting the analysis and reasoning about a system to be developed. As the context and needs of software change, advances are needed to respond to changes. For example, today's systems and their environments are more varied and dynamic, and accommodate more local freedom and initiative [9].

For these reasons, agent orientation emerged as a new paradigm for designing and constructing software systems [8, 9]. The agent oriented approach advocates decomposing problems in terms of autonomous agents that can engage in flexible, high-level interactions. Much like the concepts of activity and object that have played pivotal roles

in earlier modelling paradigms - Yu [9] argues - the agent concept can be instrumental in bringing about a shift to a much richer, *socially-oriented ontology* that is needed to characterize and analyze today's systems and environments.

The notions of institution, organization and role are part of this socially-oriented ontology. It is not clear, however, if the ontological assumptions behind this kind of entities are the same which underlie objects and agents. Many approaches recognize as properties of social entities their being the addressee of obligations [10], like agents are, the delegation mechanisms among roles [11], *etc*. Organizations are modelled as collections of agents, gathered in groups [1], playing roles [8, 12] or regulated by organizational rules [2]. We focus instead on the distinguishing properties of social concepts of organization and role.

Consider, for example, an organization which is composed by a direction area and a production area. The direction area is composed by the CEO and the board. The board is composed by a set of administrators. The production area is composed by two production units; each production unit by a set of workers. The direction area, the board, the production area and the production units are *sub-organizations*. In particular, the direction area and the production areas belong to the organization, the board to the direction area, *etc*. The CEO, the administrators and the members of the production units are *roles*, each one belonging to a sub-organization, e.g., the CEO is part of the direction area. This recursive decomposition terminates with roles: roles, unlike organizations and sub-organizations, are not composed by further social entities. Rather, roles are played by other agents, real agents who have to act as expected by their role.

Besides the decomposition structure, as we argue in [6] in organizations we have relations among the components of the organization which specify which are the powers of each component to modify the institutional properties of the other component institutions. This relation does not necessarily matches the decomposition hierarchy. For example, the senior board member has the power to command other members of the board to participate to a board meeting, even if it is at the same decomposition level of the other members. Moreover, the head of a department can give commands to the other members of the department even if they are roles all at the same level. Viceversa, the CEO and the board can take decisions for the whole organization they belong to, for example, committing it to pay for a purchased good.

Is it possible to model such structures in the object oriented paradigm? The object oriented paradigm is based on the idea that software design and implementation can be inspired by our commonsense view of the reality made of objects. For Booch [13] a basic property of objects is that they can be decomposed. Decomposition allows coping with complexity: "the most basic technique for tackling any large problem is to divide it into smaller, more manageable chunks each of which can then be dealt with in relative isolation". Isolation is the idea that code should be encapsulated in classes hiding the implementation of the objects' state; thus, other objects can access an object's state only via its public interface. Decomposition means that an object can include other objects which exist independently of it, like they were parts of the object. But even the components of an object can access it only via its public interface (and vice-versa) to preserve the encapsulation principle.

In case of organizations, the situation is different. First, in the decomposition structure: the components of an organization do not exist independently from the organization itself. For example a department does not exist without the organization it belongs to. If an organization goes bankrupt its departments do not exist anymore and similarly the roles in them (there is no CEO nor employee anymore). Viceversa, an organization can close a department without necessarily giving up its identity. Second, the notion of power inside an organization conflicts with the encapsulation principle of objects.

One alternative could be to see whether organizations can be modelled as agents, but again some difficulties arise. First of all, organizations can have organizations as their parts, while it is debatable whether agents can have parts which are homogeneous with the whole. Moreover, agents can play roles but they cannot have roles as their parts.

However, some form of decomposition should be added to multiagent systems, as noticed by Zambonelli *et al*. [2]: agents alone, and also roles, are not sufficient to deal with the complexity of a system; an organizational structure added to a multiagent system fosters modularity and encapsulation.

A bigger problem is that while agents are autonomous, organizations and roles are not, in two senses. First of all, roles' decisions are taken by the players of the roles: the actions of their players count as decisions of the roles. Analogously, the organizations take a decision on the basis of the decisions of their roles (e.g., the CEO) or sub-organizations (e.g., the board). These relations among decisions are expressed via constitutive rules. Second, a role is not autonomous in the sense that it cannot decide which goals to adopt. Rather, the goals representing the responsibilities of a role are delegated to it by other roles or by the organization itself. For example, an employee can be commanded to perform a task by its director. Analogously, the role's beliefs are assigned to it by other roles and organizations: consider the case of an advocate in a trial who has to show to believe and to support the belief that his client is innocent even if he privately believes otherwise.

We do not consider in this paper, instead, the control structure of organizations, which, e.g., [11] discuss.

Again these relations are expressed via constitutive rules, saying, e.g., that a decision of another role counts as the adoption of a goal by a role. Constitutive rules have been introduced by Searle in its construction of social reality:

> "Some rules regulate antecedently existing forms of behaviour. For example, the rules of polite table behaviour regulate eating, but eating exists independently of these rules. Some rules, on the other hand, do not merely regulate an antecedently existing activity called playing chess; they, as it were, create the possibility of or define that activity. The activity of playing chess is constituted by action in accordance with these rules. The institutions of marriage, money, and promising are like the institutions of baseball and chess in that they are systems of such constitutive rules or conventions" ([14], p. 131).

For Searle, regulative and constitutive norms are related via institutional facts like marriage, money and private property. They emerge from an independent ontology of "brute" physical facts through constitutive rules of the form "such and such an X counts as Y in context C" where X is any object satisfying certain conditions and Y is a label

that qualifies X as being something of an entirely new sort. E.g., “X counts as a presiding official in a wedding ceremony”, “this bit of paper counts as a five euro bill” and “this piece of land counts as somebody’s private property”.

As we say in [6, 15] constitutive rules define the powers among roles and organizations. Powers are behaviors which affect the internal state of another entity (the decision and obligations of an organization, the goals and beliefs of a role, *etc.*) [6]. In particular, in the example above we can distinguish three kinds of power of roles, which we extend here also to sub-organizations:

- Actions of a sub-organization or of a role that are recognized as actions of the organization: e.g., a CEO’s signature on a buy-order, or a decision of the board, is considered as a commitment of its organization to pay for the requested good.
- Actions of the agent playing the role that can modify the state of the role itself. E.g., a director can commit itself to new responsibilities.
- Interaction capabilities among sub-organizations and roles in the same organization. The CEO or the board can send a message to another role, e.g., a command to an employee.

Powers do not only violate the autonomy of organizations, but they violate also the standard encapsulation principle in object orientation described above: a sub-organization or a role which are part of an organization can access the private state of the organization they belong to and of other roles and vice versa (but not the state of the agents playing them, which are autonomous).

If we consider the current ontological analyses of social reality, we find that further differences between organizations and objects and agents have been identified. When roles are considered as predicates like natural kinds (from the linguistic analogy between “John is a person” and “John is a student”), as e.g. [7] do, then there is an asymmetry: John can stop being a student, but he cannot stop being a person. A role like student is anti-rigid because persons are only contingently students. This is a problem for the notion of class used in agent and object orientation which lacks of dynamic reclassification.

Furthermore, roles and sub-organizations are defined in relation to the organizations they belong too. In contrast, the other kinds of entities are defined independently of one another’s definition (albeit in their definitions other concepts are used). This is called *definitional dependence*. This property cannot be accounted for by the current view of object orientation and agent orientation.

Finally we will not consider here the problem of collective acceptance of institutions: institutions do not exist by themselves but they exist only if their definitions in terms of constitutive rules are collectively accepted by the community of agents.

## 3 Background

Masolo *et al.* [7] present a formal framework for developing axiomatical ontologies of socially constructed entities, and study the ontological nature of roles. Social entities and roles exist just because of social conventions, i.e., constitutive rules accepted by

communities of agents: these can be social concepts like organization, nation, money, or social individuals like the DALT workshop or the FIAT company.

In Masolo *et al.* [7] roles are 'properties' according the position defended by Sowa [16]: roles can be 'predicated' of different entities, i.e., different entities can play the same role. The basic properties of roles are the anti-rigidity and being founded. According to Guarino and Welty [17] the definition of foundation is: "a property $a$ is founded on a property $b$ if, necessarily, for every instance $x$ of $a$ there exists an instance $y$ of $b$ which is not 'internal' to $x$". The notion of 'internalness' is complex: e.g., if $x$ is a car, things internal to it can be parts of it (its wheels), but also constituents of it (the metal it is made of) or qualities of it (its color). To avoid all trivial cases, Fine [18] introduces another notion of dependence: "to say that an object $x$ depends upon an $F$ is to say that an $F$ will be ineliminably involved in any definition of $x$".

This notion can be generalized to properties considering that a property $a$ is *definitionally dependent* on a property $b$ if, necessarily, any *definition* of $a$ ineliminably involves $b$. To model this fact 'definitions' are explicitly introduced in the domain of discourse. [7] consider 'reified' social concepts and roles, as well as their descriptions, i.e, the 'social conventions' that define them. This allows to formally characterize in a first-order theory the relationships among all these entities and to talk of roles as 'first-class citizens', similarly to more common entities like objects, events, *etc.*

[7]'s approach is based on a distinction between the properties and relations in the ground ontology (like DOLCE [19]) and those at the object level representing the social reality. The former ones are represented as predicates and therefore assumed as static, rigid, extensional, and not explicitly defined or linked to a description (i.e., the primitive predicates of the theory). The latter ones (called "concepts") are reified and not necessarily static, rigid, and extensional and for which it is possible to explicitly describe some aspects of the conventions that define them (called "descriptions").

Social concepts, denoted by $CN(x)$ are defined ($DF$) or used ($US$) by descriptions ($DS$) and they classify ($CF$) other individuals: $DF(x, y)$ stands for "the concept $x$ is defined by the description $y$" to deal with the social, relational, and contextual nature of social concepts. $US(x, y)$ stands for "the concept $x$ is used by the description $y$"; they introduce a temporalized classification relation to link concepts with the entities they classify, while accounting for the dynamic behavior of social roles: $CF(x, y, t)$ stands for "at the time $t$, $x$ is classified by the concept $y$" or, more explicitly, "at the time $t$, $x$ satisfies all the constraints stated in the description of $y$".

In the axioms defining [7]'s theory, $ED(x)$ stands for "$x$ is an endurant", i.e., an entity that is wholly present at any time it is present, e.g., a book, Hakodate, a law, some metal, *etc.* $NASO(x)$ stands for "$x$ is a non-agentive social object", i.e., an endurant that: (i) is not directly located in space and, has no direct spatial qualities; (ii) has no intentionality; (iii) depends on a community of intentional agents, e.g., a law, an organization, a currency, an asset *etc.*; $TL(x)$ stands for "$x$ is a temporal location", i.e., a temporal interval or instant; $P(x, y)$ stands for "$x$ is part-of $y$", for perdurants and temporal locations; $PRE(x, t)$ stands for "$x$ is present at the time $t$".

We report here the most important axioms of their theory. Concepts, and descriptions as well, are non-agentive social objects; concepts are linked to descriptions by the relations used-by ($US$) and defined-by ($DF$). Theorem T2 below captures the fact that

a concept must be defined by a single description. This is not true for the $US$ relation: concepts can be used by different descriptions.

(A1) $DS(x) \supset NASO(x)$
(A2) $CN(x) \supset NASO(x)$
(A3) $DS(x) \supset \neg CN(x)$
(A4) $US(x, y) \supset (CN(x) \wedge DS(y))$
(A5) $DF(x, y) \supset US(x, y)$
(A8) $(DF(x, y) \wedge DF(x, z)) \supset y = z$
(T1) $DF(x, y) \supset (CN(x) \wedge DS(y))$
(T2) $CN(x) \supset \exists! y(DF(x, y))$
(A11) $CF(x, y, t) \supset (ED(x) \wedge CN(y) \wedge TL(t))$
(A14) $CF(x, y, t) \supset \neg CF(y, x, t)$
(A15) $(CF(x, y, t) \wedge CF(y, z, t)) \supset \neg CF(x, z, t)$

The properties of anti-rigidity ($AR$) and foundation ($FD$) for roles can be defined in this formalism. A concept is anti-rigid if, for any time an entity is classified under it, there exists a time at which the entity is present but not classified under the concept:

(D1) $AR(x) \equiv_{df} \forall y, t(CF(y, x, t) \supset \exists t'(PRE(y, t') \wedge \neg CF(y, x, t')))$

A concept $x$ is founded if its definition involves (at least) another concept $y$ (definitional dependence) such that for each entity classified by $x$, there is an external entity classified by $y$:

(D2) $FD(x) \equiv_{df} \exists y, d(DF(x, d) \wedge US(y, d) \wedge$
$\forall z, t(CF(z, x, t) \supset \exists z'(CF(z', y, t) \wedge \neg P(z, z', t) \wedge \neg P(z', z, t)))$

Roles are anti-rigid and founded:

(D3) $RL(x) \equiv_{df} AR(x) \wedge FD(x)$

Masolo *et al.* [20] extend [7]'s framework introducing explicitly a relation between an institution and a role to express that a role like student is relationally dependent, e.g., for a person to be a student it requires the existence of another entity, namely a certain university, to which this person is related by an enrollment relation. As Steimann [21] shows, this view of roles as anti-rigid and relationally dependent predicates is supported by the vast majority of approaches in the conceptual modeling and object-modeling literature.

Roles can be defined on the basis of a relation whose arguments are characterized by specific properties. For example, the role of 'being a student' can be defined as: "a student is a person enrolled in a university". In this case, 'being a student' is defined on the basis of 'being enrolled in', 'being a person', and 'being a university'. Formally, considering the previous properties as predicates, this definition can be formulated as:

$Student(x) \equiv_{df} Person(x) \wedge \exists y(enr(x, y) \wedge University(y))$

But given a specific relation $r$ of arity $n$, it is possible to define $n$ different predicates. For example, in the case of the relation $enr(x, y) \supset (Person(x) \wedge University(y))$, the predicate $EnrollingUni$ can be defined as:

$EnrollingUni(x) \equiv_{df} \exists y(enr(y, x))$

Hence the authors are aware that there is an asymmetry in the relation defining roles. $EnrollingUni$ has exactly the "same logical form" as $Student$, but this does not imply that $EnrollingUni$ is a role. Let us assume a theory containing an axiom stating that, necessarily, universities enroll at least one student, i.e., when a university loses all its

students, it ceases to be a university. In this theory, 'being an enrolling university' is a rigid property of universities, and therefore it cannot be a role (assuming $University$ as rigid). In addition, the two predicates $EnrollingUni$ and $University$ coincide from an extensional point of view (since all universities are enrolling universities) and they cannot be distinguished by means of the theory. In this case, the predicate $EnrollingUni$ seems "redundant" with respect to the predicate $University$ because they are provably equivalent.

To extend [7]'s framework to take into account the reification of $n$-ary relations, [20] introduce a classification relation where a relation $r$ is considered in the domain of quantification: $CF(x_1, \ldots, x_1, r, t)$ stands for "at the time $t$, the individuals $x_1, \ldots, x_n$ are classified by the relation $r$". Second, they extend the primitives $DF$ and $US$ to the reification of predicates in general, i.e., both concepts and relations.

The fact that $Student$ and $EnrollingUni$ are concepts defined on the basis of the same relation $enr$ is represented by the fact that $Student$, $EnrollingUni$, and $enr$ are used in the same description $d$. Moreover, a link between a relation and the concepts it defines is necessary to avoid the symmetry with the other arguments of the relation. They thus introduce the predicate $df$, with $df(x, y)$ standing for "the (relational) concept $x$ is defined by the relation $y$". Clearly, in order to define a relational concept $x$, a description needs to use the relation $y$ by which $x$ is defined:
$(DF(x, d) \wedge df(x, y)) \supset US(y, d)$.

## 4 The ontology of organizations

### 4.1 Ontological requirements

Summarizing the discussion in Section 2, the basic properties of institutions, organizations and roles are that, first, organizations have an organizational structure in terms of sub-organizations and roles. Second, roles are defined by the organizations they belong to. The decomposition hierarchy of the organizational structure, however, is not based on the part-of relation of objects. In particular, it is transitive (a role in a department is part of the organization the department belongs to), but the parts do not exist without and before the whole. Third, there is another type of relation among the parts of an organizations, specifying which components have power on other components.

The formal framework of Masolo *et al.* [7] is the suitable starting point for defining a foundational ontology of organizations and roles. Our requirements, however, are not fully satisfied in their axiomatization.

First of all, they do not consider the structure of social entities. They do not define sub-organizations nor roles as parts of organizations. So a social entity does not have a recursive decomposition structure. Roles have been recognized as depending on some other entity which is used in their definition, but they are not defined in the entity they depend on. Moreover, we need to extend this dependence relation to specify that also sub-organizations, and not only roles, depend on the organizations.

Moreover in [7] there is no notion of power, that is the possibility that the components of an organization can affect the state of each other. However, they offer the notion of a description defining an institution, which we will use for introducing power.

The extended framework of [20] is a closer starting point for our axiomatization. The introduction of an explicit relation between an institution and a role explains the link between them. But still they do not capture the fact that a role is part of the institution and it is defined by it as we claim.

We will fulfill the above requirements in our ontology in the following way. The organizational structure of an institution is defined exploiting the fact that a social entity is defined by a description. We say that a sub-organization or a role are defined by a description which is part of the description defining the institution they belong to. This explains also why the relations associating roles to institutions are asymmetric and why roles are part of the institution and not only involved in a relation with the institution.

Concerning power, we have to model the fact that a behavior of an organization or role can access the state of another organization or role where institutional facts are represented as private behaviors or private properties. A behavior can be an action, in an agent setting, or a method, in an object oriented one which makes an institutional change; a property can be a goal, a belief, an obligation of an organization or role, *etc*. The fact that a description of an organization contains the description of a sub-organization allows the components of the organization to access each other. The idea is that all components of an organization are defined at the same time and by the same author, thus it is safe that the private methods or actions and private properties of a component can be accessed by another component's methods or actions. A behavior or a property can be accessed by a behavior not only when it is public, but also when it is private. The condition is that the accessed entity is an organization and the entity who is accessing it is a component of that organization or belongs to the same organization (e.g., when a role accesses another role). In these cases we say that the behavior is a power.

### 4.2 Concepts and relations

In the ontology we define the following predicates used in the definitions below:

- The predicates social concept $CN$ and description $DS$ are borrowed from [7]. Moreover, we need the concept of behavior *BH* and property *PROP* to model methods or actions and properties of entities, either real or social.
- The part-of relation $P$ is extended to hold between descriptions: a description $d$ of a concept $c$ can use $US$ other concepts, but it can also include the definition of another concept. We assume $P$ is a transitive property and that a part (pre)exists independently of the whole:
  $P(a, b) \supset \exists t(PRE(a, t) \wedge \neg PRE(b, t))$
- The classification relation $CF$ is extended as in [20] to relations; we omit the temporal index when it is not necessary.
- The relation *defined-by* relates concepts and descriptions $DF(c, d)$: the concept $c$ ($CN(c)$) is defined by the description $d$ ($DS(d)$). The defined-by relation is used also to define the relation *MDF* which identifies a minimal description of a concept $c$: a description which cannot be reduced without being unable to define the concept.
  $\mathit{MDF}(c, d) \equiv_{df} DF(c, d) \wedge \neg\exists d' P(d', d) \wedge DF(c, d')$

Note that to have non-minimal descriptions we have to change Axiom A8 of [7] (and thus theorem T2), so that only minimal descriptions are required to be unique:
(A8') $(\mathit{MDF}(x,y) \wedge \mathit{MDF}(x,z)) \supset y = z$

– Besides describing concepts, descriptions define relations between concepts and their properties and behaviors (e.g., methods and actions). We distinguish two kinds of relations between a concept and a property or behavior: $public$ and $private$. This captures the idea usual in programming languages or in modelling languages like UML that some properties and behaviors are accessible (properties can be visible or modified, and behavior invoked) by other entities while some others are not. In Section 4.4 we show that privately accessible properties and methods play a role in the definition of powers of organizations and roles.
Thus, in order to define accessibility we reify the two special relations *private* and *public*. $CF(c, i, \mathit{private})$ means that the concept $c$ and the property or behavior $i$ are classified by the *private* relation defined by description $d$ $DF(\mathit{private}, d)$.
$CF(c, i, \mathit{private}) \supset ED(c) \wedge (\mathit{BH}(i) \vee \mathit{PROP}(i))$
– The access relation specifies when behaviors associated to entities can access the behaviors and properties of other entities:
$access(x, a, y, b) \supset \mathit{BH}(a) \wedge (\mathit{BH}(b) \vee \mathit{PROP}(b))$
This access relation is expressed in terms of public properties and behaviors, but it is also defined in more complex terms when we have organizations.

### 4.3 The structure of organizations

The first requirement of a foundational ontology is that organizations are institutions which have a structure. We do not introduce here a primitive part-of relation between organizations and suborganizations, nor we can use $P$ since we need different properties, like the fact that the parts do not exist without the whole. An organization $c$ is part-of *IP* another organization $c'$ if it is defined inside the minimal description defining the other one. Note that we need a minimal description, otherwise we could have a description $d$ which is the union of two (minimal) descriptions $d'$ and $d''$ defining two unrelated concepts. Requiring a minimal description thus means that the definition of $c$ is essential to define $c'$.

$\mathit{IP}(c, c') \equiv_{df} \exists d, d'\ \mathit{MDF}(c, d) \wedge \mathit{MDF}(c', d') \wedge P(d, d')$

Since the $P$ relation between descriptions is transitive, also the *IP* relation is transitive: a role which is part of a sub-organization of an organization, it is also part of the organization.

The following axiom states that if a sub-organization $c$ is part of organization $c'$ then the concept $c'$ is used in the definition of $c$.

(B1) $\mathit{IP}(c, c') \supset \exists d\ \mathit{MDF}(c, d) \wedge US(c', d)$

We can use the *IP* predicate to define our notion of definitional foundation *DFD*. Our definition is a revised version of the founded $FD$ predicate of [7]. It captures the idea that an instance of sub-organizations and roles is not only an instance of a concept which is part of (*IP*) another concept, but it requires the existence of an instance of such concept.

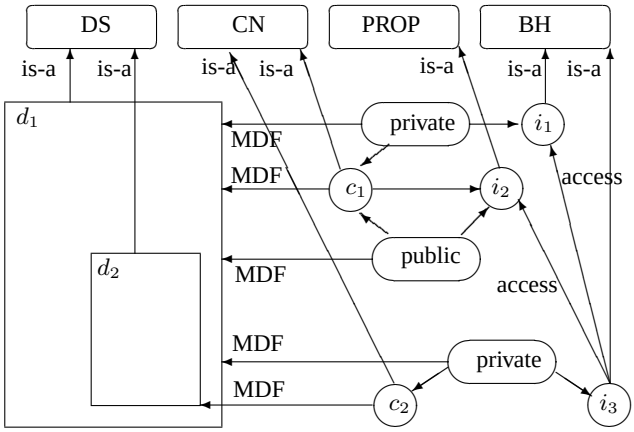


**Fig. 1.** An example of organization.

**Definition 1 (Definitional foundation).**

*DFD*$(x) \equiv_{df}$
$\exists y$ *IP*$(x,y) \wedge \forall z,t\ (CF(z,x,t) \supset \exists z'(CF(z',y,t) \wedge \neg P(z,z',t) \wedge \neg P(z',z,t)))$
*We write also:*
*DFD*$(x,y) \equiv_{df}$
*IP*$(x,y) \wedge \forall z,t\ (CF(z,x,t) \supset \exists z'(CF(z',y,t) \wedge \neg P(z,z',t) \wedge \neg P(z',z,t)))$

The difference with respect to the $FD$ predicate of [7] is that it does not require that a concept is used in a definition of $x$, but that the definition is part of another concept.

Which is the relation between the two definitions? The *DFD* property is stronger than $FD$ since we assume Axiom B1.

**Theorem 1.**

*From Axiom B1 and from the fact that MDF*$(x,d) \supset DF(x,d)$ *we have:*
*DFD*$(x) \supset [\exists y, d\ DF(x,d) \wedge US(y,d) \wedge$
$\forall z,t\ (CF(z,x,t) \supset \exists z'\ (CF(z',y,t) \wedge \neg P(z,z',t) \wedge \neg P(z',z,t)))] \supset FD(x)$

We can introduce now our definition of institutions, organizations and roles. Institutions are simply social concepts defined by descriptions, organizations are institutions which have sub-organizations and roles as their parts, sub-organizations are organizations which are definitionally founded on some organization and roles are anti-rigid definitionally founded concepts, and there is no institution dependent on them.

**Definition 2 (Institutions, organizations and roles).**

*INST*$(x) \equiv_{df} CN(x)$
*ORG(x)* $\equiv_{df}$ *INST*$(x) \wedge \exists y$ *DFD*$(y,x)$
*S-ORG*$(x) \equiv_{df}$ *ORG*$(x) \wedge$ *DFD*$(x)$
*RL(x)* $\equiv_{df} AR(x) \wedge$ *DFD*$(x) \wedge \neg\exists y$ *DFD*$(y,x)$

In the following example a simple organization composed by one institution with one role is illustrated:

*Example 1.*

$c_1$ is an organization which is minimally defined by description $d_1$ (see Figure 1). It has a private behavior $i_1$ and a public property $i_2$. Description $d_1$ includes also a subdescription $d_2$ which is the minimal description of the concept $c_2$, a role of $c_1$. $c_2$ has a private property $i_3$.

$DS(d_1)$, $DS(d_2), CN(c_1), CN(c_2),$ ***BH***$(i_1),$ ***PROP***$(i_2),$ ***BH***$(i_3) P(d_2, d_1)$
***MDF***$(c_1, d_1)$, ***MDF***$(c_2, d_2)$, ***MDF***$(d_1,$ *private*$)$, ***MDF***$(d_1,$ *public*$)$, ***MDF***$(d_2,$ *private*$)$[1]
$CF(c_1, i_1,$ *private*$)$, $CF(c_1, i_2,$ *public*$)$, $CF(c_2, i_3,$ *private*$)$

Thus, $c_2$ is a part of $c_1$: ***IP***$(c_2, c_1)$

In [6] we consider another definitional property of roles: the fact that a role can play a role. This property is implicit in the fact that nothing prevents that a role, as a social concept, can be classified by another role. Note that this is in contrast with the position about role playing roles stated in [7], even if their model allows also the alternative we choose.

### 4.4 Powers

Properties and behaviors associated with organizations cannot be all freely accessed by any agent. Some of them, e.g., the building where an organization is officially located, are physical properties which every agent can manipulate. In contrast, other properties have only a social character, and thus are immaterial: the id number of the employees, the action of firing an employee, making the organization buy some goods, obliging an employee to do something, changing the structure of the organization. Since these institutional properties and behaviors are immaterial, how can they be manipulated? As discussed in Section 2 institutional properties are controlled by counts as rules. In our ontology we model counts as rules defining powers as behaviors of social entities (organizations and roles) which access properties. To represent the fact that institutional properties can be manipulated only from inside an institution we model them as private properties of institutions.

The problem to be solved is the behaviors of which entities can access a private property of an institution, since the visibility rules in organizations are different than in objects. The notion of power is thus based on the definition of an $access$ relation defining a sort of scope for behaviors and properties.

We do not describe here behaviors. For example, actions could be described by plan operators and methods by programs. We represent, however, that a behavior accesses other behaviors (since they are actions in a plan or invoked by a program) or some properties (the value of the property is needed for executing the behavior or it is changed by the behavior).

[1] This does not mean that *public* is defined twice, but that its extension (i.e., the tuples of entities classified $CF$ by it) is determined by both descriptions.

An entity classified by a concept $c$ can access a property $p$ of another entity, if it is a public property, or it is a private property of a concept $c'$ whose definition defines also the concept $c$ (i.e., $c$ is part of *IP* $c'$) or of another concept which depends on $c'$ too.

**Definition 3 (Powers).**

*The access relation is defined as:*

$access(x, a, y, b) \equiv_{df} public(y, b) \vee superaccess(x, y, b) \vee peeraccess(x, y, p)$

*A method or property $b$ of individual $y$ is public if it is a public behavior or property of a concept subsuming $y$:*

$public(y, b) \equiv_{df} \exists c\, CN(c) \wedge CF(y, c) \wedge CF(c, b, \textit{public})$

*A method or property $b$ of individual $y$ can be accessed from individual $x$ if the concept subsuming $x$ is part of another concept subsuming $y$ (or vice-versa) and $b$ is a private behavior or property of the latter.*

$superaccess(x, a, y, b) \equiv_{df} \exists c, c'\, CF(x, c) \wedge CF(y, c') \wedge \textit{IP}(c, c') \wedge CF(c', b, \textit{private})$

*A method or property $b$ of individual $y$ can be accessed from individual $x$ if the concept subsuming $x$ is part of another concept which has as its part a concept subsuming $y$ and $b$ is a private behavior or property of this concept.*

$peeraccess(x, a, y, b) \equiv_{df} \exists c, c', c''\, CF(x, c) \wedge CF(y, c') \wedge \textit{IP}(c, c'') \wedge \textit{IP}(c', c'') \wedge CF(c', b, \textit{private})$

*A behavior of an entity is a power if it can $superaccess$ or $peeraccess$ another behavior.*

$\textit{POW}(x, a) \equiv_{df} \exists y\, superaccess(x, a, y, b) \vee peeraccess(x, a, y, b)$

If we impose that *IP* is a reflexive relation, then we have that a behavior of an entity can access the private behaviors and properties of itself and that the behaviors of an organization can access the private state of its components.

Note that the $access$ relation specifies which behaviors can access other behaviors and properties. This does not mean that in an actual organization every behavior accesses every other behaviors or properties. The fact that a behavior accesses some other behavior or property depends on how this behavior is defined in the description by means of plans or programs. As we said, the author of the definition of the organization is the author of the definitions of its components, so the access definition is safe. Nothing prevents, however, that a more restrictive definition of access is given to respect the organizational structure. For example, it can be defined on a non-transitive part-of relation, so that each component can have powers only on its direct super or sub components or on its siblings.

*Example 2.* In Figure 1 behavior $i_3$ can access both behavior $i_1$ and property $i_2$ even if the former is private, since *IP*$(c_2, c_1)$.

Note that dealing with visibility rules in a programming languages is a complex issue. In this model we do not want to propose to define a general notion of accessibility, but to study the peculiarities of accessibility in organizations.

## 5 Applications

In this section we explain how the foundational ontology presented here matches our previous work, and, in particular, how it can be used to introduce organizations and roles not only in multiagent systems but also in the object oriented paradigm.

We study normative systems [15, 22] and organizations composed of sub-organizations and roles [5, 6] using the so called agent metaphor. The agent metaphor allows to describe social entities, like normative systems, as they were agents, and thus attributing them mental attitudes like beliefs and goals. What corresponds in the agent metaphor to the basic primitives of our foundational ontology? First of all, we have to explain the structure of an organization. An agent does not have parts which are agents themselves, so an organization-as-an-agent cannot have other organizations as its parts. Rather, to structure organizations we exploit the idea that an agent can attribute mental attitudes to other entities via the agent metaphor, also to entities which are not agents. Since an organization is described as an agent, then it can attribute mental attitudes to other entities. In this way, it can define sub-organizations and roles by describing them as agents, in a recursive way.

As Searle claims, social entities are defined by means of constitutive (and regulative) rules. In [15]'s model beliefs attributed to a social entities correspond to the constitutive rules and goals the regulative rules. Thus, describing a social entity as an agent amounts to defining it. A definition of a sub-organization is included in the definition of the organization it belongs to since, in the definition of the latter are present not only the beliefs and goals attributed to them, but also the beliefs and goals which it attributes to sub-organizations and roles.

Like in our foundational ontology, powers arise from the fact that all the structure of the organization is defined in the same definition: so that the constitutive rules of a sub-organization can refer to other sub-organizations as well.

Even if at first sight can be surprising, our foundational ontology of organizations can be used to model organizations by means of standard object oriented representation languages, like UML. Rather than adding primitives to UML, we use a pattern. This does not mean that it is not useful to introduce some primitives which are based on this pattern. As a consequence, institutions can be introduced also in object oriented programming languages like Java. Thus, in [23, 24] we present an extension of Java, called powerJava, where suitable constructs are introduced to represent roles.

The basic idea is that the description of a concept in object orientation corresponds to a class and in UML and some programming languages a class can contain other classes, called inner classes. Outer classes correspond to descriptions having other descriptions as parts. An inner class can contain further inner classes as well, thus allowing a recursive decomposition structure. Moreover, inner classes have the features we need for modelling institutions: dependence and powers. First, an instance of an inner class does not exist without an instance of the outer class, since it has a reference to an instance of the outer class. Second, the methods of an inner class can access the state of the outer class and of other sibling inner classes. Powers thus can be modelled just as the methods of an inner class.

The difference between sub-organizations and roles is that roles do not have further inner classes inside them and that they are associated to a player via a reference. Roles

are anti-rigid because they are associated to their players by a reference, rather than being modelled as sub or super classes (like other proposals for representing roles, e.g., [25], do instead). Thus an inner class representing a role has always two references to two objects: the institution that defines it and the player that plays it.

## 6 Conclusions

In knowledge representation, and more specifically in the field of description logics, the term 'role' is nowadays synonymous of an arbitrary binary relation (often a function) used to characterize the structure of a concept. The concept 'person', for instance, may have the role 'likes', which represents the relationship between a person and what she likes best. But this is not what is meant by social roles.

In multi-agent systems (MAS) roles are generally viewed as descriptions of agent's acting and interacting, where agents include also societies or organizations of agents. The characterization of this kind of social roles (in the restricted sense) is founded on theories of action and behavior (involving tasks, goals, plans, *etc*.) and deontic notions. In [2] a role is viewed as an "abstract description of an entity's expected function" which is defined by four attributes: responsibilities (that determine the functionality of the role), permissions, activities, and protocols. Pacheco and Carmo [26] clearly distinguish roles from agents (agents can act, and roles cannot). But these descriptions do not tell much about what distinguish roles from objects or agents.

In object-oriented programming languages the focus has been on technical issues (multiple and dynamic classification, multiple inheritance, objects changing their attributes and behaviors, *etc*.), rather than what are the roles' distinguishing properties.

In this paper we propose a foundational ontology of organizations and roles which extend Masolo *et al.* [7]'s proposal. Institutions are social concepts which exist because of descriptions defining them, which are collectively accepted. Organizations are institutions which have a structure in terms of sub-institutions. Sub-organizations are organizations which are parts of other organizations. Finally, roles are components of organizations which do not have further organizational structure and which can be played by agents.

This work builds on our previous work on normative multiagent systems and organizations based on the agent metaphor. In [4] we present the agent metaphor to build a cognitive ontology. Here, instead we present an axiomatic ontology built in an analytical style. This work aims at isolating the essential properties which distinguish social concepts from other kind of entities and to justify the choices made in previous works. Moreover, we show that this ontology can be used to extend current representation languages like UML and object oriented programming languages.

## References


1. Ferber, J., Gutknecht, O., Michel, F.: From agents to organizations: an organizational view of multiagent systems. In: LNCS n. 2935: Procs. of AOSE'03, Springer Verlag (2003) 214–230
2. Zambonelli, F., Jennings, N., Wooldridge, M.: Developing multiagent systems: The Gaia methodology. IEEE Transactions of Software Engineering and Methodology **12(3)** (2003) 317–370

3. Searle, J.: The Construction of Social Reality. The Free Press, New York (1995)
4. Boella, G., van der Torre, L.: An agent oriented ontology of social reality. In: Procs. of FOIS'04, Amsterdam, IOS Press (2004) 199–209
5. Boella, G., van der Torre, L.: Organizations as socially constructed agents in the agent oriented paradigm. In: LNAI n. 3451: Procs. of ESAW'04, Berlin, Springer Verlag (2004) 1–13
6. Boella, G., van der Torre, L.: The ontological properties of social roles: Definitional dependence, powers and roles playing roles. In: Procs. of LOAIT workshop at ICAIL'05. (2005)
7. Masolo, C., Vieu, L., Bottazzi, E., Catenacci, C., Ferrario, R., Gangemi, A., Guarino, N.: Social roles and their descriptions. In: Procs. of KR'04, AAAI Press (2004) 267–277
8. Jennings, N.R.: On agent-based software engineering. Artificial Intelligence **117(2)** (2000) 277–296
9. Yu, E.: Agent orientation as a modelling paradigm. Wirtschaftsinformatik **43(2)** (2001) 123–132
10. Dastani, M., van Riemsdijk, B., Hulstijn, J., Dignum, F., Meyer, J.J.: Enacting and deacting roles in agent programming. In: Procs. of AOSE'04, New York (2004)
11. Grossi, D., Dignum, F., Dastani, M., Royakkers, L.: Foundations of organizational structures in multiagent systems. In: Procs. of AAMAS'05. (2005)
12. McCallum, M., Norman, T., Vasconcelos, W.: A formal model of organisations for engineering multi-agent systems. In: Procs. of CEAS Workshop at ECAI'04. (2004)
13. Booch, G.: Object-Oriented Analysis and Design with Applications. Addison-Wesley, Reading (MA) (1988)
14. Searle, J.: Speech Acts: an Essay in the Philosophy of Language. Cambridge University Press, Cambridge (UK) (1969)
15. Boella, G., van der Torre, L.: A game theoretic approach to contracts in multiagent systems. IEEE Transactions on Systems, Man and Cybernetics - Part C (2006)
16. Sowa, J.: Knowledge Representation: Logical, Philosophical, and Computational Foundations. Brooks/Cole, Pacific Growe (CA) (2000)
17. Guarino, N., Welty, C.: Evaluating ontological decisions with ontoclean. Communications of ACM **45(2)** (2002) 61–65
18. Fine, K.: Ontological dependence. Proceedings of the Aristotelian Society **95** (1995) 269–290
19. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening ontologies with dolce. In: Proc. EKAW 2002, Siguenza (SP) (2002)
20. Masolo, C., Guizzardi, G., Vieu, L., Bottazzi, E., Ferrario, R.: Relational roles and qua-individuals. In: Procs. of AAAI Fall Symposium Roles'04, AAAI Press (2005)
21. Steimann, F.: On the representation of roles in object-oriented and conceptual modelling. Data and Knowledge Engineering **35** (2000) 83–848
22. Boella, G., van der Torre, L.: Security policies for sharing knowledge in virtual communities. IEEE Transactions on Systems, Man and Cybernetics - Part A (2006)
23. Baldoni, M., Boella, G., van der Torre, L.: Bridging agent theory and object orientation: Importing social roles in object oriented languages. In: Procs. of PROMAS'05 workshop at AAMAS'05. (2005)
24. Baldoni, M., Boella, G., van der Torre, L.: Roles as a coordination construct: Introducing powerJava. In: Procs. of MTCoord'05 workshop at COORDINATION'05. (2005)
25. Albano, A., Bergamini, R., Ghelli, G., Orsini, R.: An object data model with roles. In: Procs. of VLDB'93. (1993) 39–51
26. Pacheco, O., Carmo, J.: A role based model of normative specification of organized collective agency and agents interaction. Autonomous Agents and Multiagent Systems **6** (2003) 145–184