Aligning Models of Normative Systems and Artificial Societies: Towards norm-governed behavior in virtual enterprises

Paul Davidsson and Andreas Jacobsson

Department of Systems and Software Engineering, School of Engineering,
Blekinge Institute of Technology, 372 25 Ronneby, Sweden
{paul.davidsson, andreas.jacobsson}@bth.se

Abstract. The purpose is to explore how norm-governed behavior within agent societies can be achieved in the context of Virtual Enterprises. We analyze a number of formal models from the agent research field, of which three models focus on the society aspects and three models focus on norms. A general observation is that the models reviewed are not concordant with each other and therefore require further alignment. A number of additions that may enrich the norm-focused models are suggested. It is also concluded that the introduction of different types of norms on different levels can be applied to ensure sound collaboration in agent-supported virtual enterprises. Moreover, the deployment of norm defender and promoter functionality is suggested to ensure norm compliance and punishments of norm violations.

1 Introduction

Artificial societies are typically characterized by agents that interact with each other in accordance with common rules or norms. Similarly to a human society, members of the artificial society must be allowed to coexist in a shared environment and to follow their respective goals in the presence of others. Here, the application of norms serves an important purpose in that they govern the rules of participation and provide important measures to achieve the desired behavior in a society.

We will here explore how norm-governed behavior within agent societies can be achieved in the context of virtual enterprises. In agent-supported virtual enterprises, the agents represent real interests and real entities, e.g., different agents have different owners, goals, interests, and preconditions for collaboration. A virtual enterprise, or more generally a Collaborative Network, may include a variety of entities (e.g., software, organizations and people) that are largely autonomous, geographically distributed, and heterogeneous in terms of their operating environment, culture, social capital, and goals. A virtual enterprise is typically described as "a temporary alliance of enterprises that join together to share skills or competencies and resources in order to better respond to business opportunities, and whose cooperation is supported by computer networks" (Camarinha-Matos et al., 2005). In this paper, a comparative study of formal models of agent societies and normative systems is undertaken in order to find out what types of norms and norm-enhancing mechanisms to include in agent-supported virtual enterprises.

In the next section, brief descriptions of the formal models reviewed are provided. This is followed by a discussion of how the various models are related to each other and of how the targeted models and the area in general can gain from model

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/908

alignment. Based on this exploration, we present some conclusions and suggestions for future work.

2. Formal Models

We analyze a number of formal models from the agent research field, of which three models focus on the society aspects and three models focus on norms. Each of the two groups includes two theoretical (and general) models and one specific application model.

2.1 Society Focus

2.1.1 Theory: Artificial Societies (AS)

Based on the work by Artikis and Pitt (2001) and Johansson (2002), Davidsson and Johansson (2006) suggest a formal characterization of agent societies that includes the following entities:

- a set of agents,
- a set of constraints on the society,
- a communication language,
- a set of roles that the agents can play,
- a set of states of affairs that hold at each time in the society,
- a set of owners (of the agents),
- a set of agent designers,
- the environment (computation and/or communication infrastructure)
- an environment owner, and
- an environment designer.

An agent is here defined as "a software entity that typically acts on the behalf of a person or an institution". Artikis and Pitt (2001) describe the set of constraints as "constraints on the agent communication, on the agent behavior that results from the social roles they occupy and on the agent behavior in general." The owner of the agent is the person or institution on whose behalf the agent acts. According to Johansson (2002) it has the power to launch the agent, provide it with preferences, as well as make run-time decisions regarding updating of preferences and when the agent should be terminated. Moreover, he defines the agent designer as the person(s) who has designed (and possibly implemented) the action selection and execution mechanisms of the agent. Davidsson and Johansson (2006) state that the environment owner is the person or organization that has the power to decide which agents may enter the society, which roles they are allowed to occupy, what communication language should be used, and the set of constraints on the society. Similarly, the environment designer is the person(s) who has designed and possibly implemented the conditions (mechanisms for controlling which agents may enter the society, what possible roles they may have, the space of constraints provided, etc.) under which the agents act in the environment.

2.1.2 Theory: Agent-supported Virtual Enterprises (AVE)

It has been argued that a promising approach to implement virtual enterprises is to use software agents that act as an "interface" and represent the organizations involved in a virtual enterprise. Jacobsson and Davidsson (2007) formally describe an agent-supported virtual enterprise a as a tuple:

$$ve_i = \langle A_i, R_i, AR_i, CI_i, Si, G_i, \lambda_i \rangle$$

where

- $A_i = \{a_1,..., a_n\}$ is the set of actors (typically enterprises) in ve_i. An actor can be described as a tuple:

  $$a_i = \langle I_i, T_i, C_i, G_i, \beta_i \rangle$$

  Where I_i are the relevant information systems needed in ve_i, T_i is the set of resources of the actor, C_i is the set of core competencies of the actor, βi is the agent acting on the behalf of the actor in ve_i, and G_i is the set of individual goals of the actor.
- $R_i = \{r_1,..., r_m\}$ is the set of roles that the actors can play in the ve_i. Each actor in the virtual enterprise can play one or more roles, e.g., innovator, supplier/provider of, e.g., goods, services, expertise, etc. The choice of role depends on the virtual enterprise goal(s), the actor's core competencies, resources and individual business goals.
- AR_i is a set of triples $\langle a_k, r_j, O_j^k \rangle$ where $a_k \in A$ and $r_j \in R$ i.e., the actors and their roles in the virtual enterprise, and the set of obligations, O_j^k, that is associated with the actor's role in the virtual enterprise.
- CI_i is a set of communication infrastructures needed for operating the virtual enterprise.
- S_i is a set of states of affairs that hold at each time in ve_i.
- λ_i is the agent communication language used by the agents β. We will assume that λ_i includes a set of relevant interaction protocols, a set of relevant ontologies, and possibly other things necessary to perform useful communication.
- G_i is a set of goals of the virtual enterprise that is derived from the business opportunities that motivate the initiation of the virtual enterprise.

2.1.3 Application: Plug and Play Business Communities (PPBC)

The concept of Plug and Play Business (Jacobsson and Davidsson, 2007) supports the formation and operation of agent-supported virtual enterprises. An important concept for implementing Plug and Play Business is *Internet communities* where persons (or organizations) that share some common interests (in this case, to find partners and do business) can meet virtually. Formally, an Plug and Play Business community, p_i, can be described as a tuple:

$$p_i = \langle A_i, R_i, VE_i, S_i, l_i, CI_i, gk_i \rangle$$

where

- $A_i = \{a_1,..., a_n\}$ is the set of actors (typically enterprises) in the community. An actor in the Plug and Play Business community can be described as a tuple:

$$a_i = \langle I_i, T_i, C_{i,} G_i, h_{i,} b_i \rangle$$

Compared to the definition of actors in agent-supported virtual enterprises, we add h_i, which is the person representing the actor/enterprise, and b_i, which is the Plug and Play Business client software (an intelligent agent supporting the (agent) communication language, l_i) acting on behalf of the actor/enterprise.

- $R_i = \{r_1, ..., r_m\}$ is the set of roles that the actors can play,
- $VE_i = \{ve_1, ..., ve_l\}$ is the set of virtual enterprises currently active in the community,
- S_i is a set of states of affairs that hold at any time in p_i,
- l_i is the agent communication language used by the agents *B*. We will assume that l_i includes a set of relevant interaction protocols, a set of relevant ontologies, and possibly other things necessary to perform useful communication,
- CI_i is a set of communication infrastructures needed for operating the community, and
- gk_i is the gate-keeper facility that regulates the entering (and leaving) of actors to (and from) the community. In order to become a member of p_i there is a set of criteria that must be fulfilled, e.g., VAT number must be declared, the roles it is willing to play should be stated, and information systems must be specified. Thus, some of the aims of the gate-keeper are to ensure that this type of information is available to the Plug and Play Business community and to verify the identity of the actors. Possibly, the gate-keeper may also be equipped with capabilities of handling different levels of memberships with different sets of norms in order to cope with the varying needs of potential participants and members. The gate-keeper could also inform the potential member about what general rules that hold in the community, and require the potential member to comply with them.

Plug and Play Business supports the three critical phases of virtual enterprises: In the *definition phase*, a member of the Plug and Play Business community, typically an innovator, may at any time initiate an attempt to form a collaborative coalition in-between the members. This process may be viewed analogous to crystallization, where a catalyst (innovator) initiates a process resulting in a precise form of collaboration, i.e., the formation of a virtual enterprise. In this phase, the catalyst, Ω, where $\Omega \in A$, describes the business opportunity in terms of goals, G and roles, R, of the virtual enterprise. Since this is a highly complex task, the human representative h_Ω will be the main contributor whereas b_Ω will just support the process. Thus, in this phase the degree of norm autonomy of b_Ω is rather low, whereas in other phases it may be higher (cf. adjustable autonomy).

The *creation phase* (where crystallization takes place) consists of three subtasks and is initiated by Ω:

- The function of *finding* requires that Ω has a list of the roles that must be filled in order to get an operating virtual enterprise. This list is provided by h_Ω, i.e., the person representing Ω in the definition phase. Then, for each of the roles, the task for b_Ω is to find the set of candidate actors K where $K \subset A$ that are able to play the role.

- In the *evaluation* task, Ω should rank the actors in K according to a set of requirements Q_r where $Q_r = \{q_1, q_2, ...q_k\}$ (provided by h_Ω). Based on this, Ω selects the actors with the highest rank k where $k \in K$ for negotiating on terms for virtual enterprise operation.
- The goal of *negotiation* is to establish an agreement between Ω and k concerning k's set of obligations, O_k. These obligations should of course be consistent with the set of goals, G of the ve and the set of goals of G_k.

When the creation phase is finished and a virtual enterprise is formed, the *operation phase* begins. Plug and Play Business supports two levels of collaboration: *administrational* and *operational*. They are defined by the type of *interaction protocols* they support. Administrational collaboration includes only protocols using the "weaker" *performatives*, such as, *ask*, *tell*, *reply*, etc. Operational collaboration supports protocols also using the performatives that actually manipulate the receiver's knowledge, such as, *insert*, where the sender requests the receiver to add the content of the message to its knowledge base, and *delete*, where the sender requests the receiver to delete the content of the message from its knowledge base.

2.2 Norm Focus

2.2.1 Theory: Normative MAS (NMAS)

López y López, Luck, and d'Inverno (2006) present a normative framework for agent-based systems. In their formal model, a *normative multi-agent system* consists of the following entities:

- a set of normative agents (*members*), where a normative agent consists of:
  - a set of goals,
  - a set of capabilities (actions that the agent can perform),
  - a set of motivations (preferences),
  - a set of beliefs,
  - ability to rank the goals according to preferences,
  - a set of adopted *norms*, some of which the agent has decided to comply with (*intended*) and some of which it has decided to reject (*rejected*).
- a set of general norms that govern the behavior of these agents (*generalnorms*).
- a set of norms issued to allow the creation and abolition of norms (*legislationnorms*)
- a set of norms dedicated to enforcing other norms (*enforcenorms*),
- a set of norms directed to encouraging compliance with norms through rewards (*rewardnorms*),
- the current state of the environment represented by the variable *environment*.

In addition, they identify a number of authorities:

- a set of *legislators* (agents that are entitled to create, modify, or abolish norms),
- a set of *defenders* (agents that are directly responsible for the application of punishments when norms are violated), and

- a set of *promoters* (agents whose responsibilities include rewarding compliant addressees).

The framework has been built upon the idea of autonomy of agents, i.e., it is intended to be used by agents that reason about why norms must be adopted, and why an adopted norm must be complied with.

Norms are formally defined to be composed of the following entities:

- a set of *normative goals*, which capture the purpose of the norm
- a set of *addressees*, which are the agents directly responsible for the satisfaction of the normative goals.
- a set of beneficiaries, which are the agents that benefit from the satisfaction of the normative goals,
- the *context*, which specifies the situations (environmental states) in which addressee agents must fulfill the norm,.
- *the exceptions*, which represent the situations in which addressees cannot be punished when they have not complied with the norm.
- *rewards* (expressed as a set of goals) to be given when normative goals become satisfied, or
- *punishments* to be applied when they are not.

2.2.2 Theory: Normative Systems (NS)

According to Boella and van der Torre (2004a), a normative multiagent system is composed of the following entities (we are here focusing on entities, not on how they are described, e.g., that a set of literals and rules are used to describe beliefs, desires and goals of the agents, and that there is a function, *MD*, which makes this mapping):

- a set of agents, *A*, where an agent could be either human or artificial. *A* is modeled in terms of:
  - a set of beliefs (*B*),
  - a set of desires (*D*),
  - a set of goals (*G*),
  - a set of decision variables (*X*), which represent an agent's actions,
  - a function agent description (*AD*), which maps each agent to the sets of beliefs, desires, intentions and decision variables, and
  - a priority relation ($\geq$), which expresses each agent's characteristics and how it resolves its conflicts, i.e., rank the importance of the agent's desires and goals .
- a normative agent, **n**, which is a member of *A*,
- a set of roles, *R*, that the agents can play,
- a norm description (*V*) function that represents the norms recognized by the agents, and
- a goal distribution (*GD*) function that corresponds to the goals of the agent that it is responsible for.

Moreover, they distinguish between *regulative* norms, described as obligations, prohibitions and permissions, and *constitutive* norms, such that regulate the creation of institutional facts as well as the modification of the normative system itself. In particular, regulative norms are formalized as goals, and constitutive norms are formalized as beliefs of the normative system. Regulative norms are based on the notion of conditional obligation with an associated sanction. Obligations are defined in terms of goals of the normative agent, prohibitions are obligations concerning

negated variables, and permissions are specified as exceptions to obligations. Constitutive norms introduce new classifications of existing facts and entities, called institutional facts, or they describe the legal consequences of actions on the normative system. Roles are used to specify the powers of agents to create institutional facts or to modify the norms of the system. Thereby, constitutive norms specify both the behavior and the evolution of a system in that they introduce or remove norms from the system.

2.2.3 Application: Virtual Communities of Agents (VCA)

Boella and van der Torre (2004b) investigate the use of design policies composed by prohibitions, permission and authorizations for virtual communities of agents on a computational grid. This work partly builds on Boella and van der Torre (2004A). They define a virtual community as a large, multi-institutional group of individuals who use a set of rules, i.e., a policy, to specify how to share their resources. In a virtual community, agents can play both the role of resource consumers and the role of resource providers. Resource providers retain the control of their resources and they specify in local policies the conditions for use of their resources thereby giving rise to a third role, authorization, in their model.

In virtual communities, a single set of agents (A), where each can play one or more roles, is defined so that each agent of the agent set can play three roles:

1. *Resource consumer*, denoted as $c(a_i)$, is an agent who manipulates a resource by means of some action. It can access resources to achieve its goals, is subject to norms regulating security, prohibitions and permissions, and also endowed with authorizations to access resources.
2. *Resource provider*, denoted as $p(a_i)$, can provide access to the resources it owns. This is referred to as the normative role, since it can issue norms, i.e., prohibitions and permissions about the access of a resource, and enforce their respect by means of sanctions, and delegate the power to authorize resource consumers.
3. *Authority*, denoted as $u(a_i)$, can declare resource consumers authorized when they are requested to do so. They know that their declarations are considered authorizations by the resource providers since they have been delegated the power to authorize resource consumers on behalf of resource providers.

Prohibitions and *permissions* are specified in terms of goals and desires of the bearer of the norm and of the normative role. A prohibition is defined as a goal of resource providers whereas a permission is the behavior which is not considered by a provider as a violation and thus is not sanctioned. A third concept, *authorization*, is a belief of a provider which appears as a condition in some permission it issued. Thereby, an authorization has a meaning only if it appears among the conditions of a permission. These concepts are then supplemented with two concepts, namely *violation* and *sanction*. The agent holding the normative role (i.e., the resource provider) can decide if some action is to be regarded as a violation. The possibility to punish violations by means of some sanction is among the preconditions for creating a prohibition. Sanctions provide motivation to fulfill the norms, since it is not possible to assume that all agents are cooperative and that they respect the norms. Thereby, a sanction is an action negatively affecting an agent, i.e., the agent desires the absence of the sanction.

3 Comparison and Model Alignment

We will now compare and try to align the different formal models described above. A more concise version of the comparison can be found in the table in Appendix A.

In this analysis, the object of study in the first set of models is the society, where norms play a small yet important part; whereas the second set of models reviewed take their starting points from normative perspectives, where the other aspects (e.g., agent ownership, agent roles, system state, etc.) play secondary roles. In the normative frameworks, different types of norms for different types of contexts are defined. By contrast, the AS model just considers one type of norms ("constraints") on only one level. Moreover, the AVE model specifies norms ("obligations") only between actors whereas the PPBC model does not include any norms at all. We argue that the use of other types of norms and on more levels than one can enrich the formal models of agent societies so that they are able to capture the types of norm-governed behavior that are necessary in many complex applications. For instance, we intend to introduce norms that regulate the interaction between the agents in the AVE as well as in the PPBC model. These can be both in terms of specific obligations or permissions between individual agents and in terms of general norms for all the agents.

Although a set of agents is defined in all the models reviewed, there are some differences in the views on what constitute an agent. Three different perspectives can be identified: (i) a norm-autonomous artificial or human entity as in the NSA and VCA models, (ii) a norm autonomous software entity as found in the NMAS model, and (iii) just a software entity. These different views obviously affect the treatment of agents and norms in artificial societies. In most current applications, agents are not norm autonomous, which may either be due to the complexity of implementing norm autonomous agents, or due to that, in some applications, it is desired that the agent owner is involved in decisions regarding what norms to follow, etc. In the PPBC model, a human representative is assisting the agent in decision situations. Moreover, it may be possible to use the theories that assume norm-autonomous entities also in current applications, but this requires that both the agent and its owner are included in the norm autonomous entity.

The agent owners are not regarded in NMAS, NS and VCA, which also may be due to that completely norm-autonomous agents are assumed in those models. Since the agent owner has the power to, apart from deciding what norms to follow, release and terminate an agent and to provide it with goals even during runtime, the inclusion of agent owners would seem to improve those formal models.

There are some suggestions on norm-enforcements amongst the reviewed frameworks. For instance, NMAS uses defender agents that are responsible for the application of punishments when norms are violated, and promoter agents that monitor norm compliance. The latter is corresponding to the external observer agent used in Kamara et al. (2005) in order to detect whether interacting agents operate in compliance with the norms or not. Moreover, they also use an admission protocol, which allows nodes to create, enter and exit the agent society. Similarly, a gate-keeper is used in PPBC to regulate the entering to and leaving from the community. A possible improvement of the PPBC would be to include also defender and promoter functionality as in the NMAS model to monitor the behavior of member actors in order to ensure norm compliance, and to impose punishments to those that violate the norms.

We can observe that an agent communication language is included only in the first set of models (AS, AVE and PPBC); however it may be implicitly assumed in the

other models. Obviously, the lack of a language would severely limit the application of norms, but even in the case where a language is used, the language may put restrictions on what norms can be expressed and communicated. However, further investigations are needed before any conclusions can be drawn.

Goals are included in all formal models except the AS model (agent preferences are discussed in the paper but not explicitly specified in the formal model). However, the goals in AVE and PPBC are not associated to agents but to actors and virtual enterprises. Since goals are closely related to norms, the introduction of norms on the agent level should be accompanied by the inclusion of goals in these models.

It can be observed that the physical environment is included in all models apart from the theoretical norm frameworks (NMAS and NS). The environment owner is only recognized in the AS model. This is interesting since the environment owner can have a large impact on what norms that hold for the society. Typically, the environment owner has control over gate-keepers, defenders, promoters, legislator, etc. Therefore, it is likely that the formal models can be enriched by including the environment owner.

6 Conclusions and Future Work

We have undertaken a comparative analysis of six formal models, describing artificial societies or normative systems. A general observation is that the models reviewed are not concordant with each other. For instance, completely norm-autonomous agents are assumed in the norm-focused frameworks, but in the society-focused models the notion of an agent owner is specified. In the norm-focused models, entities like agent communication language, the physical environment are often not regarded. Moreover, norm-enhancing mechanisms are only included in two models (PPBC and NMAS). Based on these findings, we have discussed how model alignment can foster both areas (modeling of agent societies and normative systems) in general and how the inclusion of missing entities may improve the various formal models in particular.

With respect to the AVE and PPBC models, we can conclude that these models can be enriched by, for instance, the introduction of norms (both on general and specific levels) that regulate the interaction between agents, and that these types of norms should be accompanied by the specification of goals. On norm enforcement, some opportunities for improvements in the PPBC model are to include defender and promoter functionality in order to ensure norm compliance and punishments of norm violations.

6.1 Future Work

Based on our analysis, there are a number of issues that need further study, e.g.:

- Investigate what types of norms (obligations, permissions, prohibitions, etc.) and on what levels (general or specific) to include in the formal models of both AVE and PPBC.
- Explore the possibility to include the owner in norm-autonomous entities in the NMAS, NS and VCA models, so that these models can be applied also to societies populated by agents that are not norm-autonomous.
- A possible improvement of the PPBC would be to include also defender and promoter functionality as in the NMAS model.

References

Artikis, A, and Pitt, J (2001) A Formal Model of Open Agent Societies. In: Proceedings of the 5th International Conference on Autonomous Agents.

Boella, G, and van der Torre, L (2004a) Regulative and Constitutive Norms in Normative Multiagent Systems. In: Proceedings of 9th International Conference on the Principles of Knowledge Representation and Reasoning (American Association for Artificial Intelligence).

Boella, G, and van der Torre, L (2004b) Virtual Permission and authorization in policies for virtual communities of agents. In: Proceedings. of Agents and P2P Computing Workshop at AAMAS'04.

Camarinha-Matos, LM, and Afsarmanesh, H (2005) Collaborative Networks: A New Scientific Discipline. In: Journal of Intelligent Manufacturing, 16: 439-452.

Davidsson, P (2001) Categories of Artificial Societies. Engineering Societies in the Agents World II, Lecture Notes in Computer Science, Vol. 2203, Springer Verlag, Berlin Germany.

Davidsson, P, and Johansson, S (2006) On the Potential of Norm-Governed Behavior in Different Categories of Artificial Societies. In: Computational and Mathematical Organization Theory, 12:169-180.

Jacobsson, A and Davidsson, P (2007) A Formal Analysis of Virtual Enterprise Creation and Operation. To appear in eds. Król, D, and Nguyen, NT: Intelligence Integration in Distributed Knowledge Management by Idea Group Publishing.

Johansson, SJ (2002) On Coordination in Multi-Agent Systems. Ph.D. Dissertation Series No. 05/02., Department of Software Engineering and Computer Science, Blekinge Institute of Technology, Sweden.

Kamara, L, Pitt, J, and Sergot, M (2005) Towards Norm-Governed Self-Organizing Networks. In: 1st International Symposium on Normative Multiagent Systems.

López y López, F, Luck, M, and d'Inverno, M (2006) A Normative Framework for Agent-Based Systems. In: Computational and Mathematical Organization Theory, 12:227-250.

Appendix A

	Theory: AS	**Theory: AVE**	**Application: PPBC**	**Theory: NMAS**	**Theory: NS**	**Application: VCA**
Agents	A set of agents	A set of agents (*B*)	A set of agents (*B*)	A set of normative agents (*members*)	A set of agents (A)	A set of *n* agents (*A*)
Norms	A set of constraints	A set of obligations for each role that the actors can play (O^{role}) thus this is not on the agent level	-----	Four sets of norms: *generalnorms*, *legislationnorms*, *enforcementnorms*, and *rewardnorms* (which in turn may be obligations, prohibitions, social commitments or social codes)	Four sets of norms: obligations, permissions, prohibitions, and institutional facts	Three sets of norms: prohibitions, permissions and authorizations
Communication language	A communication language	λ_i is the agent communication language used by the agents *B*	l_i is the agent communication language used by the agents *B*	-----	-----	-----
Agent roles	A set of roles that the agents can play	The role of participant is implicitly assumed	The roles of initiator and potential participant are implicitly assumed	With respect to a norm, an agent can play either the *addressee* or the *beneficiary*	A set of roles, *R*, that the agents can play	An agent can play three roles *resource consumer*, *resource provider* or *authority*
System state	A set of states of affairs that hold at each time at the society	S_i is a set of states of affairs that hold at each time in the virtual enterprise	S_i is a set of states of affairs that hold at each time in the Plug and Play Business community	The current state of the environment is represented by the variable *environment*	*Parameters* (*P*) describe both the state of the world and *institutional facts*	-----
Agent owners	A set of owners (of the agents)	A set of actors (*A*)	A set of actors (A), which have human representatives (*H*)	-----	-----	-----
Agent owner roles	-----	A set of roles (*R*)	A set of roles (*R*)	-----	-----	-----
The Physical Environment / Infrastructure	The environment (computation and/or communication infrastructure)	A set of communication infrastructures needed for formation and collaboration (*CI*)	A set of communication infrastructures needed for formation and collaboration (*CI*)	-----	-----	A grid infrastructure is used for the virtual communities
Norm-enhancing mechanisms	-----	-----	Gate-keeper agent (regulates the entering and leaving to and from the community)	*Defenders* (agents that are responsible for the application of punishments when norms are violated) *Promoters* (agents that monitor compliance with norms) *Legislators* (agents that define norms)	-----	-----
Environment owners	An environment owner	-----	-----	-----	-----	-----
Goals	-----	Each VE has a set of goals (G_{VE}) and each actor has a set of goals (G_{actor})	Each VE has a set of goals (G_{VE}) and each actor has a set of goals (G_{actor})	Each agent has a set of goals (*goals*)	Each agent has a set of beliefs (B_a), desires (D_a), and goals (G_a)	Each agent has a set of beliefs (B_a), desires (D_a), and goals (G_a)

	Theory: AS	**Theory: AVE**	**Application: PPBC**	**Theory: NMAS**	**Theory: NS**	**Application: VCA**
Definition of agent	A software entity that typically acts on the behalf of a person or an institution	-----	-----	Normative agent: - a set of goals - a set of capabilities - a set of motivations (preferences) - a set of beliefs - ability to rank the goals according to preferences - a set of norms - a set of intended norms - a set of rejected norms	Human or artificial. An agent is defined as: -Beliefs (B) -Desires (D) -Goals (G) -Decision variables (X) -Agent description variables (AD) -A priority relation ($\geq$)	Agent design is inspired by the BOID architecture
Definition of norm	Constraints on the agent communication language, on the agent behavior that results from the social roles they occupy, and on the agent behavior in general	-----	-----	Components are: -Normative goals -Addressee agents -Beneficiary agent -Context -Exception -Have not (complied with norms) -Immunity -Rewards -Punishments	Norms are either *regulative* (defined as obligations, prohibitions, and permissions) or *constitutive* (defined as institutional facts)	Prohibitions and permissions are defined in terms of goals and desires of the bearer of the norm and of the normative role, together with the concepts of violation and sanction.