Ontology Based Law Representation for e-Government System Public Services

Dhori Terpo
University “E. Çabej”, Faculty of Natural Sciences
Department of Mathematics & Informatics, Gjirokastra, Albania

Endrit Xhina
University of Tirana, Faculty of Natural Sciences
Department of Informatics, Tirana, Albania

Fabjan Lashi
Canadian Institute of Technology, Faculty of Engineering
Department of Software Engineering, Tirana, Albania

Copyright © 2015 Dhori Terpo, Endrit Xhina and Fabjan Lashi. This article is distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract
Process automation software empowers government institutions to provide e-services and to adapt quickly to legislative changes. In complex government systems however process design is a complex task which requires knowledge of the legal framework which regulates the service, process design methodology and expertise on using the processes automation software. Processes design should guarantee that the new process would be compliant to legal acts. Generally legal acts are stored as text files which make it difficult to make software based analysis of them with the purpose to assist the process design activity. This paper presents an ontology based representation of legal acts which regulate a list of services provided by higher educational institutions in Albania. The paper focuses on the enabling tools for reading the ontology and producing a human readable format of the ontology content. The ontology is created using Protégé and exported in owl format. Reading the structured file through a programming language, namely C# is illustrated in order to produce a human readable format of the legal act.
**Keywords:** Legal ontologies, Web Ontology Language, RDF/XML, XSLT

1 **Introduction**

An ontology \[1\] defines the terms used to describe and represent an area of knowledge. Ontologies are used by people, databases, and applications that need to share domain information (a domain is just a specific subject area or area of knowledge). Ontologies include computer usable definitions of basic concepts in the domain and the relationships among them. They encode knowledge in a domain and also knowledge that spans domains. In this way, they make that knowledge reusable.

The Web Ontology Language (OWL) has been standardized by the World Wide Web Consortium (W3C) \[2\] as a powerful knowledge representation language for authoring ontologies. Modeling information in OWL has two practical benefits: as a descriptive language, it can be used to express expert knowledge in formal way, and as logical language, it can be used to draw conclusions from this knowledge.

The OWL standard defines different syntaxes \[3\] based on RDF(S), XML and proprietary text format. The OWL RDF/XML syntax allows for an encoding of an OWL ontology within the RDF(S) framework in RDF/XML serialization. The OWL XML presentation syntax provides a more compact XML format for OWL ontologies, independent from RDF(S). In contrast to these machine-oriented serializations, the OWL abstract syntax serves as a human readable text format to present OWL ontology to knowledge engineers.

2 **Legal Ontologies**

Legal ontologies represent legal knowledge. There have been developed several legal ontologies. Among the best known legal ontologies the following ones can be mentioned:

**FOLaw** (Functional Ontology of Law) \[4\][5], based on normative knowledge, word knowledge, responsibility knowledge, reactive knowledge, and creative knowledge.

**LRI-Core Legal Ontology** \[6\], based on object, processes, physical entities, mental entities, agent, and communicative acts.

**Frame-Based Ontology** \[7\][8] based on norms, acts, and description of concepts.

**LKIF** (Legal Knowledge Interchange Format) **Core Ontology** developed in the Estrella\[1\] project that has two main roles: enable the translation between legal knowledge bases written in different representation formats and formalisms and secondly, as a knowledge representation formalism that is part of a larger architecture for developing legal knowledge systems.

---

1 Estrella is a 6th European Framework project (IST-2004-027665). See also: http://www.estrellaproject.org
3 Albanian Higher Education Law Ontology

An ontology based representation of the Albanian Higher Education Law is developed using Protégé. The Figure 1 represents some of the main concepts of the Albanian Higher Education Law ontology.

![Figure 1 Albanian Higher Education Law Ontology](image)

4 RDF/XML representation of the ontology

The ontology is exported in RDF/XML format. The representation of classes, attributes and individuals is illustrated by the code in Listing 1.

**Listing 1.** Representation of the ontology in RDF/XML format

```xml
// Classes
<owl:Class rdf:about="http://example.org#HigherEducationInstitutions">
  <rdfs:label>HigherEducationInstitutions</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
</owl:Class>
// Data properties
<owl:DatatypeProperty rdf:about="http://example.org#nrOfFaculty">
```
5 Reading RDF/XML content through C#

The RDF/XML content is read through C# code and is projected in a GridView in an ASP.NET web page. The GridView (as shown in Figure 2) contains the subclasses of the HigherEducationInstitutions Class. Listing 2 illustrates the reading of the content of the ontology.

Listing 2. Reading RDF/XML content

```csharp
private void HigherEducationInstitutions()
{
    // Create DataTable
    DataTable dt = new DataTable();
    // Put some columns in it.
    dt.Columns.Add(new DataColumn("Nr", typeof(string)));
    dt.Columns.Add(new DataColumn("Institution", typeof(string)));
    // Create an instance of the XmlDocument class that represent the DOM-based
    // parser of the .NET Framework and Load the ontology file into memory
    XmlDocument xmlDoc = new XmlDocument();
    xmlDoc.Load(Server.MapPath("ontology.owl"));
    XmlNamespaceManager ns =
        new XmlNamespaceManager(xmlDoc.NameTable);
    ns.AddNamespace("owl", "http://www.w3.org/2002/07/owl#");
    ns.AddNamespace("rdf", "http://www.w3.org/1999/02/22-rdf-syntax-ns");
    ns.AddNamespace("example", "http://example.org#");
    ns.AddNamespace("rdfs", "http://www.w3.org/2000/01/rdf-schema");
    // Select Specific Nodes Using the SelectNodes() Method
    XmlNodeList nodeList =
        xmlDoc.DocumentElement.SelectNodes("//owl:Class", ns);
    int i = 0;
    foreach (XmlNode node in nodeList)
    {
        // Process each node
    }
}
```
Ontology based law representation for e-government system public services

{  
  {  
    i = i + 1;  
    // Create the record  
    DataRow dr = dt.NewRow();  
    dr["Nr"] = i;  
    dr["Institution"] = node.SelectSingleNode("rdfs:label", ns).InnerText;  
    dt.Rows.Add(dr);  
  }  
}  
//Bind the GridView to the data in the data table for display.  
GridInstituciones.DataSource = dt;  
GridInstituciones.DataBind();  

Figure 2 Reading the content of RDF/XML

The programming language could be used to validate the ontology based workflow representation in order to discover workflow steps that might not be compliant with the legal act.

6 Producing a human readable format of the RDF/XML content

XSLT (Extensible Stylesheet Language Transformations) would enable a quick and easy transformation of the RDF/XML content in an HTML representation. Listing 3 and Listing 4 illustrate an example of displaying the ontology’s content in a browser (as shown in Figure 3).

Listing 3 Transformation of the ontology file into HTML content

protected void Page_Load(object sender, EventArgs e)  
{  

string xmlOntology = File.ReadAllText(Server.MapPath("ontology.owl"));

// Define the contents of the XML control
Xml1.DocumentContent = xmlOntology;

// Specify the XSL file to be used for transformation
Xml1.TransformSource = Server.MapPath("ontology.xslt");

Listing 4 The ontology.xslt file used for transformation

<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:msxsl="urn:schemas-microsoft-com:xslt" exclude-result-prefixes="msxsl"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:example="http://example.org#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
>
  <xsl:template match="/"
      <xsl:for-each select="rdf:RDF/owl:NamedIndividual"
          <xsl:value-of select="example:hasText" disable-output-escaping="yes"/>
      </xsl:for-each>
  </xsl:template>
</xsl:stylesheet>

Figure 3 Producing a human readable format of the RDF/XML content
7 Conclusions

A computer readable format for legal act representation helps the process of e-government public services design. An ontology based representation of legal acts is proposed and used to represent the Albanian Higher Education Law. The ontology stored in a structured RDF/XML format could be processed by high level programming languages such as C#. Producing a human readable format from the structured RDF/XML content could be done through a simple XSLT transformation, translating the file in HTML content.

References


Dhori Terpo, Endrit Xhina and Fabjan Lashi

http://dx.doi.org/10.1006/ijhc.1999.0300

Received: March 27, 2015; Published: April 30, 2015