Collaboration behavior enhancement in co-development networks

Shadi, M.

Link to publication

Creative Commons License (see https://creativecommons.org/use-remix/cc-licenses):
Other

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
Chapter 3

Norm Specification in Virtual Organizations

The list of publications based on which this chapter is written follows:


3.1 Introduction

An agile Virtual Organization (VO) can be effectively launched within a Virtual organizations Breeding Environment (VBE), to respond to emerging opportunities, either for innovation or to satisfy new market needs. The VO partners while obliged to collaborate to achieve its common goals, are autonomous, heterogeneous, and geographically distributed organizations, that behave differently in different collaborative situations. Research publications address that in most cases the failure of the VOs is strongly related to and/or caused by the behavioral factors of the involved VO partners. Moreover, the behaviors of VO partners affect the level of both their eagerness to collaborate as well as establishing trust relationship among the partners. Defining a framework aimed at monitoring, controlling, and enhancing partners’ behavior in VOs is therefore relevant and necessary. Among others, such a system can aim at identification and prediction of conflicts, and selection of the best-fit partners to take over risky tasks and better assignment of roles and responsibilities for handling the conflicts.
For this purpose, we propose a new norm-based framework to monitor the partners’ behavior in VOs. Within this framework, we introduce the S3C model, which characterizes four kinds of norms for organizations involved in a VO, including: **Socio-regulatory norms**, **Co-working norms**, **Committing norms**, and **Controlling norms**.

The **socio-regulatory norms** have the VO consortium agreement at their base, which specifies some regulations and limitations on partners’ behavior, e.g. partners are obliged not to share other partners’ information.

The **co-working norms** are specified by general terms of operational collaboration among the partners involved in the VO. These norms are typically negotiated among partners via the contracts they sign during the VO formation stage, indicating that the partners together fulfill the VO’s objectives. Considering the dynamic and adaptive nature of the VO, and even some potential changes in VO sub-goals, most contract terms do not and/or cannot specify details of day-to-day (fine-grained) activities of the VO partners, and typically indicate only coarse-grained activities, which should be performed by a group of VO partners. In other words, VO contract typically consists of several joint commitments (our so-called joint-promises) on coarse-grained activities assigned to groups of partners to perform them together.

The **committing norms** constitute the requirements that should be fulfilled by each partner to perform a fine-grained sub-task at the VO. Our framework supports the VO coordinator and its partners to agree on these fine-grained sub-tasks, based on the specifications provided in the VO contracts. Each of the commitments therefore represent the clear partial responsibility of a partner toward fulfilling a part of one coarse-grained task in the VO contract. To define sub-tasks and their associated commitments (our so-called promises), first a template is extracted from the VO contract, in which the inter-dependencies between goals and tasks as well as responsible partners are specified. This template is called the Goal-Task-Interdependency-Template (GTIT), which provides the base for definition and assigning of day-to-day sub-tasks activities to each individual partner, throughout the VO operation phase. However, before any sub-task activity assignment is made, it needs to be planned by the VO coordinator and agreed among the VO partners. During the VO operation phase, based on the GTIT and joint-commitments (our so-called joint-promises) made at the VO creation phase, partners make new agreements (translated and formalized into promises) to perform their sub-tasks. Section 3.4.1 provides our formal definition of promises and joint-promises.

The **controlling norms** constitute the requirements imposed by the VO coordinator, to control the partners’ behavior and coordinate the tasks of each partner. In our approach, the requirements for partners’ behavior to be trusted by the VO coordinator are called **trust-related norms**, the requirements on not having work overload are called **workload-related norms**, and the requirements on the required communication level of partners are called **communication-**
3.2. Related Work

This section surveys some existing works that we use to answer the following questions: (1) how the contract terms are defined as norms in our multi-agent system, (2) how the partners’ behavior can be monitored against the defined norms, and (3) how norm-related concepts are formalized.

In [36], the authors propose a two-layer agent society model, including the institutional layer and the operational layer. The institutional layer refers to the place in which the social norms of the society are defined, while in the operational layer, the overall objectives of the society and its intended actions are specified through some norms. The focus of their reported work is on three different coordination models which specify different models of societies, including their so-called: market, network, and hierarchical models. In [85], an agent-based framework is proposed in which a contract is specified by a set of norms and sanctions for the involved agents. They use logical constructs, which applies deontic operators, to automate contractual relationships. In their research, agents’ intentions are shared in order to explain what they expect from the other agents in order to fulfill their contractual norms. In [26], a hierarchical organization of norms is introduced for Virtual Enterprises, including institutional, constitutional, and operational levels of norms. Institutional norms are typically pre-existent, while constitutional and operational norms are defined when a cooperative agreement is achieved. Institutional norms, placed at the top of the hierarchy, provide a background for establishing the agents’ commitments. The second place is dedicated
Chapter 3. Norm Specification in Virtual Organizations

to constitutional norms which are created when a new VO (or cooperation agreement) is formed. Finally, operational norms are specified through operational contracts signed between a subset of the VO partners. In [47], the authors formalize a contract with some models, including (1) concept model to define some concepts existing in contracts, (2) process model to specify the internal structure of work flow process, (3) enactment model to provide monitoring information through services, such as execution monitoring service, and (4) usage model to define where the contract is used.

In our approach, four kinds of norms are defined including the soci-regulatory norms, co-working norms, committing norms, and controlling norms. The first category consists of generic obligations and prohibitions described in a VBE and specific ones described for each VO. The fact of keeping joint-promises made by the VO partners to perform coarse-grained activities jointly corresponds to the co-working norms. Definition and formalization of joint-promises are to the best of our knowledge novel in all related VO research. Because of VO’s dynamism, the details of day-to-day activities are not and cannot be typically determined at the VO creation phase, therefore during the VO operation phase some agreements (translated into promises) are made among the partners to perform the fine-grained activities (sub-tasks). The fact of keeping a promise is considered as a committing norm. The requirements on trust, workload, and communication are together considered as controlling norms. Our approach differs from the above approaches due to addressing specifications of the VO’s environments, such as the VO’s dynamism and joint responsibilities, which are supported through our formalization of co-working norms and committing norms.

3.2.1 Promise Formalization

In multi-agent systems area, there are some research focused on formal specification and verification of norms, e.g. [85] and [35]. In [85], operators of deontic logic are used to represent and model the contracts consisting of a set of permissions and obligations. Similarly, [35] uses deontic logic as the base for formalizing different kinds of obligations, and defining the semantics of deadline obligations. It should be noticed that deontic logic is a kind of symbolic logic, in which notions, such as permissible, must, claim, and impermissible are taken into account.

Modal logic has been also applied to represent promises [102]. Zhao and Lin have expressed a model to reason about Belief, Capability and Promise of agents named BCP. They use modal logic, which is intuitively acceptable for modeling the multi-agent interactions, and add some modal operators such as $P_{ij}$ to the logic. $P_{ij}\alpha$ means that agent $j$ makes a promise to agent $i$ to achieve the state $\alpha$ [102]. However, the existing representations in modal logic and deontic logic unnecessarily complicate reasoning related to the life cycle of promises, which is needed in our proposed framework without adding advantages. In fact, in our proposed approach, we introduce many more states for promises, while only
3.2. Related Work

the violation and obedience are considered in other works mentioned here. In [14], different formalism for promises is discussed. It states that usual modal logic cannot show that a promise is shared between a sender and a recipient. Moreover, a graphical scheme as a rule-based construction is also proposed to analyze promises. Our approach adopts some definitions related to promise and its states from [14].

### 3.2.2 Joint-promises Formalization

There are a few literature addressing joint-promises. In [39], some dynamic aspects of social and collective attitudes of agents involved in Cooperative Problem Solving (CPS) are described. After defining social commitment between two agents, the collective level of cooperation is considered in this work. The concrete manner of achieving the intended goal by a team is reflected by a collective commitment. Their representation of collective commitment is plan-based. Such a plan includes negotiating and determining who will do what, i.e. every part of the plan is within somebody’s responsibility. In [38] also, collective commitment is formally modeled similar to [39]. In this work, authors consider different versions of collective commitments reflecting different aspects of cooperative problem solving. These aspects are related to the kind of awareness that are present within a team as a higher level of complexity. In [27], different kinds of commitment including internal, social and collective commitments are considered and defined. An internal commitment (I-commitment) refers to a relation between an agent and its action. They consider a social commitment (S-commitment) as a relation between an individual and a group, i.e. it shows a relation between at least two agents. Finally, a collective commitment is defined as the I-commitment of a collection of agents. This implies:

- S-commitment of each member to the group.
- S-commitment of each member to another member.
- I-commitment of each member to perform its action.

In [24] and [74], it is mentioned that the establishment of a joint-commitment obliges each party, against all other parties, to do its part and these obligations are created simultaneously, when a joint-commitment is appeared. A joint commitment is fulfilled, if all mentioned obligations are fulfilled, so by realizing the violation of one of these obligations, the joint-commitment is violated. In [14], an assumption is proposed for a joint-promise called cooperative promise rule. The assumption is ”for two agents to guarantee the same promise, each agent requires a special type of a promise: the promise to cooperate with the other agent, about the basic promise themes”. In [39], [38], and [27], the authors consider the internal states and operations of individual agents. However, the
difference between our approach in relation to joint-promises and all of the above work is that we do not know how the agents work internally as we only monitor the external actions of the agents. Moreover, the focus of other works is on the construction/representation of a joint-commitment rather than its evaluation. In other words, dissimilarity between these works and our work is related to how the joint-commitment is addressed and handled. The other works carefully address the planning time, while we focus on the monitoring at the execution time. However, there are similarities between our definition of joint-promises and the definitions in the previous works (except for [14]). This is related to the fact in a joint-promise there is at least one part of the joint-action considered for each involved agent to do.

3.3 Virtual Organizations as Normative Multi-agent Systems

In a multi-agent system, the behavior of agents is constrained and regulated by a set of norms to achieve the system’ objectives. An enforcement mechanism can be implemented for handling the norms, to detect when a norm is active and to detect violations of the active norms. The properties of VO partners, such as autonomy and heterogeneity, match with those in multi-agent systems; therefore, multi-agent systems provide the required environment to monitor and control the behavior of VO partners against their defined norms.

Four specific kinds of norms are already identified and addressed in Section 3.1, including socio-regulatory norms, co-work norms, committing norms, and controlling norms.

The socio-regulatory norms in the model are defined as the set of common social rules for all partners, and set each partners’ expectations of others, through applying the coercive power of a joint VO consortium agreement. They consist of regulations on authorizations, prohibitions, and obligations, defined for partners, e.g. partners are obliged not to share other partners’ information with a third party. Although these norms are very important in VOs, they are not directly addressing day-to-day operational goals of the VOs, rather supporting the collaboration atmosphere and the infrastructure that is required for fulfilling its operational goals.

The co-working norms set the specific requirements defined at the VO level, to be obeyed by all partners involved in joint-tasks with others, in order for partners involved in each joint-task to jointly keep the responsibility of performing that task together.

The committing norms on the other hand are the requirements defined for performing one sub-task by each individual partners.

Our approach to co-working and committing norms introduces new formalizations and mechanisms for organizations to make promises related to performing
their individual sub-tasks and joint-promises related to performing joint-tasks. Thus the VO partners committing/volunteering themselves, in a bottom-up manner, to perform their tasks, as opposed to the VO coordinator assigning tasks to them in a top-down manner. The bottom-up manner is more in line with the collaboration nature in VOs that resembles federated partnerships among organizations.

We introduce GTIT representing Goal-Task-Interdependency-Template that is extracted from the VO contract. GTIT represents the joint-responsibilities of agents in the VO, in which besides hierarchical inter-dependencies (full dependency), associative inter-dependencies (partial dependency) among goals, sub-goals, and tasks assigned to agents are also specified.

When the operation phase of the VO starts, considering the established and recorded joint-promises, and the GTIT extracted from the VO contracts, specific day-to-day activities (sub-tasks) needed to be performed by each contractual partner are defined, as Figure 3.1 illustrates. In other words, before the VO operation phase, based on the information related to the main planned activities in the VO contract, the VO’s GTIT is generated. However, GTIT gets continuously instantiated and enhanced during the operation phase of the VO, to reflect specific day-to-day activities of the VO partners. The instantiation of GTIT is called Partner-Responsibility-Interdependency-Tree (PRIT). During the VO operation phase, also in case of any evolution, PRIT is dynamically updated to represent all new inter-dependencies.

In order to illustrate the role of GTIT and PRIT, assume a VO established for a new R&D project. In the project’s contract with the funding agency, the main objectives of the project are usually divided into several sub-objectives, to be fulfilled within a number of Work Packages (WPs), where a set of VO partners are jointly responsible for each WP. In turn, each WP consists of several tasks, where each task represents a subset of WP activities, planned to be jointly performed by a group of its partners. Furthermore, every task contains several sub-tasks. Figure 3.2 shows some examples from an R&D contract. In this contract, the descriptions of the WPs including their specified deadlines, their involved partners, WP leaders, as well as the specification of each WP’s tasks and task leaders, etc. are provided. However, due to the nature of federated cooperation from involved partners and being research related, many details of sub-tasks and their individual responsible party cannot be formally specified in the contract before the VO operation phase gets started. Also, the task definitions are typically not well-defined in advance and need to be later defined and adapted. Moreover, in VOs, sub-tasks typically evolve during the operation phase and thus the inter-dependencies among the sub-tasks would change, even if they were concisely defined prior to the VO operation phase. As shown in Figure 3.2 contract terms are translated into joint-promises which are later decomposed into partners promises during the VO operation phase, through negotiation between the partners and the task leaders.
Furthermore, the controlling norms are defined as requirements for coordinating the tasks of each partner, and they are imposed/enforced by the VO coordinator during the VO operation phase. The measurements of trustworthiness, work overload, and communication level of partners are carefully applied in these norms, as later addressed in Chapter 5 and 6. For each of these three measurements, the VO coordinator defines a threshold, and VOSAT identifies whether a partner violates one of these norms. Due to the fuzzy nature of trust, the requirements on trust level of partners are specified as fuzzy norms, our so called trust-related norms. The VO coordinator defines the minimum required/tolerated trust level for partners in the VO as the threshold. For example, if the trust level of a partner, measured through applying fuzzy comprehensive evaluation method based on AHP (Analytic Hierarchy Process), is lower than the minimum level tolerated in the VO, that partner’s trust-related norm is violated, indicating the lack of trust in that partner. Consequently, if any controlling norm, i.e. trust-related norm, communication-related norm, and workload-related norm is violated for a VO partner, then our introduced risk prediction mechanism issues a warning to the VO coordinator. Therefore, using the controlling norms
3.3. Virtual Organizations as Normative Multi-agent Systems

Figure 3.2: An example of VO contract and GTIT for R&D projects.
together with monitoring the responsibility inter-dependencies, it is possible to measure the probability of failure in each of the planned sub-task, task, sub-goal, and thus the general goal of the VO. This is one main target of the research in this thesis.

3.4 Conceptual Model of Normative VOs

A conceptual model is defined for a VO as a normative multi-agent system, as shown in Figure 3.3, according to which we have developed a tool to monitor the agent’s behaviors and to evaluate them with respect to the specified norms. For this purpose, some of the main VO related concepts are defined below:

**VO** - This is a temporary goal-oriented network of agents, formed when a business opportunity is emerged, and dissolved when its goals are achieved.

**Agent** - Individuals, organizations, and intelligent machines involved in a VO that are autonomous and perform different actions.

**Goal** - It captures some desired states of the world, which is divided into several sub-goals at the VO creation phase. A group of agents make a joint-promise to
fulfill each sub-goal.

**Contract** - It addresses achievement of the main tasks (course-grained activities), for which involved partners jointly are committed. Considering the dynamic and adaptive nature of VOs, contracts do not represent details of day-to-day activities of the VO agents.

**Consortium agreement** - It contains a set of common social behavior requirements for all partners, and sets the main partners’ expectations of others in a cooperative environment.

**GTIT** - It is extracted from the contract, in which inter-dependencies among goals, sub-goals, and tasks assigned to agents are specified. When the operation phase starts, considering the established/recorded joint-promises and the GTIT extracted from the VO contracts, the specific day-to-day activities (sub-tasks) are defined. GTIT is instantiated into PRIT (Partner-Responsibility-Interdependency-Tree) at the beginning of the operation phase, and from that point on, the PRIT both continuously extends with more detailed tasks to be implemented by the VO partners, and is kept up-to-date reflecting any needed changes and/or any backtracking of the failed tasks. The latest PRIT is the only one valid at each time.

**Norm** - It is defined either as an obligation or a prohibition, which limits the agents’ behavior in multi-agent systems. In this framework, S3C model characterizes Socio-regulatory norms, Co-working norms, Committing norms, and Controlling norms. Socio-regulatory and controlling norms have only two states of violation and obedience while the other two categories, the co-working norms and committing norms, are directly related to promises made, where promises have six terminated states (see Table 3.1).

**Joint-Promise** - It is the formalized specification of a group of partners jointly promising to perform a task, which is defined for achieving a sub-goal.

**Promise** - It is the formalized specification of each party’s responsibility in a joint-promise made toward fulfilling a coarse-grained activity (a joint-task). This is the result of the involved partners negotiation, where each partner agrees to perform a particular part of the coarse-grained activity, and thus a promise will be made by that partner.

**Action** - Agents are autonomous and can expose different actions. The interactive actions that change the state of agents’ norms are discussed in Chapter 4.

**External Event** - This can affect the state of a norm. For example, the passing of time is a kind of external event.

### 3.4.1 Norm-related Concepts

Norms are considered as behavioral standards to steer agents to achieve their goals. The main norm-related concepts in S3C model, including promise, joint-promise, obligation and prohibition are concisely defined as follows:
Definition 1: A promise is defined as a tuple \( < x, y, p, d, q, d' > \), where \( x \) is the promiser agent, \( y \) is the promisee agent, \( p \) is the proposition which a promiser should bring about before the deadline \( d \), \( q \) is the pre-condition for realizing \( p \) by \( x \), and \( d' \) is the deadline before which \( q \) should be realized.

A promise is read as follows: \( x \) promises to \( y \) to bring about \( p \) before deadline \( d \) if \( q \) is brought about before \( d' \). It is not important who brings about \( q \) before \( d' \). In the next and other definitions, we assume \( x \) and \( y \) to be agent identifiers, \( p \), \( q \), \( d \), and \( d' \) to be state propositions. For simplicity, we assume propositions to be atomic, because our aim is to focus on the concepts involved in VO rather than the expressiveness or the logic of VO. Our definition of a promise is an extension of the definition of promise in [14], based on which different states can be considered for a promise, as shown in Table 3.1. Some labels are used, for example C stands for Conditional state. Some of these states are rooted in [14].

<table>
<thead>
<tr>
<th>Promise States</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conditional (C)</td>
<td>Some pre-conditions are needed to be fulfilled.</td>
</tr>
<tr>
<td>Unconditional (UC)</td>
<td>All pre-conditions are fulfilled before their deadlines.</td>
</tr>
<tr>
<td>Kept (K)</td>
<td>The proposition whose fulfillment is promised, is realized.</td>
</tr>
<tr>
<td>Withdrawn (W)</td>
<td>The promise is canceled by the promiser.</td>
</tr>
<tr>
<td>Invalidated (In)</td>
<td>The promise has not been fulfilled due to some reasons beyond the promiser’s control.</td>
</tr>
<tr>
<td>Released (R)</td>
<td>Promise is canceled by the promisee.</td>
</tr>
<tr>
<td>Dissolved (Dis)</td>
<td>When the pre-conditions are not fulfilled before their specified deadlines.</td>
</tr>
<tr>
<td>Not Kept (NK)</td>
<td>The proposition whose fulfillment is promised, is not realized before the deadline.</td>
</tr>
</tbody>
</table>

Table 3.1: Different states of a promise. All states, except Conditional and Unconditional are considered as terminated states.

One of the advantages of considering different states for a promise is to generate different responses in different states. For example, a promiser whose promise is in Invalidated state may receive less sanction than the promiser whose promise is in Not Kept state, because the first case indicates that a failure has happened beyond the control of the promiser. The other advantage is to support the VO’s dynamism, for example, whenever the VO sub-goal gets modified, then we need to only modify the states of specific promises relevant to that sub-goal.

Definition 2: A joint-promise is defined as a promise in which the promiser is replaced by a group of agents. In other words, we define a joint-promise by a tuple \( < G, y, p, d, q, d' > \), where \( G \) is a group of agents, \( y \) is the promisee agent, \( p \) is the proposition which \( G \) should bring about before the deadline \( d \), \( q \) is the pre-condition for realizing \( p \) by \( G \), and \( d' \) is the deadline before which \( q \) should hold.

Joint-responsibility addresses one of the most important features in VOs. Partners are engaged in a cooperative activity, and make an agreement to collec-
tively perform certain task, toward achieving certain sub-goals of the VO. In our framework, this agreement is represented and formalized by a joint-promise. The establishment of a joint-promise obliges simultaneously each party to do its own part as well as guaranteeing the fulfillment of the part for which other partners are responsible. This can sometimes cause one partner having to even perform the task of another partner. One reason for a partner to decide to fulfill the obligations of other parties can be its reputation and financial concerns for success of the joint task. A joint-promise is fulfilled when all mentioned obligations are fulfilled. As such, the violation of one of these obligations causes the violation of the entire joint-promise (see [24]).

**Definition 3:** An obligation is defined as a tuple $<x, p, d>$, where $x$ is the agent for which an obligation is defined, $p$ is the proposition which is obliged to be brought about by $x$ before deadline $d$.

**Definition 4:** A prohibition is defined as a tuple $<x, p, d>$, where $x$ is the agent for which a prohibition is defined, $p$ is the proposition which is prohibited for $x$ during the deadline $d$.

It should be noticed that socio-regulatory norms and controlling norms include some obligations or prohibitions for agents. In our framework, the proposition $p$ can be either crisp or fuzzy. For example, we define the agent’s trust-related norm as one form of fuzzy norm, which is explained in Chapter 5. It should be noticed that while a set of common prohibitions and obligations can be defined in VOs to be obeyed by all agents, our approach also supports the possibility of their customization for every distinct agent, in order to further generalize the mechanism.

### 3.5 Conclusion

This Chapter addresses the work behavior of partners, considering both their individual and joint tasks assignments. Moreover, a framework for virtual organizations is addressed to assist with controlling the behaviors of agents involved in VOs, through monitoring norms and imposing appropriate plans when agents fail to be compliant with norms and norms are violated. The S3C model is proposed, which characterizes four kinds of norms, i.e. socio-regulatory norms, co-working norms, committing norms and controlling norms. Our approach to this model introduces new definitions and formalizations for norm related concepts, such as promise, joint-promise, obligation, and prohibition. Based on this framework, the VO supervisory assisting tool is designed and developed.