Proceedings of the Workshop on

Modelling Policy-making (MPM 2011)

in conjunction with
The 24th International Conference on

Legal Knowledge and Information Systems (JURIX 2011)

December 14, 2011
University of Vienna
Vienna, Austria

Adam Wyner and Neil Benn (eds.)

Department of Computer Science
University of Liverpool

Institute of Communications Studies
University of Leeds

JURIX Conference Committee Chairs

Katie Atkinson	Program Committee Chair
Erich Schweighofer	Organisation Committee Chair

Workshop Co-Chairs

Adam Wyner	University of Liverpool, UK
Neil Benn	University of Leeds, UK

Workshop Programme Committee

Katie Atkinson	University of Liverpool, UK
Trevor Bench-Capon	University of Liverpool, UK
Bruce Edmonds	Manchester Metropolitan University, UK
Tom van Engers	University of Amsterdam, Netherlands
Euripidis Loukis	University of the Aegean, Greece
Thomas Gordon	Fraunhofer FOKUS, Germany
Ann Macintosh	University of Leeds, UK
Gunther Schefbeck	Austrian Parliamentary Administration
Maria Wimmer	University of Koblenz-Landau, Germany
Radboud Winkels	University of Amsterdam, Netherlands

Conference Website

`http://wyner.info/LanguageLogicLawSoftware/?p=1157`

Copyright

Preface

In conjunction with the 24th International Conference on Legal Knowledge and Information Systems Conference (JURIX 2011) in Vienna, Austria, we organised a Workshop on Modelling Policy-making (MPM 2011), held on December 14, 2011. Broadly, the aim of the workshop was to provide a forum in which researchers could present their most recent research on the topics of computational models of policies and the policy-making process. This volume contains the papers presented at the workshop.

As the European Union develops, issues about governance, legitimacy, and transparency become more pressing. National governments and the EU Commission realise the need to promote widespread, deliberative democracy in the policy-making cycle, which has several phases: 1) agenda setting, 2) policy analysis, 3) lawmaking, 4) administration and implementation, and 5) monitoring. As governments must become more efficient and effective with the resources available, modern information and communications technology (ICT) are being drawn on to address problems of information processing in the phases. One of the key problems is policy content analysis and modelling, particularly the gap between on the one hand policy proposals and formulations that are expressed in quantitative and narrative forms and on the other hand formal models that can be used to systematically represent and reason with the information contained in the proposals and formulations. The workshop provided a forum for computer scientists, political scientists, legal professionals and government administrators to discuss these pressing problems related to policy modelling.

The workshop invited submissions of original research about the application of ICT to the early phases of the policy cycle, namely those before the legislators fix the legislation: agenda setting, policy analysis, and lawmaking, seeking to address the problems previously noted. The aim was to solicit papers focusing particularly on using and integrating a range of subcomponents—information extraction, text processing, representation, modelling, simulation, reasoning, and argument—to provide policy making tools to the public and public administrators.

The papers presented at the workshop covered the intended range of approaches and topics, but from diverse research traditions. For example, Benn and Macintosh introduce a tool that visualises the arguments in public-policy debates, thus supporting the increased public understanding of arguments and rationale behind policy decisions. Boer et al. build on the extensive literature in knowledge modelling research to propose a problem solving framework for policy making which posits that policy arguments are a result of a multi-agent model-based diagnosis problem, with participants competing for attention for their diagnostic hypotheses. Szoke et al. tackle the problem of analysing and enriching legal texts and present their prototype system that combines an ontology-based approach to enriching legal texts with a Linked Data approach to connect structured legal texts, ontologies, and rules, to increase the probability of finding relevant legal texts. Wyner et al. also investigate the use of ontologies in their work. Specifically, they present an abstract semantic model and an ontological realisation of this semantic model, which can then be used to underpin tools for automated policy-making. Scherer and Wimmer tackle the problem of bridging text artefacts (where policies are typically expressed) with formal policy models. They present a *meta*-model and a prototype tool that supports the semi-automatic transformation of conceptual models of a policy context to formal policy models. Gordon presents a tool that combines aspects of legal rule-based expert systems with computational models of arguments to help users to analyse and understand the legal effects of alternative policies in particular cases. Gonzalez-Conejero et al. present their work from the emerging discipline of Governance, Risk Management, and Compliance, and introduce a framework that can be used to support organisations with legal compliance of privacy and data security policies. Finally, Engers et al. take us away from computer science concerns and provide us with a necessary alternative perspective of the policy-making process from the administrative science perspective. The provide a timely critique of the rational nature of the policy-making process.

We wish to thank the JURIX conference committee chairs, the workshop program committee members, and all the authors for their contribution to the success of this workshop.

We also thank the IMPACT Project, which supported us while we organised the workshop (FP7-ICT-2009-4 Programme, Grant Agreement Number 247228).

Liverpool, UK and Leeds, UK — A.W. and N.B.
December 2011

Contributed Papers

Using PolicyCommons to support the policy-consultation process: investigating a new workflow and policy-deliberation data model

Neil BENN[1] and Ann MACINTOSH
Institute of Communications Studies, University of Leeds, Leeds, LS2 9JT, UK
{N.J.L.Benn, A.Macintosh}@leeds.ac.uk

Abstract. This position paper describes our ongoing research into developing a Web-based tool for visualizing the arguments in public-policy debates. In our research, we are investigating how such a tool can be used to support the policy-consultation process of both governmental and non-governmental agencies as they operate increasingly in a political environment that demands greater openness, accountability, and public participation in governance. This paper briefly outlines a new policy-consultation workflow making use of our tool and discusses certain constraints regarding the underlying data model of the tool with respect to its use in this workflow.

Keywords. Policy-consultation workflows, argument visualization, online policy-deliberation

Introduction

This position paper describes ongoing research and development on a Web-based tool, PolicyCommons, which is used for visualizing arguments about policies as browsable debate maps. These proposals are typically made by a governmental or non-governmental agency, which then typically engages the services of a policy-analyst to conduct a consultation with members of the public and/or interested stakeholders. In our research, we will investigate how PolicyCommons can be used to support this policy-consultation process.

PolicyCommons is one of a set of tools being developed within the EU-funded IMPACT project[2]. One aim of the IMPACT project is to address the four overarching research and development issues identified in [1]:

1. How can the various actors determine the relationships between contributions to policy development, whether taken from expert papers, consultations or public forum discourse, and appreciate how these contributions are taken through to decisions?

---

[1] Corresponding author.
[2] IMPACT stands for **I**ntegrated **M**ethod for **P**olicy making using **A**rgument modelling and **C**omputer assisted **T**ext analysis: http://www.policy-impact.eu/

2. How can the unstructured text from the various information sources be analyzed to enable the reconstruction of formal arguments?
3. How can the actors understand better what critical questions to ask in order to determine the validity of the information put forward?
4. Given the large, dynamic nature of the information base, how can the actors identify which issues are of importance to them and how can they be supported to make reasoned contribution to the policy development?

The paper begins by briefly describing a possible new and improved policy-consultation workflow that involves the use of PolicyCommons (Section 1). It then goes on to describe technical details of the PolicyCommons tool, particularly how its underlying data model supports the functionality of the tool (Section 2).

1. An improved policy-consultation workflow

In describing our proposed new policy-consultation workflow, we make use of the scenario outlined for the IMPACT project in IMPACT Deliverable D1.1 [2]. In this scenario, the agency initiating the policy-consultation process is the EU Commission who are seeking contributions from the public and interested stakeholders on policy proposals dealing with Copyright in the Knowledge Economy.

Taking this as a starting point for investigating a new policy-consultation workflow, we imagine there is a Policy Analyst called Bernd Gröninger, say, who is given a PolicyCommons account. He creates a Group called "Copyright in EU Knowledge Economy debate contributors" and then creates accounts on behalf of each contributor to the debate (e.g. "Aston University [created by Bernd Gröninger]" and "BBC [created by Bernd Gröninger]"). Note that this is a compromise solution for the current state of development of the PolicyCommons tool. Ideally, in the tool there should be an entity-type called 'Contributor' which is separate from the user-account system (where policy-analysts conducting the consultation sign up).

Continuing with the workflow, at this stage Bernd Gröninger uses the Argument Reconstruction Tool from the IMPACT toolbox to help him extract the arguments from each contribution to the policy consultation and then convert these arguments into the Legal Knowledge Interchange Format (LKIF), an XML exchange format for representing arguments and legal reasoning developed during the EU-funded ESTRELLA Project [3].

Next, Bernd Gröninger creates a new "Debate Map" in PolicyCommons called "Copyright in the EU Knowledge Economy". Currently, "Debate Map" is just a type of node in PolicyCommons, which the user interface will treat in a special manner (in other words, there is no new "Debate Map" entity being created in the back-end, but see next section for discussion of why this is a constraint of the current system and a discussion of alternative solutions).

Next, Bernd Gröninger models the Green Paper on Copyright in the Knowledge Economy in PolicyCommons – specifically he models the Issues raised in the Green Paper which then frame the subsequent debate. As stated in [4], the policy-consultation process begins with the publishing of the Green Paper by the EU. The issues raised in the Green Paper 'seed' the subsequent consultation, bounding what counts as a relevant response from stakeholders.

Finally, Bernd Gröninger imports each contributor's LKIF-represented arguments into PolicyCommons – specifically the arguments are imported into the Debate Map he created previously entitled "Copyright in the EU Knowledge Economy". This map is then published with PolicyCommons, via the IMPACT toolbox architecture, so that the public is able to browse the debate complete with the contributions made by all the stakeholders during the consultation process.

2. Revisiting the underlying data model for representing online policy-deliberation

Rather than develop the PolicyCommons tool from scratch, we have decided to reuse the Cohere platform [5]. Cohere[3] is an open source, Web2.0 tool for argument analysis and argument visualization. We have decided to use Cohere as a platform for the PolicyCommons because it already supports a number of features that we believe the PolicyCommons should provide. These features include enabling users to create Web-based argument maps; to add, delete, and edit nodes and relations in an argument map; and to browse and zoom argument maps, making use of hyperlinks embedded in nodes to access further information (e.g. the original source data from which the node is derived). At the same time, as an open source tool it can be extended to include the new features as envisaged by our specific IMPACT project usage scenarios.

However, the workflow outlined in the previous section requires that we revisit the underlying data model of the platform. As we envisage in the workflow, PolicyCommons is about allowing users to visualize and browse debates of policies. Debates contain issues/statements/arguments (nodes) and links between these. However, the Cohere data model, which PolicyCommons has inherited, only consists of nodes (Ideas) and links (connections). It does not have the idea of nodes and links being contained within a wider debate. This section explores options for overcoming this limitation.

2.1. Modeling 'Debate Map': option 1

The first option is to keep the underlying database schema exactly the same but introduce "Debate Map" as a special Cohere/PolicyCommons node-label and "contains" as a special relation between the new "Debate Map" node-label and other node-labels (such that, e.g., a "Debate Map" contains an "Argument"). Then, whenever the front-end PolicyCommons app retrieves nodes labeled "Debate Map" then these are treated differently, purely in terms of visualization, to any other node label[4].

The main advantage of this option is that no changes have to be made to the underlying database schema or to the built-in Cohere API that accesses the database. Another advantage is that should PolicyCommons need to share data with the original Cohere application then this can be done easily since the two applications would still be using the same underlying data model.

[3] http://cohere.open.ac.uk/

[4] Of course, if the original Cohere front end is used then "Debate Map" is just displayed the same as any other node label

2.2. Modeling 'Debate Map': option 2

The second option is to create a new entity in the underlying data model called "Debate Map" (so that now, in addition to things called Nodes and Connections in the data model, we would now have something called Debate Maps). This would then mean changing the database schema so that there is a table of Debate Maps and the necessary relationships between the Nodes and Connections tables and this new maps table. The specifics of achieving this could even be adapted from the Compendium system[5] since the Compendium data model has the idea of maps, which contain other elements.

The main advantage of this option is that now we have the flexibility of giving Maps any attributes that we choose – e.g. contributors, status (such as 'draft', 'public', 'closed'), and recent updates (which is useful for generating RSS feeds of a debate). The main disadvantage of this option is that there is the initial cost of updating the database schema and updating the API that accesses the underlying database. Also, it would no longer be possible to directly exchange data with the original Cohere application. It would not be impossible to share data at all, but now we would have to do some minor transformation to make the PolicyCommons model compatible with the Cohere model.

3. Conclusion

This paper briefly described ongoing research and development on a Web-based tool, PolicyCommons, which is used for visualizing arguments about policies as browsable debate maps. The paper outlined a new policy-consultation workflow making use of PolicyCommons. Finally, the paper discussed certain constraints regarding the underlying data model of the tool with respect to its use in this workflow and how these constraints might feasibly be overcome.

References

[1] Macintosh, A. (2009). Moving Towards "Intelligent" Policy Development. *IEEE Intelligent Systems , 24* (5), 79–82.

[2] IMPACT Project. (2011). Deliverable D1.1 - Requirements Analysis Report. http://www.policy-impact.eu/file-cabinet/IMPACT-D1.1.pdf?attredirects=0

[3] ESTRELLA Project. (2008). *Deliverable 4.1 – The Legal Knowledge Interchange Format (LKIF).* http://carneades.berlios.de/files/LKIF-Specification.pdf.

[4] Benn, N., & Macintosh, A. (2011). Argument Visualization for eParticipation: Towards a Research Agenda and Prototype Tool. In E. Tambouris, A. Macintosh, & H. de Bruijn (Ed.), *ePart 2011* (pp. 60-73). Berlin: Springer.

[5] Buckingham Shum, S. (2008). Cohere: Towards Web 2.0 Argumentation. *2nd International Conference on Computational Models of Argument (COMMA '08)* (pp. 97–108). IOS Press.

[5] http://compendium.open.ac.uk/

A Problem Solving Model for Regulatory Policy Making

Alexander BOER [a,1], Tom VAN ENGERS [a], and Giovanni SILENO [a]
[a] *Leibniz Center for Law, University of Amsterdam, The Netherlands*

Abstract.
In this paper we discuss how the interests and field theory promoted by public administration as a stakeholder in policy argumentation, directly arise from its problem solving activities, using the framework for public administration problem solving we proposed in [1,2]. We propose that calls for change of policy in public administration mainly arise from model-based diagnosis problem solving.

1. Introduction

Most of us have experience with policy making processes. Policy making covers anything from planning family holidays to deciding on tax evasion and CO_2 emissions. A policy is a commitment to a rule or principle that guides decisions during the period that the policy is adopted. The regulatory instrument, the law, is one of several types of policy instrument used by government.

Policy making is based on a policy field theory and a policy effects theory. The first describes the causal relationships between phenomena that play a role in the problem domain, while the latter describes the instruments that could be used to influence the problem domain. Both the policy field theory and the policy effects theory are subject to argumentation. In policy making processes the stakeholders involved have different interests and perspectives, and the optimal policy is usually a compromise between various interests. Still, argumentation from adversarial positions is not the sole useful abstraction of policy making. Instead of reconstructing field theories from arguments put forward by stakeholders, we try to show in this paper how the interests and perspectives that public administration, as a stakeholder, promotes in policy argumentation, arise from its problem solving activities, using the framework for public administration problem solving that we proposed in [1,2], and our forthcoming Jurix 2011 paper.

In policy areas where regulatory solutions play a key role, policy field and policy effects theories prominently address agents, behavior, norms, beliefs, desires and intentions, because we use these concepts to model the processes directly affected by regulation. Since agents produce the effects of regulatory policy, the multi-agent approach, proposed in [1] to explain expectations about the effects of alternative policies, makes obvious sense. In this paper we discuss our three-level problem-solving model for public administration, as a complex agent acting in an environment with other agents, addressing

[1] Corresponding Author: E-mail: A.W.F.Boer@uva.nl.

the dynamics of government and public administration policy making from the point of view of the organization. Our framework for public administration is based on the *suite of problem types* in [3], an analysis of generic problem and task decompositions found in knowledge-based system literature [4,5,6], combined with the multi-agent paradigm, and a stratification of knowledge into three interacting spheres of activity. We discuss these elements in the next sections, concluding in the final section that policy making is driven by model-based diagnosis.

2. Three Spheres of Activity in Public Administration

We distinguish three abstraction levels of action and problem solving in public administration. The three levels represent in the first place three spheres of activity, based on different conceptualizations of the domain, and often with different participants. The three levels are only imperfectly reducible to each other, as is generally the case with decompositions of nontrivial problems. From a managerial perspective it is plausible to see product and process development as a result of implementation of legislation, and activities on the work floor as the result of implementation of products and processes. The model is suggestive of cybernetic control models of organizations in administrative science, like the influential model of Stafford Beer in [7], familiar in management circles, but these models have a different origin, and are based on scientific claims of another kind. We only claim these three perspectives on the organization prominently exist.

Fig. 1 reflects our interpretation of development and policy making as core activities of the organization. Public administration and policy makers are mutually dependent. Public administration is in charge of efficient operationalization, and the policy maker wants effective rules. Process and product development is a core activity in public administration that consumes a large and increasing part of its resources. Due to systematic overspending on IT projects and reorganizations, and slow policy making processes, the interaction between both deeper spheres of activity attracts increasing attention.

At the shallowest level of problem solving – *operations* – a plan selection based on service request classification takes place. This is a routine approach to the interpretation/modeling problem of [3]: the domain conceptualization is selected, not constructed. This is a purely reactive adoption of predefined agent roles based on communication events like the client service request that the agent reacts to. The agent role adopted determines the problem definition for planning service delivery, and the suspicious events to monitor for. Suspicious events, if any arise, determine the problem definition for diagnosis, and the diagnosis determines the problem definition for a refined interpretation of what is going on. The case may for instance be reclassified as a potential legal nonconformance problem. Diagnosis may however also trigger a deeper mode of problem solving, on the product and process development level.

On the *product and process development* level, design and resource scheduling take place, progressively refining agent role descriptions, knowledge sources used by agents, and systems, structures, and other resources supporting agent activity. Development activity is often triggered by diagnostic processes on the operational level, and translated into requests for reallocation of resources or change of agent role descriptions, supporting knowledge sources, systems, and structures on the design level.

When design level problems cannot be solved within the constraints of existing policy, diagnostic processes on the *policy making level* may lead to proposals for new policy.

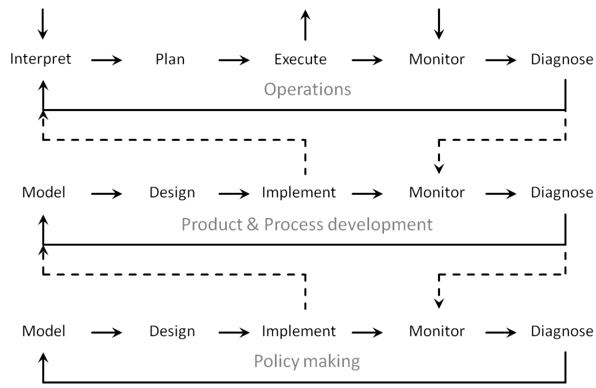

Figure 1. The *operations*, *development*, and *policy making* problem solving cycles.

This is usually an informal line of communication (in the sense of usually being outside the scope of communication models of the development departments) from a committee of domain experts responsible for analysis of legal problems to the competent legislative drafting department and the responsible minister that has to sponsor the proposal.

3. Generalized Problem Solving Activity

The function of the suite of problem types in [3] is primarily to classify knowledge sources used in the organization as belonging to, and taking their requisite structure from, a generic problem definition. [3] argues (amongst others) that:

1. the availability of structural and behavioural models in a domain determines which problems can be defined and solved, and
2. there are recurrent functional dependencies between problem types characterized by the (input and output) roles that knowledge plays in the problem definition.

The suite of problem types presents us with a generic problem solving cycle, and two different vocabularies for it, depending on the type of model of the domain, design-oriented and planning-oriented, that is available:

1. *Model* → *Design* → *Implement* → *Monitor* → *Diagnose/Assess*: When we feel able to control a domain by encapsulating processes into fixed structures and resources, we decide on a *design* for dealing with a type of problem, and implement that design.
2. *Interpret* → *Plan* → *Execute* → *Monitor* → *Diagnose*: When we feel this is not feasible, given the characteristics of the domain or problem, we make situation-specific *plans* to address a problem, using any fixed structures and processes available.

Problem solving activities will partially be automated, and partially allocated to human experts. Often we see a distinction between a heavily automated *happy flow* that deals with the great bulk of service requests, and manually handled hard cases that drop out of the happy flow. Moreover, the first half of the problem solving cycle dominates in task modeling: automated planning and scheduling systems without automated diagnosis

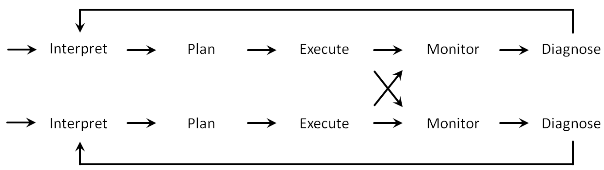

Figure 2. Generic relationship between problem solving activities, without the assumption of control.

functions to deal with important failure modes are common in practice. The happy flow tends to be better modeled by design departments, even if it is not where the organization actually spends most of its resources. The happy flow gives the best return on investment for design activity in the operations sphere.

4. Agents, Multi-agent Systems, and Control

The sort-of problem decomposition into three spheres of activity is based on the assumption that deeper levels have access to, and control over, the diagnosis and modeling problem definitions at shallower levels (the dashed lines in Fig. 1). In other words, it assumes that at shallower levels diagnostic feedback is not a result of planning with the intent to influence problem definitions at deeper levels. Communication between levels is fundamentally asymmetric: the deeper levels directs, and the shallower level informs. Executive (sub)organizations are not naturally conceived of as stakeholders and game players in policy making. Fig. 1 idealizes the organization as one single cybernetic organism capable of learning new policies. The intentional, argumentative aspect of diagnosis feedback from shallower levels is obscured by an illusion of control, over employees, over departments, etc.

Fig. 2 gives a more general representation of the relation between problem solving processes. It applies to the organization in relation to its clients and network partners in the environment, but also to the agents in the organization, and in the state as a whole, who may all work to influence operations, development, and policy making.

The product and process development suborganizations, as maintainers of the existing structures and systems, most knowledgeable on sunk costs in the state's organizations, are legitimate stakeholders in policy making. A theory of public administration's influence in policy making would be based on an inversion of control arrows in Fig. 1, making the policy making sphere the playing field, in which operational and development units of the organization exercise influence through argumentation, about the policy field, and policy effects.

5. Conclusions

To us, our problem solving framework suggests that policy arguments are a result of a multi-agent model-based diagnosis problem, with participants competing for attention for their diagnostic hypotheses. From the perspective of public administration, the field theory underlying policy making problems is produced in two different spheres of action, operations and development, that draw attention to specific problematic cases, and to system-level performance constraints, respectively. Policy arguments made by public administration often boil down to 1) problems distinguishing between diagnostic hypotheses in operations due to lack of evidence, as argued in [2], or 2) a desire in development units to extend the potential performance envelope in current operations.

The viable system model in [7] gives a definite meaning to productivity, potentiality and performance in this context, proposing that *normative planning*'s main purpose in organizations is to push the potential performance envelope. The viable systems model in [7] is representative of the managerial, normative, rational, control-oriented view on policy making in organizations. In government regulatory policy competing interests play a role, but public administration's role as stakeholder in policy making prominently includes performance. The organization may of course choose to represent the interests of its network partners and clients, and it often does, but its perspective on the field will be limited by the boundaries of the system it monitors, and its methods of sampling information from that system.

Policy argumentation frameworks, like the recent work reported in [8], are more realistic about the policy production process, and the competing interests and perspectives that play a role, but tend to obscure the origin – in other problem solving activities – of policy field and policy effect theories.

In this paper we relate the two perspectives in a unifying concept, pointing out that the main difference is the assumed direction of control in Fig. 1, and that the policy field theory is a derivative of knowledge used in diagnosis, including its biases towards evidence, as we pointed out in [1]. A major part of the return of investment for model-based diagnosis design activity in public administration is generated in the development and policy making spheres.

References

[1] Boer, A., van Engers, T.: An agent-based legal knowledge acquisition methodology for agile public administration. In Ashley, K.D., van Engers, T.M., eds.: The 13th International Conference on Artificial Intelligence and Law, Proceedings of the Conference, June 6-10, 2011, Pittsburgh, PA, USA, ACM (2011)

[2] Boer, A., van Engers, T.: Diagnosis of multi-agent systems and its application to public administration. In Abramowicz, W., ed.: Business Information Systems Workshops - 14th International Conference, BIS 2011, Poznan, Poland, June 15-17, 2011. Volume 97 of Lecture Notes in Business Information Processing, to appear., Springer (2011)

[3] Breuker, J.: Components of problem solving and types of problems. In Steels, L., Schreiber, G., Van de Velde, W., eds.: A Future for Knowledge Acquisition. Volume 867 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (1994) 118–136

[4] Chandrasekaran, B., Johnson, T.R.: Generic tasks and task structures: History, critique and new directions. In David, J.M., Krivine, J.P., Simmons, R., eds.: Second Generation Expert Systems. Springer Verlag (1993)

[5] Clancey, W.J.: Heuristic classification. Artificial Intelligence **27** (1985) 289–350

[6] Steels, L.: Components of expertise. AI Mag. **11**(2) (1990) 30–49

[7] Beer, S.: Diagnosing the system for organizations. The Managerial cybernetics of organization. Wiley (1985) Stafford Beer. ill. ; 24 cm. "Companion volume to Brain of the firm and The heart of enterprise." Includes index.

[8] Cartwright, D., Atkinson, K.: Using computational argumentation to support e-participation. Intelligent Systems, IEEE **24**(5) (sept.-oct. 2009) 42 –52

Linking Semantic Enrichment to Legal Documents

Ákos SZŐKE [a,b], András FÖRHÉCZ [a], Krisztián MÁCSÁR [a], György STRAUSZ [b]
[a] *Multilogic Ltd., Budapest, Hungary*
[b] *Dept. of Measurement and Information Systems*
Budapest University of Technology and Economics, Budapest, Hungary.

Abstract. Regulations affect every aspect of our lives. Policy-makers are usually facing the problem of finding the relevant parts of the regulations with their impacts in order to change the legal texts consistently and coherently. To propose a solution to these demands, we present a semantic enrichment approach which aims at 1) decreasing the ambiguousness of legal texts, 2) increasing the probability of finding the relevant legal materials, and 3) utilizing the application of legal reasoners. To evaluate the the approach, a preliminary study was carried out in a large organization.

Keywords. legal xml, ontology, SWRL, rdf, semantic enrichment, linked data

1. Introduction

Regulations affect every aspect of our lives, from education to health, environment, privacy and democracy itself. Considering private sector, corporate laws within large multinational organizations are also increasing. Despite there are attempts to increase harmonization and freeing from regulations by deregulation, the extent and complexity of law constitute an extremely complex legal environment [1,2].

1.1. Related Work

National governments and large organizations have been published laws on the web or intranets in order to provide information for their citizens, businesses and their employees [3,4,5]. Although these web sites are great legal sources, they usually do not provide easy access to the relevant information for the enquirer.

Metadata schemes attempt to solve this problem. Some public prominent examples are the British Legal and Advice sectors Metadata Scheme (LAMS) [6] and the Australian Justice Sector Metadata Scheme (JSMS) [7]. The problem of these initiatives is twofold. First, they apply fixed categorizations on documents ordered by legislative domain, and they do not consider the cross branches of law. Second, they do not consider the structure of the document which not only set back the cooperation of different government bodies during drafting (e.g. referencing is difficult) but (semi-) automatic processing of the documents is also hindered.

The efficiency of information processing of legal documents can be improved by applying XML technology. It not only provides extensible structure encoding of the contents of legal documents, but supports enriching documents with metadata. It enables smart applications such as intelligent data retrieval. Two prominent initiatives are CEN MetaLex [8] and the Akoma Ntoso [9]. MetaLex also includes an OWL scheme that enriches the documents with bibliographic concepts and metadata.

1.2. Motivation

Policy-making is a process by which governments / corporations translate their vision into programmes and actions. Policy can take a range of different forms, including non-intervention, licensing, regulation, encouragement of voluntary change as well as the direct service provision. To be effective, policy-making usually involves finding the relevant part of the regulation, finding their precise meaning and their connection with other parts of the legal body. These requirements ensure that new policies can be inserted into the appropriate parts, they can be adapted to the context, and their effects can be reviewed and evaluated to guarantee coherency and consistency.

Regarding policy development or review and considering the previous requirements, we identify the following actors: *legislator*, *civil servant*, *legal professional* and *business person*. Their needs can be summarized as:

N1. *Advanced search in the body of law*: Any ordinary citizen/business person need fast and easy access to laws, regulations or judgments for their own purposes.

N2. *Precise understanding of the law*: Although legislations are drafted very carefully by civil servants/business persons, the understanding of the text depends on their choice of words and on the consistent usage of them.

N3. *Finding related legal text or materials for a law*: For a legal professionals / business person who executes regulations the laws are not self-explanatory, and they usually consult a wide variety of additional sources to interpret them.

N4. *Consulting with a legal expert about a legal question*: There are cases when the arising legal question is complex and requires consultation with a legal expert.

The previously mentioned approaches (e.g. Akoma Ntoso, MetaLex) fulfill **N1**, but **N2-N3** are just partially supported. Although, these approaches can provide information how to interpret a legal text, (logically) precise understanding (**N2**) cannot be achieved with their apparatus. To achieve this aim, we propose *i*) a hybrid knowledge base-based enrichment method of legal texts and *ii*) a linking method to increase the probability of finding the relevant legal materials (**N2**) and their formal descriptions (**N3**). Moreover, on this basis, application of legal reasoners can be utilized to provide computer-aided answers to arising legal questions (**N4**).

The rest of the paper arranged as follows: Sec. 2 presents the enrichment approach to legal texts; Sec. 3 introduces the linking method; and finally Sec. 4 concludes the paper.

2. Legal Text Enrichment

In our approach, the expressiveness and semantic richness of a legal text are increased in two steps: adding formal description of terminology using ontologies and then adding logical rules to legal text. These steps are detailed in the next sections.

2.1. XML Representation

Without loss of generality, we can assume that legal documents conform to an XML Schema (e.g. MetaLex). Otherwise they can be converted semi-automatically [9]. Schema-based (or simply *structured*) legal texts provide a technology-neutral electronic representation of legal text. These approaches structure the content into meaningful elements that can be read and understood by software applications (i.e. machine readable) and humans as well. A legal text excerpt is shown in Code 1. (N.B. `lang` denotes it is in Hungarian, and `id` is an identifier of the text fragment.)

```
<Sentence>
    <TextVersion xml:lang="hu" id="ca527">
        A bírálati folyamatban lévõ másik ügylet az </TextVersion>
    <TextVersion xml:lang="hu" id="ca528">
        összkockázati kitettség </TextVersion>
    <TextVersion xml:lang="hu" id="ca529">
        ben figyelembevételre kerül.</TextVersion>
</Sentence>
```

Code 1: An XML snippet that represents an excerpt of a legal text.

2.2. Enriching with Ontologies

The legal text completion with ontologies is decreasing the ambiguousness of legal texts (i.e. **N2**). Ontologies that are described using OWL2 can be processed by a description logic (DL) reasoner (i.e. **N4**). Major services offered by these reasoners is *subsumption testing* (whether or not a concept is a specialization of another concept) that can provide an inferred ontology class hierarchy from an asserted one, and *consistency checking* (whether or not all concepts refer to valid, non-empty categories). These services provide mathematical precise definitions of legal terms and their relations.

Figure 1 shows an ontology that was constructed interpreting a corporate law of an international bank. This figure indicates hierarchical ($is-a$) and equivalent ($\equiv$) relationships between terms, which are narrower and broader or equivalent in scope. These properties are derived from their definition and with the help of the DL reasoner's subsumption service. The textual definition of the Hungarian term ('ÜgyfélKitettség' – which is equivalent to 'ÖsszKitettség') is shown in the lower right corner.

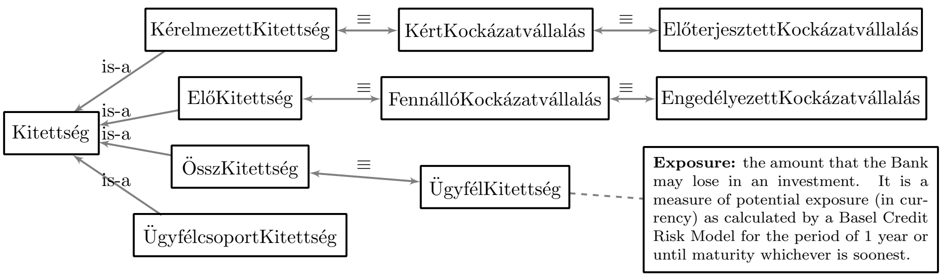

Figure 1. Some concepts of a regulation represented as an OWL2 ontology.

2.3. Enrichment with Logical Rules

One of the most fundamental distinctions in legal theory is that between 'positive' and 'normative' legal theory. The main important characteristic of the latter one is that they can be *evaluated* by their nature [10]. As a consequence, the normative legal statements can be modeled with mathematical logic rules.

As a second step of our enrichment, we propose building logical rules on top of ontologies. This hybrid approach (such as SWRL) enables powerful rules based on a combination of DL-safe Horn rules and description logic that can be considered as a straightforward extension of the semantics for OWL2 [11,12]. By building rules on top of ontologies not only provides powerful reasoning capabilities (i.e. **N4**), but it enables linking normative legal statement to logical ones to clarify the texts (i.e. **N2,N3**).

The following code snippet continues the previous example. It expresses with normalized SWRL syntax how the total exposure ('ÖsszKitettség') is calculated according to the text. Namely, if the client is retail one (SME or private person) and the actual exposure is `?value1`, the requested exposure is `?value2` then the total exposure is their sum. (Please note, `Rule-1` is the identifier of the SWRL rule.)

```
Rule-1:
  RetailÜgyfel(?x) and
  ?x.élõKitettségÖsszege = ?value1 and
  ?x.kérelmezettKitettségÖsszege = ?value2
    => összKitettségÖsszege(?x, ?value1 + ?value2)
```

Code 2: An SWRL snippet that represents a normative legal rule.

3. Linking the Elements

The Linked Data paradigm is powerful enabler for the transition of the current document-oriented web into a web of interlinked data [13]. The idea of Linked Data can be utilized to connect the previously mention structured legal texts, ontologies and rules (i.e. resources) to increase the probability of finding the relevant legal materials (i.e. **N1,N3**). First, it requires the ability to refer resources and their parts with URIs. Second, it mandates a mechanism that can connect this heterogeneous information.

RDF data model is designed for the integrated representation of heterogeneous information. It represents information as statements, which called RDF triples. A triple consists of three parts – as subject, predicate and object – and follows the structure of a simple sentence. In the legal context, this data model can be utilized as a link that connects resources. Continuing the previous examples, a link can be established between a term of a legal text ('összkockázati kitettség') and the related ontology entity (Code 3):

```
<base URI#ca528> <base URI#relatedTerm> <base URI#ÖsszKitettség> .
```

Code 3: An RDF statement that represents a link between a legal term and an ontology entity.

In the previous example, the `relatedTerm` link type is used. However, any types of a link can be defined among documents, ontologies and rules.

4. Conclusions

Policy-makers are expected to know the laws that are under revision. Although governments and organizations have been published laws on the web, they usually do not provide easy access to the relevant information.

Considering the identified actors needs (**N1-N4**), we propose a hybrid knowledge base-based enrichment method of legal texts and a linking method. They 1) increase the probability of finding the relevant legal materials (**N1,N2**), 2) find formal descriptions of legal sources (**N3**), 3) and can utilize legal reasoners (**N4**). Additionally, as a triple having identifiers from different sources at subject and object position, and a predicate identifies the type of link, we can visualize the linkage of resources. Therefore, this graphical view can provide a visual overview. Moreover, since we apply RDF as a data model of linking it makes it possible to use standard query language technologies such as SPARQL (**N1**).

To obtain a proof-of-concept, we implemented a prototype in GWT technology [14] – named Emerald. With this prototype, we carried out a preliminary study on corporate law documents of an international bank. The study included 22 different type of documents with different languages and time versions and more than 3K links. The related ontologies contained more than 200 OWL classes with their object / data properties.

We think that our proposed method is a plain combination of the present theories and methods thus it leads us to generalize our findings beyond the preliminary study. However as we conducted a simple study, further examination of the method is recommended.

Acknowledgements. The work is supported in part by the KMOP-2009-1.1.1. grant.

References

[1] A. Boer, R. Hoekstra, R. Winkels, T. van Engers, and F. Willaert. ^{META}lex: Legislation in XML. In *Jurix 2002: The Fifteenth Annual Conference*, Frontiers in AI and App., pages 1–10, 2002. IOS Press.

[2] Rinke Hoekstra. The MetaLex Document Server - Legal Documents as Versioned Linked Data. In *Proceedings of the 10th International Semantic Web Conference ISWC 2011*, page 16. Springer, 2011.

[3] Dutch government website. `http://www.overheid.nl` Accessed on 10 Oct 2011.

[4] Hungarian govermental portal. `http://www.magyarorszag.hu` Accessed on 10 Oct 2011.

[5] The UK national archive. `http://www.legislation.gov.uk` Accessed on 10 Oct 2011.

[6] Advice Services Alliance. Legal and advice sectors metadata scheme (LAMS), 2010. `http://www.advicenow.org.uk` Accessed on 10 Oct 2011.

[7] Department of Attorney General and Justice. Justice sector metadata scheme (JSMS), 2010. `http://www.lawlink.nsw.gov.au` Accessed on 10 Oct 2011.

[8] European Commitee for Standardization. MetaLex (open xml interchange format for legal and legislative resources), 2010. `http://www.metalex.eu/` Accessed on 10 Oct 2011.

[9] UN Department of Economic and Social Affairs (UNDESA). Akoma ntoso: XML for parliamentary, legislative and judicuary documents. `http://www.akomantoso.org` Accessed on 20 Oct 2011.

[10] R. West. *Normative Jurisprudence: An Introduction*. Cambridge Introductions to Philosophy and Law. Cambridge University Press, 2011.

[11] Peter F. Patel-Schneider et al. OWL 2 Web Ontology Language Direct Semantics, 2009. `http://www.w3.org/TR/2009/REC-owl2-direct-semantics-20091027/` Acc. on 10 Oct 2011.

[12] Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, and Mike Dean. SWRL: A semantic web rule language combining OWL and RuleML. Technical report, W3C, 2004.

[13] Sören Auer, Jens Lehmann, and Axel-Cyrille Ngonga Ngomo. Introduction to linked data and its lifecycle on the web. In *Proceedings of the 7th int. conf. on Reasoning Web 11*, pages 1–75, 2011. Springer.

[14] Google Web Toolkit. `http://code.google.com/webtoolkit/` Acc. on 30 Oct 2011.

Semantic Models and Ontologies in Modelling Policy-making –A Position Paper–

Adam WYNER [1], Katie ATKINSON, and Trevor BENCH-CAPON
Department of Computer Science,
University of Liverpool, Liverpool, United Kingdom

Abstract. In modelling policy-making, we propose the use of formal semantic models and ontologies to structure the analysis of informal policy statements, specify the domain, generate the logical space of arguments and counter-arguments about a policy proposal, and underpin tools for automated policy-making.

Keywords. policy-modelling, semantics, ontologies

Introduction

In this position paper, we discuss the role of semantic models and ontologies in modelling policy-making. Policy making can be viewed as a cyclical, multi-stage process [1,2] with several stages: evaluation, agenda setting, policy formulation, decision, implementation. We focus on the policy formulation stage, where the policy proposal is set and critiqued.

While current approaches make some use of IT, e.g. e-petitions to get feedback on a policy proposal, these tend to be too *coarse-grained* to identify what, in particular, citizens agree or disagree with. Moreover, there is a lack of *structure* to the proposal in the sense that the components and their relationships have to be identified. Consequently, it is difficult to apply automated techniques and tools to support policy formulation, which would make it more efficient, transparent, and accessible.

In recent work [3,4], formal semantic models are introduced to represent knowledge about the policy domain and to provide components that are used in forming practical reasoning arguments about policy proposals. Such arguments are extracted from comments on policy proposals, formalised, then presented to users in a web-based policy consultation tool, where users are queried about their opinions about particular components of the policy proposal [5].

In our view, formal semantic models are advantageous in that they provide a precise basis for organising, distinguishing, and reconciling the diverse opinions

[1] Corresponding Author: University of Liverpool, Department of Computer Science, Liverpool, United Kingdom; E-mail: adam@wyner.info

from the original documents, for constructing initial arguments based on this collective understanding, for evaluating these arguments by identifying conflict and inconsistency, and for assimilating additional information. It disciplines the policy analysts' analysis of the source material by providing a clear, specific structure into which the arguments from the source must be cast, thereby clarifying alternatives and making implicit information explicit. Furthermore, the model, once given, can be used to generate the logical space of arguments found in the source materials along with their systematic inter-relationships. The semantic model also supports and makes transparent the policy analysts' evaluation of given arguments in light of the responses to the survey, which may endorse, oppose, or introduce further particular elements of the arguments. A formal model specifies a computer program that can generate arguments and their relationships, allowing in-depth representation and automated reasoning. In sum, the semantic model helps us to have a greater understanding of the meaning and implications of the policy as well as how they might specifically critique or contribute to it.

Aligned with the semantic model, we propose to present an ontology for the domain, where the ontology is systematically derived from the semantic model. The ontology has been used to generate arguments using the *Legal Knowledge Interchange Format* (LKIF) [6], an XML specification for the representation of legal rules, and Carneades [7], a tool which supports argument representation and evaluation. The role of the ontology is to act as a bridge between the formal model and the executable code. This bridge could also be realised in other ways, such as a database or even directly in code, e.g. Prolog.

While we have some understanding of the semantic models and ontologies, we intend to deepen and broaden it, elaborating on the derivations, relationships, and applications of the models and ontologies as well as extending our current representations to incorporate models and ontologies for related aspects of policy-making such as supporting arguments concerning credible sources of information and causation. In the following sections, we outline elements of our current understanding of the formal semantic model and ontology, while at the workshop, we propose to present elaborated and extended analyses.

1. Practical Reasoning and a Semantic Structure

In public policy discussions, participants recommend and justify what should be done. We can represent such arguments with the practical reasoning argumentation scheme (PRAS) [8,9], wherein the proponent justifies an action:

> **PRAS**: In the current circumstances (R), action Ac should be performed, since this will bring about a new set of circumstances, the consequences (S), in which some goal (g) is realised. Goal g is desirable as it promotes a particular social value (v).

To give a formal semantic basis to the scheme, we make use of a computational model based on the Action-Based Alternating Transition System with Values (AATS+V) [10,9,3]. Here we provide an informal statement of the AATS+V and its association with the PRAS (see [3] for a full, formal specification and a range of examples), where main elements of the structure are:

- Q, a set of *states*, where a state is a consistent conjunction of literals. Current circumstances S and consequences R in the PRAS are states.
- Ag is a set of *agents*, Ac_i is the set of *actions* available to a particular agent, ag_i, and J is the set of *joint actions*, assuming agents execute actions jointly. In the PRAS, agents appear as part of Ac.
- The *state transition function* defines the state that results from the execution of each joint action in a given state. Transitions are implied in the PRAS by the change from circumstances to consequences.
- A *goal* is a literal that holds after execution of a joint action, where the negation of the literal holds in a circumstance before execution.
- V is the set of *values* relevant to the scenario.
- The *valuation function* defines the status (promoted +, demoted -, or neutral =) that labels the transition between the two states. The values and function are given in the PRAS.

The AATS+V is an abstract semantic structure that we have instantiated [5]. However, to operationalise and automatically reason with the arguments generated from the semantic representation, we must represent the instantiated semantic model in a machine readable form.

2. The Ontology

We have represented our instantiated semantic model in an OWL ontology using the *Protege* ontology editor and knowledge acquisition system. For the purposes of this position paper, we do not give a full presentation of the ontology, but indicate the elements of the semantic model (given in *italics*) followed by ontological classes with data and object properties (indicated in **bold**), briefly discussing the relationship between elements of the semantic model and the ontology.

- Q - class **State** of individuals such as q_0,...,q_6. Every element of Q is an individual in **State**.
- q_0 - class **CurrentState** with a single individual. The class has **hasTime** data property with range to type **string** set to value 0 and an object property **stateRelation** with domain **CurrentState** and range **State**. The object property associates the **CurrentState** individual with a **State** individual, thus indicating the time of the **State** individual.
- Ag - For each agent in Ag, we have an object property with domain **JointAction** and range **Action**, for each of the available action subclasses, e.g. j_0 **jointActionGovernment governmentAction1**.
- Ac_i - **Action** with disjoint subclasses for each agent in Ag, e.g. **Government**, and given individuals. These subclasses have a data property **hasAction** with domain **Action** and range of type **string**.
- *System transition function* τ - a class **Transition** with object properties **transitionJointAction** with range **JointAction**, **transitionSource** with range **State**, and **transitionTarget** with range **State**.
- *Action pre-condition function* ρ - this is expressed derivatively in the ontology as individuals of **Action** do not themselves have preconditions. The

class **Transition** has an object property **transitionJointAction**, which gives the joint actions associated with individuals in **Transition**. **JointAction** has an object property that identifies the actions associated with the **JointAction**, e.g. **jointActionNature natureAction1**. In addition, the class **Transition** has object property **transitionSource**, which gives an individual of **State** as source of an individual in **Transition**. Thus, individuals of **Action** implicitly are associated with a source state in virtue of object properties to transitions.

- *Atomic propositions* - a data property with range **State** and range type **boolean**. For each atomic proposition of ϕ, there is a separate data property, e.g. **hasPropertyP**.
- *Interpretation function* π - this is given by assignment of a boolean value to each of the data properties associated with each proposition with respect to a state, where **hasPropertyP** is a data property with domain **State** and range **boolean**. For example, q_1, an individual from **State**, has the **hasPropertyP** data property set to **false**.
- *Joint actions* - a class **JointAction** with several joint action object properties, one for each of the **Action** subclasses associated with an agent, e.g. **jointActionGovernment governmentAction1**.
- V - a class **Value** that has individuals such as **Budget**, **Freedom**, etc.
- *Valuation functions* δ - demotes and promotes object properties from **Transition** to **Value**.

One of the key differences between the AATS+V semantic model and the OWL ontology is that the ontology can only be expressed within the constraints of Description Logics, while the AATS+V is expressed in First-order Logic and functions. In addition, not every aspect of the semantic model is translated directly into the ontology, e.g. the Action pre-condition function has no direct correlate in the ontology. As well, though the semantic model represents propositions and their interpretation (truth-value) as is standard in logic, there are alternative ways to achieve the same representation in the ontology. These (and other) issues remain to be explored further.

Here we have presented the ontological realisation of the semantic model. Other operational representations are possible in a database or in a Prolog program, offering different *views* on the semantic information and supporting different *functionalities* relative to purpose.

3. Conclusion

In this position paper, we have outlined some of our motivations and uses for a semantic model and ontology for policy-making. Some of the formal elements are sketched. In future work, we deepen and broaden our understanding of representations of policy-making, for example, by adding argumentation schemes for causation or credible source along with supporting semantic models and ontologies.

Acknowledgements

This work was partially supported by the FP7-ICT-2009-4 Programme, IMPACT Project, Grant Agreement Number 247228. The views expressed are those of the authors and are not necessarily representative of the project.

References

[1] Bench-Capon, T.J.: Knowledge based systems applied to law: A framework for discussion. In Bench-Capon, T.J., ed.: Knowledge Based Systems and Legal Applications. Academic Press (1991) 329–342

[2] Birkland, T.A.: An Introduction to the Policy Process: Theories, Concepts, and Models of Public Policy Making. M.E. Sharpe (2010)

[3] Atkinson, K., Bench-Capon, T., Cartwright, D., Wyner, A.: Semantic models for policy deliberation. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Law (ICAIL 2011), Pittsburgh, PA, USA (2011) 81–90

[4] Wyner, A., Atkinson, K., Bench-Capon, T.: Towards a structured online consultation tool. In: Electronic Participation - Third International Conference (ePart 2011). Volume 6847 of Lecture Notes in Computer Science (LNCS)., Berlin, Germany, Springer (2011) 286–297

[5] Pulfrey-Taylor, S., Henthorn, E., Atkinson, K., Wyner, A., Bench-Capon, T.: Populating an online consultation tool. In Atkinson, K., ed.: Proceedings of the 24th International Conference on Legal Knowledge and Information Systems (JURIX 2011), IOS Press (2011) to appear.

[6] Gordon, T.: Constructing legal arguments with rules in the legal knowledge interchange format (LKIF). In P. Casanovas, N. Casellas, R.R., Sartor, G., eds.: Computable Models of the Law. Number 4884 in Lecture Notes in Computer Science. Springer Verlag (2008) 162–184

[7] Gordon, T., Prakken, H., Walton, D.: The carneades model of argument and burden of proof. Artificial Intelligence **171** (2007) 875–896

[8] Walton, D.: Argumentation Schemes for Presumptive Reasoning. Erlbaum, Mahwah, N.J. (1996)

[9] Atkinson, K., Bench-Capon, T.J.M.: Practical reasoning as presumptive argumentation using action based alternating transition systems. Artificial Intelligence **171**(10-15) (2007) 855–874

[10] Wooldridge, M., van der Hoek, W.: On obligations and normative ability: Towards a logical analysis of the social contract. Journal of Applied Logic **3**(3-4) (2005) 396–420

Conceptual Models Supporting Formal Policy Modelling: Metamodel and Approach

Sabrina SCHERERand Maria A. WIMMER
University of Koblenz-Landau, Koblenz, Germany

Abstract. A novel approach for engaging stakeholders in policy making is developed and implemented in the OCOPOMO project. Scenario texts generated by stakeholders as well as other background documents are used to inform formal policy modelling, which applies declarative rule-based agent modelling paradigms. To bridge the gap between narrative texts of scenarios and formal policy models, the OCOPOMO approach introduces the Consistent Conceptual Description (CCD) as a "modelling middle-layer". This paper presents the CCD metamodel, i.e. the underlying vocabulary to describe policy contexts, and it describes how the CCD metamodel supports the semi-automatic transformation of conceptual models of a policy context to generate formal policy models.

Keywords. Conceptual Models, Metamodel, Model Driven Development, Agent-Based Policy Modelling and Simulation

1. Introduction

In the OCOPOMO[1] project, a novel approach for engaging stakeholders in policy development is conceptualized and implemented [8]. Stakeholders are collaboratively involved in the development of scenario texts relevant in the context of a policy under discussion. By policy we refer to strategic areas of complex decision-making with various stakeholders having potentially diverging interests. In OCOPOMO, public policies are investigated and modelled such as renewable energy policy of the Kosice region in Slovakia, housing policy of the city of London or the distribution of structural funds in the Campania region in Italy. The overall OCOPOMO policy development process consists of six phases, where the first two phases involve stakeholders to generate scenarios of potential policy aspects, which are complemented with background documents to evidence statements in the scenarios. In phases three and four, the policy case is conceptualized and modelled by modelling experts, while phase five runs simulations to generate outcomes. In phase six, the results of the simulation are exposed to the stakeholders, who compare their scenarios (of phase 2) and the simulation outcomes in order to either update their scenarios or accept the insights from the simulation and agree that these are consistent with the inputs the stakeholders provided in the scenario generation phase. For more details on the OCOPOMO policy development process the reader is referred to [8].

[1] Open Collaboration Policy Modelling, http://www.ocopomo.eu/

As indicated before, the scenarios as well as other background documents inform formal policy models. The formal policy models are developed applying declarative rule-based agent modelling paradigms [5]. The main aims of formal policy modelling are to evaluate consistency of the simulation, to elicit and generate insights on the policy case which could not be detected solely from the narrative and conceptual models, and therewith to inform decision makers with a consistent formal and narrative representation of a potential policy to be decided. The overall OCOPOMO approach thereby ensures participation and involvement of stakeholders, and collaborative policy development. It increases transparency by making complex policy cases more understandable through (i) a set of artefacts representing distinct policy options, and (ii) enabling traceability of arguments put forward in the scenarios or reported in background documents and embodied in formal policy models.

OCOPOMO requires a multi-disciplinary approach for eliciting opinions and viewpoints of stakeholders, for analysing and mining large text sources, for conceptualizing the policy context, and for developing and running simulation models. The project therewith faces two challenges as outlined in [3]: "In individual developments, it is difficult that one person has all the expertise required; in teams, there are communication problems". In order to solve or reduce communication and integration challenges, "the use of agent-oriented software engineering methods and tools, relying on a modelling middle-layer" is proposed [3]. To tackle these challenges and to bridge the gap between narrative texts of stakeholder-generated scenarios (evidenced through background documents of the policy to be discussed) and formal policy models (generating model-based scenarios), the OCOPOMO approach introduces the Consistent Conceptual Description (CCD) as "modelling middle-layer". A CCD enables different stakeholders in the OCOPOMO process to better understand the policy context and to support semi-automatic transformation of text statements into formal statements and agent descriptions. Thereby, the CCD is exposed to the following needs and requirements:

Consistent By relying on visualization and ontological structures, the CCD aims to make modelling decisions understandable and traceable for stakeholders. In the CCD, text phrases from source scenarios and background texts are linked with the conceptual descriptions. Semi-automatic transformation of a CCD into imperative and declarative programming code (thereby also conveying references to scenarios and background documents) allows that simulation results can be linked with original text phrases. Through the conceptualization, the CCD supports in developing consistent policy models.

Conceptual The CCD aims to provide a conceptual model of a policy case i.e. conceptualising actors, policies, beliefs, aims etc. and their relations relevant in a policy case, which are described in scenarios and background documents. Concepts are further transferred into concrete programming code elements based on a Model Driven Development (MDD) approach known from software engineering in agent-based simulation and modelling as e.g. described in [1,4].

Description Along the transformation to formal policy models, the CCD plays a role of an intermediary between scenarios and simulation models. Several scenarios can form input to the CCD of a policy domain and further lead to a formal simulation model. Likewise, expertise of policy analysts may lead to particular knowledge constructs in the CCD. The CCD may also inform the scenario development of stakeholders by visualising particular knowledge gaps in the existing scenario descriptions. Finally, the CCD content may be revised or enriched based on input from analysing simulation results (model-based scenarios, i.e. interpretations of simulation outcomes, and supportive graphical representations of model outcomes). [8]

In this paper, we present the CCD metamodel, i.e. the underlying vocabulary and relations to describe policy contexts. We further sketch the concept for semi-automatic transformation of conceptual policy descriptions of the CCD into formal policy models. The remainder of the paper is as follows: Section 2 describes the CCD metamodel. Section 3 introduces related work and highlights the differences of the CCD metamodel from these concepts. In Section 4, the technical implementation to support the OCOPOMO process in bridging narrative text with formal policy models via the CCD tool is briefly introduced. The paper concludes with a brief reflection of the added value of the approach and needs for further research.

2. Metamodel for the Consistent Conceptual Description (CCD-Metamodel)

A CCD represents the conceptualisation of a policy case. To ensure well-defined and structured CCD models of a policy domain, the CCD metamodel serves as a blueprint or vocabulary for describing a policy case with the requirements mentioned in Section 1. Formally, a metamodel can be defined as a conceptual description of a modelling, which illustrates the model concepts as well as their usage [9]. Thereby the CCD metamodel defines a "set of representational primitives with which to model a domain of knowledge or discourse" as typically an ontology does [2]. Representational primitives of an ontology are typically "classes (or sets), attributes (or properties), and relationships (or relations among class members)" [2]. Based on the approach for ontology of [2] and the rationale of metamodels according to [9], a set of elements of the CCD metamodel are defined. The core elements for the CCD metamodel are derived from insights into the three policy cases in OCOPOMO as well as from discussions with policy modellers in the project. The following (static) elements have been defined for the CCD metamodel: *Concepts* with *Actors*, *Objects* (representing particular entities or sets), *Attributes* and *Relations*. Facts for the simulation model are encoded with *Instances* of a concept *Actor* or *Object*. Dynamic aspects are encoded as *Action*. Each concrete *Action* has an actor, inputs and outputs (*ActionInputOutput*). This allows to model behaviour and interaction in one model. All elements can be linked with each other. Figure 1 shows the CCD metamodel with its core elements. The elements are further described in Table 1. In addition to the core elements of the CCD metamodel, the class *Annotation* links a concept or concrete instance of the CCD with its basic information in a scenario or background document. Two types of annotations are possible: text annotations and expert annotations. An annotation consists of a URL to the document referenced, a start position and the length of the phrase forming the annotation. The annotation is directly stored in the CCD.

3. Model Driven Development For Agent-Based Modelling and Simulation

Approaches applying MDD in agent-based modelling are e.g. described in [7,1,4,6]. The different approaches use different vocabularies / metamodels for conceptual models and transformation procedures. The annotation component in OCOPOMO differentiates the CCD metamodel from those metamodels found in literature. The CCD metamodel fol-

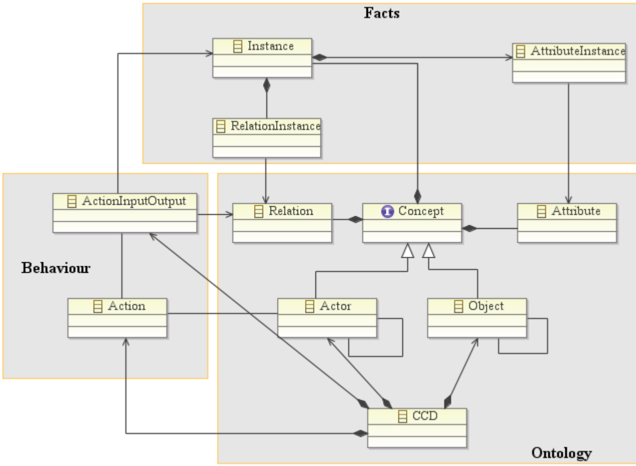

Figure 1. The CCD metamodel

Table 1. Elements of the CCD metamodel

Element	Description
CCD	Representation of the environment with its actors, objects, and actions.
Concept	Concept is an abstract class. A Concept can have instances, relations and attributes. There are two concrete concepts available: Actor and Object
Actor	Representation of a class of actors in the policy case - i. e. the agents in the multi-agent based simulation model. Actors are active and perform pro-active actions. Actions executed by a class of actors can be linked with the actor class.
Object	Representation of a class of objects in the policy case, i. e. all objects that are passive. Hence actions are not linked with an object class. However, objects can be part of an ActionInputOutput object and therefore be changed by an action - such an action is called a reaction.
Relation	Representation of relationships between two actors, two objects, or an actor and an object.
Attribute	Representation of attributes/properties for a class of actors or objects.
Instance	Representation of a concrete occurrence of an actor or object class. Instances have concrete attribute values (AttributeInstance) or have relationships (RelationInstance) with other instances. Instances define facts.
Action	Representation of tasks or actions in the policy case. Pro-active actions are executed by actors and pursue an objective. Reactions have no actor defined. Each Action can have several inputs and several outputs represented by ActionInputOutput objects. ActionInputOutput represents an input or output of an action. An ActionInputOutput as output of one action can be used as input for another action. This way, behaviour of and interaction between actors are modelled. This construct also provides the basis for the rule-dependency graph in the declarative agent-based modelling.

lows a rather descriptive and ontology-based approach than most of the metamodels in literature. In addition to actors (agents), the CCD metamodel allows the definition of objects and instances and their relations; however, without specifying the types of objects - i.e. the conceptual modeller has freedom. The freedom of the modeller makes the transformation from conceptual descriptions into programming code more complex. Yet, the descriptive character of the conceptual model is seen by the OCOPOMO project partners as more important than the automatic code transformation (i.e. a semi-automatic transformation is accepted). From a conceptual viewpoint, the transformation of conceptual models into simulation models is done by building analogies between elements of the CCD metamodel and elements of the simulation model (the declarative rule-based agent modelling system (DRAMS) as introduced in [5]). Thereby a kind of transformation model is built, which is dependent on the CCD and DRAMS metamodels. The CCD metamodel and DRAMS metamodel themselves are independent from each other. The transformation model allows the specification of details relevant for the simulation model already in the conceptual description. To bridge the conceptual models and the formal simulation models, OCOPOMO foresees the following transformation steps: 1. Transformation of text into CCD, 2. Transformation of conceptual models (CCD) into conceptual simulation models 3. Transformation of conceptual simulation models into programming code.

4. Technical Implementation

The CCD Tool supports Facilitators and Policy Modellers in developing a stakeholder-accessible formalisation of a policy model and thereby ensuring and documenting the consistency via the CCD. It supports the following main functions: a) Creating, editing and saving a consistent conceptual description (CCD) for a policy case, b) Linking the concepts of a CCD with background documents and scenarios, and c) Creating formal policy models that can be further programmed in the Simulation Engine. The CCD Tool[2] as part of the OCOPOMO ICT toolbox has a modular design with the following components:

Conceptual Description (CD) Tool: supports the creation, editing and storing of a CCD file (XML) which is representing the conceptual description of a policy case. The structure of the file is defined in the CCD metamodel.

Annotation Tool: is used to annotate and link relevant text phrases from background documents and stakeholder scenarios with relevant elements in a CCD file.

Transformation Tool (CCD2DRAMS Tool): supports (draft) source code generation from a CCD file for the Simulation Model in DRAMS. Thereby the CCD2DRAMS Tool supports traceability of simulation results by linking code fragments with concepts of the underlying CCD file.

5. Conclusions and outlook

In this contribution, we have outlined the concept for bridging narrative text artefacts and formal policy models via a conceptual description to support policy modelling. We introduced the CCD metamodel and argued its differences from other meta-modelling concepts. Finally, we briefly outlined the implementation setup for the CCD tool. The main added value and benefits of the CCD tool (with the underlying CCD metamodel) are to enable non-expert policy developers to extract aspects of the policy domain under consideration, to conceptualize and structure the domain and to provide an understandable conceptual model of a policy domain, which can be used by expert policy model programmers to generate initial code fragments for the simulation model. A key added

[2]Implemented as Eclipse Plug-in using EMF (Eclipse Modeling Framework, http://www.eclipse.org/emf/) and GMP (Graphical Modeling Project, http://eclipse.org/modeling/gmp/

value for the stakeholders and the programmers is the concept of traceability, i.e. being able to trace an agent description, a fact description or rules firing in the simulation runs back to the descriptions from where these have been derived (the evidences). This way, the current request for open government with the principles of collaboration, openness and participation is supported to a large extent. The CCD tool (incl. the annotation, conceptual model development and transformation into DRAMS code) is currently tested along the three use cases of OCOPOMO. During the tests, the CCD metamodel is analysed and evaluated for different criteria: its comprehensibility by different project actors, its completeness and eligibility to model the policy cases, its practicability to generate formal policy models. The evaluation will result in revisions of the CCD metamodel presented in this paper. Further research is ongoing to implement the traceability which is to be enabled across different ICT tools, as the CCD toolbox builds on model-driven architecture. As this concept allows individual modules to be replaced by other software modules, further research will be needed to support adaptability and transformation of the tracing concept into other software modules that can be integrated in the OCOPOMO policy development process.

Acknowledgements

OCOPOMO is co-funded by the EC within FP 7, contract No. 248128. The authors acknowledge the contributions of and express their gratitude to the OCOPOMO project partners, especially Scott Moss, Klaus Troitzsch and Ulf Lotzmann. The content of this paper represents the view of the authors, respectively. The European Commission cannot be made liable for any content.

References

[1] J. Gómez-Sanz and J. Pavón. Agent oriented software engineering with INGENIAS. In *Proceedings of the 3rd Central and Eastern Europe Conference on Multiagent Systems, Springer Verlag, LNCS*, volume 2691, pages 394–403. Citeseer, 2003.

[2] T. Gruber. Ontology. In L. Liu and M. T. Özsu, editors, *Encyclopedia of Database Systems*. Springer-Verlag, 2009.

[3] S. Hassan, R. Fuentes-Fernández, J. Galán, A. López-Paredes, and J. Pavón. Reducing the Modeling Gap: On the use of metamodels in agent-based simulation. In *6th Conference of the European Social Simulation Association (ESSA 2009)*, pages 1–13, 2009.

[4] T. Iba, Y. Matsuzawa, and N. Aoyama. From conceptual models to simulation models: Model driven development of agent-based simulations. In *9th Workshop on Economics and Heterogeneous Interacting Agents*. Citeseer, 2004.

[5] U. Lotzmann and R. Meyer. DRAMS - A Declarative Rule-based Agent Modelling System. In *Proceedings of 25th European Conference on Modelling and Simulation*, 2011.

[6] F. Okuyama, R. Bordini, and A. da Rocha Costa. ELMS: An environment description language for multi-agent simulation. *Environments for Multi-Agent Systems*, pages 91–108, 2005.

[7] J. Pavón, J. Gómez-Sanz, and R. Fuentes. Model Driven Development of Multi-Agent Systems. In A. Rensink and J. Warmer, editors, *Model Driven Architecture Foundations and Applications*, volume 4066 of *Lecture Notes in Computer Science*, pages 284–298. Springer Berlin / Heidelberg, 2006.

[8] M. A. Wimmer, K. Furdik, M. Bicking, M. Mach, T. Sabol, and P. Butka. Open Collaboration in Policy Development: Concept and Architecture to integrate scenario development and formal policy modelling. In Y. Charalabidis and S. Koussouris, editors, *Empowering Open and Collaborative Governance*. Springer Berlin / Heidelberg, 2012.

[9] A. Winter. *Referenz-Meta-schema für visuelle Modellierungssprachen*. Deutscher Universitäts-Verlag, Oktober 2000.

The Policy Modeling Tool of the IMPACT Argumentation Toolbox

Thomas F. Gordon
Fraunhofer FOKUS, Berlin

Abstract. In this paper we present an overview of the policy modeling tool of the argumentation toolbox being developed in the European IMPACT project. The tool is a web version of the Carneades argumentation system extended with support for comparing policy alternatives in deliberative democracy application scenarios. The tool can be also be used as a legal expert system shell, for example in applications for helping citizens to assess their rights to social benefits, in much the same way as some commercial rule-based systems, but goes further towards realizing the vision of isomorphic modeling by being based on the state-of-the-art in Artificial Intelligence and Law and Computational Models of Argument fields.

Keywords. policy deliberations, computational models of argument, e-participation

1. Introduction

In this paper we present an overview of the policy modeling tool of the argumentation toolbox being developed in the European IMPACT project. IMPACT is a European Framework 7 project on the ICT for Governance and Policy Modeling theme. The project began January 1, 2010 and will run for three years.[1] IMPACT is conducting original research to develop and integrate formal, computational models of policy and arguments about policy, to facilitate deliberations about policy at a conceptual, language-independent level. These models will be used to develop and evaluate a prototype of an innovative argumentation toolbox for supporting open, inclusive and transparent deliberations about public policy on the World-Wide-Web.

Four prototype tools are being developed for the IMPACT argumentation toolbox:

1. Argument Reconstruction Tool
2. Structured Consultation Tool
3. Policy Modeling Tool
4. Argument Visualization and Tracking Tool

The first three of these tools support the process of getting arguments and assessments of arguments into the system. The argument reconstruction tool supports analysts with the task of finding and modeling relevant arguments in large numbers of articles and comments in natural language on the Web. The structured consultation tool uses formal models of argumentation schemes to generate surveys on the web which help the general public to voice their opinions, evaluate previous arguments, and ask

[1] Grant Agreement No 247228. For further administrative details about the project, see the project website: http://www.policy-impact.eu.

critical questions, by simply completing interactive forms. The argument visualization and tracking tool helps users to browse, understand and keep track of arguments which have been entered into the system using the other tools. The tool is a web application for displaying, browsing and querying interactive diagrams of arguments, called "argument maps".

The focus of this paper is the policy modeling tool. It uses computational models of policies, applying methods from Artificial Intelligence and Law and Computational Models of Argument, to help users to analyze and understand the legal effects of alternative policies in particular fact situations or cases. It helps users to "get arguments into the system" in an indirect way. By helping users to better understand the proposed policies, they are better able to contribute informed arguments to the policy debate. The tool is an interactive web application that works much like a rule-based expert system or "wizard". Users engage in a simple kind of dialogue with the system, using menus and forms.

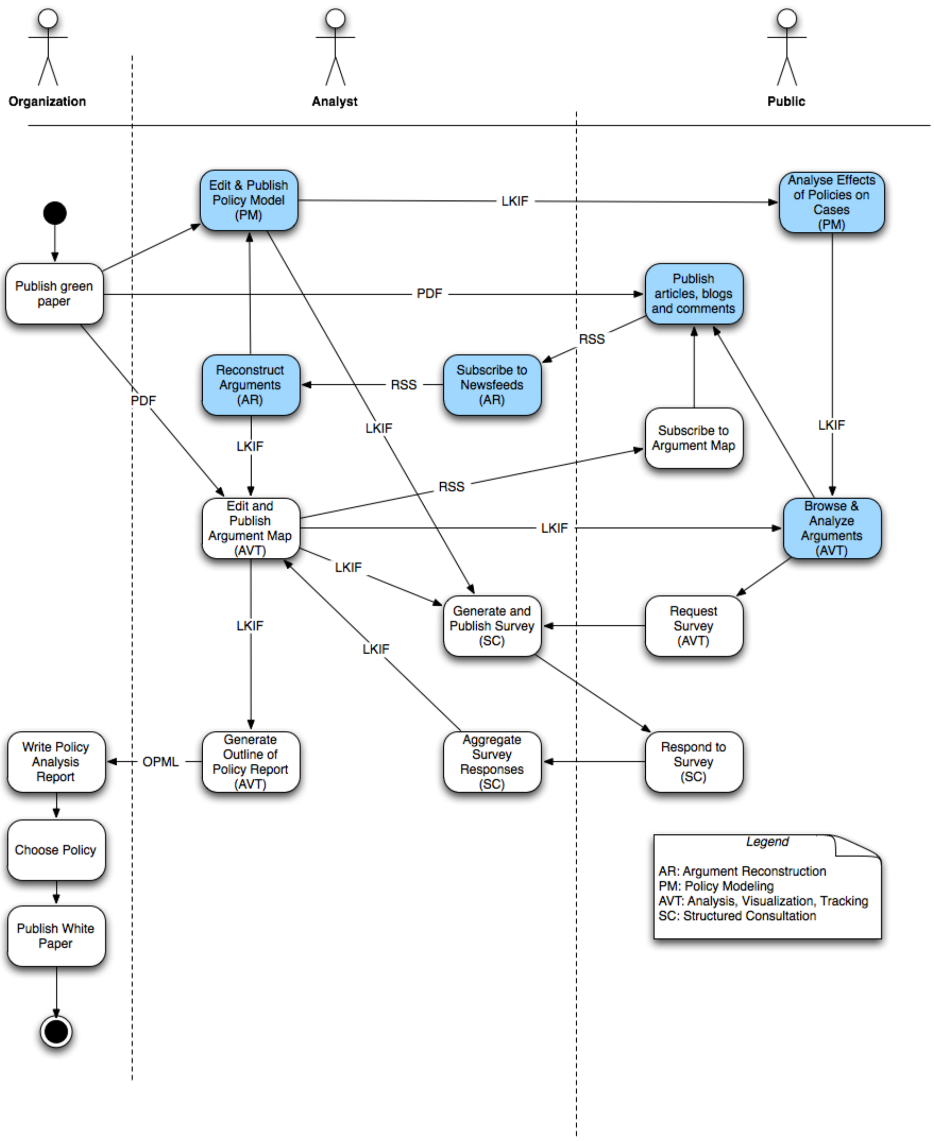

Figure 1. Process Model

Figure 1 is a process model showing how all the tools of the IMPACT system are used together to support policy deliberations. The highlighted tasks are relevant for the policy modeling tool. Suppose that welfare benefits are at issue and that several new welfare policies have been proposed. First, a computational model of these welfare policies is constructed by the analysis and published on the web ("Edit and Publish Policy Model"). Next, public users (citizens and other stakeholders) can use these models to assess the legal effects of the alternative welfare policies on cases, for example to determine whether single mothers would be entitled to a benefit, by using the web user-interface of the tool ("Analyze Effects of Policies on Cases"). This is done by selecting and loading models of the policies and entering the relevant facts of the case by completing a series of forms. Figure 2 shows a screen shot of the web user interface for entering case facts of the current prototype. Questions are asked by the system in a goal-directed way, using the rules of the policies, to assure that only relevant questions are asked. When sufficient facts have been gathered, the system produces a diagram visualizing relationships and dependencies between the facts of the case, the rules of the various policies, and the legal conclusions that can be drawn from these policies ("Browse and Analyze Arguments"). Figure 3 shows a screen shot of the argument map displayed by the current prototype.[2] The policy modeling tool is able to compute preferred policies serving the interests of the user, by achieving desired legal effects.[3] The cases entered by the user will be able to be saved back to a database on the server and published on the Web, to make them available to other participants in the policy debate. Care will be taken to assure that the privacy of users is protected, by not storing any personal information. The Policy Modeling tools also will provide a way for users to take part in a survey, to express their opinion about which policy is preferable, and to view the aggregated results of the survey. After having analyzed the effects of the proposed policies, the user should be in a better position to make an informed contribution to the policy deliberations, for example by posting an argument on his web log ("Publish articles, blogs and comments.") The argument can include links to any cases he constructed and published using the policy modeling tool. Clicking on such a link would launch the policy modeling with this case displayed. The facts of the case can then be modified, in order to explore the effects of the policies on other cases, without having to enter all the facts from scratch.

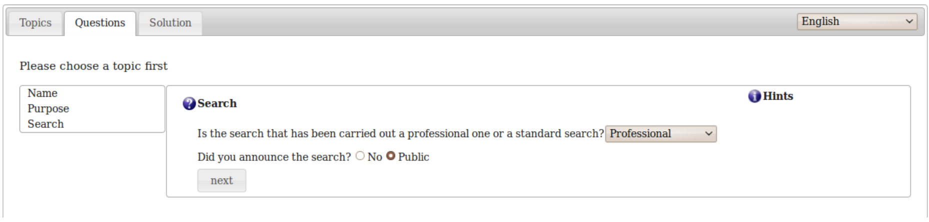

Figure 2. User interface for entering case facts

[2] In this prototype the statements are shown in a formal language. In the final version, the statements will be displayed in the user's choice of natural language. The analyst can manually provide templates for the most important languages. If a template has not been provided, an external translation web service is used to produce text in the requested language.

[3] 1The preferred policies are computed using a form of "abduction".

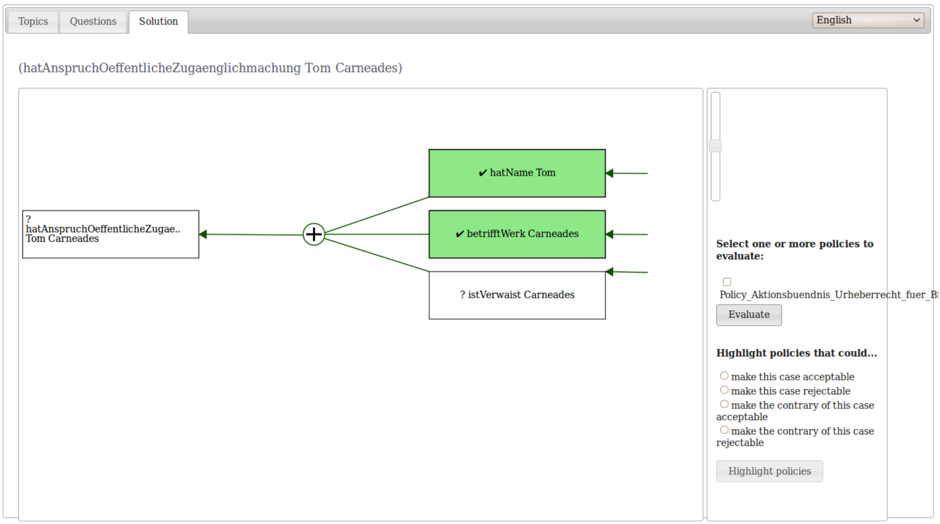

Figure 3. User interface displaying a map of the resulting arguments

2. System Architecture and Implementation Status

The system architecture of the IMPACT argumentation toolbox, shown in Figure 4, has been designed to ensure the usability, interoperability and portability of all of the argumentation tools being developed in the project, while complying with relevant standards.

Formal models of arguments are interchanged among the four tools using an XML Schema based on and derived from the argumentation part of the Legal Knowledge Interchange Format (LKIF) (Gordon 2008). LKIF is an XML schema for representing and interchanging rules and arguments that was developed by some of the IMPACT partners in a prior European project, ESTRELLA (IST-2004-027655).

The tools of the argumentation toolbox are web applications, with a three-tiered, client-server architecture, with a relational database backend. The server-side of the tools are packaged and published as RESTful web services, following World-Wide Web Consortium (W3C) standards.

The client-side of the IMPACT argumentation tools are implemented as Rich Internet Applications using W3C standards, in particular Asynchronous JavaScript and XML (Ajax). Web components are packaged as web widgets using Googles iGadget specification. Gadgets are small web applications that can be stored locally on the client computer and run outside a browser in a special web container or may be included by other platforms supporting the gadget specification. This enables the web user interfaces of the IMPACT tools to be published on pages of eParticipation and other web sites and portals and be used with any standards-compliant web browser, without requiring plug-ins.

To ensure that all IMPACT web clients have a common look and feel, they are all implemented using the jQuery JavaScript library. A custom stylesheet has been developed, using the jQuery User Interface CSS Framework, to enable widgets of the tools to have a common, attractive look and feel. The jQuery library was chosen because it is open source, mature, well documented and widely used. A set of guidelines and principals for the developers of IMPACT tools has been developed, to address usability, ergonomics, accessibility, security, internationalization and other issues.

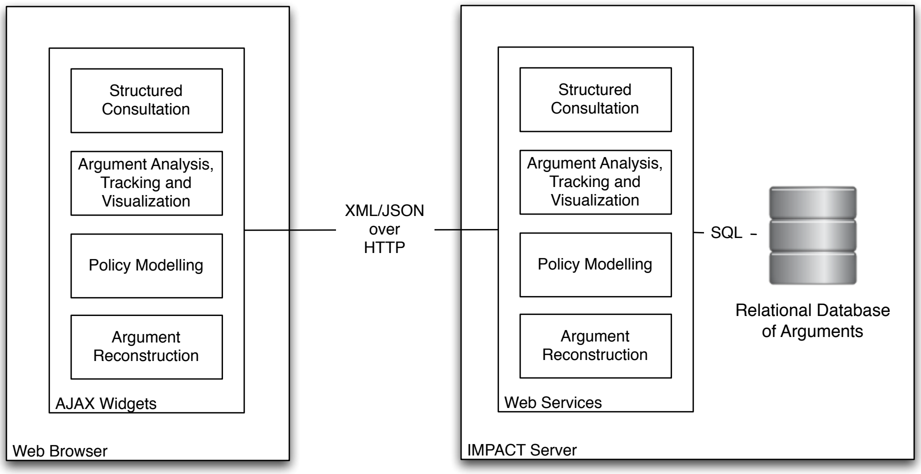

Figure 4. System Architecture

The system architecture described above applies to all the tools of the IMPACT toolbox. However, the way the web services of the tools are implemented varies from tool to tool. The policy modeling web service is based on our prior work on the Carneades argumentation system (Gordon 2010), but with the following major changes and extensions:

- Carneades was first ported from Scheme to Clojure[4], another Lisp dialect, to enable it to run on the Java Virtual Machine and make use of the extensive set of available Java libraries, for example for interacting with databases via the Java Database Connectivity library (JDBC) and for implementing web services and applications.
- We have developed and implemented an original method for a kind of "abduction" from a set of arguments, interpreted as defeasible rules in a propositional (nonmonotonic) logic (Ballnat and Gordon 2010). This work provides the foundation for reasoning about the differential effects of alternative policy proposals in the policy modeling tool.
- Next we have refactored Carneades to more cleanly separate its modules for argument construction and argument evaluation. It is now easier to implement and "plug in" modules for other computational models of structured argument, such as ASPIC+ (Prakken 2010). We intend to validate this architecture during the IMPACT project by implementing

[4] http://www.clojure.org/

and offering ASPIC+ in addition to Carneades' own model, called Carneades Argument Evaluation Structures (CAES).

- We have designed and implemented a Domain Specific Language (DSL) in Clojure for visualizing directed graphs, similar to the "dot" language used by the Graphviz system (Ellson et al. 2001), called LACIJ[5], for the purpose of generating visualizations of argument graphs for the Web, using the Structured Vector Graphics (SVG) web standard for 2D vector graphics[6].
- A new XML format for interchanging arguments, called the Carneades Argument Format (CAF), has been developed. CAF is based on the LKIF schema for arguments, but has been extended to provide better support for metadata about the sources of arguments and simplified by omitting the parts of LKIF for representing rules. Metadata is represented using the Dublin Core Metadata Element Set.[7]
- Policies are represented in a new Domain Specific Language (DSL) for defeasible inference rules (also called "argumentation schemes") in Clojure[8]. The language provides a way to specify templates for translating formulas into positive or negative assertions, as well as questions, in several natural languages, as required by the IMPACT application scenarios for European policy debates. The language supports the isomorphic modeling of legislation, as a hierarchy of sections. Rules ("schemes") can be included at every level of the hierarchy. Metadata, using the Dublin core attributes, can be associated with the rulebase as a whole, as well as each section and scheme. This metadata also enables links to the original legal sources to be included within the model.
- Carneades now has a relational database backend for storing and managing arguments, along with metadata describing and linking the arguments to their source documents. The database schema is isomorphic to CAF, to enable CAF files to be imported into a database, and exported from a database, with no loss of information.
- A RESTful web service[9] for creating, reading, updating and deleting arguments from the database has been implemented. Data is exchanged between the web service and clients using the JSON language[10], which is based on JavaScript and thus ideal for Rich Internet applications written in JavaScript using AJAX.[11] This web service will be extended to provide access to all features of Carneades for constructing, evaluating and visualizing arguments on the Web.

---

[5] https://github.com/pallix/lacij

[6] http://www.w3.org/Graphics/SVG/

[7] http://dublincore.org/documents/dces/

[8] Carneades no longer uses LKIF or some other XML format for representing legal rules. An OASIS Technical Committee has been proposed for developing an XML standard for legal rules, called LegalRuleML, which we intend to support when it is finished. (See http://lists.oasis-open.org/archives/tc-announce/201111/msg00014.html).

[9] http://en.wikipedia.org/wiki/Representational_state_transfer

[10] http://www.json.org/

[11] http://en.wikipedia.org/wiki/Ajax_(programming)

A prototype of the policy modeling tool with all of the above features is nearing completion and can be demonstrated.

3. Related Work

The Carneades argumentation system, upon which the policy modeling tool is based, and which is being extended during the course of the IMPACT project to meet requirements of policy deliberation application scenarios on the Web, has been inspired by and builds upon a large body of work in the fields of AI and Law, computational models of argument and web-based groupware for argumentation. It would exceed the bounds of this research abstract to mention all relevant prior work. It will have to suffice to list some of the main influences.

The inference engine for constructing arguments from defeasible rules (argumentation schemes) is based on prior work on nonmonotonic logics and legal reasoning in the AI and Law field. The main direct influences include (Prakken 1997; Hage, Verheij, and Lodder 1993) not to mention my own prior work beginning with Oblog (Gordon 1987) and later the rule language of the Pleadings Game (Gordon 1994). A good summary of this line of research is (Prakken et al. 1998).

The Carneades computational model of argument is one of the few systems to model legal proof standards, such as preponderance of the evidence and beyond reasonable doubt (Gordon and Walton 2009). This work was inspired by (Freeman and Farley 1996). Recently, the ASPIC system has been extended to support proof standards (Prakken and Sartor 2011). For some time, the relationship between Carneades and the leading computational model of argument, Dung abstract argumentation frameworks, was unclear. But this relationship has been clarified in a series of articles (Brewka and Gordon 2010; Governatori 2011; van Gijzel and Prakken 2011).

Finally, the new parts of Carneades for supporting argumentation on the web were inspired and informed by a number of systems, in particular Cohere (Buckingham-Shum 2008), the vision of a World-Wide Argument Web (Rahwan, Zablith, and Reed 2007), Gregor Betz' ArguNet system[12], as well as my own prior work on the Zeno system (Gordon and Richter 2002).

4. Conclusions and Future Work

This short paper has outlined our ongoing work in the IMPACT project on building a policy modeling and analysis tool, based on state-of-the-art methods from Artificial Intelligence, Computational Models of Argument and groupware for argumentation on the World Wide Web. The policy modeling tool builds on our prior work on the Carneades argumentation system, but to meet the new requirements of the

[12] http://www.argunet.org/

IMPACT project for supporting policy deliberations on the web, Carneades was ported to the Java platform, to take advantage of its better support for building web applications, extended with support for a kind of abduction, useful for deriving policies with desired effects, and repackaged as a web service with a relational database backend. The services provided by the policy modeling tool for evaluating alternative policies will help participants in policy debates to better understand the effects of policies on cases, more easily formulate informed opinions about policy issues and contribute higher quality, constructive and rational arguments.

The IMPACT project continues for another year. In this time, we will develop the second prototype of the policy modeling tool. This next version will support the collection of feedback from users on their preferred policies and provide a convenient way for users to "vote" on the policies and have their votes recorded, anonymously, in the Carneades database on the IMPACT toolbox server. In addition, a way will be provided for users to publish cases in the database, in an anonymous form that respects privacy, along with their policy preferences, to enable other users to access and reuse the cases. The published cases will be assigned a URL by the system, to enable anyone to reference and link to the cases on the Web, for example in weblog and discussion forum articles about the policy issues. This feature will provide an easy way for users to back up their policy arguments with evidence and enable others to reproduce, understand and confirm the claimed effects of policies on cases.

References

Ballnat, Stefan, and Thomas F Gordon. 2010. Goal Selection in Argumentation Processes — A Formal Model of Abduction in Argument Evaluation Structures. In *Proceedings of the Third International Conference on Computational Models of Argument (COMMA)*, ed. Guillermo Simari, 51-62. Amsterdam: IOS Press.

Brewka, Gerhard, and Thomas F Gordon. 2010. Carneades and Abstract Dialectical Frameworks: A Reconstruction. In *Proceedings of the Third International Conference on Computational Models of Argument (COMMA)*, 3–12. Amsterdam: IOS Press.

Buckingham-Shum, Simon. 2008. Cohere: Towards Web 2.0 argumentation. In *Computational Models of Argument – Proceedings of COMMA 2008*, 97-108.

Ellson, John, Emden Gansner, Lefteris Koutsofios, Stepen C North, and Gordon Woodhull. 2001. Graphviz --- Open Source Graph Drawing Tools. In *Proceedings of the 9th International Symposium on Graph Drawing (GD 2001)*, 483-484. Vienna, September.

Freeman, Kathleen, and Arthur M Farley. 1996. "A Model of Argumentation and Its Application to Legal Reasoning." *Artificial Intelligence and Law* 4 (3-4): 163-197.

van Gijzel, Bas, and Henry Prakken. 2011. Relating Carneades with abstract argumentation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI-2011)*, in press.

Gordon, Thomas F. 1987. Oblog-2: A Hybrid Knowledge Representation System for Defeasible Reasoning. In *Proceedings of the First International Conference on Artificial Intelligence and Law*, 231-239. Boston.

———. 1994. “The Pleadings Game – An Exercise in Computational Dialectics.” *Artificial Intelligence and Law* 2: 239-292.

———. 2010. An Overview of the Carneades Argumentation Support System. In *Dialectics, Dialogue and Argumentation. An Examination of Douglas Walton’s Theories of Reasoning*, ed. Christopher W. Tindale and Chris Reed, 145-156. College Publications.

Gordon, Thomas F, and Gernot Richter. 2002. Discourse Support Systems for Deliberative Democracy. In *eGovernment: State of the Art and Perspectives (EGOV02)*, ed. Roland Traunmüller and Klaus Lenk, 248-255. Berlin / Heidelberg / New York: Springer Verlag.

Gordon, Thomas F, and Douglas Walton. 2009. Proof Burdens and Standards. In *Argumentation in Artificial Intelligence*, ed. Iyad Rahwan and Guillermo Simari, 239-260. Berlin, Germany: Springer-Verlag.

Gordon, Thomas F. 2008. The Legal Knowledge Interchange Format (LKIF). The ESTRELLA Project.

Governatori, Guido. 2011. On the Relationship between Carneades and Defeasible Logic. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Law (ICAIL-2011)*, in press, this volume. ACM Press.

Hage, J C, H B Verheij, and A R Lodder. 1993. Reason-Based Logic: A Logic that Deals with Rules and Reasons. In *Working Papers NAIC*, ed. J M Akkermans and J A Breuker, 293-304.

Prakken, Henry. 1997. *Logical Tools for Modelling Legal Argument : a Study of Defeasible Reasoning in Law*. Dordrecht ; Boston: Kluwer Academic Publishers.

———. 2010. “An abstract framework for argumentation with structured arguments.” *Argument & Computation* 1: 93-124.

Prakken, Henry, Gerard Vreeswijk, Grote Kruisstraat, and T S Groningen. 1998. “Logics for Defeasible Argumentation 1 Introduction.” *Argumentation*.

Prakken, Henry, and Giovanni Sartor. 2011. On modelling burdens and standards of proof in structured argumentation. In *Legal Knowledge and Information Systems (Jurix 2011)*, in press. IOS Press, Amsterdam.

Rahwan, Iyad, Fouad Zablith, and Chris Reed. 2007. “Laying the foundations for a World Wide Argument Web.” *Artificial Intelligence* 171 (10-15) (July): 897-921. doi:10.1016/j.artint.2007.04.015. http://linkinghub.elsevier.com/retrieve/pii/S0004370207000768.

Ontologies for Governance, Risk Management and Policy Compliance[1]

Jorge GONZÁLEZ-CONEJERO [a,2], Albert MEROÑO-PEÑUELA [a] and David FERNÁNDEZ GÁMEZ [b]

[a] *Institute of Law and Technology, Universitat Autònoma de Barcelona, Spain*
[b] *S21sec, Barcelona, Spain*

Abstract. The Internet and Information Systems evolution have dramatically increased the amount of information held by companies and public administrations. These data can be very sensitive, specially regarding personal data, so governments and international organizations promote acts and guidelines in order to ensure privacy and data security. Thus, on the one hand, companies have to consider legal and Information Technology compliance. On the other hand, Governance, Risk Management and Compliance (GRC) is an emerging discipline which consists of a holistic approach to these three areas of an organization. In this work we introduce the OGRC framework, a software application based on legal and security ontologies that aims at providing organizations with legal compliance support. The main features are: i) the *automation* of some compliance evaluation processes; and ii) *flexibility* to add or modify policies.

Keywords. Governance, Risk Management, Compliance, Ontologies

Introduction

Nowadays, Internet has become the most important channel to share information with the whole world. Most traditional activities including music, film, television, newspapers or books have been reshaped or redefined by the Internet. The Internet has also enabled new forms of human interactions through instant messaging, forums and social networking. Moreover, e-commerce and financial services on the Internet have boomed in the recent years. Nevertheless, the use of these services entails providing many personal information to companies and public administrations.

The main issues that arise in this scenario are: i) *privacy*, namely the use that organizations make of these data; and ii) *security*, since sensitive information could be accessed illicitly. Therefore, the most common approaches to face these issues are acts and international standards. On the one hand, governments have developed acts to regulate this scenario. In Spain, the responsible for this area is the Data Protection Agency (AGPD)[3].

[1]This work has been partially supported by the Spanish Ministry of Science and Innovation (Ministerio de Ciencia e Innovación) under grant IPT-430000-2010-43.

[2]Corresponding Author: Institute of Law and Technology, Universitat Autònoma de Barcelona, B Building, 08193 Bellaterra (Barcelona), Spain; E-mail: jorge.gonzalez.conejero@uab.es

[3]Agencia Española de Protección de Datos `http://www.agpd.es`

AGPD audits organizations to impose a fine when they fail to comply with applicable law. On the other hand, international standard organizations have developed guidelines to ensure the privacy and security of this kind of data.

Governance, Risk Management and Compliance (GRC) [1] is a holistic approach to these three areas of an organization. GRC is increasingly being integrated in a convergent and more abstract layer, which tries to avoid conflicts, face uncertainty and reduce overlaps and gaps between business processes in order to gain efficiency. In the literature, several approaches that are able to evaluate GRC requirements have been introduced (see [2] and [3]). For instance, the Unified Compliance Framework (UCF) [4] is an effort to integrate many compliance policies from the Information Technology department. Modulo Risk Manager [5] implements effective solutions for Compliance based on a wide range of relevant regulations and standards. Finally, Oracle Fusion GRC [6] is a suite of applications designed to work as a complete enterprise GRC solution.

In this work, we introduce the OGRC project, which states for Ontologies for Governance, Risk Management and Compliance. This project will provide organizations with an integrated software solution intended to monitor processes, assets, risks and requirements. Moreover, the OGRC framework is supported by semantic data models aimed at promoting desirable features as *automation* and *flexibility*.

The work is organized as follows: section 1 describes the OGRC project. Acts and standards considered in the OGRC framework are presented in section 1.1, the most important tools operated are described in section 1.2, the architecture is depicted in section 1.3, and how the non-compliant situations are detected is addressed in section 1.4. Finally, section 2 points out some conclusions.

1. Ontologies for Governance, Risk Management and Compliance

Currently, compliance of legal and international standards is verified mostly by experts, usually auditors or consultants, and it is still a manual task. This compliance assessment process can be extraordinarily expensive. The OGRC project aims at providing organizations with legal and automatic compliance support through Bitacora [4] [7] and ontologies. Bitacora is a S21sec's[5] SIEM (Security Information and Event Manager) that collects information from IT systems, applications, users and external intelligence data feeds into a centralized data warehouse. The OGRC framework joins features from Bitacora and ontologies to provide organizations with a tool suitable for: i) automatically evaluate compliance for many policies –automation–; and ii) providing a flexible platform for monitoring business processes, were policies can be added or modified accordingly –flexibility–.

1.1. Acts and Standards

The first step is to set the policies that organizations consider as the most useful, in order to integrate them in the OGRC framework. At this point, the consultancy services from S21sec pointed out three different policies. The first one is a Spanish act: the Spanish

[4] Bitacora^{TM} is a trademark
[5] S21sec^{TM} is a trademark

Data Protection Act (LOPD). The LOPD[6] aims at protecting sensitive and personal data from citizens. This act is specifically devised to guarantee two main issues: i) honor; and ii) personal and family privacy. The scope includes mediums where these sensitive data are managed. The act also involves both citizen's rights and obligations of those persons or/and companies responsible for managing this kind of data.

The second one is also a Spanish act: the National Security Scheme or ENS[7]. The ENS states the principles and requirements of a security policy regarding the use of electronic tools, ensuring an adequate protection for information. In addition to scoping systems, services and security measures that are present in most real scenarios, the ENS is intended to increase security of services and information provided by systems held by public administrations, as well as protecting communications between citizens and these systems. ENS does so identifying sensitive assets, setting security dimensions and its levels, deciding categories for systems and selecting suitable security measures. This process generates a number of documents whose compliance must be certified.

The third policy is an International Security Standard: the Payment Card Industry Data Security Standard (PCI DSS)[8]. The PCI DSS was developed to enhance cardholder data security. The keystone is the PCI DSS, which offers robust and comprehensive standards as well as supporting materials to enhance and facilitate the broad adoption of consistent data security measures globally. These materials include a framework of specifications, tools, measurements and support resources to help organizations ensure the safe handling of cardholder information. PCI DSS applies to all entities involved in payment card processing such as merchants, acquirers, service providers, etc.

1.2. Tools: Ontologies, Protégé and Pellet

In the OGRC framework, ontologies are designed to represent the extracted knowledge from policies presented in section 1.1 in a machine-readable format. Furthermore, its XML Schema allows the execution of a reasoner algorithm against the ontology structure to determine the compliance status of sensitive assets.

An ontology describes the concepts and relationships that are important in a particular domain, providing a vocabulary for that domain as well as a computerized specification of the meaning of terms used in the vocabulary. Gruber defined an ontology as a formal, explicit specification of a shared conceptualization [8]. The main features that make ontologies suitable for our GRC platform are: i) ability to share common information; ii) enabling reuse of knowledge; iii) resilience to changes in the acquired knowledge; and iv) reasoning to determine compliance.

Protégé [9] is a suite of tools for ontology development and use, and it is the main framework used in our projects to implement ontologies. Additionally, many applications developed for the Semantic Web require some kind of reasoning capability. Providing complete reasoning services is essential for many of these applications to function properly. Pellet [10] has a number of features either driven by OWL requirements or Semantic Web issues.

[6] Ley orgánica 15/1999 de 13 de diciembre de protección de datos de carácter personal, (LOPD).
[7] Real Decreto 3/2010, de 8 de enero, por el que se regula el Esquema Nacional de Seguridad
[8] PCI DSS home page `http://www.pcisecuritystandards.org`

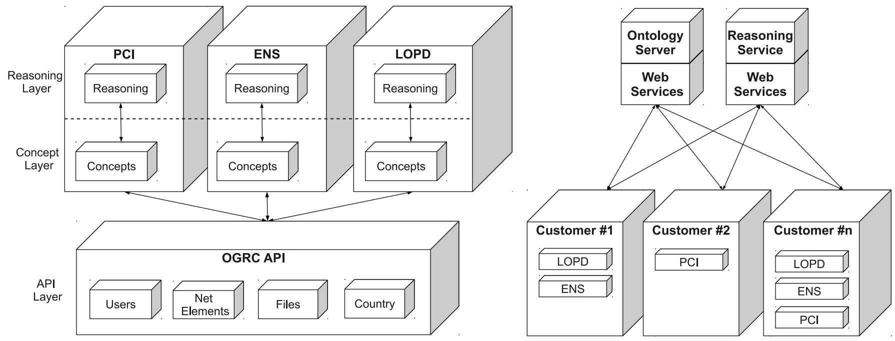

Figure 1. *Left*: Ontologies for each regulation divided in three different layers: API, Concept and Reasoning. *Right*: Architecture falling in a client/server paradigm, including ontology and reasoning modules.

1.3. Architecture

In this section we focus on the designed architecture concerning two main issues of the OGRC framework: the development of ontologies and the updating process for the customers' software. Ontologies design method fall on a three-layer paradigm. Fig. 1 *Left* depicts the structure. From bottom to top, the first layer is an Application Programming Interface (API) composed by common elements, which are shared between specifically modeled policies. Next layer provides the acquired knowledge from each policy modeled independently. The last layer joins acquired knowledge in the previous layer and suitable elements to obtain reasoning capabilities. In this architecture, changes are propagated from lower layers to upper layers and the modular structure eases the inclusion of new policies. Both benefits provide the flexibility feature to the OGRC framework. Besides, the reasoning capabilities obtained from the third layer also favor the automation feature.

The architecture of the OGRC framework is specifically designed to manage ontologies and reasoning algorithm executions through a client/server architecture. This is depicted in Fig. 1 *Right*. The main idea is to create a structure where users –client side– are able to update and download new policies from the server. In addition, a reasoning machine is enabled to release the user machine from the reasoning process.

1.4. Detection of Non-Compliance Situations

The ontologies developed in OGRC project have different individuals that represent users, passwords, machines, servers and so on. Therefore, the main reason for the described ontology-stack approach in section 1.3 is to enable the detection of ontology individuals that allow the discovery of non-compliance situations. Hence, ontologies are able to detect non-compliant situations. The classification is achieved in three different steps: i) modeling a special subhierarchy of undesired or problematic situations besides the fundamental domain concepts; ii) putting these problem-defining subclasses in the Reasoning layer of the ontology-stack together with the logical rules; and iii) deploying dedicated services in the reasoning machine that execute the reasoner against the ontology-stack. When this last step is taken, individuals in the Reasoning layer are classified into the problem-defining subhierarchy, according to their properties. Later stages inspect this classification, looking for individuals denoting possible non-compliance situations.

2. Conclusions

Over the last years, sensitive information that organizations manage from citizens has increased exponentially. Therefore, governments and international organizations have developed regulations to ensure privacy and security of sensitive data. In this work we introduce the OGRC framework, which implements semantic data models in the form of ontologies. Ontologies provide two main features: i) *automation* of assets compliance evaluation; and ii) *flexibility* to incorporate modifications and new policies.

Currently, semantic models for two Spanish acts and one international standard are included in the OGRC framework. The ontology architecture falls on a three-layer paradigm. From bottom to top, the first layer is aimed at sharing common knowledge. Then, the second layer is specifically designed for acquisition of knowledge from each developed policy. The last layer provides elements needed in order to run a reasoning algorithm to determine the assets' compliance. Finally, the OGRC framework features a client/server architecture with an ontology repository in the server side, which allows users to connect to the server to update or download new policies.

Acknowledgments

This work was conducted using the Protégé resource, which is supported by grant LM007885 from the United States National Library of Medicine.

References

[1] A. Tarantino, *Governance, Risk, and Compliance Handbook: Technology, Finance, Environmental, and Int'l Guidance and Best Practices*, Business & Economics, John Wiley and Sons, Hoboken, NJ, 2008.

[2] N. Racz, E. Weippl and R. Bonazzi, *IT Governance, Risk & Compliance (GRC) Status Quo and Integration: An Explorative Industry Case Study*, IEEE World Congress on Services (SERVICES), IEEE Proceedings, (2011), 429–436.

[3] N. Racz, E. Weippl and A. Seufert, *Governance, Risk & Compliance (GRC) Software – An Exploratory Study of Software Vendor and Market Research Perspectives*, 44th International Conference on System Sciences (HICSS), IEEE Proceedings, (2011), 1–10.

[4] Network Frontiers, *Unified Compliance Framework (UCF)*, (2011), available on-line at: http://www.unifiedcompliance.com.

[5] MODULO Solution for GRC: Comprehensive Solutions for Governance, Risk and Compliance Management, *Modulo Risk Manager*, (2011), available on-line at: http://www.modulo.com/risk-manager.

[6] Oracle, *Oracle Fusion GRC: The New Standard for Risk Management and Compliance*, (2011), available on-line at: http://www.oracle.com/us/solutions/corporate-governance/index.html.

[7] S21sec, *Bitacora v5*, available on-line at: http://www.s21sec.com/productos.aspx.

[8] T.A. Gruber, *A translation approach to portable ontology specifications*, Knowledge Acquisition **5**(2), (1993), 199–220.

[9] Stanford University, *Protégé*, available on-line at: http://protege.standford.edu.

[10] E. Sirin, B. Parsiaa, B.C. Graua, A. Kalyanpura, Y. Katza, *Pellet: A practical OWL-DL reasoner*, Web Semantics: Science, Services and Agents on the World Wide Web, **5**(2), (2007), 51–53.

Policy Making: How rational is it?

Tom M. van Engers, Wouter van Haaften[a,1] and Ig Snellen[b]
[a] *Leibniz Center for Law, University of Amsterdam*
[b] *Erasmus Univeristy Rotterdam*

Abstract. The last two decennia developers have created a wide variety of tools that are to support policy-making. These tools vary from supporting voting procedures to agenda setting, gathering issues and structuring arguments. While some of these tools may have some merits in specific contexts of use, uptake of such tools in actual policy making is still rather limited. The reason for this may be that most of these tools seem to be built with an oversimplified conceptualization of policy-making in mind. In this paper we'll present some essential elements of policy-making and from this characterization we'll derive the requirements for support tools. Furthermore we'll characterize one of the recent tools that are claimed to support policy-making, Parmenides.

Keywords. Policy-making, Group Decision Support Systems, Argumentation, Decision-making, Rationality.

Introduction

From the beginning of the nineteen eighties developers have created tools to support policy-making. Many of these tools resulted from research projects and have been tested in settings that were to reflect actual policy-making before being made available. From a tool perspective we can categorize these tools as Group Decision Support Systems (GDSSs). Research on these systems started in the early eighties when personal computers and computer networks became a commodity (see [1]). The GDSS community translated traditional group techniques into a computer-supported form, with the clear advantage that the place and time restrictions could be relaxed, claiming that this would also enhance synergy and objective evaluation of ideas amongst others. The question relevant for us is in how far GDSS supported decision-making contributes to effective decision-making, more specifically effective policy-making.
The increased interest in policy-making and support tools for policy-making is caused by, are due to the raised interest in electronic democracy also known as eDemocracy and eParticipation. eDemocracy and eParticipation are proposed to be an answer to the diminishing participation of citizens in politics. Especially the decreased number of citizens becoming member of political parties and the declining number of people that participate in elections, are a threat to the democracy and to the legitimacy of the law. For this reason many European member states as well as the European Committee have started various initiatives to 'reconnect' the citizen. Enabling participation by electronic means, particularly using Internet technology, is expected to increase such citizens' participation. This participation should not be limited to voting for or against certain

[1] Corresponding Author can be reached at vanEngers@uva.nl .

proposals (sVoting), or the creation and distribution of them (ePetition), but allow for a broader form of participation, allowing the 'professional' policy-makers to 'use the wisdom of the crowds' and understand which interest group supports what element of a policy. This way governments hope to build support for their policies and hence increase legitimacy of the legislation created to effectuate these policies.

1. Policy-making

An eDemocracy and eParticipation focus may lead to a shift in functional requirements for the supporting GDSSs. In order to understand these requirements we should first decide on what policy-making actually is. In this paper we define policy-making as the process that leads from identifying a problem to the formulation of a policy to solve it. This process usually is following a procedure, i.e. a principle or rule, to guide decisionmaking and achieve a rational outcome. A policy is a statement of intent or a commitment for those that adhere to the policy. For that reason at least, the decision-makers can be held accountable for their policies.

Policies are often an answer to a complex problem. Public policies, as well as political struggles about policies, are based on sets of assumptions, which together form the theory of the policy concerned. The aim of a policy-making process is to produce a solution to that problem, which requires construction of such theory and specifically the formulation of possible actions and selecting the appropriate ones, given certain selection criteria.
In order to separate the problem space from the solution space, we separate the policy theory in a 'policy field theory' and a 'policy effects theory'. A policy field theory has to give answers on questions like: which actors and factors do create problems and possibilities in a certain policy field, which require the attention of the policy makers. As such, a policy theory has a causal component and a normative component.
A policy effects theory describes the effects of possible actions that are assumed to provide a solution to the problem at hand. The connection between these actions and the problem is through factors that have a causal relationship to the problem. The policy-making process is aimed at finding and deliberating possible alternative solutions, i.e. alternative actions.
The heart of policy-making is therefore the construction of the causal component of the policy field and making an inventory of the wishes and desires playing a role in the normative component. Consequently we can construct a policy effects theory describing possible measures and interventions and there assumed effects on the causal components described in the policy field theory.

Both the policy field theory and the policy effects theory can be subject to discussion, requiring argumentation and deliberation. Furthermore in actual policy-making processes, e.g. in politics, stakeholders are not completely independent. Actually various dependencies-relations may exist between them as they may have conflicting interests in one problem area but have the mutual interests in another. Consequently in democracies and societies in general the optimal policy is usually a compromise between various interests and preferences. It may still be the case that the strongest arguments win, but what is 'best' is less obvious and certainly not as straightforward as suggested by the typical school book argumentation examples that are often the used by argument support tool developers and scientist studying computational argumentation (see e.g. [5]). Also choosing between policy options is not merely counting votes which is demonstrated in everyday practice where is widely accepted that within democracys the majority cannot always merely dictate its view to a minority. Snellen [3] distinguishes within his policy effect theory four perspectives, which he calls rationalities; political, legal, economical and technical/scientific . The stakeholders in a policy-making process may be biased towards one of them. The question is if it is possible to design GDSSs that can assist in disentangling the various rationalities that need to be analyzed in order to understand the problem at hand, and support the deliberation on the possible solutions to that problem based on the pros and cons thereof.

One recently developed tool that is aiming to support policy-makers is Parmenides [4]. Parmenides is developed in a EU sponsored 7th Framework Programme and can consequently be expected to incorporate the most recent research findings and state of the art technology. The developers of Parmenides focus on the consultation aspect and the structuring support of the various elements of policy-making and consequently refer to their tool as a structured consultation tool (SCT) rather than a GDSS. From our perspective it is nevertheless a GDSS with specific target audience. After introducing some policy issues in an actual example, we will show how far Parmenides fits in to such policy-making process and we will analyze in how far Parmenides could support it.

2. A policy-making example

In this section we'll introduce an actual policy debate about road pricing, that was recently discussed in the Netherlands. This example will clarify the nature of both policy field theory and policy effect theory, and their application in policy making. The central problem to solve is road congestion. From a policy field theory perspective this problem is a result of a growing need for transport influenced by the technical, economical and demographical developments. In fulfilling the need for private transport road use has increased over the past decades, more than public transport, leading to road congestion, specifically on certain motorways and during rush hours in the morning and the early evening. This road congestion leads to a vast amount of vehicle loss hours, causing substantial economical damage. Since the problem has a clear supply and demand character solutions could be found in both areas, increasing road capacity, or decreasing car use, specifically during rush hour.
In search for a solution for the road congestion problem most traffic economists consider road pricing a sustainable way to solve it. This solution will reduce the demand for road space. Potentially it could also be tailored to particular place and time, thus making the instrument more accurate in solving the congestion problem. The alternative would be to increase the supply by building extra lanes and roads, but this would be costly and eventually insufficient.
This policy field theory can be visualized as a set of assumptions regarding the chain of causes and consequences determining the (problematic) developments in the policy object. This visualization does not indicate loops, interdependencies etc., but gives a straightforward chain of causes and consequences. Visualizations like this are

commonly used in qualitative modeling (see e.g. [2]). The problem field theory relating to the road pricing policy is depicted in figure 1.

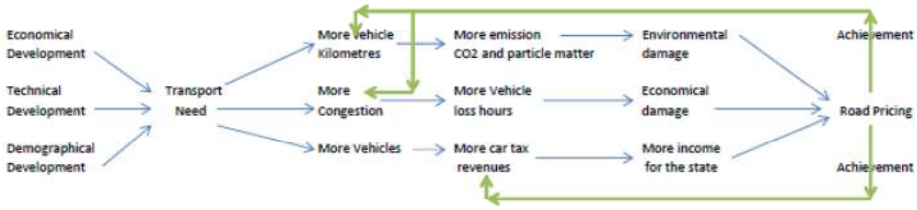

Figure 1. This picture describes the causal factors and their relationships that constituent the policy field theory related to the road pricing policy.

After the policy field has been described a policy effects theory pointing out the instruments and their effects need to be created. The effects and connection to the problem space are explained and demonstrated with the help of the policy field theory. The constituents of the policy field theory serve as the intervention variables. The policy effect theory contains the actions that will have to be taken to influence those variables. The choice for the road pricing instrument will lead to fewer kilometers driven. For the environment this results in less emission of CO_2 and of particle matter, serving the environmental policy value. As far as road congestion is concerned this will generally be reduced, and so will be the amount of vehicle loss hours, thus serving this policy value. More reduction could be achieved by introducing an additional charge specifically during rush hours and in congestion areas. Finally road pricing will lead to an increase of income for the government because every kilometer driven will have to be paid for. Compensation can be found in the decrease of sales tax on cars, in order to achieve the desired Government budget neutrality.

3. A Structured Consultation Tool; Parmenides

The Parmenides e-participation forum structures proposals for political action, to allow the results to be computationally analyzed. Rather than starting from a policy field theory Parmenides starts with a proposal, i.e. an instrumental view or actions that should be taken in order to solve the problem and then helps to establish if the actions chosen will actually contribute to the solution and to the preferred values. In other words the policy effect theory, describing possible actions are input for Parmenides. Consequently these actions can be scrutinized. Parmenides primarily handles proposals and it can only handle one action at a time. If multiple instruments are proposed (as part of the policy effect theory) multiple Parmenides sessions will have to be instantiated. Each proposal invites the community addressed to comment. In these comments the effects of the actions on the policy field theory (does the action serve this policy value(s)) will be scrutinized. This way Parmenides could be a helpful tool to test the validity of road pricing as the answer to the various desires like a) less car traffic b) less vehicle loss hours, c) less emission of CO_2 and particle matter and e) a more 'pay as you drive' like taxation on cars.

As it comes to rationalities – political, juridical, economical and technical/scientific – one could say that when using Parmenides for e-Democracy purposes the rationality is merely a political one. Knowing how their voters feel about a certain proposal is most interesting to politicians, but this may be less interesting to economists or legal experts in the field involved. Because the opinion of voters is only a modest part of the political trade off, even with respect to this rationality the usability of Parmenides is limited. It may however be possible to use Parmenides confronting the proposal separately with jurists, economists and technical experts, each within their own realm. In that case various sessions will have to be run, each with its own focus on the rationality applied.

4. Conclusions

In this paper we have characterized policy-making and described an approach from administrative science perspective. We also described a recent tool development, the SCT Parmenides, for which development AI-techniques such as computational argumentation and formal semantics are used. This tool is used for argument reconstruction and argument visualization, supports the analysis of possible causes and consequences and start conditions of intervention variables and the semantic analysis thereof. Parmenides is intended to involve a wide audience in consultation process, and is yet too limited to support some important aspects of policy-making. Expanding the problem space and solution space, i.e. the respective policy field theory and policy effects theory are hindered by taking a policy proposal, i.e. an instrumental view, as a starting point. This might have people focus on the wrong problems or lead to neglecting alternative solutions. Despite this Parmenides is a promising development, and future extensions could help to turn this in an applicable tool in actual policy-making processes. Some aspects related to the policy field theory and the policy effects theory, such as the trustworthiness, consistency and back up evidence are useful extensions of the current administration theoretical framework. Further research after the applicability of instruments like Parmenides should be conducted. Particularly when dealing with policy proposals that have to be examined by experts from various rationalities, in order to get a more complete picture of the pro's and con's of a proposal.

References

[1] G. DeSanctis, B. Gallupe, Group Decision Support Systems: A new frontier, in *ACM SIGMIS Database Volume 16 Issue 2*, Winter (1984), ACM New York, NY, USA.

[2] J.A.M. Vennix, Group Model Building: tackling messy problems. *System Dynamics Review 15 (4)* (1999), *379-401*.

[3] I.Th.M. Snellen, Conciliation of rationalities: the essence of public administration,. Administrative Theory & Praxis, Vol. 24, June 2002, No. 2.

[4] A. Wyner, K.Atkinson, T Bench-Capon, Towards a Structured Online Consultation Tool, University of Liverpool, Liverpool L69 3BX, UK, {azwyner, Katie,tbc}@liverpool.ac.uk, http://www.csc.liv.ac.uk/.

[5] D. Cartwright, K. Atkinson and T. Bench-Capon (2009): **Supporting argument in e-democracy.** In *A. Prosser (editor): Proceedings of the Third Conference on Electronic Democracy (EDEM 2009)*, pages 151-160. Vienna, Austria.