Norms as a Basis for Governing Sociotechnical Systems

Munindar P. Singh and Kartik Tadanki
North Carolina State University

June 16, 2010

Abstract

We define *governance* as the administration of a multistakeholder system by the stakeholders themselves. In this regard, governance is a peer-to-peer notion and contrasts with traditional management, which is a top-down hierarchical notion. Traditionally, there is no computational support for governance and it is achieved through out-of-band interactions among system administrators. Not surprisingly, such approaches simply do not scale up to pervasive computing systems that consist of tens of thousands of computational resources.

We develop an approach for governance based on a computational representation of norms in organizations. Our approach is validated in the Ocean Observatory Initiative, a thirty-year $400 million project, which supports a variety of resources dealing with monitoring and studying the world's oceans. These resources include autonomous underwater vehicles, ocean gliders, buoys, and other instrumentation as well as more traditional computational resources. Our approach has the benefit of directly reflecting stakeholder needs and assuring stakeholders of the correctness of the resulting governance decisions while yielding adaptive resource allocation in the face of changes in both stakeholder needs and physical circumstances.

1 Challenges in Sociotechnical Systems

A cyber-physical system is a kind of pervasive system that brings together a number of computational and physical resources, usually in a specific social context. A major value of such systems is in expanding human and social capabilities in dealing with a complex environment. First-generation cyber-physical systems have been largely focused on low-level aspects such as sensors and effectors. Existing approaches assume that a single organization (e.g., a hospital or a disaster recovery team) owns or controls all the resources.

In contrast, our interest lies in *sociotechnical systems*, which we define as multistakeholder cyber-physical systems. Sociotechnical systems feature autonomous stakeholders whose interests are at best imperfectly aligned. We address the challenge of enabling stakeholders to *(self-)govern* such systems in a manner that supports adaptation in accommodating the exceptions and opportunities that arise in a complex environment.

Our participation in the recently launched Ocean Observatories Initiative (OOI) [Arrott et al., 2009], a paradigmatic sociotechnical system, has reinforced our motivation for the above challenges. The OOI facilitates the efforts of scientists and research institutions in acquiring, storing, analyzing, and sharing information from the world's oceans. Its stakeholders include oceanographers, educators, members of the public as well as research laboratories and universities. The stakeholders own and share resources such as Underwater Autonomous Vehicles (UAVs), buoys, ocean sensors, and research databases.

The OOI is designed to be operated for decades with initial funding for a thirty-year period. Consequently, we expect that nearly all implementation technologies deployed today will be obsolete within the lifetime of the system. Thus, not only must we accommodate changes in stakeholder needs, we cannot rely upon any specific technology to provide a stable notion of correctness. Further, OOI is a large system: about $40 million or 10% of its budget is for IT. It is worth noting that OOI itself would not own most resources involved in the collaborations it will help administer. The OOI is conceived of as a system with thousands of stakeholders, tens of thousands of physical resources such as ocean gliders, and potentially millions of virtual resources such as datasets. At those scales, adaptation is essential for administering resources according to the preferences of the stakeholders.

How can we accommodate stakeholder needs that are continually *changing*? How can multiple stakeholders function collaboratively in a sustainable, efficient manner? How can individual ownership and control be respected as autonomous parties interoperate? How can resources be added or dropped dynamically at runtime? How can dynamic coalitions be constructed and enacted to optimally share resources while entertaining challenges such the stakeholders' needs changing unexpectedly, as in an emergency? How may we accomplish all of the above *adaptations* over a range of resource granularities and timescales?

1.1 Self-Governance: Interplay of Norms and Organizations

The above challenges come together in the problem of *(self-)governance*. Briefly, governance is how autonomous entities administer themselves. Governance contrasts with traditional top-down management, which presumes authority (superior to subordinate) relationships. In the systems of interest, the collaborating parties are autonomous peers and none has authority over the others. Today, governance is carried out "by phone call"—by ad hoc negotiations among humans. Such manual techniques can work in small settings where a few resources need to be shared over long timescales. In contrast, the (pervasive) sociotechnical systems of interest involve large numbers of resources and require decision making at fast timescales. Manual negotiations would simply not scale to such settings.

We observe that from the perspective of governance, the stakeholders of a sociotechnical system are themselves *participants*. Recognizing the autonomy of the participants of sociotechnical systems, we observe that we cannot prescribe a decision-making strategy for each participant. Instead, each system can prescribe its rules of encounter via a set of norms. Informally, a *norm* characterizes sound interactions among the participants of a social group. Fundamentally, a norm defines what is normal, thereby distinguishing sound from unsound interactions, and reflecting the mutual expectations of the participants. We emphasize interactions because we have no interest in regimenting private behavior that has no effect on others. Two examples of norms in a scientific setting are putting an instrument in a power-save mode at the end of an experiment and closing unneeded datastreams from sensors. We are not concerned here with how norms arise, whether through bottom-up conventions or top-down legislation. We further restrict ourselves to norms that have some contractual force, so that their satisfaction or violation is significant.

Based on the above intuition, we formalize a sociotechnical system as an *organization* that involves two or more roles, each specified in terms of the norms applying to it. To this end, we formalize norms not as amorphous properties of the "system"—whatever that might be—but as directed normative relationships between participants in the context of an organization. Our formal model reflects this *essential duality* of organizations and norms: an organization is defined via norms and a norm is defined in an organization. Importantly, our approach accommodates *open* settings where a party may act outside the scope of a sociotechnical system while remaining subject to the norms defined in the system.

1.2 Principles of Adaptation in Sociotechnical Systems

Our approach seeks to engineer a sociotechnical system in such a manner as to support adaptation, both (1) in its configuration (and implementation) and (2) in its enactment realized through the interactions of its participants. The twin challenges of ensuring adaptation and achieving rigor lead us to adopt the following main principles.

- *Centrality of Norms:* A normative, as opposed to an operational, characterization of acceptable interactions is minimally constraining and thus essential in a long-lived system.

- *Autonomy and Policies:* The participants are autonomous, though subject to applicable norms. Each participant applies its local *policies* to decide how to interact; its policies reflect its autonomy.

The foregoing emphasis on autonomy and adaptation suggests that our computational system must incorporate *agents*, active entities that represent individual participants and organizations. The internal implementations of the agents are not relevant to governance, but their interactions are. The agents are subject to norms, which govern their interactions. The agents are only partially regimented. Where appropriate, we prefer to develop agents that respect the applicable norms, but recognizing the autonomy of the agents means that any agent may violate a norm. Therefore, our norms support a relaxed notion of correctness wherein correctness can potentially be restored after a violation.

1.3 Contributions and Claims

We develop a novel approach for governance that respects the needs of the stakeholders and is computationally realized. Our contributions are two-fold.

- A *formal model* for governance that incorporates a rich set of normative clauses promoting adaptability and reuse. This model provides a natural mapping to computations and can be realized generatively. It also supports useful kinds of analysis of particular organizations and norms.
- An *architecture* that realizes the above model and is instantiated in a prototype to demonstrate our approach on significant use cases arising in the OOI setting.

We claim that our model and architecture (1) enable the construction of a *flexible* sociotechnical system that can naturally (2) adapt in its *configuration*, thereby accommodating changes in stakeholder needs by reconstituting its rules of encounter and (3) adapt in its *enactment*, thereby accommodating the dynamics of a sociotechnical system.

For simplicity and brevity, we limit the scope of this paper to the aspects of the model and architecture that specifically focus on governance. In particular, we elide the substantial efforts within the OOI project on ontologies, resource models, and programming effort on instrumentation, data management, and a cloud-based computing infrastructure.

Section 2 introduces important governance scenarios from OOI. Section 3 describes our formal model for a sociotechnical system. Section 4 shows how to enact specifications in our formal model. Section 5 evaluates our approach with respect to the scenarios of Section 2. Section 6 discusses some general themes and directions for future research.

2 Application Scenarios and Varieties of Adaptation

Let us consider some simple scenarios that convey a sense of how we conceptualize the OOI being put to use, and illustrate the tension between regimentation and adaptability that is an essential characteristic of sociotechnical systems.

2.1 Collaboration through Resource Sharing

The stakeholders of OOI include a broad range of users, such as researchers, educators, students, and enthusiasts, with varying interests and expectations. Say, a teacher from a school near Chesapeake Bay would like to have his students conduct a project that exposes them to real-world data from their local environment. The teacher discovers an OOI member willing to share data from her salinity sensors in the Bay. Elsewhere, a researcher plans a comparative study of seasonal variations in salinity in Chesapeake Bay and Monterey Bay and its effect on algae. Although both the teacher and the researcher seek collaborations, the two engagements would differ in duration, exclusivity, and permissions over data use. Configurational adaptation: The researcher observes that scientific and educational engagements only account for 40% of her instrument's capacity. To maximize its impact, she begins to participate in a community of enthusiasts (members of the public). Operational adaptation: Because of an oil spill, there are suddenly new requests from researchers and enthusiasts. The instrument owner prioritizes even new requests from researchers over ongoing engagements with enthusiasts. Accordingly, she pulls back her instrument from the enthusiasts' community but lets the enthusiasts access a datastream from the instrument.

2.2 Affiliation

Research institutions and laboratories are central to the scientific effort and are first-class participants in OOI. Recognizing the benefits of sharing ocean instruments and curated datasets on a regular basis, the Chesapeake and Monterey laboratories, become *affiliates* of each other. Hence, the research staff of one can access resources from the other. But, each laboratory would keep some data and analytical tools private, e.g., because such data and tools are part of an ongoing study whose results the laboratory wishes to be the first to publish. Configurational adaptation: The laboratories expand their affiliation to include their respective zoological databases and students on a reciprocal basis. Operational adaptation: Monterey learns that Chesapeake has hired a researcher who was involved in some controversy about publishing premature results. At Monterey's behest, the two modify their affiliation to forbid unilateral publishing of results arising from collaborative studies.

2.3 Sanction

Individual collaborators or laboratory affiliates agree to specific terms, some of which restrict their actions. For example, a collaborator may be forbidden from changing the firmware on an instrument that is temporarily checked out to him or from externally publishing the results of a joint experiment. The participants in OOI are autonomous, meaning that they have an existence outside of the OOI system. Thus they can violate the terms of an agreement through actions that OOI cannot prevent, e.g., because they have physical control of an instrument or use an external web site to publish some data. However, such breaches may eventually be detected by the concerned parties, who can complain to the OOI, viewed as an authority. In such cases, OOI would subject the responsible party, if identified and found culpable, to specified sanctions, such as having to replace the instrument or issue a public retraction. OOI could cancel the account of a malfeasant participant. Configurational adaptation: The above engagement may be modified to allow revealing the data externally provided it is to a research sponsor to fulfill deliverable requirements. Operational adaptation: When a severe algae bloom occurs hidden beneath the surface of the Bay, a researcher unilaterally reports it to the press. The sanctioning process absolves the researcher because of extenuating circumstances: in this case, the researcher's violation was necessary to protect the health and safety of the public.

2.4 Requirements Induced from the Above Cases

The above scenarios indicate the need for flexibility in configuring engagements among individuals and institutions, because no static solution would accommodate the dynamic nature of stakeholder needs. For example, a researcher must be able to specify her requirements for sharing her ocean instruments. Even though such requirements would fall into a few typical patterns, the best practices patterns themselves would change over the course of years, if not of decades. Therefore, instead of legislating fixed policies, we must provide a flexible means to govern collaborations that naturally supports adaptation while ensuring a rigorous notion of correctness. In essence, we must lift the architecture from considerations of control or data flow among software components to considerations of norms among autonomous participants. In particular, given the autonomy of the participants, we cannot assume that no norm will be violated. This is because it would often be impossible to regiment all interactions of the participants. Thus each participant should potentially have recourse in case one of the other participants violates a norm, even if it does so outside the operational scope of the OOI.

Singh et al. [2009] identify three main elements of a service engagement: *transactional* (what the engagement accomplishes for its participants); *structural* (how the engagement is organized); and *contextual* (the broader rules of encounter to which the engagement is subject). We adopt the idea of Desai et al. [2009] to classify changes in requirements in terms of the above three elements. Whereas Desai et al. consider cross-organizational business processes, here we consider the norms broadly and consider more subtle situations of how the engagements in question are arranged. Viewed in the above light, the adaptations in the resource usage, affiliation, and sanctioning scenarios correspond to the transactional, structural, and contextual elements, respectively.

3 Modeling a Sociotechnical System

The foregoing use cases suggest two main requirements: the need for adaptivity and the need for rigor. On the one hand, the autonomy of the participants and the fact that they carry out long-lived collaborations across institutional scopes means that we must accommodate change. On the other hand, the same features mean that we must do so in a rigorous manner because otherwise it would be impossible to guarantee appropriate outcomes in such a complex setting. We develop a normative approach to address the above challenges. The norms are founded upon the idea of stakeholders being modeled as autonomous principals, represented computationally as agents, who carry out loosely coupled interactions.

3.1 Conceptual Model of a Sociotechnical System

Figure 1 illustrates the conceptual model that underlies our approach for governance. The notion of an Org is crucial in formulating interactions in terms of norms. Indeed, in our approach, all norms arise with an Org as a backdrop. In simple terms, an Org is recursively constructed: its members are principals that could themselves be Orgs. A principal may be a member of more than one Org: thus Orgs can have overlapping memberships. For simplicity, we assume that the membership relation between Orgs and principals is well-founded, e.g., two Orgs are not members of each other.

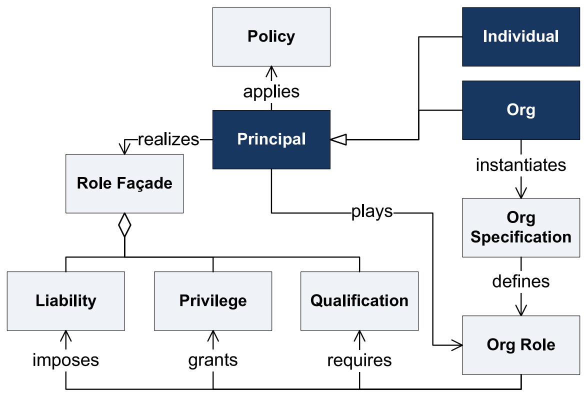


Figure 1: Simple conceptual model geared toward governance. (Here, white text indicates active entities; black text indicates representations.) This model is centered on the notion of *principal*, which corresponds to a participant in a system. A principal may be an *individual* (a researcher or even a laboratory if understood as an atom) or an *Org* (a structured entity such as a resource sharing community, an affiliation, or even the OOI itself). In either case, a principal is a locus of autonomy and potentially chooses its own *policies*, which reflect its autonomy. For example, a researcher may decide through her local policies whether to contribute usage of her glider to a community of enthusiasts. A laboratory may decide whether to admit an educator to access a sensor datastream.

Principals communicate and collaborate within the scope of an Org of which they are members. The most important purpose of an Org in our architecture is that an Org systematizes the norms among its members and potentially provides an authority to which the members may complain regarding norm violations by others. An Org may apply any appropriate sanctions on any of its members; such sanctions typically include canceling the membership of, or escalating a complaint against, a principal it judges malfeasant.

Orgs are finely structured through the notion of a *role*, which codifies a set of related interactions that a member of an Org may enact. To be a member of an Org means to play at least one role in that Org. In principle, a principal may concurrently play more than one role in the same Org or in different Orgs. However, some roles may limit such flexibility, e.g., to ensure a separation of duties. Each Org is specified by defining the rules of encounter for each of its roles. Together these rules of encounter may be understood as a multiparty contract. However, the elements that concern an individual role are most relevant to a principal who plays that role. For each role, we collect these elements into what we term the *façade* of that role. Each façade comprises three major components.

Qualification A prerequisite or eligibility requirement for a principal to play the specified role. Example: A user who wishes to participate as an *educator* in a continuing education Org for school teachers must be a credentialed and currently employed teacher.

Privilege A liberty, broadly understood, accorded to a principal who plays the specified role. Example: A teacher who is admitted as an *educator* to a summer camp Org is authorized to access all datasets available within the camp and is empowered to further admit a student as a *pupil* to the camp.

Liability A demand imposed on a principal who plays the specified role. Example: Continuing with the above scenario, a teacher who becomes an *educator* must entertain help requests from a student who is a *pupil*. Likewise, a *pupil* who introduces a virus into the camp's computers would risk sanctions, including possible expulsion.

Playing a role is thus a natural path for a principal to enter into norms with other principals. Moreover, principals may form additional norms through explicit negotiation. However, even the latter type of norms are governed by the norms that arise from the roles in an Org. For example, a teacher as an *educator* gains access to datasets but not to instruments. To be able to use an instrument owned by a scientist, a teacher may agree on additional terms and conditions, such as that he would not reboot the instrument. Such agreements would arise in the scope of the same Org, and their violation could have consequences such as the impact of sanctions defined in the *educator* façade.

The model of Figure 1 posits that an Org is a principal and can thus participate in other Orgs. We now further posit that an Org *qua* principal may also interact with and enter into norms with its own members. For example, when researcher Ryzard joins OOI, not only is he subject to OOI's norms but he may also expect OOI to keep his private information safe. We capture the above intuition by postulating a distinct *self* role for each Org. In any Org, this role is played by exactly one principal, namely, the Org itself. Further, this role is instantiated simultaneously with the Org coming into being. In conceptual terms, an Org as *self* interacts with all its members, handles their requests to discover other members and resources, entertains their complaints about each other, adjudicates on the norms between them (in its capacity as context for such norms), and enforces any applicable sanctions.

3.2 Normative Concepts

Based on an analysis of sociotechnical systems, especially OOI, we postulate the following normative concepts as the key elements of a role façade. These concepts are not new to our approach but we characterize them differently from previous work and show (in Section 4) how to operationalize them in a way that applies naturally to sociotechnical systems.

When employed as a design construct, a norm codifies desired properties of interactions among principals. In simple terms, a norm captures the sense of how an interaction *ought* to proceed and thus regulates the interactions of the principals involved. By providing a rich set of constructs with which to express the norms, we enable encoding the essential properties of interactions in a manner that is flexible (any enactment that satisfies the norms is acceptable) yet rigorous (there is a precise computational notion of when a norm is violated). The flexibility helps ensure correctness while supporting adaptation in configuration (to accommodate changes in stakeholder requirements) and during enactment by the principals. During enactment, the norms progress because of the principals' interactions: e.g., they may be activated, satisfied, or violated. A snapshot of the norms taken together constitutes the *normative state* of the sociotechnical system. Figure 2 shows how our norm representation generalizes over the representations of Singh [1999, 2008].

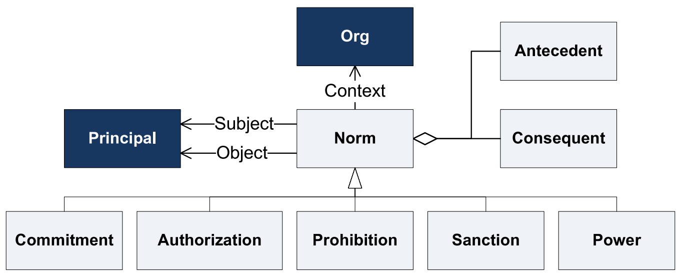


Figure 2: The unified logical form of a norm. Each norm involves a *subject* (the principal on whom the norm is focused), an *object* (the principal with respect to whom the norm arises), a *context* (the Org within whose scope the norm arises), an *antecedent* (expressing the conditions under which the norm is fully activated and brought into force), and a *consequent* (expressing the conditions under which the norm is fully satisfied and deactivated). In many practical cases, we set the antecedent to true to indicate an unconditional norm. Further, the context could be the same as the subject or the object, whereas the subject and object are always distinct.

Commitment An active commitment means the subject (i.e., debtor) is committed *to* the object (i.e., creditor) within the scope of the organizational context [Singh et al., 2009]. It means that if the antecedent holds, the debtor commits to bringing about the consequent. And when the consequent holds, the commitment is satisfied and deactivated. Example: A researcher who borrows an instrument for a study commits to returning it within one hour of being requested to do so.

Authorization This specifies that, with respect to their interactions within the given context, the object authorizes (i.e., permits) the subject to bring about the consequent when the antecedent holds. Bringing about the consequent if the antecedent remains false is a violation. Example: An instrument owner authorizes a colleague to use the instrument between 7:00PM and 9:00PM.

Prohibition This specifies that, with respect to their interactions within the given context, the object prohibits (i.e., forbids) the subject from bringing about the consequent provided the antecedent holds. Bringing about the consequent if the antecedent holds is a violation. Examples: An instrument owner prohibits a borrower from changing the firmware on the instrument. A dataset curator prohibits a reader from publishing any of the data on an external web site.

Sanction This specifies that, with respect to their interactions within the given context, the object would sanction (i.e., punish) the subject for bringing about the consequent provided the antecedent holds. Examples: An instrument owner would sanction a borrower who illicitly changes the firmware on a borrowed instrument by giving the borrower a poor rating. A dataset curator would sanction a reader who publishes any of the data externally by complaining to the Org. The resource sharing Org would sanction a reader who publishes any of the data externally by ejecting him from the Org.

Power This specifies that, with respect to their interactions within the given context, when the antecedent holds the object empowers the subject to bring about the consequent at will. Loosely following Hohfeld [1919], we treat a power as the ability to alter the norms between two or more principals, usually those playing specific roles. We follow Jones and Sergot [1996] in treating power as an *institutional* construct, meaning that a power exemplifies the so-called *counts-as* relation between a low-level (physical) ability and a high-level (institutional) action. This intuition generalizes to the idea of making a norm concrete [Aldewereld et al., 2010]. Importantly, a principal may be empowered to do something but not be authorized to do so. A simplification our setting supports is that the physical action is a communication: thus when the antecedent holds, the subject need only "say so" to bring about the consequent. Examples: The Chesapeake Bay Org is empowered to admit or eject its members by declaring so. An instrument owner is empowered to contribute her instrument to a resource sharing Org, also by declaring so. A system administrator is empowered to admit new people into OOI by creating their accounts, but is—crucially—prohibited from creating accounts (and admitting members) without approval from the membership department. However, because the administrator has the power, her creation of a new account will succeed, though it might later be deemed illicit and revoked, and the administrator sanctioned for exercising the power illicitly.

Governance involves modeling not only the norms but also how the norms are *manipulated*. For example, when a principal joins a resource sharing Org as a *user*, he acquires the norms in the *user* façade. Table 1 maps the above concepts to the components of a role façade and its caption explains how they are manipulated. Notice that qualifications are treated merely as credentials even when they happen to refer to privileges in other Orgs.

Taking on a role creates the associated norms; exiting a role cancels or releases (as appropriate) its norms; a sanction may create an additional commitment to pay a penalty and cancel current authorizations to use any instruments within the Org; and so on. The operations on norms as well as events in the Org cause a progression of norms. Figure 3 shows the life cycle of a norm in terms of its key states and transitions.

As Section 2.3 illustrates, a sociotechnical system is inherently open in that its autonomous participants have an external existence. In general, each Org is open and cannot

Table 1: Mapping normative concepts and operations on them to role façades. We generalize Singh's [1999] commitment manipulation operations for all norms. Any norm may be created (directly by the liable party or via a causal chain leading back to the creation of another norm by the same party), discharged (satisfied by the liable party), canceled (terminated by the liable party, though at risk of violation), released (terminated by the privileged party, because it does not care), delegated (by the liable party to a new liable party), or assigned (by the privileged party to a new privileged party).

Normative Concept	**Subject's Façade**	**Object's Façade**
Commitment	Liability	Privilege
Authorization	Privilege	Liability
Power	Privilege	Liability
Prohibition	Liability	Privilege
Sanction	Liability	Privilege

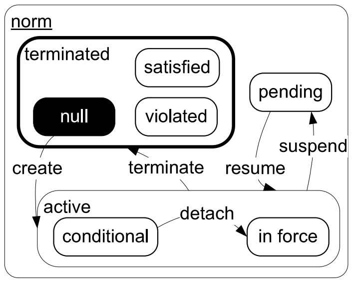
regiment all the actions of its participants. We address the above challenge through a simple approach that consists of two parts: (1) representing the appropriate norms for each Org, as below, and (2) enacting the norms appropriately, as in Section 4.

In general, a principal ought to perform only such actions for which it is authorized and not prohibited. We distinguish interactions that occur within the scope of an Org from those that occur without. We adopt the following design pattern that simplifies modeling and enactment. We treat authorizations as applying exclusively to the internal interactions and prohibitions as applying exclusively to the external interactions. The internal interactions are architecturally regimented by authorizations and therefore never occur unless authorized. The external interactions are subject to prohibitions but cannot be architecturally regimented. Therefore, for each prohibition we need to specify a sanction in case it is violated, but not so for any of the authorizations.

Figure 4 summarizes an important part of our vocabulary. Notice that the states of a norm are part of our vocabulary and can be referenced from other norms. For example, consider a commitment $c_1 = \mathsf{C}(d, c, o, p, q)$ in the form introduced in Figure 2. Then we can express a commitment from Org context o to creditor c that if c_1 is violated, Org o will compensate creditor c by ensuring r as $c_2 = \mathsf{C}(o, c, \mathsf{OOI}, \mathsf{vio}(\mathsf{C}(d, c, o, p, q)), r)$ [Singh et al., 2009]. Self-referential or mutually referring norms are not permitted by our syntax.

3.3 Outline of a Modeling Methodology and Specification Snippets

To understand our modeling methodology, consider the resource sharing scenario of Section 2.1 again. A resource sharing Org admits principals who may play one or both of the roles *user* or *owner*. Any principal who owns resources may accept the *owner* façade and thus enroll in the Org. An *owner* may contribute a resource to the Org: the Org would list it in a directory. Similarly, a *user* may search the resource directory maintained by



If terminated in		Then				
ant	con	Com	Aut	Pro	San	Pow
false	false	null	null	null	null	null
false	true	sat	vio	null	null	null
true	false	vio	null	sat	null	vio
true	true	sat	sat	vio	sat	sat

Computing the substate of a terminated norm (abbreviated to three letters). In the case of a power, a vio occurs upon the failure of an attempt to bring about the consequent.

Figure 3: The unified life cycle for a norm in the UML state diagram notation. (A nested state indicates that the norm could be in any of its substates.) Here null is the initial state and terminated is the final state. A norm is active when created: it is in force when its antecedent holds and conditional otherwise. A norm may be suspended, e.g., as when a subject of a commitment delegates it, and resumed, e.g., as when the delegate fails and the subject activates the commitment again. A norm is terminated because its subject, object, or context explicitly deactivate it or because of timeouts. The table specifies the appropriate terminating substate of a norm depending whether its antecedent and consequent are true or false there.

the Org, request access to, and use resources contributed by others. A *user* and an *owner* may negotiate usage terms, possibly creating additional norms. The negotiation may be as simple as an *owner* requiring a *user* to accept a disclaimer about the quality of the resource. An *owner* may withdraw a resource it previously contributed, but only when no *user* is actively using the resource. Further, a *user* may not share a resource obtained from this Org with any entity external to the Org. However, if the *user* wishes to do so, the Org cannot prevent it. Therefore, we express a prohibition against external sharing along with a sanction of possibly ejecting violators from the Org. Applying our methodology on this scenario yields a specification of the Org in our formal language. For brevity, we embed some illustrative snippets of the specification below (a question mark indicates a variable).

- *Identify the roles in the scenario:* *user* and *owner* as well as *self* (needed for each Org).

- *Identify the interactions:* A principal interacts with the OOI Org to discover an Org for accessing data about the water chemistry of Chesapeake Bay. The principal discovers resources contributed by the members of this Org. Alternatively, or in addition, the principal may also contribute resources to the Org. The foregoing yields interactions for discovering, negotiating for, using, contributing, and withdrawing resources.

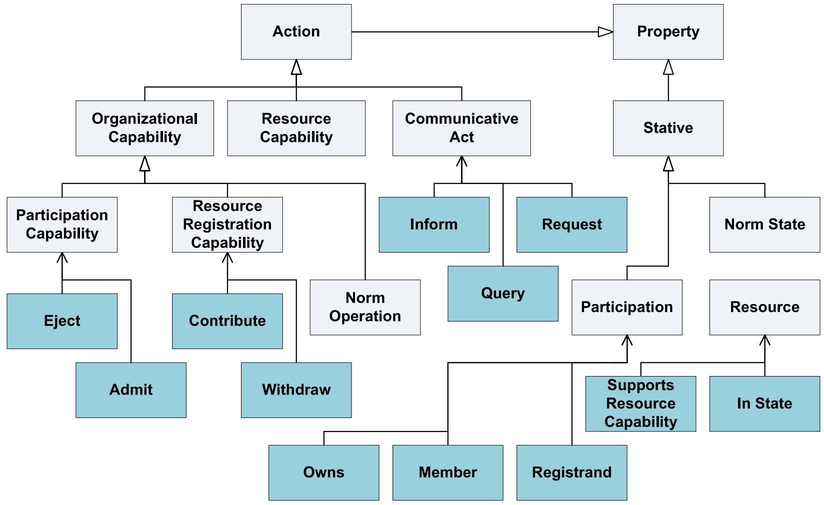


Figure 4: A fragment of the vocabulary used in expressions (antecedents and consequents of norms). Each property is applied with the requisite number of arguments. The unshaded boxes are general classes corresponding to our model; the shaded boxes are important examples, some specific to OOI. Resource capabilities would be highly domain specific. The actions bring about changes in state and the statives help refer to relevant states from within norms. To reduce clutter, we elide the norm states and operations, which are as described in Table 1 and Figure 3. Nominally, the antecedents of norms would involve statives and their consequents actions.

- *Identify resource capabilities:* A glider may be dived, surfaced, recharged, and read.

```
Capability:Resource (?aResource, DepthControl)
```

- *Identify the façade of each role:* The *user* and *owner* façades include the following.

  - Qualifications: A *user* must be a member of OOI.

  ```
  Qualification OOI:Public(?user)
  ```

  - Privileges: *self* is empowered to admit members into the Org. An *owner* is empowered to contribute or withdraw a resource. An *owner* may only contribute a resource that she owns and may withdraw a resource that she owns only when it is not currently in use. An *owner* may authorize a *user* to apply a resource capability.

  ```
  Power {
    If SharableResource(?resource)
     AND Owns(?owner, ?resource)
     AND Supports(?resource, ?capability)
    Then Contribute(?owner, self, ?resource, ?capability)}

  Power {
  ```

```
    If Contributed(self, ?resource, ?capability)
     AND Owns(?owner, ?resource)
    Then Withdraw(?owner, self, ?resource, ?capability)}

  Authorization {
   If Contributed(self, ?resource, ?capability)
    AND NOT inState(inUse, ?resource, ?capability)
   ThenMay Withdraw(?owner, self, ?resource, ?capability)}

  Power {
   If Contributed(self, ?resource, ?capability)
    AND Owns(?owner, ?resource)
    AND Participant(self, ?user, ?role)
   Then Authorize(?owner, ?user, Apply(?role, ?resource, ?capability))}
```

- Liabilities: A *user* may not externally share a capability on a resource accessed through this Org. A *user* who violates the above prohibition is subject to the sanction of being potentially ejected from the Org.

```
Prohibition {
  If NOT Participant(self, ?outsider, ?norole)
   AND Contributed(self, ?resource, ?capability)
  ThenMayNot Share(?user, ?outsider, ?resource, ?capability)}

Sanction {
  If Violated
    (Prohibition {
      If NOT Participant(self, ?outsider, ?norole)
       AND Contributed(self, ?resource, ?capability)
       ThenMayNot Share(?user, ?outsider, ?resource, ?capability)})
  Then Eject(self, self, OOI:member, ?user)}
```

- *Validate the set of norms:* No principal should be prohibited from satisfying a commitment. A sanction must be applied by a principal who possesses the requisite power and authorizations. For example, an aggrieved principal may sanction by escalating the dispute to the Org, which would impose its own sanction on the malfeasant principal. For brevity, we accounted for this above.

Formally, given the design pattern introduced above, we need an authorization for every power. Therefore, the following permissive authorization is automatically generated for a power to bring about P, for which no other authorization is specified.

```
Authorization {
 If true
 ThenMay P}
```

Our language supports role inheritance so that one role may extend another role. This enhances reusability. Specifically, *owner* extends *user* since it grants additional privileges and imposes additional liabilities.

4 Enacting a Sociotechnical System

Because we understand sociotechnical systems as involving autonomous participants, a clear requirement is that we enact such a system in a fully decentralized manner. To this end, an *agent* as a computational surrogate of a principal. An agent is not autonomous with respect to its principal, but is autonomous as viewed from the perspective of other agents. An agent applies its local policies, presumably based on its principal's preferences, in deciding how to interact with other agents.

Each principal's agent helps with the bookkeeping of the norms in which it features as subject, object, or context. The agent helps determine if the principal itself is complying with the applicable norms and, equally importantly, if the principals with whom it deals are complying as well. The agent maintains its local view of the normative state by continually updating the relevant norms.

We address the following challenges: (1) developing an agent so that it respects the façades of roles its designer would like it to play; (2) judging if an interaction complies with the specified norms; and (3) during enactment, having an agent compute what actions it ought or ought not to perform.

4.1 Computing with Rules

In architectural terms, our approach is neutral as to whether an agent is implemented in a more or a less restrictive manner, ranging from traditional software to a general-purpose planner. As a practical matter, we adopt a rule-based approach because it offers a happy middle between flexibility and ease of implementation. Note that any domain-specific reasoning could be realized through a traditional imperative language even though we account for norms through a rule-based language. To this end, we model each agent as maintaining a *belief store*. We use the term belief instead of fact or knowledge, because these are representations of the agent's local view. An agent acts according to its beliefs, but norms are inherently interactive and compliance in general is not based on what an agent believes but solely on how it interacts. Hence a design requirement is to ensure both that agents have true beliefs and can reason properly from them.

An agent updates its belief store by asserting or deleting beliefs as it performs communications. We capture actions on resources as messages sent and observations from the environment as messages received. The beliefs in an agent's belief store represent the current snapshot of the physical state of the system, e.g., that a glider is broken or that a network connection to a buoy has a throughput of 2kbps. We separate out elements of the normative state, e.g., that the agent has an active commitment to reporting the failure of the glider to the (agent of the) owner of the glider or that its commitment is pending because it was delegated to another agent. The beliefs occur as propositions within the antecedents and consequents of a norm. Each agent ideally tracks each norm in which its principal features, whether as subject, object, or context. Potentially, any action that an agent chooses to perform or omit may have repercussions on the satisfaction or violation of its norms: in some cases immediately and in other cases a long time into the future. Therefore, an agent may evaluate and filter its options with respect to the norms it tracks.

Figure 5 illustrates our reference agent architecture. The decision maker attempts actions. The normative filter checks all of the agent's attempted actions for proper authorization and forwards along exactly those that it judges to be in compliance with the applicable norms. The communicator receives and sends messages, thereby applying the agent's attempted action if approved by the filter. In either case, it updates the beliefs accordingly.

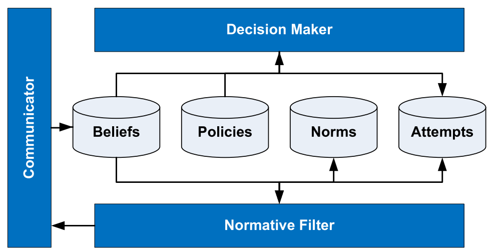


Figure 5: A simplified representation of our agent architecture. An agent has four main stores (for beliefs, policies, norms, and attempts) and three active modules. The communicator reports incoming and outgoing messages to the belief store. The decision maker applies its policies given the beliefs to compute possible actions, which it attempts. The normative filter maintains the agent's norm store. In particular, it adjudicates on the suitability of the attempted actions and determines whether other agents are complying with their norms—those of relevance to the agent. The agent acquires or modifies norms, such as occurs when it takes on or resigns any role.

4.2 Mapping Norms to Generative Rules

We now discuss how we systematically map a role façade to an enactable agent specification. The agents, being autonomous, apply their local policies. However, each role that an agent plays constrains it based on the role's façade. Since we conceive of sociotechnical systems in which the agents are broadly cooperative though not fully trusted, we propose a straightforward means by which we can ensure that an agent complies with its norms. This involves placing some regimentation into the computational system as a way of ensuring that each agent respects its authorizations. However, we leave open the possibility of an agent not complying with an applicable norm.

Since the commitments where an agent is the subject (i.e., debtor) specify what it must perform, we use them to structure the decision maker component of the above architecture. A commitment maps to the following forward-chaining rule template for its subject. Here the variables in the antecedent are bound when the commitment is in force, i.e., detached, and additional variables needed in the consequent are bound through the agent's policy.

```
If Antecedent (?x ?y)
 AND Policy (?x ?y ?z)
Then Attempt (Consequent (?x ?y ?z))
```

Each policy is itself captured through one or more backward-chaining rules accounting for how the programmer wishes the agent to reason in this case. In general, there would be at least as many policies as commitments. If the antecedent holds and the policy evaluates to true for some bindings of variables, the decision maker attempts the consequent action. In cases where the consequent of a commitment involves exercising a power, i.e., the consequent of a commitment includes the consequent of a power, we generate an alternative rule template whose Then clause is the communication corresponding to the consequent.

An additional rule template corresponds to handling messages from others. Each such template checks if the sender is suitably empowered and authorized for the given interaction. Also, where access is given to a resource (as by an *owner* to a *user* for an instrument), the authorization is placed on a proxy for the resource and verifies that incoming requests are valid. In either case, failure flags a violation.

The normative filter verifies if the action being attempted is authorized and passes it along to the communicator if and only if it is. We model two kinds of attempts: *now or never* (discarded on failure) or *good till canceled* (retried repeatedly until it executes once or the decision maker annuls it). In addition, in our default operational model, the normative filter also verifies whether the action being attempted would violate any prohibitions. Doing so improves the quality of a collaboration. In general, an agent cannot assume others will not violate their prohibitions, because the agents are not all implemented or controlled by us. Specifically, users may cause their agents to violate a norm or, as explained above, may act externally to the Org. Thus the "trust but verify" dictum applies in our setting.

Determining whether an attempted action is authorized is nontrivial, because some actions can have additional consequences, and some of those consequences might not be authorized. In particular, when the agent is empowered to create a new norm, it may not exercise such a power unless the norm to be created is authorized. For example, the Monterey Org should not commit Ryzard to reboot Alice's instrument without her permission. To this end, the normative filter computes the *power closure* of an action and verifies that all actions in the closure are authorized.

Additionally, the normative filter tracks the states of all applicable norms in which the given agent features. Specifically, it updates the normative state based on any powers (of this or other agents) that might be exercised when an outgoing or incoming message occurs. If it detects a violation of a norm by another agent, it applies the specified sanction. In particular, a common sanctioning pattern is for the agent to generate an escalation, i.e., a complaint, to the Org that is the context of the violated norm, and for the Org to exercise stricter sanctions. In addition, depending on the applicable role façade, the given agent may also carry out a sanction such as giving the violating agent a poor rating.

5 Evaluation

We now address the claims asserted in Section 1.3 by returning to the OOI application scenarios introduced in Section 2.

5.1 Realizing the Scenarios of Section 2

5.1.1 Collaboration

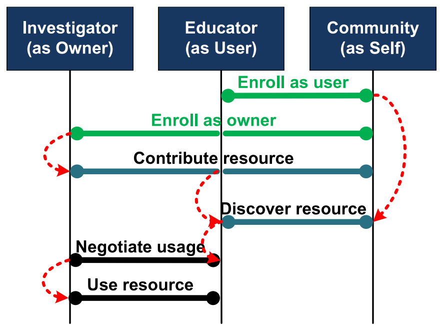


Figure 6: Illustration of governance in a resource sharing Org. The notation is loosely based on message sequence charts. The horizontal lines show governance interactions that create or modify norms among the parties whose life-lines they connect. (In general, a governance interaction may involve more than two parties.) Any temporal order requirements are captured via the dashed arrows that connect some pairs of the horizontal lines.

Figure 6 shows how governance may be flexibly captured in terms of the creation and manipulation of norms among principals. Doing so yields clarity in understanding and validating the model as well as flexibility in operational terms. Specifically, even something as simple as enrollment can potentially be operationalized in multiple ways, including by having either the prospective enrollee or the prospective enroller take initiative by, respectively, requesting membership or inviting the enrollee.

5.1.2 Affiliation

Figure 7 illustrates the structural and contextual scenarios. The Chesapeake and Monterey Orgs *qua* principals play the *affiliate* role in the Salinity and Algae Org, whose norms support the formation and maintenance of norms between Ryzard, a Monterey *user*, and Bejan, a Chesapeake *owner*.

5.1.3 Sanction

Figure 7 treats OOI itself as a principal that acts as an overarching authority for all interactions within its scope. As the *root* Org, OOI defines the identities for the principals involved and provides the basic rules of encounter for all constituent Orgs. In this setting, if Ryzard misuses Bejan's instrument, Bejan can complain to the Chesapeake Org; his complaint is forwarded via the Salinity and Algae Org to the Monterey Org, which may sanction Ryzard or risk itself being ejected from the Salinity and Algae Org.

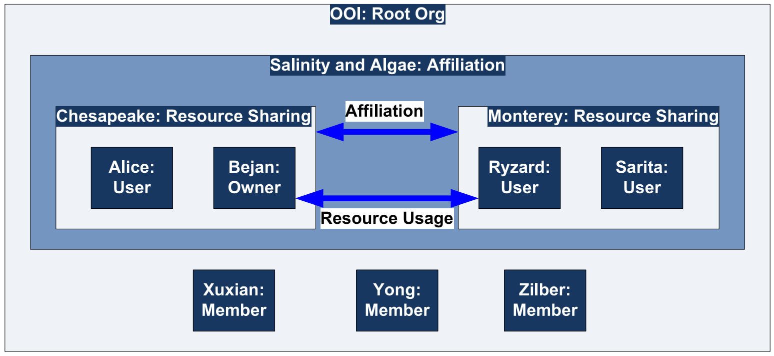


Figure 7: A schematic of two resource sharing Orgs affiliating with each other through the aegis of another Org. In this scenario, the root Org has seven members, two of whom have enrolled in the Chesapeake Org and two in the Monterey Org. The Chesapeake and Monterey Orgs affiliate to form the Salinity and Algae Org and function as principals within it.

5.2 Flexibility in Configuration

5.2.1 Transactional Adaptation

The researcher merely enrolls in a community for enthusiasts to which she contributes instruments that have spare capacity. She limits the contributed capabilities so an inexperienced user cannot inadvertently harm her instrument.

5.2.2 Structural Adaptation

The Salinity and Algae affiliation Org is expanded so that each laboratory (1) exposes its zoological databases, enabling their discovery by members of the other, and (2) entertains discovery and usage requests from principals playing the *student* role in the other.

5.2.3 Contextual Adaptation

The collaborators decide that a prohibition against sharing data externally would prove onerous. They decide to remove that prohibition with respect to deliverables of datasets to satisfy their respective research sponsors. This modifies the prohibition and in essence reconfigures the subsequent engagement.

5.3 Flexibility during Operation

5.3.1 Transactional Adaptation

The researcher simply applies a policy that leads her to exercise her power as *owner* to withdraw the instrument that is attracting high demand from the community of enthusiasts.

According to the rules of encounter, she can deny additional usage requests for the instrument immediately, but must wait to withdraw the instrument until its current usage sessions have ended.

5.3.2 Structural Adaptation

In the middle of the ongoing affiliation, one of the parties proposes a modification of one or more role façades. Each participating Org (viewed as a principal) acts autonomously with respect to the other and the modification takes place only if both agree. If they do not agree, the proposing Org may terminate the affiliation according to the existing norms. Notice that the Orgs are autonomous with respect to each other, but need not be autonomous with respect to their members. For instance, based on each Org's membership norms, a designated role could have the power to decide on its behalf or the Org could conduct a referendum of its members. Specifically, the two Orgs may use completely different mechanisms to arrive at their respective decisions.

5.3.3 Contextual Adaptation

We support this scenario by dynamically modifying the sanctioning norm of the Chesapeake Org. The Org's configuration would not change but the policy under which the Org exercises applicable sanctions is altered (through a decision mechanism similar to the one alluded to above).

6 Discussion: Literature and Future Work

This paper has made the case that self-governance is a natural metaphor for the administration of multistakeholder sociotechnical systems, treating the stakeholders as active participants. First, our approach respects the autonomy of the participants while supporting adaptations in their mutual interactions. Thereby, it enables the automation of what would otherwise be manual out-of-band administrative processes. Second, being founded in norms, our approach naturally provides an elegant way to realize governance by providing a measure of correctness that emphasizes interactions and is independent of implementation details. Third, a benefit of our approach is that through the composition of Orgs, it supports modularizing the norms and the agents' policies with respect to norms. As a result, it simplifies reusing Org specifications as well as the policies through which agents enact their roles. We have applied our approach on real-life scenarios from the specification and operation of a large sociotechnical system. As Table 2 shows, it is conceptually quite straightforward to accommodate a rich variety of adaptations in our approach.

Table 2: Summary of the main types of adaptations accommodated by our approach.

	Element	Example (from Section 2)	Approach (from Sections 5.2 and 5.3)
Configuration	*Transactional*	Support enthusiasts with spare capacity instruments	Join an additional Org corresponding to a community of enthusiasts
	Structural	Expand affiliation to introduce new resources and participants	Agree to make additional resources available to the affiliate and to entertain requests from an expanded pool of roles
	Contextual	Allow deliverables to be sent to a research sponsor	Weaken the prohibition against sharing and no longer prohibit sharing data with a sponsor
Operation	*Transactional*	Reallocate resources during oil spill	Withdraw a resource with high demand from the (low priority) community of enthusiasts
	Structural	Modify the affiliation to forbid unilateral publishing of results	Introduce a new norm into the façades of the appropriate roles
	Contextual	Disregard sharing prohibition during algae bloom emergency	Modify the sanctioning policy to account for situations threatening public safety

6.1 Relevant Literature

The relevant literature falls into two major classes: (1) on autonomic computing and policy-based systems and (2) on normative multiagent systems.

6.1.1 Autonomics and Policy

Our principles and approach for adaptation respect but enhance autonomic computing [Kephart and Chess, 2003]. In particular, our configurational adaptations capture their notion of self-configuration. Brazier et al. [2009] identify synergies between autonomic computing and multiagent systems, which this paper partly illustrates. A point of distinction from the above works is that we emphasize multistakeholder systems, where self-governance is a better metaphor than self-management.

Curry and Grace [2008] motivate the need for self-representation as a basis for autonomic computing. They describe the implementation pattern used in their GISMO system for messaging middleware. We agree on the importance of self-representation for adaptivity. However, we emphasize multistakeholder systems wherein the self-representation and the concomitant patterns are more naturally expressed at a high level, in terms of norms.

It is clear that policies are crucial to governance. A key difference from existing work is that we consider distinct policies for different principals because there is no central point that controls each principal. Liu et al. [2009] focus on event-condition-action (ECA) policies, which they term "obligations" because ECA policies specify actions that must be performed when the event occurs and the condition holds. Liu et al.'s main contribution is formalizing high-level events as patterns over low-level events. Shankar et al. [2006] generalize ECA policies by explicitly modeling the pre- and postconditions of actions. Doing so facilitates computing a correct order in which to apply the policies. The above approaches complement our approach and their representations and compilation techniques could be combined with it. However, we go beyond previous work in modeling a system with multiple autonomous parties, capturing norms explicitly, and in having each participant base its actions on the applicable norms. In particular, whereas obligation policies require immediate action whenever they match, by representing norms explicitly, we can decouple the creation of a commitment or other norm from acting on the norm.

6.1.2 Norms

The EU ALIVE project too addresses organizational adaptivity in terms of norms [Álvarez-Napagao et al., 2009]. The key points of difference are in our directed representations of norms, explicit treatments of operations on norms, and an emphasis on the duality of organizations and norms. Tinnemeier et al. [2010] study schema and instance changes in norms, which correlate with contextual adaptation in terms of configuration and operation. They assume that a norm change is somehow specified, but do not consider the governance processes by which principals would agree to a specific norm change.

Campos et al. [2009] propose an adaptation mechanism for electronic institutions that employs "staff" agents to monitor members' behavior and if necessary autonomically reconfigure the system. Overbeek et al. [2010] propose an approach that supports both direct control by a regulator and self-regulation as ways to ensure norm fulfillment. In contrast, our approach emphasizes the participants' autonomy, so no staff or regulators can control an agent or reconfigure an Org. We formulate the *self* role, which captures the Org as a participant, and projects the Org's identity uniformly inside and outside of its scope. We address adaptations through decentralized creation and manipulation of norms. As a result, we can accommodate configurational and operational adaptations in a simplified, uniform framework. However, Overbeek et al.'s value-based methodology is compelling, especially for a multistakeholder system. It would be useful to layer it over our governance model.

Fornara and Colombetti [2009] have similar intuitions about norms as us. They too argue that agents should not be regimented and should be able to violate their norms, albeit at the risk of facing sanctions. Our architecture offers the benefit of combining regimentation, which is efficient and reliable, with sanctioning, which is essential in open settings. Fornara and Colombetti offer an expressive temporal language but consider only commitments. Their approach can be combined with ours in this regard.

Vasconcelos et al. [2007] apply norms for modeling e-Science organizations. They treat norms as being applied by the organization and do not emphasize the autonomy of the principals; also, their norms are not directed from one to another thereby losing some expressive power. They do not have an account of external violations and sanctioning against them. However, Vasconcelos et al. address the important problem of resolving conflicts among norms, which can arise when a principal plays two or more roles. It would be useful to combine their approach with ours so that an agent can analyze its norms before taking on any roles.

6.2 Directions for Future Work

We expect that an agent would satisfy all its norms. Therefore, detecting inconsistencies among norms and computing acceptable actions for consistent sets of norms is a crucial challenge. We can expect the designers of an Org to create consistent norms. However, a principal may play roles in multiple Orgs. Further, privileges can sometimes function as liabilities. For example, assume principal Yong, but not Zhang, is empowered to publish a report merely by sending an email to a web site. Then Yong's power could prove undesirable for him because he might violate a prohibition, whereas Zhang would be protected from such a violation. Therefore, support for norm consistency (as inspired by [Vasconcelos et al., 2007]) and other validity checks are a key challenge.

Techniques for authoring agent policies and verifying them with respect to the norms that govern a given agent are crucial. A related challenge is verifying whether the specified norms are supported by a given operational description such as might be described via sequence diagrams. Telang and Singh [2010] address this problem for commitments; we leave it as future work to extend it to the full range of norms introduced in this paper.

We outlined a simple methodology for the design of sociotechnical systems. However, a more extensive approach is needed that would accommodate considerations such as stakeholders' goals, which underlie governance (and other) requirements, as in Tropos [Bresciani et al., 2004]. Our approach agrees in spirit with Tropos but begins from a first-class status for norms among autonomous principals. Penserini et al. [2007] address the challenge of high-variability design from the standpoint of Tropos. Therefore, it is only natural that in future work we attempt to develop a methodology based on Tropos but extended to deal with the special challenges of norms and organizations.

Acknowledgments

We thank Matthew Arrott, Amit Chopra, Emilia Farcas, Prashant Kediyal, Michael Meisinger, and Von Welch for useful discussions.

References

Huib Aldewereld, Sergio Álvarez-Napagao, Frank Dignum, and Javier Vázquez-Salceda. Making norms concrete. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 807–814, Toronto, 2010. IFAAMAS.

Sergio Álvarez-Napagao, Owen Cliffe, Julian A. Padget, and Javier Vázquez-Salceda. Norms, organisations and semantic web services: The alive approach. In *Proceedings of the Second Multi-Agent Logics, Languages, and Organisations (MALLOW)*, volume 494 of *CEUR Workshop Proceedings*, Aachen, 2009. CEUR-WS.org.

Matthew Arrott, Alan D. Chave, Claudiu Farcas, Emilia Farcas, Jack E. Kleinert, Ingolf Krueger, Michael Meisinger, John A. Orcutt, Cheryl Peach, Oscar Schofield, Munindar P. Singh, and Frank L. Vernon. Integrating marine observatories into a system-of-systems: Messaging in the US Ocean Observatories Initiative. In *Proceedings of Oceans, the MTS-IEEE Conference on Marine Technology for our Future: Global and Local Challenges*, pages 1–9, Los Alamitos, October 2009. IEEE Computer Society.

Frances M. T. Brazier, Jeffrey O. Kephart, H. Van Dyke Parunak, and Michael N. Huhns. Agents and service-oriented computing for autonomic computing: A research agenda. *IEEE Internet Computing*, 13(3):82–87, 2009.

Paolo Bresciani, Anna Perini, Paolo Giorgini, Fausto Giunchiglia, and John Mylopoulos. Tropos: An agent-oriented software development methodology. *Journal of Autonomous Agents and Multi-Agent Systems*, 8(3):203–236, May 2004.

Jordi Campos, Maite López-Sánchez, Juan Antonio Rodríguez-Aguilar, and Marc Esteva. Formalising situatedness and adaptation in electronic institutions. In *Proceedings of the International Workshops on Coordination, Organizations, Institutions and Norms in Agent Systems*, pages 126–139, Berlin, 2009. Springer-Verlag.

Edward Curry and Paul Grace. Flexible self-management using the model-view-controller pattern. *IEEE Software*, 25(3):84–90, May 2008.

Nirmit Desai, Amit K. Chopra, and Munindar P. Singh. Amoeba: A methodology for modeling and evolution of cross-organizational business processes. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 19(2):6:1–6:45, October 2009.

Nicoletta Fornara and Marco Colombetti. Specifying and enforcing norms in artificial institutions. In *Declarative Agent Languages and Technologies VI, Revised Selected and Invited Papers*, volume 5397 of *LNCS*, pages 1–17, Berlin, 2009. Springer.

Wesley Newcomb Hohfeld. *Fundamental Legal Conceptions as Applied in Judicial Reasoning and other Legal Essays*. Yale University Press, New Haven, CT, 1919. A 1919 printing of articles from 1913.

Andrew J. I. Jones and Marek Sergot. A formal characterisation of institutionalized power. *Journal of the IGPL*, 4(3):429–445, 1996.

Jeffrey O. Kephart and David M. Chess. The vision of autonomic computing. *IEEE Computer*, 36(1):41–50, 2003.

Zhen Liu, Anand Ranganathan, and Anton Riabov. Specifying and enforcing high-level semantic obligation policies. *Journal of Web Semantics*, 7(1):28–39, 2009.

Sietse Overbeek, Virginia Dignum, and Yah-Hua Tan. Towards a value-sensitive system to support agents in norm fulfillment and enforcement. TR, Delft University, June 2010.

Loris Penserini, Anna Perini, Angelo Susi, and John Mylopoulos. High variability design for software agents: Extending Tropos. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 2(4):16:1–16:27, November 2007.

Chetan Shankar, Vanish Talwar, Subu Iyer, Yuan Chen, Dejan Milojicić, and Roy Campbell. Specification-enhanced policies for automated management of changes in IT systems. In *Proceedings of the 20th Large Installation System Administration Conference*, pages 103–118, Washington, 2006. Usenix.

Munindar P. Singh. An ontology for commitments in multiagent systems: Toward a unification of normative concepts. *Artificial Intelligence and Law*, 7(1):97–113, March 1999.

Munindar P. Singh. Semantical considerations on dialectical and practical commitments. In *Proceedings of the 23rd Conference on Artificial Intelligence (AAAI)*, pages 176–181, Menlo Park, July 2008. AAAI Press.

Munindar P. Singh, Amit K. Chopra, and Nirmit Desai. Commitment-based service-oriented architecture. *IEEE Computer*, 42(11):72–79, November 2009.

Pankaj R. Telang and Munindar P. Singh. Specifying and verifying cross-organizational business models: An agent-oriented approach. TR 12, North Carolina State University, May 2010.

Nick A. M. Tinnemeier, Mehdi Dastani, and John-Jules Ch. Meyer. Programming norm change. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 957–964, Toronto, 2010. IFAAMAS.

Wamberto Vasconcelos, Martin J. Kollingbaum, and Timothy J. Norman. Resolving conflict and inconsistency in norm-regulated virtual organizations. In *Proceedings of the 6th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 632–639, Honolulu, May 2007. IFAAMAS.