MetaLex: An XML Standard for Legal Documents

Radboud **Winkels** <winkels@lri.jur.uva.nl>
Alexander **Boer** <aboer@lri.jur.uva.nl>
Rinke **Hoekstra** <rinke@lri.jur.uva.nl>

Abstract

This paper presents a proposal for an open XML standard for the markup of legal documents: METALex. The standard provides a generic and easily extensible framework for the XML encoding of the structure and contents of legal and paralegal documents. MetaLex is first and foremost meant as an interchange format for legal documents. It differs from other existing metadata schemes in two respects: It is language and jurisdiction independent and it aims to accommodate uses of XML beyond search and presentation services.

Table of Contents

1. Introduction

European citizens and enterprizes are confronted more and more with rules and regulations, affecting various aspects of their daily business. These regulations come from international, European, national and local authorities. Despite attempts at harmonization and de-regulation, the amount and complexity of the potentially relevant body of 'law' increases. This is a problem for administrations too, legislative and executive bodies alike. The process of drafting consistent and coherent legislation is getting more complicated, as is that of upholding and applying valid law. ICT has the potential of supporting both the government and citizens in dealing with this increasing body of law. In the E-POWER project[1] we are trying to provide such support. A necessary precondition for this is the electronic availability of legal sources in a structured and standard format. That is why we developed such a format in XML, called MetaLex.[2]

The standard intends to provide a generic and easily extensible framework for the XML encoding of the structure and contents of legal and paralegal documents. This obviously includes legislation and case law, but also written public decisions, internal and external business regulations (for instance ship classification rules as in [Winkels et al. 1999]), and contracts. XML elements and structure are defined in schemas that can be used to validate a document. Since there is a great variety of legal documents that cannot be covered by one normative standard, the standard consists of multiple schemas defining vocabularies that can be mixed in a document.

[1] An IST project, see Acknowledgements at the end of this paper.
[2] See: http://www.metalex.nl

While the standard aims to cover all possible legal sources, the focus of current work is on Dutch legislation: the 2001 Dutch law on income tax in the context of the E-POWER project, and the Dutch penal code of 1881 in the context of the e-COURT project. Later we will cover the structure of (Italian and Polish) court room transcripts (for e-COURT) and case law. The standard differs from other existing metadata schemes for legal documents in two respects; it is language-independent and it aims to accommodate uses of XML beyond search and presentation services.

1.1. XML Standards for Legislation

The efficiency of managing and processing information in legal documents can be dramatically improved by applying XML techniques. As a part of the more general idea of an integrated *semantic web*, documents are enriched with *metadata* to enable smart applications such as (intelligent) retrieval and reasoning. XML schema and metadata definition efforts in the legal domain are initiated either by *legislators* or by *legal publishers*. Examples of initiatives by governments are the British Legal and Advice sectors Metadata Scheme (LAMS) for 'Just Ask!' and the Australian Justice Sector Metadata Scheme (JSMS).[3] These efforts take the *citizen* as a target audience, which does not necessarily make them suitable for other user groups.

Legal publishers have developed standards mostly for internal use, by contract (or as an agency of) a legislator, and are strongly market oriented. This means they often cater for only one language (unless the country they cater for is multilingual, e.g. Belgium), and that they are focused on layout, versioning and references.

In both fields, access to documents is mostly through a search engine interface where documents are positioned in a *fixed categorization*, ordered by legislative domain. Such a fixed categorization creates a potential maintenance issue: The values of attributes may change over the lifetime of a legal document, even if the document itself does not, as the concepts employed in the document change over time and become associated to (disassociated from) other concepts (see e.g. [Rissland et al. 1995]). Also, metadata is often not *extra*: it mostly concerns information already contained in the document itself, or in another document that refers to it. The classification level presupposes that the user of the classification system can read the document to find out why the classification was attached. Although such domain classification schemas have worked for jurists for centuries - most of them predate the storage of legal information on computers -, they are not necessarily adequate for electronic use.

Attributes used in the classification are mostly fairly traditional: *author, creation, modification and promulgation dates, jurisdiction, legal status and language*. As these attributes are rather crude in meaning, the resulting classification lacks a lot of relevant detail, which renders its usefulness questionable for automated reasoning. Identification of documents by jurisdiction assumes that the user of a search service knows what jurisdictions he is in. This is not a trivial task in itself.

LAMSand JSMS require no information about the *structure* of the document. The CorpusLegis project ([Magnusson Sjöberg, 1998]), which developed an SGML DTD for legal documents for the purpose of a large database, did identify some basic structure elements in legal texts. Another small-scale example of this is an extension to the Text Encoding Initiative (TEI) SGML DTD's proposed in [Finke, 1997].

As an alternative to rigid domain classification, statements can be directly identified in the contents of a document. Documents can make statements about other documents and (fragments) of the document itself: the metadata on one document is distributed over different locations. The Resource Description Framework (RDF)[4] is designed to this purpose. The LeXML [5] initiative envisions to identify and describe similarities and differences between legal concepts in different languages by describing them in an RDF 'dictionary' ([Muller, 2002]) as conceptual prototypes. Existing schemas (like JSMS and LAMS) rely on compatibility with HTML's *meta* tag, thus allowing only RDF - *like* statements about the document in which the tag is used.

2. Design Requirements

MetaLex is a generic open standard for legislative documents specifically designed to facilitate the maintenance of decision support software used by public bodies like the applications developed by the POWER and E-POWER

[3]See: http://www.lcd.gov.uk/consult/meta/metafr.htm for more information on LAMS, JSMS and its basis AGLS

[4]See: http://www.w3.org/RDF/. RDF, like XML is an open standard from the W3C that is well-supported with free software.

[5]See: http://www.lexml.de and http://legalxml.org/Dictionary/

groups for the Dutch Tax and Customs Administration (DTCA) in the Netherlands [Engers et al. 2001]. In addition, it offers provisions for more or less traditional functionalities offered by publishers and search engines. The MetaLex XML schema aims to be a standard interchange format for legal documents for the purposes of presentation, description of the relations between legislative documents, search and filtering on meaningful levels of detail [Moens 2001] [Turtle 1995], and version management and file exchange.

It has been designed so that it can be embedded in technologies for legal knowledge representation, code generation, rule generation, and verification of legally relevant 'contents'. The standard itself does not commit to specific viewpoints on the contents of the regulation.

The professional user of legislation today has to keep an eye on regulations from several legislators (for instance municipal, water authority, provincial, national, EU, and international for a civil servant in the Netherlands), and special-purpose software to support decisionmaking processes is affected by - and may have to manipulate - legislation that conforms to many different standards for legislative drafting and is delivered in as many different formats. To achieve *independence of jurisdiction*, the operative principle can only be: *when in doubt, leave it out*. MetaLex is therefore limited to the few features that regulatory documents from these different jurisdictions share.

Application of the principle results in a simple and generic but also rather 'trivial' structure that does not meet specific requirements of potential users. To allow for these specific needs, it should then be possible to add custom extensions to the schema; MetaLex should make the easy things easy, but the hard things possible. *Extensibility* of MetaLex XML elements was realized with the XML schema language.

In general, optimal compliance with open standards and proposals of the World Wide Web Consortium (W3C) and other standardization bodies that are supported by standard or free software, reduces implementation and learning effort for XML developers and increases the usefulness of MetaLex documents. For this reason, the MetaLex standard is specified in W3C's XML schema and RDF, and supports features from standards such as (X)HTML and XML Linking Language (XLink).

Another consequence of the increasing global presence of supranational legislators like the EU is a growing need to separate regulations as such from the specific authorized translations in which they are available. We have to accept as an axiom that for instance a citation in a French text to an international treaty can be resolved to the English translation of that treaty without a change in meaning for users who prefer English. Legislation is increasingly available in multiple authorized translations as a service to immigrants and - in the case of for instance fiscal regulations - potential immigrants. The MetaLex XML schema has been designed with *multilingual regulations* and differences between the main European languages in mind.

3. Description of Legislation

For purposes of representation we distinguish three different viewpoints on the meaning of legal documents:

Form — A legal document can usually be 'recognized' and classified by certain phrases and formulas. Formal requirements on structure and phrasing mostly reflect considerations of consistency of language and ease of access [6] for the reader, but it also provides a context for the interpretation of the content of the document. This latter role is very specific for jurisdiction and timeframe and in many cases cannot be part of the MetaLex schema. Structural requirements are defined in XML schemas where appropriate.

Role — Although we may look at the phrases and formulas in a written decision to classify a document as a law, we know that it is not the structure of the document that makes it a law, but the role the document plays in the activities of public persons and bodies - most importantly the activities that produced the document. Information about the document of this nature is captured in RDF statements 'about' the document.

Content — We also classify documents depending on what its content means: It represents a type of decision. If it is just a public decision its meaning is limited to a particular occurrence or case. If it is a norm or policy its meaning extends to general class of occurrences or cases and it postulates a value theory for making

[6] That the requirements for ease of access can change over time is clear if you compare ancient legislation that was read out to a mostly illiterate and uninformed audience to modern legislation; The Act of Abjuration of 1581 in the Netherlands, for instance, is a fluent narrative that explains recent political events in detail before proclaiming decisions. Today's law is far more structured, but the explanation of motives is usually sketchy at best.

and judging decisions. This is captured in RDF statements 'about' this content: relating acts, norms , agents to (parts of) the document.

The MetaLex XML schema limits itself as much as possible to the form of legislative documents. The XML schema for regulations groups together articles in parts, and subdivides articles in subparts, bottoming out in sentences. Articles are self-contained discourses in the sense that they can be read and understood without reference to nearby articles to resolve anaphoric references. Articles, parts and subparts have an index designation - a number, e.g. 1, 1bis, II, a, B, , or a symbol used for constructing references - and optionally a title. A full sentence is not subdivided, unless it is formatted as a vertical (stacked up and indented) list with sentence fragments indexed as subparts. The introduction and conclusion part of a regulation are not subdivided in sentences, because these are not individually referenced or changed by another law.

3.1. Citation and Version Maintenance

Requirements of citation and version maintenance - one for instance never changes the title of a law - impose some 'generic' structural restrictions on regulations. Our survey of citation practices in laws of several countries and some international treaties shows some interesting patterns. The identity of a regulation in XML can be conceived of in three ways:

Stored Information	A document that is stored in a certain location that can be retrieved by a protocol. Hyperlinks on internet reference information in this way.
Publication	A document that has been published through a designated channel, and is identified by that channel, designation relative to that channel, and publication date. A publication obviously never changes and references to it remain correct.
Organic Form	A document that is a (virtual) reconstruction of a regulation designated by a globally unique citation title or acronym as it exists at a certain time point. Indexed parts or full sentences of an organic regulation can be modified, inserted, or retracted by another publication. To reference an organic regulation it must be clear whether the reference concerns the latest version of the regulation, the regulation at the time the reference was made, or the regulation at a specified timepoint.

The MetaLex schema distinguishes publications and organic documents, and facilitates the connection of organic documents (the *latest* version, for instance) to semi-permanent universal resource identifier (URI), similar to the way in which the W3C makes its standard documents and schemata available. Most web sites that publish legislation for free fail to qualify *which version* they offer. To make a correct citation to a part of an organic regulation identified by a URI is still not trivial; index symbols usually suggest an ordinal relation between indices, but there is no way to determine the size of the interval between for instance articles I and IV. A citation of the 'second article' is therefore not the same as a citation of 'article 2' because 'article 1bis' may be inserted later. [7]

The importance of capturing the identity criteria for regulations is also made apparent by considering the requirements for maintenance of a collection of organic regulations in time. Changes in laws are announced in separate decisions and publishers must keep track of all documents from certain publication channels to be able to reconstruct what the form of an organic regulation is at some time point. Similarly, if you find a written administrative decision on your doormat its status changes when a new written decision retracting it follows two days later.

To keep track of versions MetaLex provides a number of attributes for every structural XML element in the document that can be identified, selected, and thus changed; the *date-publication* of an element is the time the element is officially published or announced. The *date-enacted*, the time the content becomes applicable in decision making, is always later than or the same as date-publication, but before *date-repealed*, the time the content becomes inapplicable in decisionmaking. Between date-enacted and date-repealed the element and its content is *active*, and outside this interval it is *inactive*. Table Table 1 can be used to deduce active time intervals from the presence or absence of these attributes. The *date-version* attribute represents the date the correctness of the content and other

[7] A practice most common when printing was expensive, search engines non-existent, and correcting existing references to articles almost impossible.

dates of the XML element was last verified. The XML document looses its value as a normative reference as time progresses and the time-interval between date-version and today increases.

date-publication	date-enacted	date-repealed	active
date-publication	date-enacted	date-repealed	active
t1	t2	t3	[t2, t3]
t1	t2	-	[t2, t_future]
-	-	t3	[t_past, t3]
-	-	-	none
t1	-	-	[t1, t_future]
t1	-	t3	[t1, t3]
-	t1	-	[t1, t_future]
-	t1	t3	[t1, t3]

Table 1. Active time intervals

An XML element in a newly published regulation refers to another XML element to repeal, enact, or change it. Conversely, if a law, delegation decision, or mandate decision becomes inactive, XML elements referring to it as a source of legislative power also become inactive (in the Netherlands). MetaLex only provides representational primitives that adequately describe relations of this nature between regulations without commitment to a specific normative 'model' for updating documents; it is not sufficient to validate proposed changes or to establish the applicability of a certain article to a legally qualified fact.

4. Jurisdiction and Language

To achieve independence of jurisdiction MetaLex has been limited to common requirements of structure, reference, and identity. Specific legal jargon has been avoided as much as possible to reduce confusion between descriptive and prescriptive use of concepts. The guidelines for legislative drafting ("Aanwijzingen voor Regelgeving") in the Netherlands for instance states that a 'part' ('deel') in a regulation consists of chapters, while a 'section' ('afdeling') consists of paragraphs, articles, or one article indexed 'only article' ('enig artikel'). Copying this vocabulary in Dutch law for groupings of articles would suggest that such constraints apply, and make it impossible to translate even trivial element names for chapter, section, paragraph ('paragraaf'; always has a title in Dutch, because otherwise it is obviously an 'alinea').

The jurisdiction-neutral vocabulary is specified in a simple descriptive English, so that it will be easy to map more specific names to it. Chapter, section, part, paragraph, title, book as description for layered groupings of articles are thus all translated to MetaLex element 'part', which groups one or more parts or articles.

The MetaLex standard supports multi-lingual documents in two distinct ways: through *localization* of XML elements and by providing the means to maintain multiple *language versions* of the same document in one file.

Localization of element tags is achieved by defining a language-specific schema extension to the jurisdiction neutral vocabulary in the standard document schema. This schema extension imports the standard schema and substitutes the element names with a name specific to the target language using the *substitutionGroup* attribute:

```
    <xsd:element name="regeling"
                 type="Regulation"
                 substitutionGroup="Regulation"/>
```

Because of this one-to-one mapping, eXtensible Stylesheet Language (XSL) Stylesheets can easily translate such an extension to and from the standard document schema. This approach has the advantage that users can tag text using their own language, whereas general tools need only to be aware of the English standard schema.

The second multi-language feature supports multiple languages at the sentence level. The *TextVersion* tag allows the user to include different versions of (parts of) a document in one file. Any piece of text, including titles, indices etc., in the document can be enclosed in *TextVersion* tags. The standard *xml:lang* attribute specifies the language in which the text included between the TextVersion tags is posed:

```
    <CitationDesignation id="statute">
        <TextVersion xml:lang="en">
            Rome Statute of the International Criminal Court
        </TextVersion>
        <TextVersion xml:lang="nl">
            Statuut van Rome inzake het Internationaal Strafhof
        </TextVersion>
    </CitationDesignation>
```

On the basis of this *xml:lang* attribute, stylesheets used for generating presentation formats such as (X)HTML, can select the proper language. Figure Figure 1 shows an example of the application of such stylesheets to a MetaLex document. During the first XSL translation, Dutch MetaLex elements are translated into the standard elements, then the second translation produces either an English- or a Dutch version of the document in XHTML.

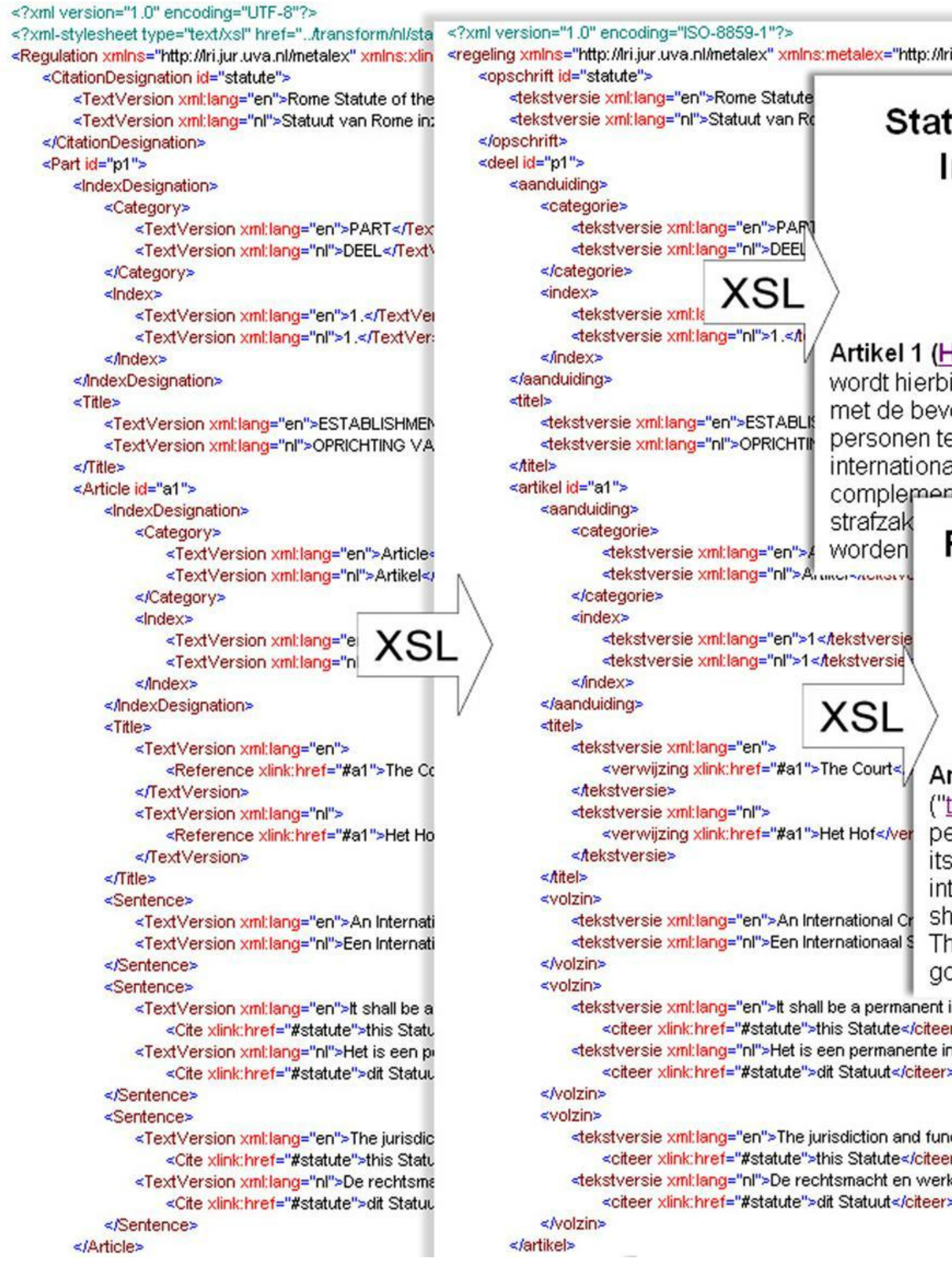

Figure 1.

5. Extensibility

New language extensions built on top of the jurisdiction-neutral English vocabulary consist of relevant language-dependent vocabulary schemas, a simple XSL transformation template that translates language-dependent schemas to standard vocabulary, and optionally some XSL templates for specialized presentations in XHTML, or other formats of interest.

In addition MetaLex XML can be embedded in other XML languages (for e.g. databases, web applications, or 'agents') and XML from other languages (in a namespace; eg. XHTML layout) can be freely embedded in text nodes (sentences, titles, etc.). Embedded XHTML layout can for instance be used to embed a table in a sentence. People can also for instance define their own (more specific) XML elements for a specific jurisdiction and embed it in a MetaLex document.

Information about a document can also be expressed in RDF statements stored outside the document. LAMS, JSMS, or Dublin Core attributes for instance could be added to MetaLex as either (external) RDF statements or as an extension to MetaLex, and copied into the header of an XHTML document by a special purpose XSL transformation. Extensions to describe the content of regulations in a logic-based framework can also be specified in RDF. As an example, an improved version of the norm language used in CLIME [Winkels et al. 1999] will be made available. RDF can also be used to store attached information for advanced search techniques (e.g. [Kohonen 2001]). Because there is no consensus on how to represent such information - no standard - there are no restrictions on content models attached to MetaLex documents.

5.1. Support of W3C Standards and Proposals

MetaLex complies as far as practical with XML-related open standards and proposals of the W3C - the organization that sets standard for the web. W3C standard are usually supported by free software for verification and delivery of web services, and sometimes built into client web software (Netscape, Internet Explorer; e.g. XML, XHTML, and XSL). W3C standards and proposals are used in the following contexts:

- Specification of MetaLex elements in XML schema language and RDF schema;

  Support of namespaces;

  XSL for transformation between language-specific MetaLex extensions, XHTML for user display, and RDF;

  Support of static URL and URN names for regulations, persons, and public bodies;

  XML Linking and XPointer support for references between resources.

5.2. Translation to RDF

MetaLex standardizes structure and designation of identity in legislation. The standard XML ID attribute can be attached to elements that represent document structure and the structure can be translated with XSL stylesheets to RDF conforming to an RDF Schema. The RDF data model for MetaLex - restricted with DARPA Agent Markup Language (DAML)+Ontology Interchange Language (OIL) schema features (e.g. [Fensel et al. 2000]) - is considered normative for identity matching because it appears to be most suitable for that purpose. The LeXML initiative assumes that describing legal concepts in XML encodings for different jurisdictions in a single RDF dictionary [Muller, 2002] will make it easier to identify similarities and differences. The MetaLex schema can be integrated in such a dictionary as a 'generic backbone' that can be exploited by MetaLex-aware tools. Figure Figure 2 shows the relationship we propose between the XML Schema-based and RDF Schema-based encoding of the same document.

A well-known limitation of standard XML is the lack of standardization of *global* object identity of elements and the interpretation of the meaning of references between elements. The ID attribute and standards for namespaces, (X)HTML, XPath, XPointer, XLink, and RDF all offer competing or complementary pieces of solutions to make XML parsing trees represent arbitrary graphs linking distinct individuals. RDF makes this underlying graph

explicit and de-couples the identity of elements from the documents in which they are serialized (only positioning the element in a namespace - which may or may not correspond to a document). If a document element is encoded in RDF statements - triples of a *subject*, *predicate*, and *object* - it can be both subject and object of statements regardless of what document it is serialized in. This perspective is certainly more suitable for a world of 'organic' regulations that may never have been entirely published in their present form. A minor disadvantage is that path-based XPointer references are meaningless strings to an RDF store; Every target of an XPointer-based link in the XML Schema-based version of a regulation must carry an ID so that it can be resolved to a 'normal' URI by the stylesheet that translates it to RDF.

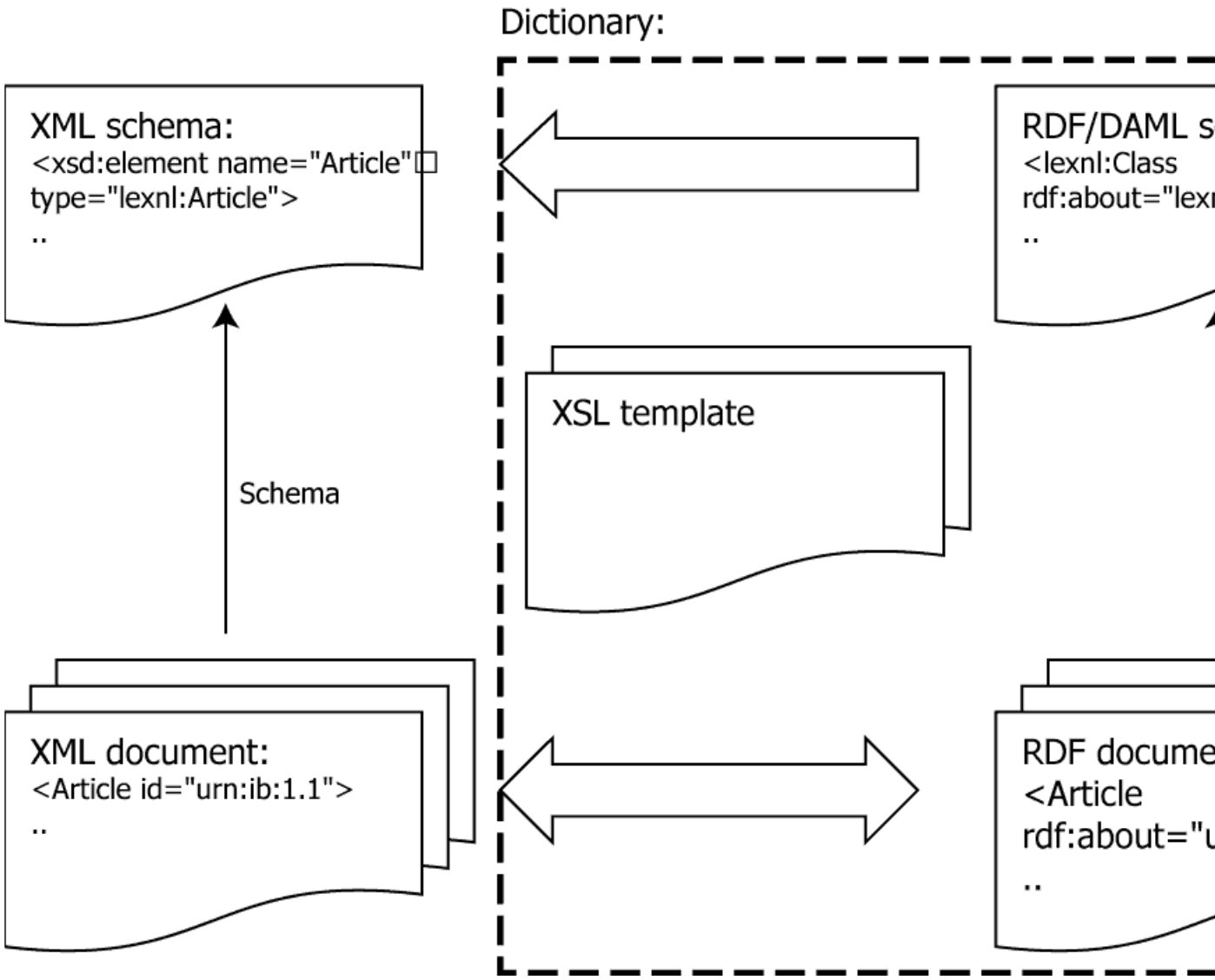

Figure 2.

Another notable difference between the MetaLex XML schemas and corresponding RDF schemas for documents is that RDF encoding requires explicit, indexed 'sequences' of e.g. articles, parts, sentences because RDF is order-independent; Any order of serialization of an RDF model into RDF/XML results in a different XML parsing tree. RDF can for instance represent the existence of an unspecified 'hole' between a first and third sentence in an article. Once the RDF version of a document contains holes, it cannot be written in normal XML Schema-based XML anymore. This notion of a hole in the document representing missing information is not the same as the notion of a hole in an index used for designation in the regulation itself; If article 1 is followed by article 3 that does not imply the 'existence' of an article 2 in a legal source during the time-interval represented by the serialized XML document. Because RDF allows this distinction, the RDF representation may be especially useful to store regulations while they are being edited.

6. Discussion

During the design of MetaLex we looked at the structure of a considerable amount of legislation from the Netherlands, and some representative pieces of legislation from Belgium, the US, and the United Nations. In addition, we consulted introductory literature on standard citation practices in the Netherlands, the UK, the US, and Hongkong. Actual markup of some of the legislation we looked at in MetaLex still revealed new and strange irregularities that we had not discovered before; It is clear that we still have to test more legislation from more jurisdictions. We hope that others will apply the schema to new legislative documents and report problems to the MetaLex discussion mailing list.

The MetaLex standard is intended to cover all formally structured *decisions* - based on attributed public powers - of public persons (meaning organizations with a 'public' legal personality) and bodies of those persons. The schema described in this paper and published on the website only covers regulations, and lacks special support for for instance amendment acts - although they can usually be understood as regulations - and mandate and delegation decisions. These other classes of decisions are relevant to explaining the status of regulations. International treaties and central labor agreements (between employer and employee representatives in the Netherlands), for instance, certainly have the appropriate structure of a regulation, but only become legally binding in a jurisdiction with a ratification decision. We are currently working on an even more general schema for capturing what notions remain for *every* type of legal decision.

We have created some XSL translations from the XML format of some legal publishers for internal use, and a plug-in for Microsoft Word from the POWER group [Engers et al. 2001] that discovers regulation structure in (Netherlands) law exports MetaLex XML. In addition, we are designing a freeware tool - in the E-POWER project - to manage, transform, and publish MetaLex documents. We are also building some standard ontologies for the legal domain, concentrating first on fiscal law for E-POWER and penal law for the eCourt project. RDF models of these ontologies will be connected to MetaLex XML as depicted in Figure Figure 2 above. At a later stage these extras will be made available at the MetaLex website.

MetaLex is intended to be an *open* standard. It was first launched in September 2002 at the DEXA conference ([Boer et al., 2002]). The MetaLex schema, documentation, and some examples are available for free at http://www.metalex.nl. The website can also be used to subscribe to the MetaLex mailing lists. We are in the process of forming a committee, consisting of representatives of public administrations, academics, publishers and other commercial enterprizes in the legal field. This standardization committee should maintain and extend MetaLex in the future. Furthermore it can certify certain extensions and tools that adhere to the standard.

Acknowledgements

The MetaLex schema has been designed in the context of the E-POWER project. The contributions of Tom van Engers of the Dutch Tax and Customs Administration and Frederik Willaert of Application Engineers have been invaluable and deserve special mention. The Netherlands LeXML committee has commented on an earlier version of the schema. E-POWER is partially funded by the EC as IST Project 2000-28125; partners are the Dutch Tax and Customs Administration, O&I Management Partners, LibRT, the University of Amsterdam (NL); Application Engineers, Fortis Bank Insurance (B); Mega International (F). eCOURT is partially funded by the EC as IST Project 2000-28199; partners are Project Automation, Ministry of Justice, CNR (I); Ministry of Justice (POL); Sema Group S.a.e. (SP); Intrasoft International (L); Université Paul Sabatier (F); University of Amsterdam (NL).

Bibliography

[Boer et al., 2002] Boer, A., Hoekstra, R., Winkels, R., Engers, T. van and Willaert, F. Proposal for a Dutch Legal XML Standard. In: *Proceedings of the E-Government Conference (at DEXA)*, Aix-en-Provence, 2002. Springer-Verlag.

[Engers et al. 2001] Engers, T. van, Gerrits, R., Boekenoogen, M., Glassée, E., Kordelaar, P.: POWER: Using UML/OCL for Modeling Legislation -an application report. In: *Proceedings of the 8th International Conference on Artificial Intelligence and Law (ICAIL 2001)*, pp. 157-167. ACM, New York, 2001.

[Fensel et al. 2000] Fensel, D., Horrocks, I., Harmelen, F. van, Decker, S., Erdmann, M. and Klein, M.: OIL in a nutshell. In: R. Dieng et al. (eds.) *Knowledge Acquisition, Modeling, and Management*, Proceedings of the European Knowledge Acquisition Conference. Lecture Notes in Artificial Intelligence, LNAI, Springer-Verlag, October 2000.

[Finke, 1997] Finke, Nicholas D., 'TEI Extensions for Legal Text', in *Proceedings of the Text Encoding Initiative Tenth Anniversary User Conference*, 1997.

[Kohonen 2001] Kohonen, T.: *Self-Organizing Maps.* Springer Series in Information Sciences, Vol. 30, 1995. Springer, Berlin. Third edition 2001.

[Moens 2001] M-F. Moens, 'Innovative techniques for legal text retrieval', *Artificial Intelligence and Law*, 9, 29-57, (2001).

[Muller, 2002] M. Muller, 'Legal RDF Dictionary', in *Proceedings of XML Europe 2002*. http://www.idealliance.org/papers/xmle02/dx_xmle02/papers/03-04-03/03-04-03.pdf

[Rissland et al. 1995] E. Rissland and T. Friedman, 'Detecting change in legal concepts', in *Proceedings of the Fifth International Conference on Artificial Intelligence and Law (ICAIL-95)*, pp. 127-136, New York (NY), (1995). ACM.

[Magnusson Sjöberg, 1998] Magnusson Sjöberg, C, *Critical Factors in Legal Document Management: A study of standardised markup languages*. Jure, Stockholm, 1998.

[Turtle 1995] H. Turtle, 'Text retrieval in the legal world', *Artificial Intelligence and Law*, 3, 5-54, (1995).

[Winkels et al. 1999] R.G.F. Winkels, D. Bosscher, A. Boer, and J.A. Breuker, 'Generating Exception Structures for Legal Information Serving', in Th.F. Gordon (ed) *Proceedings of the Seventh International Conference on Artificial Intelligence and Law (ICAIL-99)*, pp. 182-195, New York (NY), (1999). ACM.

Glossary

DAML	DARPA Agent Markup Language
DTCA	Dutch Tax and Customs Administration
JSMS	Justice Sector Metadata Scheme
LAMS	Legal and Advice sectors Metadata Scheme
OIL	Ontology Interchange Language
RDF	Resource Description Framework
TEI	Text Encoding Initiative
URI	universal resource identifier
W3C	World Wide Web Consortium
XLink	XML Linking Language
XSL	eXtensible Stylesheet Language

Biography

Radboud **Winkels**
Computer Science and Law, University of Amsterdam
Amsterdam
Netherlands
winkels@lri.jur.uva.nl

Radboud Winkels is an associate professor in Computer Science and Law at the University of Amsterdam (UoA). He has been involved in many national and international applied reserach and development projects involving Artificial Intelligence. Currently he is leading the UoA teams in the IST projects E-POWER and e-COURT. He is president of the Dutch foundation for legal knowledge and information systems (JURIX).

Alexander **Boer**
Computer Science and Law, University of Amsterdam
Amsterdam
Netherlands
aboer@lri.jur.uva.nl

Alexander Boer is senior researcher at the dept. of Computer Science and Law at the University of Amsterdam. He has worked on the CLIME project and is now involved in the IST E-POWER project.

Rinke **Hoekstra**
Computer Science and Law, University of Amsterdam
Amsterdam
Netherlands
rinke@lri.jur.uva.nl

Rinke Hoekstra is researcher at the dept. of Computer Science and Law at the University of Amsterdam. He has worked on the CLIME project and is now involved in the IST E-POWER project.