Emerald: Legal Knowledge Engineering Using OWL and Rules

András Förhécz András Millinghoffer György Strausz Gábor Kőrösi

Budapest University of Technology and Economics Multilogic Kft, Budapest

> JURIX 2009 16-18 December 2009, Rotterdam, The Netherlands

Contents

- Our approach in legal modelling
- Description logic for legal terminology
- Extending OWL with rules
- Information collection process
 - example with Duties on gifts in Hungary
- Demonstration
- Summary and future plans



Our approach in legal modelling

- Legal models for normative assessment
 - Legal qualifications & calculations (e.g. tax)
 - Explanation & justification may be more important than the answer itself!
 - logical explanation
 - justification: relevant parts of legal sources
 - other non-legal sources (e.g. common-sense)
- Prior work
 - Allex Gold
 - "old school" expert system applied in social benefits
 - Estrella: HARNESS
 - Given situation is allowed / disallowed and why?
- Goal: framework for legal application development



Using OWL for terminology and beyond

- Legal concepts and relations
 - Precise description of domain knowledge
- Qualifications, legal categories
 - Estrella HARNESS: legal qualifications, deontic notions
 - Using OWL as a KR (description logic) when applicable
- Benefits:
 - clean semantics
 - full model consistency (decidability)
 - model reuse, tool support (reasoners, SW)
- OWL2: most powerful decidable logic for terminology





Extending OWL with rules

- OWL has its own limitations
 - tree-model property: unable to handle cyclic conditions
 - missing: arithmetic, data conversion, aggregate functions (e.g. summation)
- Extending with rules
 - many examples: JessTab, Bossam, SWRL, etc.
 - we adopt DL-safe SWRL: conjunction of class and property expressions with constant / variable fillers
 - add custom features discussed here



Elements of the KB



Example for rules and questions

```
Rule: "R0 - 11/1 - obligation to pay duties"
    donation(?donation),
    subject_of_donation(?donation, ?property),
    {real_property or
        (right_of_pecuniary_value and subject_of_right some real_property)
    }(?property)
    =>
    {duty_of_transaction min 1}(?donation)
Rule: "R1 - basis of duty"
    donation(?donation),
    subject_of_donation(?donation, ?property),
    duty_of_transaction(?donation, ?duty)
    =>
    ?duty.basis_of_duty = ?property.value_of_property
```



Information collection process

- Building the UI for complex models is tedious
- Semi-automatic UI generation
 - similar to chaining in expert systems
 - major difference: OWL has an infinite data model
 - inference and question generation are separated
- Hand-crafted information:
 - set of concepts / properties
 which may occur in questions
 - expected inferences from DL reasoner