POLICYFORGE: A COLLABORATIVE ENVIRONMENT FOR FORMALIZING PRIVACY POLICIES IN HEALTH CARE

Andras Nadas
Institute for Software Integrated Systems at Vanderbilt University

Laszlo Juracz, Janos Sztipanovits
Institute for Software Integrated Systems at Vanderbilt University

Mark E. Frisse, Ann J. Olsen
Vanderbilt University Medical Center
Motivation

- Health care delivery requires the coordination of activities across many different service providers and organizations and generally requires the secure exchange of health information across organizations.

- Privacy and care management policies at the federal, state, and institutional level lead to a confusing array of potentially contradictory or subjective policy interpretations.

- A scalable solution is needed that can address the technical and human challenges.
The Problem of Policy Formalization
Policy Encoding (Current Approach)

- Policy Formalization is complex...
  - Have to understand legal domain, context and language
  - Have to understand formal domain, context and language

- ... but feasible for a small set of policies and experimentation.

- Helen Nissenbaum: Contextual Integrity
- Mitchell, Datta (Stanford and CMU’s) Policy Formalization in Prolog
Encoding of large policy sets for multiple system
(Failure of the Current Approach)

- Not feasible for
  - Large sets of policies
  - Multiple systems/semantics
  - Verification/Validation
To make it scale:

- Divide and Conquer approach for the human aspects
- Formal Model Representation for the technical issues
PolicyForge
The Pedigree of PolicyForge

- **SourceForge’s Allura**
  - Open Source version of SF’s own framework
  - Based on TurboGears – Python web framework

- **VehicleForge** (www.vehicleforge.org)
  - Adoption and extension of Allura to create a engineering model forge
  - Designed for the DARPA/FANG challenge
  - Created at ISIS/Vanderbilt

- **PolicyForge**
  - Adoption of VF platform with the addition of formal policy modeling tools
The Architecture of PolicyForge

Policy Exchange
- templates
- ontologies
- policies
- scenarios
- taxonomic search

Projects
- templates
- ontologies
- policies
- policy sets
- scenarios

Tools
- viewer
- authoring
- modeling
- composition
- analysis
- export

HIS, HIE, ACO etc.

Policy Exchange
- User management
- Profiling

Projects
- Collaboration
- Trust management

Collaborative Policy Creation and Analysis

HealthCare Organizations

Policy sets for Enforcement Points
Collaboration tools

- Projects
- User and Group Management
- Forums
- Messaging
- Issue Tracking
- Sharing and Exchange
Workflow of PolicyForge

Collect and Organize Policies And Use Cases

Extract and Markup the Policies

Model
- Ontologies
- Policies (sets)
- Use Cases

Verify and Validate
Policy Contradictions and Entailment Use Case Execution

Export (EHR, HIE, etc.) External Analysis

Define Policy Modeling Language

Match Ontologies

Anchor Language Semantics

Modeling Privacy Aware Health Information Exchange Systems
Andras Nadas, Mark E. Frisse, and Janos Sztipanovits
International Workshop on Engineering EHR Solutions (WEES), October 2012.
Policy Library

- Collect and organize
  - Federal Policies
  - State Policies
  - Institutional Policies
    - Business Agreements
    - Data Sharing Agreements
    - Privacy Notices

- Provides source and provenance for formalization
Ontology Modeling

- Ontologies represent the formal knowledge base.
- Documents from the library can be annotated by ontology terms.
- Words and expressions from the text can be used to create elements in an ontology.
- Ontologies can be imported from standards (rdf-owl and json-ld).
The challenge of relaying the meaning of a specific law to computer scientists has always been present.

Lawyers are trained to understand that a law means what it says on its face, only to the extent that it is not modified or clarified by other information: another statute, case law, etc.

This was done by breaking the text of a rule into discrete sub-rules; breaking the text of each sub-rule into sub-classes of actors, data, actions, permissions, etc.; and, then, associating each sub-rule with sub-classes about the rule’s provenance and its logical relationships with other rules.

K. Krasnow Waterman at MIT Pre-processing Legal Text: Policy Parsing and Isomorphic Intermediate Representation
Policy Modeling

- Uses imported templates as the language.
- Multiple languages can coexist and can be combined if their representation and semantics enables it.
- We prefer graphical languages but textual languages will also be supported (as long as they are formal and meets environmental constraints.)

Simple Policy Model example using Blockly visualization
Semantics and Reasoning on Policy Models

- Semantics define the inner workings of the formal policy language and its interaction with the target domain.
  - Languages can have multiple semantics anchored to them for different domains (e.g. analysis, verification or execution)
- The formal models get their semantics from the templates they are derived from.

- PolicyForge hosts a FORMULA (Formal Modeling Using Logic Programming and Analysis) engine developed by Microsoft Research to provide a state of the art analysis framework.
  - Models from languages that have semantics for analysis by FORMULA can be analyzed for contradictions and entailment.

To be published: Nadas, Levendovszky, Jackson, Madari, Sztipanovits - A Model-Integrated Authoring Environment for Privacy Policies – Science of Computer Programming Special Issue on Success Stories in Model Driven Engineering
Use Case Modeling and Validation

- Use Cases are simple clinical workflows represented on a timeline.
- Concentrates on data flows between systems and accesses by users.
- The dataflow is checked to conform to a selected set of policies.
  - The policy set is also checked for internal consistency.
Export from PolicyForge

Code Generation with Templates

- Code Generator uses Code Templates with StringTemplate v4 Library.
- With only changing the templates it's possible to generate code in different languages, like Formula, Drools, ilog IRL, etc.

Source Code Language Template:
- General Code Template to describe the language. Used to generate the ontologies and models that have static semantics.

Anchored Semantics Definition Template:
- Code Template that define the semantics of the each Policy Template and Relation.
Future Work

- Finish cloud implementation of reasoning algorithms.
- Add more content
- Build Community
- … and more content

- PolicyForge.org will open for collaborators (SHARPS) Late Fall 2013
- PolicyForge.org will open for public in Spring 2014
Thank You!

Questions?