

The Uses of Norms

Munindar P. Singh¹, Matthew Arrott², Tina Balke³, Amit Chopra⁴,
Rob Christiaan⁵, Stephen Cranefield⁶, Frank Dignum⁷,
Davide Eynard⁸, Emilia Farcas², Nicoletta Fornara⁸,
Fabien Gandon⁹, Guido Governatori¹⁰, Hoa Khanh Dam¹¹,
Joris Hulstijn¹², Ingolf Krueger², Ho-Pun Lam¹⁰,
Michael Meisinger², Pablo Noriega¹³,
Bastin Tony Roy Savarimuthu⁶, Kartik Tadanki¹⁴,
Harko Verhagen¹⁵, and Serena Villata¹⁶

- 1 North Carolina State University, USA
- 2 University of California at San Diego, USA
- 3 University of Surrey, UK
- 4 University of Trento, Italy
- 5 Vrije Universiteit, The Netherlands
- 6 University of Otago, New Zealand
- 7 Utrecht University, The Netherlands
- 8 Università della Svizzera italiana, Switzerland
- 9 INRIA Sophia Antipolis, France
- 10 NICTA, Australia
- 11 University of Wollongong, Australia
- 12 Delft University of Technology, The Netherlands
- 13 IIIA-CSIC, Spain
- 14 Deutsche Bank, USA
- 15 Stockholm University, Sweden
- 16 University of Luxembourg, Luxembourg

Abstract

This chapter presents a variety of applications of norms. These applications include governance in sociotechnical systems, data licensing and data collection, understanding software development teams, requirements engineering, assurance, natural resource allocation, wireless grids, autonomous vehicles, serious games, and virtual worlds.

1998 ACM Subject Classification I.2.11 Distributed Artificial Intelligence

Keywords and phrases Norms, MAS, Governance, Requirements engineering

Digital Object Identifier 10.4230/DFU.Vol4.12111.191

1 Introduction

This chapter presents a compendium of several uses of norms. Each writeup follows a more or less fixed pattern where it first brings out of the application scenario and its importance; second the suitability of a normative model for the problem at hand; third some technical challenges for norms brought to the forefront by that scenario; and fourth a description of its status and some speculation about its prospects. The common notion of norms in these works is that norms represent a standard of correct behavior and correspond loosely to the family of concepts that includes commitments, obligations, and prohibitions.

© M.P. Singh, M. Arrott, T. Balke, A. Chopra, R. Christiaan, S. Cranefield, F. Dignum, D. Eynard, E. Farcas, N. Fornara, F. Gandon, G. Governatori, H. Khanh Dam, J. Hulstijn, I. Krueger, H.-P. Lam, M. Meisinger, P. Noriega, B.T.R. Savarimuthu, K. Tadanki, H. Verhagen, and S. Villata; licensed under Creative Commons License CC-BY



Normative Multi-Agent Systems. *Dagstuhl Follow-Ups, Vol. 4*. ISBN 978-3-939897-51-4.

Editors: Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leendert W.N. van der Torre; pp. 191–229



DAGSTUHL Dagstuhl Publishing
FOLLOW-UPS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Germany

Note that the use of norms reported here are research efforts, in early stages of development. They are inspired by real-life applications and mostly go to demonstrate the potential value of norms in the field. We hope that a collection of these uses will provide some inspiration to researchers in norms and potentially a basis for usage scenarios that might be used in further study or evaluation.

The uses of norms presented next are organized as follows. The contributions by Singh et al., Villata and Gandon, and Fornara and Eynard all deal with norms as they relate to policies in distributed systems. The contributions by Savarimuthu and Dam, Christiaanse and Hulstijn, and Chopra relate norms to software engineering showing how to mine norms, how to map norms to an architecture, and how to base requirements on norms. The contributions by Noriega, Balke, and Governatori and Lam apply norms to modeling scenarios placing agents in real-life settings such as sharing water, wireless connectivity, and physical space (by unmanned vehicles). The contributions by Dignum and Cranefield and Verhagen discuss norms in virtual environments, such as for gaming and virtual worlds.

2 Singh et al.: Governance in Sociotechnical Systems^{1,2}

We address the challenge of administering sociotechnical systems, which inherently involve a combination of software systems, people, and organizations. Such systems have a variety of stakeholders, each in essence autonomous. Traditional architectural approaches assume that stakeholder concerns are fixed in advance and addressed out-of-band with respect to the system. In contrast, the sociotechnical systems of interest have long lifetimes with changing stakeholders and needs. We propose addressing stakeholders' needs during the operation of the system, thus supporting flexibility despite change. Our approach is based on normative relationships or norms among stakeholders; the norms are streamlined through a formal notion of organizations. We demonstrate our approach on a large sociotechnical system we are building as part of the Ocean Observatories Initiative.

We define governance as the administration of collaborations among autonomous and heterogeneous peers by themselves. Because each participant is independently implemented and operated, governance must be captured in terms of high-level normative relationships that characterize the expectations that each participant may place on the others. Norms are standards of correctness and may be aggregated into contracts.

Further, our interest lies in sociotechnical systems, which arise in a variety of domains such as scientific investigation, health care and public safety, defense and national security, global business and finance. We define a sociotechnical system as a system-of-systems (SoS). Its value and complexity arise from the combination of capabilities provided by their (heterogeneous) constituent systems.

2.1 Application Scenario

An excellent example of a sociotechnical system is the one being built as part of the NSF-funded Ocean Observatories Initiative (OOI), a thirty-year \$400 million project [54]. OOI provides novel capabilities for data acquisition, distribution, modeling, planning and control

¹ The authors of this section are Munindar P. Singh, Emilia Farcas, Kartik Tadanki, Ingolf Krueger, Matthew Arrott, and Michael Meisinger.

² Acknowledgments: This work was partially supported by the OOI Cyberinfrastructure program, which is funded by NSF contract OCE-0418967 with the Consortium for Ocean Leadership via the Joint Oceanographic Institutions.

of oceanographic experiments, with the main goal of supporting long-term oceanographic and climate research. The OOI stakeholders include ocean scientists, resource providers, technicians, operators, policy makers, application developers, and the general public.

The OOI presents system requirements that involve supporting thousands of stakeholders, tens of thousands of physical resources such as autonomous underwater vehicles (AUVs), and potentially millions of virtual resources such as datasets. The resources are independently owned and operated. Moreover, OOI facilitates virtual collaborations created on demand to share access to ocean observatory resources, including instruments, networks, computing, storage, databases, and workflows.

The stakeholders have complex needs and objectives in this setting. These include how they can benefit from individual and shared resources, monitor the health of such resources, control their functioning, and administer their usage. Additional considerations include entering into scientific collaborations, managing resource conflicts, achieving and enforcing accountability of colleagues and staff. Importantly, the specifics can differ for each stakeholder individual or organization, and are influenced by whom the stakeholder interacts with. Such concerns are not readily enumerated during design, especially when dealing with long-lived sociotechnical systems. Not treating them would waste opportunities for improving the social and scientific value of oceanographic research. Indeed, this is the current situation and its weakness has motivated the creation of the OOI.

Consider the following important OOI use cases for governance, which highlight the autonomy of the participants and the business relationships among them.

Collaboration. The stakeholders of OOI include research scientists or investigators as well as educators from middle and high schools. Consider a situation where a teacher in a school near Chesapeake Bay would like to present some information about the students' local environment. This data could be as simple as acidity levels in the Bay. Clearly, the teacher would need to access data that a researcher with the appropriate sensors would have gathered. The researcher may have entirely different interests from the teacher; for example, she may be interested in multiyear trends. To this end, the researcher would participate in a resource-sharing community where she would have shared the data streams being generated by her sensors. The teacher would also authenticate with OOI, discover the appropriate community, and enroll in it. Therein the teacher would discover the desirable data stream and extract the information he needs.

Affiliation. The stakeholders of OOI include not only investigators but also research institutions and laboratories. Two institutions may decide to share their resources on a reciprocal basis, and thus enter into a suitable contract, viewed as an aggregation of normative relationships. A researcher at one of those institutions would be able to discover with which institutions her institution is affiliated. She would then be able to access an affiliate institution and further discover a research laboratory based at the second institution. Lastly, she would be able to take advantage of resources belonging to the laboratory. Either institution may decide to end the affiliation but even its exit could be subject to the existing norms, e.g., that ongoing experiments not be aborted.

Existing IT or SOA approaches treat governance primarily as a slow, ponderous, out-of-band activity, whereby stakeholders negotiate their concerns only during the design of a system, not during its operation. Such approaches are ill-suited for specific concerns arising during collaboration. In contrast, automation is essential to improve the quality (such as the precision, timeliness, productivity, and comprehensibility) and scale of governance. For this reason, we approach governance as a central endeavor carried out in-band in a sociotechnical system.

We propose a novel approach that gives first-class status to stakeholders as principals of the resulting system and to their concerns expressed via norms and policies. A policy is what determines a principal's interactions, which may or may not comply with an applicable norm. A norm itself could be operationalized via rules applied by any principal to check if the norm is being respected. Our approach is compatible with traditional approaches, and thus helps leverage existing tools and experience where appropriate.

2.2 Suitability of a Normative Model

Our conceptual model is centered on the concept of principal. Principals include users, resources, and organizations (termed Orgs in our model). Each principal possesses a unique identity within OOI. Governance is achieved through interactions among principals: realized through their local policies and constrained by their normative relationships with each other. Each principal may adopt roles in one or more Orgs. In essence, each role corresponds to a set of norms between a principal who adopts it and the Org (also a principal). The norms constrain the subsequent interactions between two principals present in the same Org. In general, a normative relationship may arise as the result of a successful negotiation or may be implicitly imposed due to the parties adopting complementary roles in the same Org. Each norm references an Org that serves as its context [64].

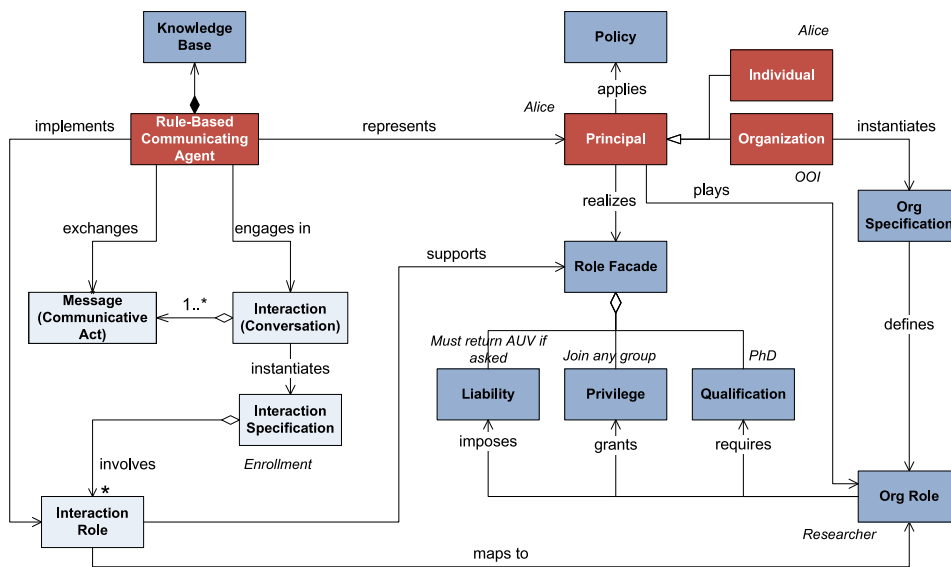
Each principal is represented in the computational system by an agent. The principal, e.g., a human, exists outside the computational system; the agent is all that exists computationally: there is no principal. Each principal's agent supports bookkeeping regarding the norms in which the principal participates. The agent helps determine (1) if its principal is complying with the applicable norms and (2) if others with whom it deals are complying as well. The agent continually tracks the state of each norm by updating the state for each observable action, such as sending or receiving a message (including making an observation of the environment).

Orgs serve multiple purposes in our architecture, specifically providing a backdrop for norms, a locus for identity, and a venue to share resources. Each Org defines the rules for adopting each of its roles. Joining an Org means adopting at least one role in that Org. Adopting a role means accepting the rules of the Org for that role. Thus, we understand enrollment in an Org as involving the creation of one or more norms and treat the subsequent interactions of the participants as arising within the scope of the given Org. An example of enrollment is someone joining eBay; an example of additional norms is when two eBay members carry out a transaction. The members are subject to eBay's rules such as accepting the price announced by eBay at the end of an auction.

The above interactions, including enrollment, inherently involve the creation and manipulation of normative relationships and can potentially be operationalized in multiple ways. For example, for enrollment, (1) the prospective enrollee may request membership; (2) the prospective enroller may invite the enrollee; (3) a third party may introduce the enrollee and enroller; or (4) a third party may require the enrollee and enroller to carry out the enrollment. Such flexibility facilitates separating stakeholder concerns from each other and from the implementation, thereby improving how stakeholders comprehend the architecture and enhancing the confidence they can place in it.

Each principal applies its own policies to determine what actions to take. Thus, a principal can decide whether to adopt a role in an Org and, conversely, the Org can decide whether to admit a principal to a role. Each principal's decisions are subject to constraints such as the requirements imposed by the roles that it has adopted.

The model from Figure 1 relates an Org specification with a set of roles. Each norm



■ **Figure 1** Overview of our governance model.

involves two or more Org roles. In effect, each Org role partitions its view of the relevant parts of the set of norms that characterize the Org. We model the role-relevant parts of each Org specification as consisting of three components, assembled into a *role façade* [65], which helps us provide a normative basis for roles:

Qualifications, which specify eligibility requirements for a principal to take on a role. For example, a professor must have a university identity to join a PhD committee.

Privileges, which specify what authorizations and powers a principal gains in adopting the role. A professor as committee member is authorized to review the student’s lab notebook and empowered to determine if the student passes.

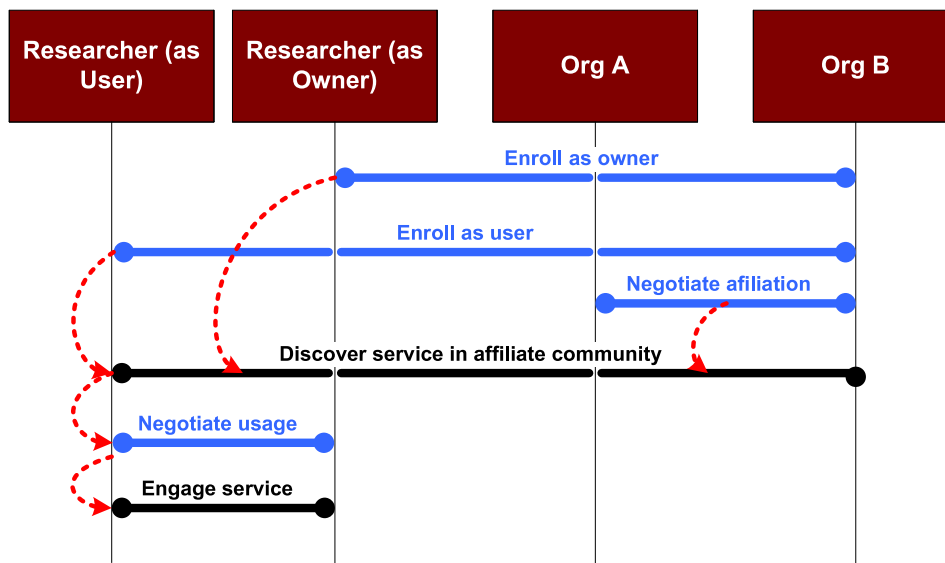
Liabilities, which specify what a principal becomes subject to in adopting the role. A committee member must attend a PhD defense.

Each principal applies its policies, to determine whether to enroll, potentially to take advantage of its privileges, and ideally to satisfy its liabilities. In general, one cannot guarantee compliance in a sociotechnical system, but we address compliance in two main ways:

- Conservatively, ensure that actions taken by a principal are compliant. This can work where the principal is not autonomous and heterogeneous. We can subject the principal to a guard that allows only the policy-compliant (attempted) actions to proceed.
- Optimistically, recognize that a principal may proceed as it would, but detect and handle noncompliant behavior. We can accomplish detection either by introducing a monitor in the architecture or through the principals monitoring each other. We can respond to detected violations by escalating them to the nearest scoping Org.

2.2.1 Architectural Case Study

OOI enables its primary stakeholders (scientists) the opportunity to seamlessly collaborate with other scientists across institutions, projects, and disciplines and to access and compose resources as needed. To address complexity, mitigate risks, and accommodate requirement



■ **Figure 2** Governance of resource sharing across affiliated Orgs.

changes, OOI uses a spiral development process, a variant of the Incremental Commitment Model (ICM) [7]. ICM includes iterative development cycles focusing on incremental refinement of system definition and stakeholder commitment and satisfaction. We have adopted selected architectural views from the Department of Defense Architecture Framework (DoDAF) [23] to document the OOI architecture.

OOI resources are distributed both physically and virtually among different organizations, each with its own policies for resource access and data delivery or consumption. We model OOI itself as an Org that is the highest scope for all OOI users and their interactions. The OOI Org serves as the root Org for the identities for all OOI principals and helps monitor and enforce the applicable norms among them.

Figure 2 illustrates the use case where two research organizations (each an Org) form an affiliation relationship with each other. Both Org A and Org B are what we term resource-sharing Orgs, and define two main roles: owner (of a resource) and user (of a resource). Each principal who adopts owner can contribute its resources to the Org, so those resources can be discovered by any principal who adopts the role user. In addition, to form affiliations, each Org supports additional roles capturing the affiliation relationship. These roles are `AFFILIATEORG` to capture the clauses for the affiliated community, and `AFFILIATEMEMBER` to capture the clauses for the members of the affiliated community, which could have weaker rights than the Org's own members.

The affiliation relationship between Orgs propagates to their respective members. As a result, a member of Org A can discover services offered by members of Org B. Once it has discovered such services, it may negotiate with and engage them as appropriate.

Our notation is similar to message sequence charts in terms of having a swim lane for each principal. However, instead of messages, we use horizontal lines to show joint (governance) actions that create or modify relationships among the (two or more) parties whose lifelines they connect. Any temporal order requirements are captured via the dashed arrows that connect some pairs of the horizontal lines. In general, the parties would realize a governance interaction such as enrollment by exchanging multiple messages, e.g., propose, counterpropose, accept, or reject.

2.2.2 Benefits of Norms and Allied Constructs

We attribute the benefits of our architectural treatment of governance to the following main principles that it respects.

- Centrality of organizations in modeling communities; modeling the OOI itself as a runtime entity or agent; specifying rules of encounter; monitoring norms; sanctioning violators.
- Autonomy of stakeholders; representing stakeholders as agents that apply autonomous policies and are subject only to the applicable organizational rules of encounter.
- Emphasizing normative relationships and modeling them explicitly to make them easy to inspect, share, and manipulate; accommodating openness of the system by recognizing that autonomous parties may violate rules of encounter and, thus, may need enforcement ex post facto, such as via sanctions.

In the OOI, policies specified in the norms within an Org govern the circumstances under which resources can be discovered, accessed, and utilized. In the example, we considered two classes of stakeholder roles: user and owner of a resource. The user is concerned with accessing a resource, without facing any hidden obligations. The owner is concerned with providing resources (with spare capacity) to expand the impact of the resources on others and to treat the resources as a basis for negotiating value in exchange.

Our governance approach addresses stakeholder (user and owner) concerns as follows:

- The resource sharing community provides access to needed resources and clarifies what restrictions are imposed on the user as a result; guarantees that the user will not be subject to the whims of the resource owner once the user begins an allowed interaction with a resource.
- The affiliation community expands resource sharing to external organizations and provides access to remote resources on a reciprocal basis.
 - The user and owner can negotiate detailed terms as norms that go beyond the basic norms imposed by being members of a community.
 - The user and owner can accommodate changing needs, renegotiate the set of norms, or may decline to continue to participate.
 - In deployment, policies are separated from the business functionality, allowing them to be changed easily over time according to stakeholder needs.

2.3 Challenges for Norms

The above exercise makes apparent some important challenge for norms.

An engineering challenge is to harmonize norms with approaches to software architecture and methodology, so that norms can be naturally incorporated into practice.

- The OOI effort builds upon methodologies such as Model-Driven Architecture (MDA) and goal-oriented requirements engineering, and goes beyond them by providing a systematic treatment of governance from the modeling level to implementation. We understand sociotechnical systems to be ultra-large-scale systems (ULSSIS) because they inherently involve multiple stakeholders use the system, contribute resources, form virtual communities, and determine the rules that govern their interactions [27]. Our approach applies naturally to ULSSIS because it dynamically captures stakeholder concerns by (1) defining patterns of interaction based on Orgs; (2) enabling stakeholders to select roles in desirable Orgs; and (3) supporting the specification and application

of policies potentially customized to each stakeholder. How can ULSSIS incorporate norms in general?

- Addressing the inherent complexities of sociotechnical systems involves going beyond traditional Service-Oriented Architecture (SOA), specifically in accommodating multiple ownership domains [26]. Following Singh et al. [66], we view services as real-life services, not computational objects. We identify principals as the participants in service engagements described in terms of the normative relationships, and define patterns on the creation, propagation, and manipulation of such norms. Our approach coheres with recent advances in goal-oriented methodologies, specifically Tropos [13]. Tropos emphasizes the goals of the actors whereas we emphasize their norms and would capture their goals both in what norms they enter and how they choose to act based on those norms. How may we expand the above-mentioned normative patterns and place them within a comprehensive methodology for developing normative systems?

Theoretical challenges are highlighted by this effort.

- To use norms for specifying sociotechnical systems, we would need algorithms that help validate normative specifications, so as to identify conflicts early in development.
- We need techniques to determine whether a principal complies with applicable norms, especially using information that other principals can access.

2.4 Status and Prospects

The OOI development effort is underway. At a conceptual level, normative thinking has guided the software architecture right from the beginning of the OOI. The early part of its development effort has dealt with providing the software infrastructure to realize scientific collaboration. Normative concepts are now being introduced into the development.

We apply the Rich Services architecture [3], a type of SOA that provides decoupling between concerns and allows for hierarchical service composition. Rich Services is a logical architecture that can be mapped to possible deployments such as Enterprise Service Buses or multi-agent systems. For the affiliation use case in OOI, each Org and the User itself are modeled as a Rich Service within the root OOI Rich Service. Infrastructure Services include identity and policy management, logging of all conversations and actions, as well as repositories for the community specification and the norms already established with other parties. Each Rich Service has its local policies and a local representation of the norms in which it participates.

Rich Services provide a clear separation between the business logic and its external constraints, supporting their composition at the infrastructure level through specialized interceptors. When requirements change during the lifetime of the system, they often affect policies and not core services; therefore, the decoupling between them allows to update Infrastructure Services without modifying the services that are composed.

A specific implementation of governance may be realized via a rule-based communicating agent, which maintains the applicable rules and information about the state of the world and any ongoing interactions in a knowledge base. Each agent represents one principal—thus the approach is decentralized. An agent represents a principal in an Org as a locus of autonomy and identity. We have prototyped such an agent using an agent platform (specifically, Java Agent Development Framework (JADE)) and a rule engine (specifically, Java Expert System Shell (JESS)). An agent platform provides a container for the execution of agents, communication infrastructure to enable agent communication, and directory services. A rule

engine maintains and applies the facts and rules for an agent and, thus, enables reasoning and reaction.

Rules lead to a simple implementation where an agent loads the rules corresponding to each role that it adopts. The rules are generated from the norm specifications for which we developed a domain Specification Language; its constructs are based on properties and predicates.

3 Villata and Gandon: Data Licensing in the Web of Data

3.1 Application Scenario

A common assumption in the Web is that the publicly available data, e.g., photos, blog posts, videos, can be reused without restrictions. However, this is not always true, even when the licensing terms are not specified. Consuming Linked Open Data includes the fact that the data consumer has to know the terms under which the data is released. The licensing terms in the Web of Data are specified by means of machine-readable expressions, such as additional triples added to the RDF documents stating the license under which the data is available. We highlight the future trends in data licensing and the possible connections with normative multi-agent systems.

The JISC Linked Data Horizon Scan³ states about the link between Linked Data and Open Data: “Linked Data may be Open, and Open Data may be Linked, but it is equally possible for Linked Data to carry licensing or other restrictions that prevent it being considered Open, or for Open Data to be made available in ways that do not respect all of Berners-Lee’s rules for Linking.”⁴ Licensing of data needs to be explicit to avoid any ambiguity in terms of use and reuse for the data consumers. The absence of clarity for data consumers about the data terms of reuse does not encourage the reuse of that data. There are many differences worldwide related to the copyright of data, and not all data is copyrightable. Some of the most popular licenses on the Web include Creative Commons⁵, GNU Free Documentation License⁶, Open Data Commons⁷, Science Commons Database Protocol⁸, and Freedom to Research: Keeping Scientific Data Open, Accessible, and Interoperable⁹.

The Linked Data cloud¹⁰ presents various examples of use of different licensing terms. Table 1 shows the absence of a common approach for data licensing in the Web of Data. This is one of the main problems in the context of licensing for the Web of Data [6].

Heath and Bizer [40] underline that “the absence of clarity for data consumers about the terms under which they can reuse a particular dataset, and the absence of common guidelines for data licensing, are likely to hinder use and reuse of data”. Therefore, all Linked Data on the Web should include explicit license, or waiver statements, as discussed also by Miller et al. [51]. In this paper, we briefly introduce the licenses schemas proposed in the Web of Data, then we describe the open challenges in data licensing for the Web of Data. We conclude by suggesting some further challenge bridging the gap between the Semantic Web and Normative Multi-Agent Systems (NorMAS).

³ <http://linkeddata.jiscpress.org/>

⁴ http://wiki.cetis.ac.uk/images/1/1a/The_Semantic_Web.pdf

⁵ <http://creativecommons.org/>

⁶ <http://www.gnu.org/copyleft/fdl.html>

⁷ <http://www.opendatacommons.org/licenses/>

⁸ <http://sciencecommons.org/resources/faq/database-protocol>

⁹ <http://sciencecommons.org/wp-content/uploads/freedom-to-research.pdf>

¹⁰ <http://richard.cyganiak.de/2007/10/lod/>

■ **Table 1** Examples from the Linked Data cloud and their licenses.

	CC	ODC	Country	GNU	Commercial	No licenses
MusicBrainz				X		
Guardian Data Store						X
OpenStreetmap	X					
BBC Backstage					X	
DBpedia	X			X		
legislation.gov.uk			X			

3.2 Suitability of a Normative Model

The applications that consume data from the Web must be able to access explicit specifications of the terms under which data can be reused and republished. The availability of appropriate frameworks for publishing such specifications is an essential requirement in encouraging data owners to participate in the Web of Data, and in providing assurances to data consumers that they are not infringing the rights of others by using data in a certain way. Initiatives such as the Creative Commons have provided a framework for open licensing of creative works, underpinned by the notion of copyright. However, as discussed by Miller et al. [51], copyright law is not applicable to all data because not all data are creative works, which from a legal perspective is also treated differently across jurisdictions. Therefore frameworks such as the Open Data Commons can be adopted to state the revise conditions.

The most diffused machine-readable licensing languages are Creative Commons, Open Data Commons, and MPEG-21 REL. The Creative Commons Rights Expression Language (ccREL) [1] is the standard recommended by Creative Commons (CC) for machine-readable expression of copyright licensing terms and related information. Miller et al. [51, 69] propose the Open Data Commons waivers and licenses¹¹ that try to eliminate or fully license any rights that cover databases and data. The Waiver vocabulary¹² defines properties to use when describing waivers of rights over data and content. A waiver is the voluntary relinquishment or surrender of some known right or privilege. As discussed by Heath and Bizer [40], “licenses and waivers represent two sides of the same coin: licenses grant others rights to reuse something and generally attach conditions to this reuse, while waivers enable the owner to explicitly waive their rights to a dataset”. In MPEG-21, a Rights Expression Language (REL)¹³ is a machine-readable language that can declare rights and permissions using the terms as defined in the Rights Data Dictionary¹⁴. Two further vocabularies which can be used also to define the licensing terms of the data on the Web are the Description of a Project vocabulary (DOAP)¹⁵, and the Ontology Metadata vocabulary (OMV)¹⁶ [55]. The former is an RDF/XML vocabulary to describe software projects, in particular open-source. It defines a property `doap:license` for defining the licensing terms of the project. The latter, instead, describes a particular representation of an ontology, and it captures the key aspects of the

¹¹<http://opendatacommons.org/licenses/>

¹²<http://vocab.org/waiver/terms/.html>

¹³<http://mpeg.chiariglione.org/standards/mpeg-21/mpeg-21.htm>

¹⁴<http://iso21000-6.net/>

¹⁵<http://usefulinc.com/ns/doap>

¹⁶<http://omv2.sourceforge.net/index.html>

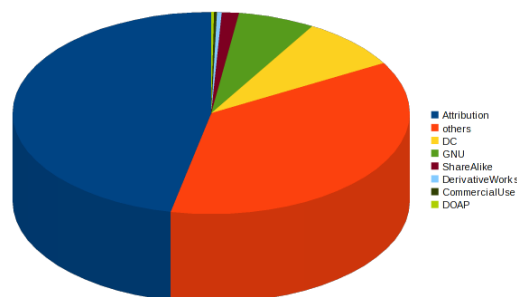
ontology metadata information, e.g., provenance, availability, statistics. OMV defines the property `omv:hasLicense` which provides the underlying license model. Moreover, OMV introduces a class `omv:LicenseModel` which describes the usage conditions of an ontology. Finally, we mention also the Dublin Core license document class `dc:LicenseDocument`¹⁷ which provides the legal document giving official permission to do something with the resource and the license property `dc:license`¹⁸. A mapping among the concepts used in these schemas is provided in Figure 3.

	Conditions of release	Rights	Law
ccREL	cc:Reproduction cc:Distribution cc:DerivativeWorks cc:CommercialUse	cc:permits cc:prohibits	cc:legalCode cc:jurisdiction
MPEG-21 REL	Terms, Conditions, Obligations	Issue, Revoke, Obtain	
Waiver	Declaration		Norms, Waivers
OMV	omv:LicenseModel		
DublinCore			dc:LicenseDocument
DOAP	doap:license		

■ **Figure 3** Mapping among the different licensing languages.

Licenses such as Creative Commons and Open Data Commons have common features, but also differ from each other. The requirement to mention the author (BY) seems to be one of the best shared features, since it is absent only in the license PDDL 1.0¹⁹—which in essence, exempts any obligation. Most legal frameworks allow commercial use: that is, they make it possible for re-users to sell public data without transforming or enriching them. Other features are adopted by some legal framework, and not by others.

Figure 4 visualizes the results of a search on Watson²⁰ of the licensing terms we have presented. The results show that the Creative Commons Attribution license is the most used one among the Creative Commons licenses. The other well diffused way to express licenses adopts the Dublin Core `license` property. These results make even clearer the lack of a uniform approach to data licensing.



■ **Figure 4** The use of licenses in the Web (Watson search).

¹⁷ <http://dublincore.org/documents/dcmi-terms/#classes-LicenseDocument>

¹⁸ <http://dublincore.org/documents/dcmi-terms/#terms-license>

¹⁹ <http://www.opendatacommons.org/licenses/pddl/>

²⁰ <http://watson.kmi.open.ac.uk/WatsonWUI/>

3.3 Challenges for Norms

The challenge about treating licenses in the Web of Data can be decomposed in a number of sub-tasks as follows:

1. Selection of n license schemas;
2. Alignment of these n license schemas;
3. Returning the requested data together with the license under which it is released.

Each of the points above presents a challenge in the research area of the Semantic Web. First, the selection of the n license schemas is a complex task since the vocabularies are not all indexed and the licensing terms of some data may be expressed by various vocabularies, and not only by ad-hoc vocabularies. Second, these vocabularies may define concepts and properties about licenses which have been already defined by other vocabularies. At this point, an alignment step is necessary to establish which are the “equivalent” licensing terms defined in the various vocabularies. Third, the aim of a license model for the Web of Data is to provide, after a user query, the data resulting from the query together with the licensing terms under which the data is available.

```
<?xml version="1.0"?>
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
<head>
...
<link href="metadataLicenses.rdf"/>
</head>
...
</sparql>
```

■ **Figure 5** A sample of the SPARQL query results XML format providing information about the licenses on the data.

A possible solution concerning the third point would be to adopt the standard SPARQL query results XML format²¹, and to introduce thanks to the `<link>` element the information about the license under which the data returned by the SPARQL query is released. The problem which arises at this point is that we cannot express more than one license on the data. Thus we should choose the more restrictive license among the set of licenses constraining the data returned by the query. If this solution is adopted, this leads to a lack of satisfaction of the less restrictive licensing terms which allow, for instance, a free distribution and reuse of the data. The possible alternative consists in changing the SPARQL query results XML format in order to associate to different sets of data different licenses. This would allow a better representation of the licensing terms, but this includes also the definition of a new standard for the SPARQL query results XML format.

The Linked Data initiative aims at improving data publication on the Web, thereby creating a Web of Data: an interconnected, distributed, global data space. The Web of Data enables people to share structured data on the Web as easily as they can currently share documents on the Web of Documents (WWW). The basic assumption behind the Web of Data is that the value and usefulness of data increases with the amount of interlinking with other data. The emerging Web of Data includes datasets as extensive and diverse as DBpedia, and DBLP. The availability of this global data space creates new opportunities for the exploitation of techniques in relation with knowledge representation and intelligent agents. In particular,

²¹ <http://www.w3.org/TR/rdf-sparql-XMLres/>

a new challenge in this view consists in having intelligent agents exploiting the Web of Data. This is a challenge which involves the NorMAS community as well concerning for instance the licenses issue. In particular, normative multi-agent systems may be adopted to support the alignment phase among the different licenses schemas. An open issue in ontology matching is how to have a consistent alignment. For instance, Santos and Euzenat propose a model based on argumentation theory [24]. An idea would be to use techniques developed in the field of normative multi-agent systems to check the consistency of an alignment of licensing schemas, following the approach proposed by Fornara and Colombetti for obligations [33]. Moreover, the reasoning techniques developed in the NorMAS community can be used to reason over the Web of Data in order to infer, starting from the links among the different schemas and datasets, further normative constraints among the datasets and further links among the (licenses) schemas.

4 Fornara and Eynard: Web-based Data Collection using Norms and Semantic Web Technologies²²

4.1 Application Scenario

Web-based data collection is becoming more and more important for many social science fields like economy, sociology, social media, market research, and psychology. This fact is clearly highlighted for example by the WEBDATANET COST Action²³. Web-based data collection is not restricted to Web surveys, but it also includes *non-reactive data*, collected by means of log files analysis, data mining, text mining, and data crawling from heterogeneous Web sources (i.e., blogs, social networks, consumer reviews, folksonomies, and search results). These approaches require new techniques, algorithms, and tools whose application to the problem of Web-based data collection represents a crucial multidisciplinary problem, involving both social scientists and computer engineers.

The design of the tools for collecting non-reactive Web data is strongly influenced by the perspectives, the constraints, and the desires of the different actors involved in the creation, publication, collection, storage and manipulation of those data. In particular it is crucial to take into account: (i) the point of view of those who will analyse these data, whose requirements are data validity, reliability, quality and, as discussed in [43], the need to be able to guarantee integrity, transparency, and to respond to privacy and confidentiality wishes of the users; (ii) the perspective of data providers (i.e., data publishers and companies or single users that produce those data), who consider essential the possibility to constrain the access to their data, together with the possibility of being aware of how they will be stored, used, and combined. Guidelines for professional ethics and practices can be found, for example, in [30]. Some of them express norms, that is, obligations (“we shall”) and prohibitions (“we shall not”) on how data can be obtained, stored, used, disclosed, and so on. Examples policies stating how the resources available on a social software like Facebook can be used for automatic data collection can be found at http://www.facebook.com/apps/site_scraping_tos_terms.php. Other examples of constraints on how data –and as a particular case Linked Data– can be accessed and reused by consumers are represented by popular licenses like Creative Commons and GNU Free Documentation License, plus many others referred and discussed in detail in the previous section.

²² Acknowledgments: Dr. Fornara is supported by the Hasler Foundation project nr. 11115-KG and by the SER project nr. C08.0114 within the COST Action IC0801 Agreement Technologies.

²³ <http://webdatanet.cbs.dk/>

4.2 Suitability of a Semantic Normative Model

Very often these guidelines, norms, policies, and constraints are expressed in natural language (e.g. English). Therefore, in order to comply with them researchers need to read, understand, and finally apply them. The problem of applying these norms is complicated by the fact that top-down policies (provided by data publishers and users) also need to be integrated by additional (bottom-up) constraints, provided by data collectors and declaring what, within a set of Web sources, can or cannot be accessed in the context of a given research. Moreover, the licenses that regulate access to datasets are often different from one another. Finally, a relevant aspect of the problem is that this application scenario involves different actors (social researchers, data collectors, data publishers, and data producers) having different interests. Therefore there is always the possibility that some of those norms are not fulfilled, thus it is important to implement mechanisms for their monitoring and enforcement.

When big amounts of data are treated for automatic extraction by means of specialized software (i.e., both site-specific and generic crawlers, or survey software), being compliant with those norms becomes very difficult. This problem even worsens when data to be used in one survey are collected from many sources by different people and organizations, with different techniques, and in different instants of time. *Formal models* of guideline, policies, and norms may be used to guide specialized software with a specific focus on (i) guaranteeing that activities performed during data collection are compliant with given norms, (ii) guaranteeing that the way those data are stored and re-used is compliant with given policies and guidelines, and (iii) providing a way to keep track on how data is accessed and used and issue warnings when a potential misbehavior, with respect to a set of norms, occurs.

Studies on normative multi-agent systems (NorMAS) and on automatic extraction and representation of knowledge from semi-structured and unstructured data may be crucial to tackle those problems. Studies on the formalization of obligations, permissions, and prohibitions in particular, and of agreements²⁴ and contracts in general [21] may be used to express norms or licenses (e.g. in the Web of Data as discussed in Section 3) for accessing, using, and storing data. Moreover if those norms are expressed and manipulated using Semantic Web Technologies, [41] like OWL as proposed in [32, 31] and in [63], it will be possible to easily merge sets of norms coming from different sources (by merging their OWL ontologies) and use techniques for OWL ontologies alignment (as discussed in the previous section) for solving possible differences and for checking the consistency of the resulting set. Similarly techniques for the monitoring of those semantic norms may be used to develop software able to issue warning when violations occurs. Studies on mechanisms for developing software that are compliant with a given set of norms [35] or on mechanisms for developing agents able to plan their actions on the basis of certain norms [19] can be used to develop software agents able to reason on semantic norms.

Those formal representation of norms can be viewed as formal data attached to semi-structured and unstructured data that is published on the Web on a daily basis. The problem of structuring knowledge can be addressed in two different ways. On the one hand, knowledge representation standards and techniques could be adopted to provide knowledge in a form that is directly consumable by machines. These techniques mainly relate to the use of Semantic Web Technologies [41] like RDF and OWL for knowledge representation. On the other hand, given the amount of data already provided on the Web as semi-structured or unstructured text, the study of tools and techniques for automatic knowledge extraction from these sources is becoming more and more important.

²⁴<http://www.agreement-technologies.eu/>

4.3 Challenges for Norms

The main challenges for norms formalization and monitoring related to the proposed application scenario are:

- The development of models and techniques to express Web-based data collection guidelines, rules and policies at different level of abstraction, that is, representing high-level general guidelines and transforming them in low-level concrete descriptions of allowed and disallowed actions;
- Given that those rules need to be automatically processed by software tools, studying how to formalize them using decidable logics, like for example the Description Logics (DLs) that are at the basis of the Ontology Web Language OWL;
- The design of systems able to plan their actions on the basis of a set of semantic norms;
- The design of systems able to keep track of how data are accessed and used in an intrinsically partial observable world like the Internet is and raise warnings when inconsistencies among the expected behavior and the actual behavior arise.

4.4 Status and Prospects

The idea of applying semantic norms to the formalization of guidelines for Web data collection and of norms and policies for regulating how, where, and from who those data may be collected is underway. We plan first of all to study models and techniques to express Web-based data collection guidelines, norms and policies at different levels of abstraction, from very high-level general guidelines to low-level concrete descriptions of allowed and disallowed actions. To this purpose we plan to extend our model of obligations [31] and norms [32], expressed using Semantic Web technologies. Based on such models and techniques we plan to design and implement a demonstrative system able to monitor processes of Web data collection, in order to guarantee that data collection guidelines, norms and policies are actually satisfied.

5 Savarimuthu and Dam: Norms in Open Source Software Repositories

5.1 Application Scenario

Extracting valuable information from Open Source Software (OSS) repositories is gaining popularity since huge volume of data is available for free. Open source projects such as Linux and Andorid OS are used by millions and developed by hundreds of developers over extended period of time have produced rich, extensive, and easy-to-access data from which valuable information can be *mined*. We believe open source repositories present an interesting context for the extraction and the study of norms. The reasons are manifold. Firstly, there are a substantial number of OSS projects in various sizes (ranging from a few developers to hundreds of contributors), different coding cultures (e.g., Java vs. C), or different application domains (browsers vs. operating systems). These projects would allow us to understand *how norms emerge and how they are enforced in different settings*. Secondly, OSS projects involve communication and coordination of contributors from different backgrounds, cultures and geographical regions, which makes OSS an exciting domain for exploring *how norms affect the success or failure of a particular software project or community*. Thirdly, such rich, extensive and readily available data from OSS projects allow us to *extract norms from different sources*. For instance, we can directly observe developer discussions, identify their contents (e.g., patches, bugs, reviews) on mailing lists or forums. We can build social

networks, and cross-check associated discussion and programming activity. In addition, we can leverage existing mining software repositories (MSR) technologies [39]²⁵ such as data preprocessing, cross-linking data from different sources for mining norms.

5.2 Suitability of a Normative Model

We believe the techniques and tools developed by NorMAS researchers to identify and extract norms can be leveraged and extended. Researchers in NorMAS [60, 61] have developed mechanisms for extracting norms from agent interaction data. Open source software repositories available in various forms such as historical repositories (e.g., SVN or CVS repositories, archived communication records, bug repositories), code repositories (e.g., Sourceforge.net or Google code), and run-time repositories (e.g., deployment and/or execution logs) remain largely unexplored in the context of norms. Since these repositories are populated by humans, these repositories contain explicit or implicit information on *norms* relevant to the communities involved in the process of software development. These repositories need to be *mined* using the techniques developed in the NorMAS community to uncover useful and important patterns and information about the normative processes associated with the development of software systems. Such information might offer insights and predictions about the future of software systems.

5.3 Challenges for Norms

This subsection discusses the research opportunities for NorMAS researchers in applying the concepts and mechanisms developed to extract different types of norms from software repositories and the associated challenges.

5.3.1 Challenge 1: Norm Types and Classification

The first challenge is to answer the question of *what types of norms exist in open source software development communities*. Several research work in NorMAS have treated both conventions and norms under the same umbrella of norms despite the differences between the two. We briefly discuss the distinction between the two using the examples from Open Source Software Development (OSSD).

Conventions of a community are the behavioural regularities that can be observed. Coding standards of a project community is an example of a convention. The specifications of these conventions may be explicitly available from the project websites²⁶ or can be inferred implicitly (e.g., a wide spread convention that may not be explicitly specified in project websites).

Norms are conventions that are enforced. A community is said to have a particular norm, if a behaviour is expected of the individual members of the community and there are approvals and disapprovals for norm abidance and violation respectively. There have been several categorizations of norms proposed by researchers (c.f. [59]). We believe that *deontic norms* - the norms describing prohibitions, obligations and permissions studied by the NorMAS community [73] is an appropriate categorization for investigating different types

²⁵ A extensive review of the work in MSR can be found from the “Bibliography on Mining Software Engineering Data” available at <http://ase.csc.ncsu.edu/dmse>.

²⁶ Refer to <http://source.android.com/source/code-style.html> for the coding guidelines for Android development.

of norms that may be present in OSSD communities. We believe most norms in software repositories will either be prohibitions or obligations.

Prohibition norms: These norms prohibit members of a project group from performing certain actions. However, when those prohibited actions are performed, the members may be subjected to sanctions. For example, the members of an open source project may be prohibited to check-in code that does not compile, and they may be prohibited to check-in a revised file without providing a comment describing the change that has been made.

Obligation norms: Obligations describe activities that are expected to be performed by the members of a project community. When the members of a community fail to perform those, they may be subjected to sanctions. For example, the members may be expected to follow the coding convention that has been agreed upon. Failure to adhere to this convention may result in the code not being accepted by the repository (e.g., based on automatic checking) or a ticket may be issued by a quality assurance personnel. Another obligation may be that the members should complete a task within a time frame. Failure to do so may result in a warning message (issued either automatically or manually) in the first instance.

We note that recognizing sanctions (a starting point to infer norms) is a key challenge since it involves natural language processing. Verbose text may be used in the construction of sanction messages. For example, the messages may involve terms that are well beyond the deontic terms such as 'should not', 'must not', 'ought not' in the case of prohibitions. One way to address this problem is to use existing tools such as WordNet [28] to extract synonyms of terms used in the text to infer deontic terms and also use information retrieval tools that offer data manipulation functions such as cleaning and disambiguating the verbose text in order to extract sanctions. Suitability of tools such as OpenCalais (<http://www.opencalais.com>) and AlchemyAPI (<http://www.alchemyapi.com>) for this purpose can be investigated. We believe recognizing sanctions is indeed a huge challenge. At the same time, it presents opportunities such as the construction of normative ontologies that can be used across projects for recognizing sanctions.

5.3.2 Challenge 2: Norm Identification

In NorMAS, researchers have proposed a life-cycle for norms [59] which broadly consists of five phases namely norm creation, identification, spreading, enforcement and emergence. Chapters 5 and 6 of this volume also presents a similar norm life-cycle model. We believe various phases of norm development can be studied based on the data available from software repositories. Specific research challenges in the context of mining software repositories for norms are given below.

- What are the modes of norm creation in a project community?
- How are prespecified norms enforced? What kinds of sanctions exist for norm violations? What is the uptake of a norm in a community (i.e., level of conformance)?
- How can emergent norms be detected? How are these norms spread? What contributes to the acceptance or rejection of these norms in a community?

5.4 Status and Prospects

This section offers some initial thoughts on addressing the research questions described in Section 5.3.2. It also reports the progress made by relevant research works.

5.4.1 Modes of Norm Creation

There could be two modes for norm creation in a software development community. They are 1) explicitly specified norms which every project member is expected to know (prespecified norms) and 2) norms that arise due to interactions between agents (emergent norms).

5.4.2 Enforcement of Prespecified Norms

In a project, both conventions and norms may exist. Conventions agreed upon by project members can be easily monitored. Examples include coding conventions and the convention of not uploading files that do not compile to a version control system. It should be noted that coding conventions can be checked for compliance by evaluating the code using an automated software program such as CheckStyle²⁷.

Norms on the other hand are enforced. Enforcement involves the delivery of appropriate sanctions. In the domain of software repositories these sanctions are present in artifacts. For example, a bug report on a module that does not deliver the functional requirements can be viewed as a sanction. Additionally, tickets issued for not resolving a bug completely can also be considered as a sanction. Therefore, sanctions that follow violations act as triggers to infer norms. Frequency of norm violations over time may provide evidence for the uptake of a norm in a society. We note that identifying and categorizing different types of sanctions from different types of artifacts is a challenge since the extraction of sanctions involves natural language processing.

5.4.3 Identifying Emergent Norms

Norms that are not prespecified but that emerge at run-time will be challenging to identify. We believe that emergent norms can be identified by identifying violations first and then inferring what the norms might be. The machinery proposed for norm identification by Savarimuthu et al. [60, 61] can be used as a starting point to infer prohibition and obligation norms. In their work, prohibition norms are identified by extracting sequence of action (or actions) that could have caused the sanction by using a data mining approach [60]. Sanctions form the starting point for norm identification. In the case of obligation norms, missing event sequence (or sequences) that was responsible for the occurrence of a sanction, is identified [61]. While these work on norm identification can be used as a starting point for the extraction of emergent norms in simple cases, the domain of MSR poses more challenges. For example, correlating or linking different types of documents containing relevant information is required before a sequence of actions can be constructed. For example, an email message may contain the sanction message exchanged between developers A and B. Let us assume that A sanctions B for not adding a valid comment to the latest version of the uploaded file. The problem in extracting the norm in this case is that, first, the verbose message sent from A to B should be understood as a normative statement which involves natural language processing. Second, a cross-check should be conducted to evaluate whether the normative statement is indeed true (i.e., checking whether the comment entered by B is invalid by investigating the log)²⁸. Third, the support for endorsements or oppositions to such normative positions need to be evaluated in order to extract this as a norm.

²⁷ <http://checkstyle.sourceforge.net/>

²⁸ In this example only two artifacts, the email message and the log are involved. But in practice, several different types of documents may need to be traversed to find the relevant information. Techniques developed in the field of MSR (e.g., [5], [53]) can be employed for cross-linking documents.

Norms that are identified through this process can then be made available to the project community (e.g., on the project websites) once it has been verified by the project administration team.

5.4.4 Need for a Norm Extraction Framework

A first step towards addressing these issues is to create a framework that can extract both conventions and norms from software repositories. The framework should be equipped with appropriate libraries for a) information retrieval techniques (including natural language processing) in order to identify sanctions b) mining software repositories (e.g., cross-linking different sources) and c) norm extraction (e.g., inferring norms from sequences of events). Additionally, it should be able to track and trace the life-cycle of a norm. For example, it should provide appropriate features to capture the waxing and waning of a norm across different periods of time.

5.4.5 Cross-Disciplinary Research Questions

The following are interesting research questions that can be considered in the future.

- How different are norms in large projects (e.g., measured based on total number of members or size of the project in kilo-lines of code) than the smaller projects? Are norm violation and enforcement patterns different in these projects?
- What are the relationships between roles of individuals in software development and norms (e.g., contributor vs. reviewer vs. module administrator)?
- Are there cultural differences within members of a project with regards to norms (inter- and intra-project comparisons) since individuals from different cultures may have different norms?
- Is there a difference between norm adoption and compliance between open-source and closed-source projects?

The above mentioned questions may interest both humanities researchers and computer scientists. Synergy between the two is required for addressing these questions. As computer scientists we can employ our expertise in several areas (i.e., normative multi-agent systems, information retrieval and MSR) to help answering these questions.

6 Christiaanse and Hulstijn: Automation of Control Measures

6.1 Application Scenario

Management of corporations will delegate tasks. Delegating tasks raises specific control problems, as studied in agency theory²⁹. In particular, delegation raises the problem of *private information* [47]: the agent to which the task is delegated has private access to information about execution, which the principal, who delegated the task, does not have. Private information problems can be of several types, namely *moral hazard* (hidden action), the agent may perform differently from what was expected, and *adverse selection* (hidden knowledge), the principal may have chosen the agent on the wrong grounds. Control problems are usually

²⁹ Agency theory is not the same as multi-agent systems theory. It is used in economics and sociology to study the delegation of tasks and ways of dealing with the resulting control problems [25]. For example, it explains the nature of remuneration and the set up of contracts.

mitigated by the principal, who may demand additional *control measures* to be implemented, such as supervision, formal procedures and guidelines, budget constraints, verification measures, software application controls, input controls, etc. Control measures must fulfill a purpose: a *control objective*. Generally, control objectives require a combination of organizational, procedural and automated control measures. A control objective corresponds to the notion of norms used in Normative Multi-Agent Systems. Just like norms, control objectives prescribe particular behavior, and clearly define deviations and violations.

However, adding control measures may have large *costs*. Consider the costs of implementing controls, the costs of resources that cannot be spent otherwise and the costs of reduced efficiency, usability or flexibility in execution of a task. In an attempt to reduce the costs of control, automated controls are becoming increasingly important. Control measures are for example built into ERP systems and workflow management systems to prevent undesirable behavior. This preventative approach may be called *compliance by design* [34]. Security logs and existing systems for monitoring process quality are extended and also used to verify effective implementation of control measures. Such a continuous approach to monitoring and detection may be called *continuous control monitoring* [72], or *continuous auditing* [46].

In general, providing assurance that an organization is compliant, involves several tasks: determining the control objectives (norms), determining specific control measures (actions) to be taken by the organization, determining control indicators (evidence), evidence collection, monitoring, warning in case of deviations, adjusting behavior when necessary, and applying sanctions when necessary. When parts of control systems are automated, the various tasks involved in providing assurance are re-distributed [14]. Tasks like data collection, monitoring, and triggering warnings can be automated. But even in a fully automated control system, an auditor must assess the appropriateness of the design and verify operating effectiveness of the system as specified.

Question: what are the effects of control automation on assurance provision?

Traditionally, auditors are responsible for providing reasonable assurance that (financial) information is free from material misstatements [45]. In doing so, auditors often use the *audit risk model*. This model helps to determine the amount and kind of substantive testing the auditor must perform for a given audit assignment, given the nature of the business, the strength of internal controls and the quality of evidence. Substantive testing is done manually, and is therefore relatively expensive.

$$\text{Audit Risk} = \text{Inherent Risk} \times \text{Control Risk} \times \text{Detection Risk} \quad (1)$$

Audit risk is the risk for an auditor that material misstatements remain undetected. Usually, an acceptable audit risk is set beforehand. *Inherent risk* is the a priori likelihood for a misstatement. This is based on the nature of the business. *Control risk* is the likelihood that (internal) controls will not prevent or detect a misstatement. This depends on the strength of preventative controls. *Detection risk* refers to the risk that an auditor will not detect a material misstatement. This depends on the amount of substantive testing and the persuasiveness of audit evidence. In general, there are six ways of obtaining audit evidence: Inspection, Confirmation, Observation, Re-performance, Analytical evidence and Client inquiry [45]. Of these types Inspection, Confirmation and Re-performance are considered most reliable, but they are also the most labour intensive and therefore the most expensive. So there is a trade-off between the quality of evidence and the costs of control.

We argue that automating controls may have two kinds of effects on the costs of control. Obviously, it will increase control effectiveness (prevention). In terms of the audit risk

formula, this means a smaller control risk. Given a fixed audit risk and inherent risk, this means that the detection risk is allowed to be higher, and less substantive testing is required, reducing the costs of control. Second, controversially, we also believe control automation will enhance the quality of evidence (detection). This means a smaller detection risk. Therefore, less additional substantive testing is needed, which will reduce the costs of control.

6.2 Example Case Study

We investigated this claim by analysis of a case study [17]. The case concerns the procurement process for care-related public transport services. The case is representative for a large class of complex purchasing processes, which can benefit from control automation.

The case involves the following parties. SRE³⁰ is the name of a public agency acting on behalf of several municipalities in the south of the Netherlands, which must purchase care-related transport services for elderly and disabled citizens. As you can imagine, care-related transport services are highly regulated. There are many legal requirements concerning the vehicle, the driver and how to deal with patients. The transport service provider (TSP) provides care-related taxi bus services on a demand basis. It receives a monthly fee from SRE for all patients being transported, as well as individual contributions from non-patient passengers. TSP keeps a record of all trips being requested, executed and cancelled, including data about individual patients and other travelers.

What is the norm? SRE must ensure the accuracy and legitimacy of the monthly invoice from TSP. The norm that needs to be verified, is whether the invoice is calculated correctly and all trips are executed according to legislation. Therefore the contract stipulates that TSP must provide a data file about the executed trips. The contract also contains a data-protocol about the expected format of the data file. In the context of the contract, the data file counts as evidence of accuracy and legitimacy of the invoice. How can SRE verify adherence to this norm? In other words: how can SRE establish reliability of the data file?

First, with the help of an auditor, SRE has set up a system of automated controls, to verify syntactic and semantic well-formedness of the data according to the data protocol. Verification is executed automatically by a PHP script. For example, for all trips a patient number must be recorded, and the patient must be known to be eligible for transport. In addition, the script can also test for coherence of the data file according to *reconciliation relationships* [67]. For example, the set of executed trips should equal the set of requested trips, minus the cancelled and no-show trips. Or, the total length of all executed trips should equal the sum total of kilometers registered by the taxi company. These reconciliations make sure that the data represent well-formed transactions.

Second, the contract stipulates that once every year, SRE must provide an audit opinion from an external auditor about the reliability of the processes and computer systems which generate both the invoice and the data file. In particular, reliability depends on segregation of duties and general IT-related control measures: change management, access control, logging and monitoring, and baseline security.

6.3 Challenges for Norms

The case study concerns *control automation*: automatic verification of evidence against a norm. Although there are many techniques for automated verifications, it is unclear how

³⁰ Samenwerkingsverband Regio Eindhoven (Cooperation Eindhoven Region)

these techniques will affect the role and responsibility of auditors in providing assurance. In fact, the case is representative of a large number of cases, where audit evidence will be provided by an automated system, partly under the supervision of the company being audited. This requires trust, that can be founded on control measures. Auditors are still important, but only at the set-up and periodic assessment of the automated collection of evidence. In a sense, auditors will now perform a meta-audit of the automated controls, rather than a direct audit of behavior. Clearly, the tasks in assurance provision can be redistributed, partly to the automated system and partly to the company being audited. How does this redistribution of tasks affect assurance provision?

The challenge is to come up with a suitable *assurance architecture*: a set of modules in a specific configuration, which together provide assurance that some process or system is ‘in control’, i.e., will meet specified control objectives (norms). Some modules may be implemented by procedures carried out by humans and others by automated systems.

6.4 Use of NMAS: Status and Prospects

Currently, automated verification of controls is often addressed within the field of business process management (BPM). Here, people tend to focus on *conformance testing* [58]: can we prove that process designs meet specific constraints? However, the basic audit questions remain relevant: who translates a general regulatory objective into appropriate process constraints? (testing of design) Who decides that a specific system does in fact implement the processes as specified? (operating effectiveness).

We believe that essentially, these questions are about the automated collection of evidence. We believe the notion of *constitutive norms* [62], can be fruitfully used to further investigate the conditions under which automatically generated evidence becomes legally acceptable. In the case study, we have seen that a monthly datafile may under some circumstances (verified to be well-formed; yearly external audit opinion) count as sufficient evidence.

The notion of *organizational roles*, which is explored at length in Normative Multi-Agent Systems [8], will also play an important role. After all, who is authorized to state that certain automatically generated data counts as evidence of compliance to norms?

Finally, the notion of a *contract* will be crucial. Much compliance issues develop between companies, although set in a legal context of enforced contract law. There are various proposals for the representation of contractual clauses, and subsequent translation into business process constraints, e.g. [38]. The process of contract negotiation between parties about the required strength of additional controls can be fruitfully studied using qualitative game theory, as developed with normative Multi-Agent Systems [8].

7 Chopra: Norms in Requirements Engineering

Requirements Engineering (RE) either treats requirements as properties of the environment [76] or as stakeholder goals [52]. In this note, I present a novel conceptual take on requirements.

I take a broader communication-centric view of RE than is customary. In this view, RE is itself a *social application* [15, 16] in which software engineers and stakeholders represent autonomous participants. Taking this perspective sheds new light on the nature of requirements.

7.1 Suitability of a Normative Model

A communication-oriented view leads naturally to the idea that a requirement is a normative relation between the communicating parties. I use the notation $R(x, y, p, q)$ to mean that x requires of y that if p then q (p and q are propositions whose satisfaction or violation can be determined by observing the environment). For example, *BestWines* requires that *CellarSys* raise an alarm if the temperature in the cellar rises above 12°C .

$$R(\textit{BestWines}, \textit{CellarSys}, \textit{temp} > 12^{\circ}, \textit{alarmRaised})$$

Requirements, like commitments [64], are established and manipulated by communications. Table 2 shows a partial list.

■ **Table 2** Communicating Requirements.

Communication	From	To	Desired Effect
CreateReq(x, y, p, q)	x	y	$R(x, y, p, q)$
ReleaseReq(x, y, p, q)	x	y	$\neg R(x, y, p, q)$
CancelReq(x, y, p, q)	y	x	$\neg R(x, y, p, q)$

The above normative view of requirements brings forth the nature of a requirement. It makes the parties to a requirement explicit. In influential RE literature dealing with the conceptual treatment of requirements, the parties are either implicit, or worse, missing altogether. There are at least two pragmatic advantages in making the parties explicit: (1) large projects may involve multiple stakeholders and engineers, and (2) contracts among parties will be established based on the requirements. Further, the above view of requirements does not make any assumptions about stakeholder goals; their existence is grounded instead in communication.

Requirements may be satisfied or violated. Further, once a system that presumably meets a requirement has been deployed, if the requirement is violated in perfect operating conditions for the system (see discussion on *domain assumptions* below), then the stakeholder will hold the engineer responsible for the violation. And lastly, a stakeholder cannot repudiate his or her requirement arbitrarily: the stakeholder is bound to the statement of the requirement (nonrepudiation does not mean that a stakeholder cannot change requirements; it just means that he or she cannot deny their communication).

7.2 Challenges for Norms

Besides requirements, RE also deals with domain assumptions and specifications [76]. A *domain assumption* describes what one can safely assume to hold in a stakeholder's operational environment. Specifications describe a machine's interface with the environment such that when an implementation of the machine is introduced in the environment, the requirements are satisfied. In other words, specifications are the bridge between RE and the rest of software engineering (SE). A normative description of RE would have to account for not just requirements, but also domain assumptions and specifications. Further, these concepts would have to be formalized so that one could perform reasoning over them.

7.3 Status and Prospects

The above normative view of requirements engineering breaks from the prevalent tradition in RE where requirements are either described in low-level terms or as the goals of stakeholders.

A communication-oriented view of RE can potentially have a big impact on the practice of both RE and SE.

- It could provide a basis for formulating business contracts between engineers and stakeholders.
- It provides a conceptual framework within which to place *requirements evolution*. Requirements would be created because of communications from the stakeholder to the engineer and would evolve only when the stakeholder communicated modified requirements. Further, any evolution would need to be accommodated by a change in the contractual relationships between the parties. The operationalization of the communication primitives would result in *requirements management systems*.
- It provides a basis for requirements evolution to be understood in the broader context of *requirements negotiation*. The stakeholder may change his requirements; however, that does not mean the engineer is committed or will commit to meeting them. The engineer would normally also take into account the cost and time required (among other things) to meet the requirements and possibly make counter-proposals.

8 Noriega: mWater as a Normative MAS³¹

8.1 Application Scenario

Water use—because of scarcity and stake-holders’ conflicting goals—is a conflict-prone domain. Not surprisingly, it is a highly regulated one. One way that water policy-makers have to foster better water use and avoid conflicts is to regulate demand, and one such way is establishing a “water bank” to trade water rights.

mWater is a normative MAS that models the use of water rights in a closed basin. It focuses, on one side, on the process of trading those rights (regulating conditions that make the rights tradable, thus affecting demand and use behavior) and, on the other, on the process of using those rights (thus affecting conflict and conflict resolution).

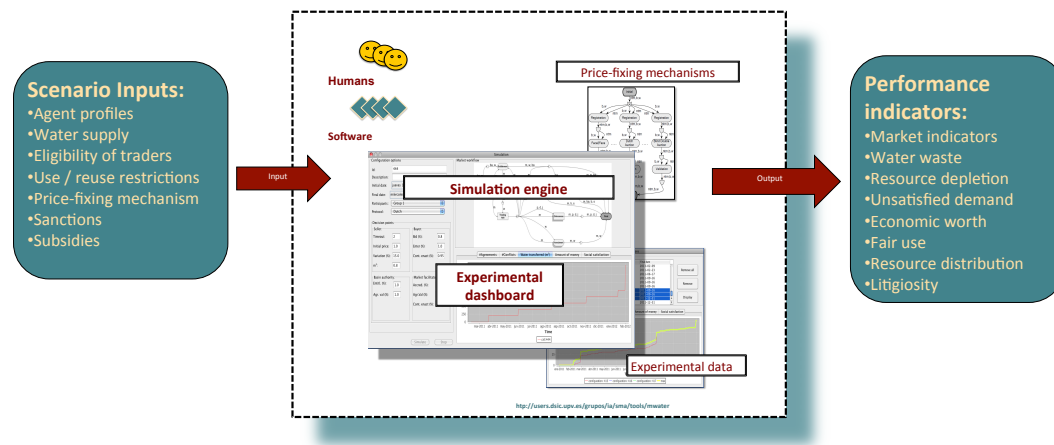
The system has three objectives:

1. As a testbed of agent technologies developed within the Spanish Agreement Technologies project.
2. To simulate effects of different normative corpora on user behavior, for water management policy design.
3. To build a realistic prototype of an on-line water bank for a closed basin.

The design is rooted on traditional practices and actual regulations, although it is slightly idealized to be a malleable platform (for testbed and simulation). The model departs from current legislation by allowing trading and usage with added flexibility (contract, use and misuse of rights, grievances and corrective actions) and under different price-fixing and conflict resolution mechanisms. The model makes obvious simplifications on the anchoring and constitutive conventions.

In abstract terms, *mWater* is defined as an agile market (mWater) of water rights and an agreement space for the management of rights. It is modeled and implemented as a regulated open MAS using the electronic institution (EI) meta-model and the EIDE platform [22].

³¹ Acknowledgments: The mWater prototype was partially funded by the Consolider programme of the Spanish Ministry of Science and Innovation through project AT (CSD2007-0022, INGENIO 2010) and MICINN project TIN2008-06701-C03-03. This research has also been partially funded by the Generalitat de Catalunya under the grant 2009-SGR-1434 and Valencian Prometeo project 2008/051.



■ **Figure 6** mWater used as a water-policy simulator.

8.2 Suitability of a Normative Model

From a normative point of view, *mWater* has the following features:

- *Norm sources*: Laws and regulations issued by governments; policies and local regulations issued by basin managers; traditional social norms.
- *Norm Expression*: Some are regimented in the EI framework or embedded in the decision-making processes of institutional agents. Other expressed are expressed in declarative form so that individuals may reason about them. The idea is that agents and designers may reason about these norms on and off-line; at design and at run time; and from the institutional (or legislative) perspective and the agent's individual perspective.
- *Issues dealt with in modeling*: Choice of expressive formalisms, institutional and social governance; norms dynamics (legislative and individual's perspectives), agent's decision-making strategies for compliance and, finally, criteria to evaluate effectiveness of norms.

The following, exemplify the types of norms that govern the mWater domain:

- Ponds in private land plots are considered part of it as long as they are used exclusively within that plot.
- Mineral and thermal waters have their own regulation.
- Any person may treat sea water provided proper administrative permission is granted and it has been proved that standards of residual disposal and quality for the intended use are met.
- When the Ministry of the Environment declares a state of emergency drought, all entitlements to water rights are suspended while the state is active.
- Rights are tagged for a type of use and may be exploited for that or for uses of higher priority. Water use priorities in descending order are: human, agriculture, energy, and recreational.
- The basin authority may at its discretion allow trading of suspended rights during states of emergency, given priority to holders of lower priority rights for higher priority use.
- Unused rights may be challenged and expropriated.

8.3 Challenges for Norms

As a normative MAS, *mWater* provides three main contexts of interest where challenges may be properly differentiated and to a large extent isolated and addressed independently:

- *Negotiation*: Protocols (conventions, eligibility of participants, tradable rights; judgment aggregation (decide when drought may be officially declared), negotiation heuristics (from an individual's decision to comply or not with given norms, suitability), agent with norm-aware architectures; argumentation about suitability, non-compliance, and others.
- *Contracting*: actual clauses of contracts, conflict identification and impact assessment. Transient character of norms, commitment conflicts, new contracts co-exist with other active contracts.
- *Agreement management*: Agreements and contracts may be contested by agreeing parties, observing third parties, authorities. System and agents capabilities to address conflict, online dispute resolution (ODR) and other forms of conflict management. Post-trading activities to study conflicts (detonators, structure, types) and conflict resolution (intervention strategies, rhetorical moves, settlement strategies).

Generally speaking, then, the application provides a convenient example to study the interplay between formal institutional components (laws, ontologies, sanctioned practices) organizations that enforce or should abide by them, individuals that form those organizations or participate in the regulated activities. An interesting, and perhaps not so frequently seen in normative MAS, *mWater* involves collective actors with their own group rules for allocating rights and solving conflicts. Collective actors that are involved in negotiation as collective entities in collective decision-making, judgment aggregation and other ways of social choice. Moreover, the problem domain needs to reflect organizational and institutional dynamics, is a good example to study an individual's immersion of norms and collective emergence of norms, and a natural context where there are empirical grounds for playing with notions like trust and reputation, moral authority, power and force.

8.4 Status and Prospects

A crude prototype of the complete system (including both trading and conflict management) was been implemented using the EIDE platform. Then a second version consisting of a more thoroughly developed trading part was implemented, and then, on top of it, a running simulator was built. Recently, a new version of this simulator with suitable agent populations is being developed using the GORMAS meta-model and tools.

On the other hand, *mWater* provided grounds for two other systems currently under development: a system for on-line trading of waste products and an "agreement space" for open innovation (in the "green economy" domain). Both have commercial interests behind and, in both cases, the need to approach the problem as a normative MAS is not only adequate but unavoidable if an effective sociotechnical system is to exist.

9 Balke: Wireless Mobile Grids

9.1 Application Scenario

The current deployment of the third generation (3G) of mobile network systems is in progress, but a quite different next generation network (called Fourth Generation or 4G) is under development. This latter is intended to bring about a paradigm shift in the cooperation

architecture of wireless communication [44]. Whereas for 3G the industry focused on technology for enabling voice and basic data communications (technology-centric-view), the emphasis in 4G is more user-centric [75]. One issue that, according to several studies [70], is of very high importance to users is the battery capacity of mobile phones.

Batteries have fixed capacity that limits the operational time for a device in one charge cycle. The increasing sophistication of mobile phones and their evolution into smart phones offering Internet access, imaging, audio and access to new services, has a significant impact on power consumption, leading to shorter stand-by times.

Fitzek and Katz [29] proposed a mechanism to address these issues with the concept of a “wireless mobile grid” (WMG), in which users share resources in a peer-to-peer fashion via the short-link connection devices built-in in the current mobile phone generation (e.g., WLAN or Bluetooth). The advantage of these is that they use significantly less power. However, the WMG idea of Fitzek and Katz requires collaboration between users that may be difficult to realize. The ensuing social dilemma is that network users can exhibit strategic behaviour, that places their own benefit above that of the collective. The main problem in WMG is that collaboration comes at a cost, in the form of battery consumption for contributing to the WMG. In consequence, rational users will prefer to access the resources without any commitment. However, if a substantial proportion of users follow this selfish strategy, the network itself would be at stake; depriving all users from the benefits, namely the potential battery saving arising from cooperation [75].³²

9.2 Suitability of a Normative Model & Challenges for Norms

Following this brief explanation of WMGs and the possible contribution problems in them, they appear to be an interesting case study of a normative models in which one could analyse how different norms and enforcement mechanisms (reputation, police agents,...) affect the behaviour of the telephone users (agents) in the system and how norms could possibly alter their behaviour.

From a normative point of view WMGs have several interesting properties. These properties, which make them well suitable for normative models and pose interesting challenges for normative MAS—the major one being the study of encouraging collaboration (e.g., by means of enforcement)—will now be explained in brief.

Complex Open Distributed Setting. WMGs are systems in which any mobile phone user with an WMG-enabled phone can join or leave at any point of time. The users are not static, but they are moving, which changes the possible collaboration groups all the time and also reduces the number of possible repeated interactions. This poses interesting challenges for reputation-based enforcement mechanisms. In addition if reputation mechanisms were to be used, the problem arises that sending information comes at a battery cost again and thus poses the research challenge to reason about enforcement and “optimal” levels of enforcement when enforcement is not for free but comes at a cost.

³² WMGs are similar to conventional P2P networks, but differ in physical aspects and emphasis. In particular, phones have specific resource restrictions (e.g., SIM card space) that limit the processing and storage capabilities of the WMG nodes. In addition, the WMG concept includes sharing processing power (omitted in the scenario discussed later for the sake of simplicity). Current scenarios from the mobile phone industry include big sport events, news and financial data in banking districts, IPTV, cooperative online gaming as well as maps and location information at airports, etc. Routing is a major issue in P2P systems, but of less significance in WMG, in part due to the relatively short-lived and transient nature of alliances. Lastly, we note that transmission failure is more common in wireless than in wired networks, making it harder to determine intent when non-cooperation is observed.

As real humans are envisioned to engage in WMGs, they need to be adequately represented in a normative MAS by means of agents. This poses huge challenges w.r.t. modelling their decision making behaviour, which is far from being rational at all times.

Beyond Micro-Macro. From a normative perspective, in a WMG both on the micro as well as on the macro level norms can be present and the norms on both levels possibly affect each other. Thus, in WMGs one expects norms to be defined at a macro-level specifying correct behaviour and for example which sanctions might be enacted if the users do defect (i.e., do not contribute to the WMG but use its resources). This is likely to influence the decision making of the users in the WMG. On the micro level, in addition norms as a result of the user interaction can emerge. These norms do also can have an effect on the users' decision making and actions and will not necessarily be in concordant with the norms defined on the macro-level. They might even effect the the macro-level norms. This results in a complex micro-macro link where both levels bi-directionally interactively influence each other. To model this continuous two-way interaction of the micro and the macro perspective is another challenge for normative MAS.

Multiple Stakeholder. The final interesting challenge that WMGs pose for normative MAS is the number of different stakeholders involved in WMGs; all of which are dependant on one another, but which have different objectives. In the WMG one for example typically finds at least the mobile phone users, mobile phone manufacturers, infrastructure providers as well as telecommunication providers. Balancing the interest of these different WMG stakeholders adds another layer to the question of finding good mechanisms that encourage WMG collaboration—a challenge that so far has not been approached in normative MAS.

9.3 Status and Prospects

From a technological point of view, much has been done in respect to WMGs. Thus, currently prototypes of WMG-capable mobile phones exist and first tests for their deployment are being conducted. This allows to obtain real WMG mobile phone data which can be used in normative MAS models. Nevertheless, from a normative perspective many questions—such as how to encourage contribution and discourage defection in these networks—still need to be addressed, before WMG could be turned into a commercial product. This real world commercial focus as well as the numerous WMG-related interesting research challenges for normative MAS outlined before are the reason why research into the normative aspects of WMGs is both important as well as at the same time challenging, but worthwhile.

10 Governatori and Lam: UAV³³

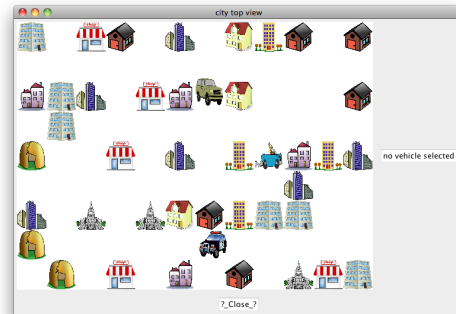
Governatori and Rotolo [36, 37] proposed a computationally oriented rule based approach to model (normative) agents. The framework, called BIO (Belief, Intention, Obligation) is an extension of defeasible logic with modal operators to model: the representation of the environment in which an agent is situated (beliefs), the norms governing the agent (obligations), and the goals of the agent (intentions). The BIO approach permits parametrised definition of different agent types, where an agent type corresponds relationships and preferences over the various modalities describing the mental attitudes of the agent and external modalities. The framework is intended to provide executable specifications for

³³ Acknowledgments: NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council.

an agent. This means the rules in which the agent and the environment are described provide the rules can be executed directly by a defeasible logic engine without the need to program the agent in an external language. To facilitate this, SPINdle [48], a modern and efficient Java based implementation of defeasible logic has been developed. The result is that the combination of of the BIO framework and SPINdle offers a flexible and agile tool for programming (normative) agent based applications.

10.1 Application Scenario

Typical complex system have to manipulate and react to different types of data (e.g., numerical and boolean), and in many occasions we have to integrate different types of reasoning process. In this sense, the focus of our research are of two-fold: (1) how a non-monotonic rules-based system can be integrated with numerical computations engines, and (2) how the behavior of an agent can be affected by the external contexts. To illustrate the combination of techniques, we have the following problem scenario:



■ Figure 7 City map model.

Given a city map with specific targets and obstacles (Figure 7), a number of UAVs has to navigate through the city from a starting location to a destination without colliding with each other. There is a GPS enabled application that informs the UAVs about the current traffic situations and the locations of other UAVs. To navigate successfully, the UAVs have to follow some guidelines about how and when they should alter their route.

The above scenario revealed how a UAV should interact with its environment. It presumes an execution cycle consisting of a phase where the UAV collects information through its sensors, decides an action and then applies this action [74].

10.2 Suitability of a Normative Model

In order to travel from one location to another, a UAV has to gather different types of information from the GPS monitor within a proximity range and detects if any traffic problems might appear. The Knowledge Base (KB) of a UAV is a well-documented limited set of behavioral interactions that describes the behavior of a UAV under different situations, in particular it contains (prescriptive) rules for determining who has right of way over who. It is complemented with formal logics (and in particular Defeasible Logic (DL)) to represent significant issues regarding the domain of operations.

10.3 Challenges for Norms

In case of a possible collision, a UAV will utilize the information in its KB and incorporate into it the set of context-related information (such as traffic situation, information about the vehicles nearby, etc) and derive a safe direction of travel or eventually to temporary stop its motion in *real-time fashion*.

Consider the scenario as shown in Figure 8 where vehicles V_3 , V_4 and V_5 are moving towards the same location (the red circle) and collisions may occur if none of the vehicles alter their route. This perception-action cycle (Figure 9) can be conceived not only as an issue of control, but also lays out the interface on how the UAVs should interact with the environment [4]. Here, the *sensors* (in our case the GPS monitor) collect information about the environment (as described above) and which is then combined them with the knowledge base for common-sense decision making. The *behavior controller* then reasons on these information and triggers the associated *actuator(s)* (whether to change its current travel direction, speed or even to stop its current motion) based on the conclusions derived.

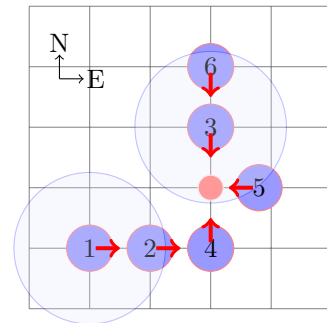


Figure 8 City with UAVs.

10.4 Status and Prospects

Besides the combination of numerical and logical techniques, in particular, an extension, which combines the ability of handling violations and degree of preferences, of BIO approach to the development of agents in DL has been established. In addition, a novel algorithm computing extensions of Modal Defeasible Logic has been devised. Readers interested please refer to [49] for details.

Future work includes the study of UAV negotiation, the use of Temporal Defeasible Logic and integrating the rule-based system with reaction-based mechanism.

The main aim of this application was to demonstrate the suitability of the use of the BIO approach to the development of agent based applications. The outcome is promising. It was possible to describe the behaviour of the UAV agents, and the norms defining the right of way in a theory of approximately fifty rules.

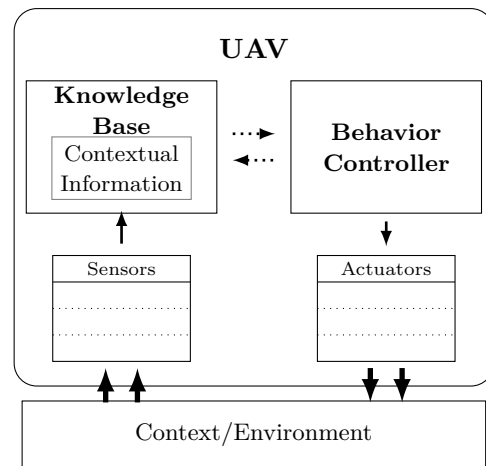


Figure 9 Behavior control under an perception-action cycle.

11 Dignum: Using Norms for Serious games

11.1 Application Scenario

When training skills like arithmetic or spelling the trainee has to learn to follow the rules that will deliver the right output for a given input. For example, $5^3 = 5 \times 5 \times 5 = 5 \times 25 = 5 \times 20 + 5 \times 5 = 100 + 25 = 125$.

In these types of serious games the rules can be implemented as constraints or given as formulas that can be used to calculate the answer. However, when training involves the behavior of other people we cannot suffice to use simple rules or constraints anymore. For example, when learning to drive a car in a simulator we also have to simulate realistic behavior of other traffic participants. Although all traffic participants are regulated by the same traffic laws this does not mean that they always obey those rules!

In order to train properly we should not just create traffic participants that do not follow the rules, but characters that violate the rules in realistic ways. For example, a kid might cross the road without looking when chasing its ball. So, this would happen if there is space next to the road where the kids can play with a ball (e.g., a park or garden), but not at a through road without playroom or houses. It would be completely weird to have a kid chasing a ball in a highway.

In order to model scenarios where characters violate behavioral rules in believable ways we need to be able to model the rules as norms which can be violated. Having the norms represented explicitly facilitates the representation of rules for violation as well. These rules indicate the type of circumstances in which violation is likely or at least possible. They might also indicate the consequences of the violation.

Using norms to represent behavioral rules in training scenarios forces the designers of the game to think carefully which type of violations (or unexpected events) the trainee should be able to handle. When some violations are not important for the purpose of the training they can be represented by fixed constraints or rules and thus will never be violated.

For many training scenarios it is exactly the coping with violation of rules that is important. In a first phase of the training one can make scenarios where every character keeps to the rules and thus the standard behavior is trained. Think about a driver that learns to cope with handling the car (gas pedal, brake, clutch, steering wheel, etc.). In the next phase the trainee is allowed to apply the behavioral rules, but, of course, can make mistakes when applying them. In this case the game should react properly in order to show the (possible) consequences of breaking a rule. For example, when a car turns right the driver should look over his right shoulder to check if a bike is approaching (typical in at least the Dutch situation). When the driver forgets to do this the game might react by having bikes appear and hit the car or fall when trying to avoid the car.

Finally, we want to model scenarios where characters violate the rules and the trainee has to react properly to the violation. For example, bikes going straight and ignoring a red light while the car has to turn right. In this case the driver might assume he does not have to look for the bikes because his light is green. However, in The Netherlands the car is still responsible to avoid the bikes even though they violate the traffic rules.

Many training scenarios follow the same pattern as sketched above. Whenever training skills in situations that involve other people it is important to be able to accurately react to violations of rules. In order to model these scenarios it is imperative to model rules as norms and have them explicitly available during the design.

Although many serious games are now used in class room situations where cognitive skills are trained there is a large interest to use serious gaming for training skills that involve social abilities and require adequate coping with people behaving unexpectedly. Training of these skills which range from the above driving lessons, to fire drills, team training and application training usually are now performed using hired actors or involving experienced professionals. These people are expensive to use for training and are only available at limited times. Therefore the future of serious games for this market is promising.

11.2 Suitability of a Normative Model

The suitability of the normative model is already discussed implicitly in the previous section. When the rules would have been implemented as constraints in the game, the trainee would have no opportunity to violate the rules. Although this might be good in the initial stage of the training, it would really impede the learning of the rules, which is an important part of training situations involving other persons.

One could argue that the system does not need to represent the norms explicit in order to react to violations of rules by the trainee. This is true. However, designing the system becomes easier if situations can be generated (or triggered) based on explicit violations of rules. This makes the system more modular and also facilitates later additions of extra scenarios. This might be necessary when certain violations occur often and can occur especially in different situations. Then one might want to design different scenarios covering all the prototypical violation situations. For example, not looking over your right shoulder when turning right in a car can happen at a traffic light, but also when changing lanes at the highway or returning in your lane after passing a car. It is easy if all these situations can be explicitly tied to the same norm. Additionally this also facilitates keeping track which rules a trainee has most trouble with and possibly giving him extra scenarios in which that rule plays a role.

Having the norms explicitly available becomes even more important when the characters in a scenario are supposed to violate the rules. Those violations should be realistic. It is very hard to implement realistic violations for many scenarios if the norms are not explicitly available and can be prioritized and balanced against other goals of the character. Thus having normative agents, that can explicitly reason about norms and make decisions based on norms and other elements of the situation becomes almost necessary in order to implement these complex scenarios.

11.3 Challenges for Norms

When using norms in serious games one of the main challenges will be the combination and prioritization of the norms. Usually there will be both general rules governing the behaviour and interactions between persons as well as specific rules that govern detailed situations. For example, in general the speed limit for cars within a build-up area is 50 km/hr, however when signs along the road indicate that the speed limit is 70 km/hr one can drive more than the 50 from the general rule. (In general, in traffic law, signs take priority over rules). However, this becomes different again when a truck drives in a highway, where the sign states that you can drive 100 km/hr. In spite of this sign a truck can still only drive 80 km/hr in the highway.

There has been quite some theory developed on deontic logic that can be used to combine and prioritize norms. However, little work has been done that makes this theory practically usable in software systems.

11.4 Status and Prospects

Although the use of norms in designing serious games seems very intuitive we have not been able to actually include it in a project yet. The main impediment is the fact that a different design methodology should be used to design the game. At the present time game developers are under too much pressure to make money in order to take any risks by using an unfamiliar methodology.

We have participated in the early stages of a serious game design project for training fire fighting on board ships. Introducing norms led to exactly the right questions about which scenarios would need to be made for the training to be effective. Thus it seemed very promising as part of the design methodology. Unfortunately, after the initial phase there was too little money to take any risk in developing the game and traditional methods were used (leading to a rather static and limited game play).

We hope to get funding from government in order to develop a real case study and get a prototype serious game that can be used as show case. Once we succeed in this the prospects are quite good. Industry indicates that games for training human skills will become more important. Normative behavior forms an integral part of this type of skills and it thus becomes imperative to include norms into the games.

12 Cranefield and Verhagen: Virtual worlds as an application area for normative multi-agent systems

12.1 Application Scenario

Consider an online meeting place in which geographically separated human users can interact with each other and with software agents in a human-friendly way, whilst also being secure in the knowledge that certain specified norms of interaction will be monitored and enforced. That is the type of scenario explored by research on extending 3D virtual worlds with e-institution and normative multi-agent systems middleware. For example, remote buyers can participate in auctions held in a real-life fish market by controlling avatars that move (when allowed by the auction house rules) through various virtual rooms representing different stages of the auction process (e.g., buyer registration, the auction itself, and settlement), using natural gestures such as raising their (virtual) hands to make bids [11]. A connection with the real-world fish market is provided by instrumenting objects in the virtual world objects with scripts that sense avatar actions and enable or disable virtual counterparts to institutional actions (e.g., doors will open to allow avatars to move between rooms if they have permission to move between the stages of the auction represented by those rooms) [12].

This scenario illustrates the potential of research on electronic institutions and normative multi-agent systems to enhance computer-mediated human interaction. While virtual worlds offer a rich medium for people to interact despite being physically distant, they currently offer little or no support for users to maintain an awareness of the social context in which they are interacting. Tools developed by researchers in the NorMAS research community hold promise to fill this gap.

12.2 Suitability of a Normative Model

While much of the use of virtual worlds such as Second Life [50] is for unstructured social interaction and exploration of novel constructed fantasy environments, virtual worlds are also used for interactions within societies or organisations with a predefined or emergent social structure. For example, virtual worlds are used as a venue for meetings, lectures, role-playing and training exercises, re-enactments of historical or fictional societies, and for buying and selling both virtual and real-world goods. All these activities involve social structure and, potentially, norms.

12.3 Challenges for Norms

The scenario above has been implemented using an existing model of *electronic institutions*. This was possible due to the use of a specially constructed virtual world environment generated from an electronic institution specification [12]. Key challenges in this approach include how to generate virtual world environments from institutions, and how to map institutional actions and the regimentation and enforcement of norms to suitable counterparts in the virtual world.

Another challenge is to integrate NorMAS technology with virtual world environments that have already been constructed and have existing social uses. Although it has been suggested that the popular virtual world Second Life does not facilitate the use of social norms as an efficient social order mechanism due to the ease of users changing their identities [68], social norms have been identified at the level of groups in Second Life [9]. NorMAS technology could therefore be applied to provide computational support for normative processes at the local group level in virtual worlds. This could be done in a way that requires users to adapt some of their real-world practices when interacting in virtual worlds, e.g. by provided instrumented virtual artifacts that users must use to ensure that their institutional actions are detected. An example would be an object that virtual meeting participants must pass between themselves to indicate who “has the floor”. However, it would be less intrusive to develop techniques for detecting the existing domain-specific significant events in virtual worlds. As virtual worlds are real-time simulations with many possible avatar movements and actions available to users, detecting significant normative events requires the recognition of complex events from a rich (and not always consistent) sensory data stream. The richness of virtual worlds may also mean that more expressive languages for representing norms will be needed.

An even greater challenge is to develop techniques for *learning* norms that are present in existing virtual world societies. In this case, the high level significant events (such as the application of sanctions or other signals that indicate norm violation) cannot be predefined and must be learned from observation, with the norms then inferred by an inductive process. Norms that are never or rarely violated will be difficult to learn, unless agents can communicate with human avatars about possible norms and use natural language analysis of text chat to detect discussions about norms. Awareness of cultural difference may also be necessary, given the diverse range of users that may be present in virtual worlds. Research in this area could be undertaken not only to provide better tools for better social awareness and control in virtual world societies, but also to study norm-related processes amongst human users. In the chapter titled “(Social) Norm Dynamics” (page 135) the effects of cultural differences are discussed in more detail.

However, the focus there is not on mixed culture groups but on potential differences between cultures as such.

12.4 Status and Prospects

Progress has been made on many of the research challenges above, but to the best of the authors’ knowledge, no NorMAS technology has yet been deployed in real virtual world applications—only research prototypes have been developed to date.

Perhaps the most sophisticated constructed virtual institution is an interactive simulation allowing users to have an immersive experience of the culture of the ancient city of Uruk in 3000 B.C., complete with agent-controlled citizens playing different a variety of roles in the society [10]. Technology developed to facilitate the construction of such virtual institutions includes middleware for managing “intelligent objects” in virtual worlds [57] and a formal approach to generating environments in virtual worlds [71]. Another recent application of this technology is a prototype of a virtual institution for trading water rights [2].

Other work has investigated monitoring for the fulfillment and violation of conditional rules of expectation in existing environments in Second Life, based on the detection of changes in avatar animations [18] or predefined domain-specific complex events [56]. Agents can use a monitor service to track their personal expectations, which describe the expected future traces of the local region of the virtual world. These may correspond to norms, team tactics,

or simply observed regularities in the environment that the agent assumes will hold, but wishes to monitor.

The concept of a *virtual nation* has been proposed “to guarantee the existence of a secure and safe virtual world” by incorporating into a virtual world real-world structures such as a constitution, government and monetary system [20]. Laws in a virtual nation are defined in natural language and then translated into technical implementations by a team consisting of (at least) a legal expert, a virtual world developer and a programmer. Advances in NorMAS technology will be necessary to make this vision a reality.

Virtual worlds also provide a generic 3D simulation platform in which games (particularly so-called “serious games”, such as training scenarios [42]), can be implemented. The application of normative multi-agent systems to serious games is discussed in Section 11.

13 Summary

The preceding sections have illustrated the wide range of applications of norms in areas of significance not only to the computing discipline but also to society at large. These sections highlight various aspects of norms, e.g., in modeling complex systems, in engineering software systems, and in applying norms to capture and regulate interactions among humans. These could be potentially extended to apply to other real-life social entities such as organizations. Although the uses of norms reported here are research efforts, they are inspired by practical considerations. To fully develop an approach to the level where it could be deployed would require substantial effort, mostly in capturing elements of the domain over which norms can be employed.

14 Authors’ Note

A remark about the writing. Like for the other chapters in this volume, a working group for this chapter was organized at Dagstuhl. The original contributors of that group (Savarimuthu, Singh, Villata) had drafted longish extended abstracts of their efforts, as did Christiaan. Subsequently, other researchers were invited to contribute brief writeups on a use of norms. For this historical reason—and not anything to do with the relative importance of the application scenarios—the contributions by the above-named four authors (and their coauthors) are longer than the others.

15 Acknowledgments

We are indebted to the Dagstuhl organizers for the opportunity and venue to meet and discuss these topics. We would like to thank the anonymous reviewers for helpful comments.

References

- 1 Hal Abelson, Ben Adida, Mike Linksvayer, and Nathen Yergler. ccREL: The creative commons rights expression language. Technical report, Creative Commons, 2008.
- 2 P. Almajano, T. Trescak, M. Esteva, I. Rodriguez, and M. Lopez-Sanchez. v-mWater: a 3D virtual market for water rights (demonstration). In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, pages 1483–1484. IFAAMAS, 2012.
- 3 Matthew Arrott, Barry Demchak, Vina Ermagan, Claudiu Farcas, Emilia Farcas, Ingolf H. Kriiger, and Massimiliano Menarini. Rich services: The integration piece of the SOA

- puzzle. In *Proceedings of the IEEE International Conference on Web Services (ICWS)*, pages 176–183, Salt Lake City, 2007. IEEE Computer Society.
- 4 David Billington, Vladimir Estivill-Castro, René Hexel, and Andrew Rock. Architecture for Hybrid Robotic Behavior. In Emilio Corchado, Xindong Wu, Erkki Oja, Álvaro Herrero, and Bruno Baruaque, editors, *HAIS*, volume 5572 of *Lecture Notes in Computer Science*, pages 145–156. Springer, 2009.
 - 5 Christian Bird, Alex Gourley, Prem Devanbu, Michael Gertz, and Anand Swaminathan. Mining email social networks. In *Proceedings of the 2006 international workshop on Mining software repositories*, MSR '06, pages 137–143, New York, NY, USA, 2006. ACM.
 - 6 Christian Bizer, Tom Heath, and Tim Berners-lee. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 5:1–22, 2009.
 - 7 Barry Boehm and Jo Ann Lane. Using the incremental commitment model to achieve successful system development. TR 715, USC Center for Systems and Software Engineering, Los Angeles, July 2007. <http://csse.usc.edu/csse/TECHRPTS/2007/usc-csse-2007-715/usc-csse-2007-715.pdf>.
 - 8 G. Boella and L. van der Torre. A game theoretic approach to contracts in multiagent systems. *IEEE Transactions on Systems, Man and Cybernetics - Part C*, 36(1):68–79, 2006.
 - 9 Tom Boellstorff. *Coming of age in “Second Life”: An anthropologist explores the virtually human*. Princeton, NJ: Princeton Univ. Press, 2008.
 - 10 A. Bogdanovych, J. A. Rodríguez-Aguilar, S. Simoff, and A. Cohen. Authentic interactive reenactment of cultural heritage with 3D virtual worlds and artificial intelligence. *Applied Artificial Intelligence*, 24(6):617–647, 2010.
 - 11 A. Bogdanovych, S. Simoff, and M. Esteva. Normative virtual environments: Integrating physical and virtual under the one umbrella. In *Proceedings of the Third International Conference on Software and Data Technologies (IC-Soft 2008)*, pages 233–236, 2008.
 - 12 Anton Bogdanovych, Marc Esteva, Simeon J. Simoff, Carles Sierra, and Helmut Berger. A methodology for 3D electronic institutions. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 358–360. IFAAMAS, 2007.
 - 13 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.
 - 14 Brigitte Burgemeestre, Joris Hulstijn, and Yao-Hua Tan. Value-based argumentation for justifying compliance. *Artificial Intelligence and Law*, 19(2-3):149–186, 2011.
 - 15 Amit K. Chopra. Social computing: Principles, platforms, and applications. In *Proceedings of the 1st Workshop on Requirements Engineering for Social Computing*, pages 26–29. IEEE, 2011.
 - 16 Amit K. Chopra and Paolo Giorgini. Requirements engineering for social applications. In *Proceedings of the 5th International i* Workshop*, volume 766 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2011. 138–143.
 - 17 Rob Christiaanse and Joris Hulstijn. Control automation to reduce costs of control. In Marta Indulska, Michael zur Muehlen, Shazia Sadiq, and Yao Hua Tan, editors, *Proceedings of CAISE workshops - (GRCIS'2012)*, volume LNBIP XYZ, pages x – y. Springer Verlag, Berlin, 2012.
 - 18 Stephen Cranefield and Guannan Li. Monitoring social expectations in Second Life. In J. Padget, A. Artikis, W. Vasconcelos, K. Stathis, V. Silva, E. Matson, and A. Polleres, editors, *Coordination, Organizations, Institutions and Norms in Agent Systems V*, volume 6069 of *Lecture Notes in Artificial Intelligence*, pages 133–146. Springer, 2010.

- 19 Natalia Criado, Estefania Argente, and Vicente J. Botti. A normative model for open agent organizations. In Hamid R. Arabnia, David de la Fuente, and José Angel Olivas, editors, *IC-AI*, pages 101–107. CSREA Press, 2009.
- 20 V. Čyras and F. Lachmayer. Technical rules and legal rules in online virtual worlds. *European Journal of Law and Technology*, 1(3), 2010.
- 21 K. da Silva Figueiredo, V. Torres da Silva, and C. de O. Braga. Modeling norms in multi-agent systems with NormML. In M. De Vos, N. Fornara, J. Pitt, and G.A. Vouros, editors, *COIN@AAMAS 2010, Toronto, Canada, May 2010, COIN@MALLOW 2010, Lyon, France, August 2010, Revised Selected Papers.*, volume 6541 of *LNCS*, pages 39–57. Springer, 2010.
- 22 Mark d’Inverno, Michael Luck, Pablo Noriega, Juan A. Rodriguez-Aguilar, and Carles Sierra. Communicating open systems. *Artificial Intelligence*, 186:38–94, 2012.
- 23 US Department of Defense Architecture Framework Working Group DoDAF-WG. Department of defense architecture framework (dodaf), version 1.5. volume i: Definitions and guidelines, April 2007. http://www.defenselink.mil/cio-nii/docs/DoDAF_Volume_I.pdf.
- 24 Cássia Trojahn dos Santos and Jérôme Euzenat. Consistency-driven argumentation for alignment agreement. In Pavel Shvaiko, Jérôme Euzenat, Fausto Giunchiglia, Heiner Stuckenschmidt, Ming Mao, and Isabel F. Cruz, editors, *OM*, volume 689 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2010.
- 25 K. M. Eisenhardt. Agency theory: An assessment and review. *Academy of Management Review*, 14(1):57–74, 1989.
- 26 Jeff A. Estefan, Ken Laskey, Francis G. McCabe, and Danny Thornton. Reference architecture for service oriented architecture. Version 1.0, OASIS, April 2008. <http://docs.oasis-open.org/soa-rm>.
- 27 Peter Feiler, Richard P. Gabriel, John Goodenough, Rick Linger, Tom Longstaff, Rick Kazman, Mark Klein, Linda Northrop, Douglas Schmidt, Kevin Sullivan, and Kurt Wallnau. Ultra-large-scale systems: The software challenge of the future. Technical report, CMU Software Engineering Institute, Pittsburgh, 2006. <http://www.sei.cmu.edu/library/abstracts/books/0978695607.cfm>.
- 28 Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.
- 29 Frank H. P. Fitzek and Marcos D. Katz. Cellular controlled peer to peer communications: Overview and potentials. In Frank H. P. Fitzek and Marcos D. Katz, editors, *Cognitive Wireless Networks*, pages 31–59. Springer, 2007.
- 30 American Association for Public Opinion Research AAPOR. The Code of Professional Ethics and Practices, (Revised May, 2010).
- 31 N. Fornara. Specifying and Monitoring Obligations in Open Multiagent Systems using Semantic Web Technology. In A. Elçi, M. Tadiou Kone, and M. A. Orgun, editors, *Semantic Agent Systems: Foundations and Applications*, volume 344 of *Studies in Computational Intelligence*, chapter 2, pages 25–46. Springer-Verlag, 2011.
- 32 N. Fornara and M. Colombetti. Representation and monitoring of commitments and norms using OWL. *AI Commun.*, 23(4):341–356, 2010.
- 33 Nicoletta Fornara and Marco Colombetti. Ontology and time evolution of obligations and prohibitions using semantic web technology. In Matteo Baldoni, Jamal Bentahar, M. Birna van Riemsdijk, and John Lloyd, editors, *DALT*, volume 5948 of *Lecture Notes in Computer Science*, pages 101–118. Springer, 2009.
- 34 G. Governatori and S. Sadiq. *The journey to business process compliance*, pages 426–445. IGI Global, 2009.
- 35 Guido Governatori, Francesco Olivieri, Simone Scannapieco, and Matteo Cristani. Designing for compliance: Norms and goals. In Frank Olken, Monica Palmirani, and Davide

- Sottara, editors, *Rule - Based Modeling and Computing on the Semantic Web*, volume 7018 of *Lecture Notes in Computer Science*, pages 282–297. Springer Berlin Heidelberg, 2011.
- 36 Guido Governatori and Antonino Rotolo. Defeasible logic: Agency, intention and obligation. In Alessio Lomuscio and Donald Nute, editors, *Deontic Logic in Computer Science*, number 3065 in LNAI, pages 114–128, Berlin, 2004.
 - 37 Guido Governatori and Antonino Rotolo. BIO logical agents: Norms, beliefs, intentions in defeasible logic. *Journal of Autonomous Agents and Multi Agent Systems*, 17(1):36–69, 2008.
 - 38 Guido Governatori and Antonino Rotolo. Norm compliance in business process modeling. In *Proceedings of (RuleML'2010)*, volume LNCS 6403, pages 194 – 209. 2010.
 - 39 A.E. Hassan. The road ahead for mining software repositories. In *The 24th IEEE International Conference on Software Maintenance, Frontiers of Software Maintenance.*, pages 48–57, October 2008.
 - 40 Tom Heath and Christian Bizer. *Linked Data: Evolving the Web into a Global Data Space*. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool, 2011.
 - 41 P. Hitzler, M. Krötzsch, and S. Rudolph. *Foundations of Semantic Web Technologies*. Chapman & Hall/CRC, 2009.
 - 42 M. Johansson, H. Verhagen, and M. Eladhari. Model of social believable NPCs for teacher training using Second Life. In *Proceedings of the 16th International Conference on Computer Games*, pages 270–274. IEEE, 2011.
 - 43 L. Kaczmirek and N. Schulze. Standards in Online Surveys. Sources for Professional Codes of Conduct, Ethical Guidelines and Quality of Online Surveys. A Guide of the Web Survey Methodology Site. <http://websm.org/>, 2005.
 - 44 Marcos D. Katz and Frank H. P. Fitzek. Cooperation in 4g networks - cooperation in a heterogeneous wireless world. In Frank H. P. Fitzek and Marcos D. Katz, editors, *Cooperation in Wireless Networks: Principles and Applications*, pages 463–496. Springer, 2006.
 - 45 W. Knechel, S. Salterio, and B. Ballou. *Auditing: Assurance and Risk*. Thomson Learning, Cincinnati, 3 edition, 2007.
 - 46 John R. Kuhn and Steve G. Sutton. Continuous auditing in erp system environments: The current state and future directions. *Journal of Information Systems*, 24(1), 2010.
 - 47 Jean-Jacques Laffont and David Martimort. *The Theory of Incentives: the Principal-Agent Model*. Princeton University Press, 2002.
 - 48 Ho-Pun Lam and Guido Governatori. The making of SPINdle. In Guido Governatori, John Hall, and Adrian Paschke, editors, *Rule Representation, Interchange and Reasoning on the Web*, number 5858 in LNCS, pages 315–322, Berlin, 2009. Springer.
 - 49 Ho-Pun Lam and Guido Governatori. Towards a model of UAVs Navigation in urban canyon through Defeasible Logic. *Journal of Logic and Computation*, 2011. to appear.
 - 50 Linden Lab. Second Life home page. <http://secondlife.com/>, 2008.
 - 51 Paul Miller, Rob Styles, and Tom Heath. Open data commons, a license for open data. In *Proceedings of LDOW*, 2008.
 - 52 John Mylopoulos, Lawrence Chung, and Eric S. K. Yu. From object-oriented to goal-oriented requirements analysis. *Communications of the ACM*, 42(1):31–37, 1999.
 - 53 Nachiappan Nagappan, Thomas Ball, and Andreas Zeller. Mining metrics to predict component failures. In *Proceedings of the 28th international conference on Software engineering, ICSE '06*, pages 452–461, New York, NY, USA, 2006. ACM.
 - 54 OOI. Ocean Observatories Initiative, 2007. <http://www.oceanobservatories.org/>.
 - 55 Raul Palma, Jens Hartmann, and Peter Haase. OMV Ontology Metadata Vocabulary for the Semantic Web. Technical report, OMV Consortium, 2008.

- 56 S. Ranathunga, S. Cranefield, and M. Purvis. Identifying events taking place in Second Life virtual environments. *Applied Artificial Intelligence*, 26(1–2):137–181, 2012.
- 57 Inmaculada Rodríguez, Anna Puig, and Marc Esteva. Cross-platform management of intelligent objects behaviors in serious virtual environments. *Journal of Visualization and Computer Animation*, 22(4):343–350, 2011.
- 58 A Rozinat and W.M.P. van der Aalst. Conformance checking of processes based on monitoring real behavior. *Information Systems*, 33(1):64–95, 2008.
- 59 Bastin Tony Roy Savarimuthu and Stephen Cranefield. Norm creation, spreading and emergence: A survey of simulation models of norms in multi-agent systems. *Multiagent and Grid Systems*, 7(1):21–54, 2011.
- 60 Bastin Tony Roy Savarimuthu, Stephen Cranefield, Maryam A. Purvis, and Martin K. Purvis. Norm identification in multi-agent societies. Discussion Paper 2010/03, Department of Information Science, University of Otago, 2010.
- 61 Bastin Tony Roy Savarimuthu, Stephen Cranefield, Maryam A. Purvis, and Martin K. Purvis. Obligation norm identification in agent societies. *Journal of Artificial Societies and Social Simulation*, 13(4):3, 2010.
- 62 J. R. Searle. *The Construction of Social Reality*. The Free Press, 1995.
- 63 Murat Sensoy, Timothy J. Norman, Wamberto W. Vasconcelos, and Katia Sycara. OWL-POLAR: A framework for semantic policy representation and reasoning. *Web Semantics: Science, Services and Agents on the World Wide Web*, 12-13:148–160, April 2012.
- 64 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.
- 65 Munindar P. Singh. Norms as a basis for governing sociotechnical systems. *ACM Transactions on Intelligent Systems and Technology (TIST)*, pages 1–30, 2013. To appear; available at <http://www.csc.ncsu.edu/faculty/mpsingh/papers>.
- 66 Munindar P. Singh, Amit K. Chopra, and Nirmal Desai. Commitment-based service-oriented architecture. *IEEE Computer*, 42(11):72–79, November 2009.
- 67 R. W. Starreveld, B. de Mare, and E. Joels. *Bestuurlijke Informatieverzorging (in Dutch)*, volume 1. Samsom, Alphen aan den Rijn, 1994.
- 68 Phillip Stoup. The Development and Failure of Social Norms in Second Life. *Duke Law Journal*, 58:331–344, 2008.
- 69 Talis. ODC Public Domain Dedication and License. Technical report, Open Knowledge Foundation, 2008.
- 70 TNS. Two-day batter life tops wish list for future all-in-one phone device. Technical report, Taylor Nelson Sofres, September 2004.
- 71 Tomas Trescak, Marc Esteva, and Inmaculada Rodríguez. A virtual world grammar for automatic generation of virtual worlds. *The Visual Computer*, 26(6-8):521–531, 2010.
- 72 M. A. Vasarhelyi, M. Alles, and A. Kogan. Principles of analytic monitoring for continuous assurance. *J. of Emerging Technologies in Accounting*, 1(1):1–21, 2004.
- 73 Roel J. Wieringa and John-Jules Ch. Meyer. Applications of deontic logic in computer science: a concise overview. In *Deontic logic in computer science: Normative system specification*, pages 17–40. John Wiley & Sons, Inc., New York, USA, 1994.
- 74 Michael Wooldridge. *An introduction to Multi-Agent Systems*. John Wiley & Sons, 2009.
- 75 Konrad Wrona and Petri Mähönen. Analytical model of cooperation in ad hoc networks. *Telecommunication Systems*, 27(2–4):347–369, October 2004.
- 76 Pamela Zave and Michael Jackson. Four dark corners of requirements engineering. *ACM Transactions on Software Engineering and Methodology*, 6(1):1–30, 1997.