REGULATING WEB-BASED COMMUNITIES

Alan S. Abrahams
Department of Operations and Information Management, The Wharton School, University of Pennsylvania
3730 Walnut Street, Philadelphia, Pennsylvania, United States of America
asa28@wharton.upenn.edu

David M. Eyers, and Jean M. Bacon
University of Cambridge Computer Laboratory
William Gates Building, 15 JJ Thomson Avenue, Cambridge, CB3 0FD, United Kingdom
{dme26, jmb}@cl.cam.ac.uk

ABSTRACT

Regulations of organizations and the contracts they enter into are generally recorded in human-readable, legal documents. Were such documents interpretable by computer, software could perform ongoing compliance checking. This would allow organizations to ensure they operate within legal bounds, or locate flaws in the regulations themselves. This paper presents early work on the CamPACE environment for contract parsing, analysis, refinement and consistency checking.

KEYWORDS

Groupware, Contract analysis, Compliance-checking, Conflict resolution, Conflict detection, Linguistic analysis.

1. INTRODUCTION

An emerging distributed groupware application is regulation definition, analysis, and compliance checking for communities of individuals and organizations. We are developing an web-based software prototype, **CamPACE**, to assist with this task. Our tool allows textual contracts and regulations to be uploaded, semantically tagged, and then assessed for conflicts. Event logs (behavior traces) can then be fed into the system to monitor for compliance against the policies previously defined.

This paper begins with an analysis of related work (§2), and then provides an overview of the **CamPACE** prototype (§3), which is being developed jointly at the University of Cambridge and the University of Pennsylvania . We conclude with a discussion of our future plans.

2. RELATED WORK

Past work in electronic contracting and regulating pertains to our current research. A multitude of electronic contracting approaches exist: (Bons et al. 1995, Daskalopulu et al. 2001, Grefen et al. 2000, Greunz et al. 2000, Kafeza et al. 2001, Kerrigan et al. 2003, Lee 1980, Merz 1998, Milosevic 1995, Morciniec 2001, OASIS 2002, Peyton-Jones et al. 2000, Reeves et al. 2002, Weigand & Hasselbring 2001) Abrahams (2002) provides a detailed review of these and others. Most of these approaches cannot express or detect conflicting provisions, and many cannot monitor active systems for contract violations. Furthermore, none of these existing approaches integrate with linguistic databases. Previous contract assessment work has been based on small, single contracts. Lee and co-workers (Bons et al. 1995) have implemented a Petri-Net-based trade procedure executor, whilst Daskalopulu (2001) and colleagues provide a finite-state-machine-based conceptual framework for assessment of a small number of obligations. Existing approaches have focused on developing small-scale conceptual solutions to the problem, rather than on studying real-world scenarios. In the field of legal expert systems, Sergot et al (1986) showed that the citizenship of an individual could be determined on the basis of approximately 500 rules of legislation. Dasakalopulu (1999) showed how

contracts may be assembled from blocks of text, but did not provide a multi-user analysis and assessment tool with semantic lookup features. StatuteExpert (SoftLaw Corporation, 2003: http://www.softlaw.com.au/) is a commercial online environment for documenting law into a rules-based expert system. The CISAU (Douglas et al. 1995) and RegNet (Kerrigan et al. 2003) projects are academic initiatives that have looked at the formalization of regulations. We believe our software environment is unique in that it offers a collaborative environment, with integrated linguistic database queries (Abrahams & Mimouni 2003), for formalizing, analyzing, and monitoring-against multiple sets of regulations. Novel inbuilt conflict-checking and resolution facilities (Abrahams 2002, Abrahams & Bacon 2002a; 2002b, Abrahams et al. 2002) are also provided.

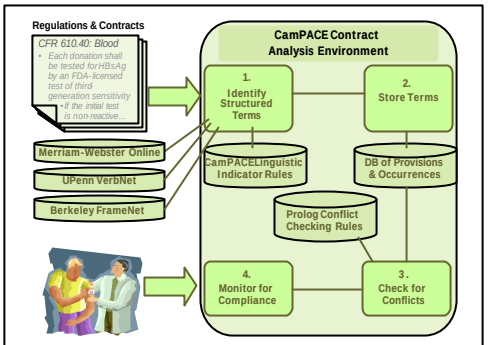

Figure 2: *CamPACE* Context of Use

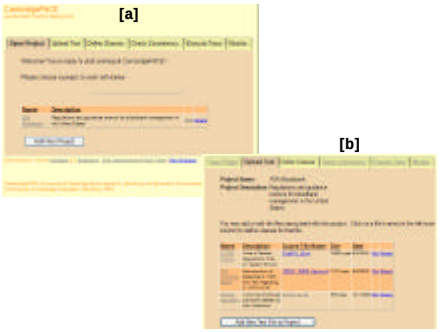

Figure 1: Open Project / Upload Files

3. CAMPACE

As a joint initiative at the Universities of Cambridge and Pennsylvania, we are developing a software prototype, CamPACE, to allow distributed users to define and analyze regulations, and assess behavior against those regulations. In our application scenario we are working with a set of blood management regulations defined in Title 21 Section 610.40 of the United States Code of Federal Regulations. These regulations have been supplemented by subsequent guidance memoranda issued by a Director of Blood and Blood Products at the Food and Drugs Administration (FDA). The goals of our prototype are three-fold:

1. Consistency-checking of these regulations, defined by multiple individuals and institutions, against each other. Both conflict detection, and case-based conflict resolution must be supported.

2. (Partial) Completeness checking of these regulations, so that regulators may be prompted to make explicit any vague, ambiguous, or incomplete directives.

3. Compliance-checking of various blood management software implementations against the federal regulations and amendments. Individual blood management software vendors develop software for processing blood donations – we would like to systematically assess event traces from their software against what is mandated by law. For instance, a software product that has allowed a blood sample which tests positive for the Hepatitis B Surface Antigen (HBsAg) to be used for non-autologous transfusion (i.e. for transfusion to an individual other than the original donor) is clearly in breach of federal regulations.

Figure 1 above shows the context of use of the CamPACE prototype. Federal regulations and guidances are uploaded to the software tool. Linguistic rules, sourced from an internal database and from external sources such as Merriam-Webster online dictionary (http://www.m-w.com/), the University of Pennsylvania's VerbNet (http://www.cis.upenn.edu/verbnet/), and Berkeley's FrameNet (http://www.icsi.berkeley.edu/~framenet/), are used to help identify semantic skeletons for the individual provisions (1). These structured provisions are then stored in a database (2) and can be checked against each other for consistency using a purpose-built Prolog engine which we have produced (3). Finally, event traces from actual blood donation scenarios can be tested for compliance against the stored regulations (4).

The following sub-sections summarize the main steps in using the CamPACE environment:

3.1 Open Project and Upload Text

Users can define a project name and specify a project description (Figure 2[a] above). They can then upload one or more text files containing regulations, guidance memoranda, or contracts (Figure 2[b] above).

3.2 Define Indicators and Exceptions

Users can define markers (prefixes, suffixes, infixes, or regular expressions) that have particular meanings in English specifications. For example, the suffix '-st' as in 'first' often means that a list is being sorted, whereas the suffix '-or' as in 'donor' implies a corresponding occurrence of donating has happened or may happen. Users can also specify exceptions to the indicator rules (e.g. in 'thirst' the '-st' suffix does not imply a list has been sorted). A list of over 400 English-language indicators and exceptions (identified originally in Abrahams, 2002 and Abrahams & Mimouni, 2003) are predefined in the CamPACE linguistic knowledge-base for the user.

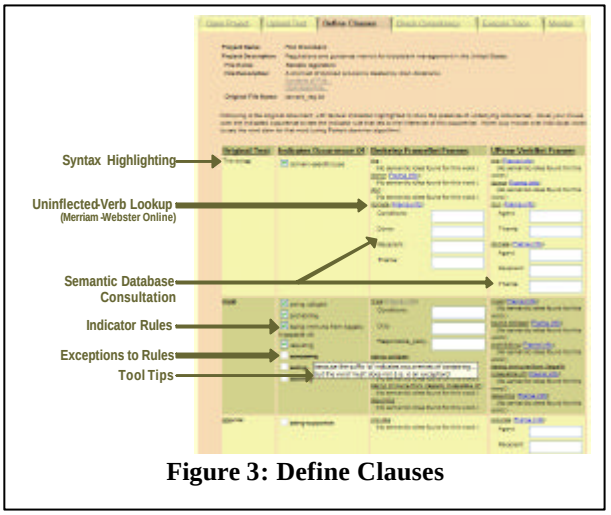

Figure 3: Define Clauses

3.3 Define Clauses

CamPACE employs its internal linguistic indicator and exception rules (§3.2), as well as external information from Merriam-Webster, FrameNet, and VerbNet (Figure 1 above), to allow the user to produce a structured representation of individual provisions. CamPACE reads the original text of the regulation, and highlights for each word the implications and associations for that word based on linguistic rules. Semantic role (frame) information downloaded from FrameNet and VerbNet helps the user ensure completeness, by prompting them to specify the contents of unfilled frame slots (roles) – see Figure 3.

3.4 Check Consistency

CamPACE feeds the structured provisions through a Prolog rules engine which defines a variety of conflict-checking rules, to check for a catalogue of safety and liveness constraints (Figure 4). Individual conflicts are flagged; conflict resolution requires the definition of additional provisions which specify how the conflict is to be avoided, or whether it is to be ignored.

3.5 Execute Trace and Monitor

Users can upload an event log to CamPACE. CamPACE will then monitor for violations of the defined regulations, and trace any violations to their originating clause (i.e. their provenance in law) and source event-scenario (i.e. evidence).

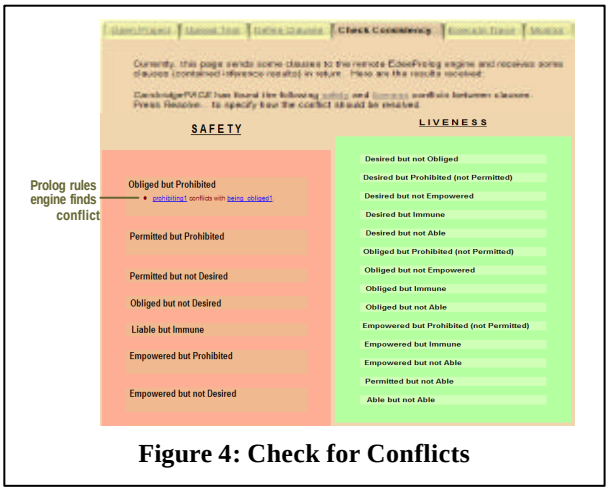

Figure 4: Check for Conflicts

4. CONCLUSION

CamPACE is still in its initial phase of development. We intend to employ data mining and natural language processing techniques (specifically, anaphora resolution mechanisms) to allow semantic roles to be semi-automatically filled by the software – currently, this laborious process is completely manual. Additional structure-determination rules, lexical and semantic definitions (e.g. syno-, anto-, mero-, hyper-, hypo-, and tropo-nym relations), and user interface enhancements are needed – for the latter, we aim in particular to provide facilities to allow operational and contractual events to be dynamically recorded and notified using SOAP-based web-services. Final testing of real-world event logs against complete, consistency-checked regulation texts can then be attempted.

ACKNOWLEDGEMENT

This research is supported by Microsoft Research Cambridge, and Cambridge Australia Trust grants.

REFERENCES

Abrahams, A.S., 2002. *Developing and Executing Electronic Commerce Applications with Occurrences*. PhD Thesis. University of Cambridge Computer Laboratory. Cambridge, England.

Abrahams, A.S. and Bacon, J.M., 2002. A Software Implementation of Kimbrough's Disquotation Theory for Representing and Enforcing Electronic Commerce Contracts. *Group Decision and Negotiations Journal*. 11(6), 487-524.

Abrahams, A.S. et al, 2002. A Coverage Determination Mechanism for Checking Business Contracts against Organizational Policies. *3rd VLDB Workshop on Technologies for E-Services (TES'02)*. Hong Kong, China. *Lecture Notes in Computer Science 2444*. Springer-Verlag. Berlin, Germany. 97-106.

Abrahams, A.S. and Bacon, J.M., 2002. The Life and Times of Identified, Situated, and Conflicting Norms. *Sixth International Workshop on Deontic Logic in Computer Science (DEON'02)*. Imperial College, London, UK.

Abrahams, A.S. and Mimouni, N.K., 2003. Semi-automated analysis of electronic commerce application specifications. *Proceedings of the 6th International Conference on Electronic Commerce Research (ICECR6)*. Dallas, Texas.

Bons, R.W.H. et al, 1995. Modelling Inter-organizational Trade Procedures Using Documentary Petri Nets. *Proceedings of the 28th Hawaii International Conference on System Sciences*.

Daskalopulu, A. et al, 2001. E-Contract Fulfilment and Agents' Attitudes. *Proceedings ERCIM WG E-Commerce Workshop on the Role of Trust in E-Business*. Zurich.

Daskalopulu, A., 1999. *Logic-Based Tools for the Analysis and Representation of Legal Contracts*. PhD Thesis. Department of Computing, Imperial College, University of London.

Douglas, S. et al, 1995. Construction Industry Specification and Understanding. *Proceedings of Language Engineering '95*. Montpellier, France.

Grefen, P. et al, 2000. CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises. *International Journal of Computer Systems Science & Engineering*. 15(5), 277-290.

Greunz, M. et al, 2000. Supporting Market Transactions through XML Contracting Containers. *Proceedings of the Sixth Americas Conference on Information Systems (AMCIS 2000)*. Long Beach, CA.

Kafeza, E. et al, 2001. View-Based Contracts in an E-Service Cross-Organizational Workflow Environment. *Proceedings of the Second International Workshop on Technologies for E-Services (TES'01)*. Rome, Italy. *Lecture Notes in Computer Science 2193*. Springer-Verlag. Berlin, Germany. 74-88.

Kerrigan, S. et al, 2003. Regulatory Information Management and Compliance Assistance. *Proceedings of the National Conference on Digital Government Research*. Boston, MA.

Lee, R.M., 1980. *CANDID: A Logical Calculus for Describing Financial Contracts*. PhD Thesis. Department of Decision Sciences, The Wharton School, University of Pennsylvania. Philadelphia, PA.

Merz, M., 1998. Electronic Contracting with COSMOS – How to Establish, Negotiate, and Execute Contracts on the Internet. *Proceedings of the 2nd International Enterprise Distributed Object Computing Workshop (EDOC'98)*. IEEE.

Milosevic, Z., 1995. *Enterprise Aspects of Open Distributed Systems*. PhD Thesis. Department of Computer Science, University of Queensland. 154-248.

Morciniec, M. et al, 2001. Towards Regulating Electronic Communities with Contracts. *2nd Workshop on Norms and Institutions in Multi-Agent Systems, 5th International Conference on Autonomous Agents*. Montreal, Canada.

Organization for the Advancement of Structured Information Standards (O.A.S.I.S), 2002. *OASIS ebXML Collaboration-Protocol Profile and Agreement Specification Version 2.0*. Available at: http://www.oasis-open.org/committees/ebxml-cppa/documents/ebcpp-2.0.pdf

Peyton-Jones, S. et al, 2000. Composing contracts: An adventure in financial engineering. *International Conference on Functional Programming*. Montreal, Canada.

Reeves, D.M. et al, 2002. Automated Negotiation from Declarative Contract Descriptions. *Computational Intelligence*. 18(4), 482-500.

Sergot, M.J. et al, 1986. The British Nationality Act as a Logic Program. *Communications of the ACM*. 29(5), 370-386.

Weigand, H. and Hasselbring, W., 2001. An Extensible Business Communication Language. *International Journal of Cooperative Information Systems*. 10(4), 44-56.