The MetaLex Document Server
Legal Documents as Versioned Linked Data

Rinke Hoekstra[1,2]

[1] Leibniz Center for Law, Faculty of Law, University of Amsterdam
hoekstra@uva.nl
[2] Computer Science Department, VU University Amsterdam
r.j.hoekstra@vu.nl
http://www.rinkehoekstra.nl

Abstract. This paper introduces the MetaLex Document Server (MDS), an ongoing project to improve access to legal sources (regulations, court rulings) by means of a generic legal XML syntax (CEN MetaLex) and Linked Data. The MDS defines a generic conversion mechanism from legacy legal XML syntaxes to CEN MetaLex, RDF and Pajek network files, and discloses content by means of HTTP-based content negotiation, a SPARQL endpoint and a basic search interface. MDS combines a transparent (versioned) and opaque (content-based) naming scheme for URIs of parts of legal texts, allowing for tracking of version information at the URI-level, as well as reverse engineering of versioned metadata from sources that provide only partial information, such as many web-based legal content services. The MDS hosts all 28k national regulations of the Netherlands available since May 2011, comprising some 100M triples.

Keywords: metalex, rdf, legal xml, law, linked open data, open government

1 Introduction

Where open government data is concerned, the rules and regulations a government imposes on its citizens are arguably close to the top of the list of every open data enthusiast. Law is the oldest form of open government information in existence. For it to be effective, the adage holds that "every citizen is expected to know the law" – 'knowing' in the sense of 'having access to'. Legislation and court rulings grow in importance. Policy makers are increasingly inclined to 'govern by regulation'[3], EU directives form an ever more complex legal framework that shapes national policies and regulations, giving citizens easier access to supra-national legislatures such as the European court. Businesses are subject to highly detailed regulations concerning e.g. financial reporting, safety and security. Gartner estimates that the cost of meeting regulatory compliance needs will

[3] A recent example in The Netherlands was the threat of the Minister of the Interior to draft legislation that would force municipalities to accept a budget cut and carry out tasks previously belonging to national government.

pose severe problems for smaller banks by 2013.[4] Compliance affects businesses and government agencies alike: how to ensure the minimally required alignment of internal business processes with (external) regulations?

Regulations are at the heart of modern society, they affect every aspect of our lives, from public safety, to education, health, environment, food, civil disputes, traffic, privacy and democracy itself. It is therefore not surprising that many national governments have been publishing legislation and court rulings on the web for quite some time now. The National Archive's Legislation.gov.uk was at the forefront of the linked open government data wave that hit shore in 2009.[5] It set the standard for what governments should do to provide 5-star access to legal documents.[6].

In the Netherlands, the 'wetten.nl'[7] portal was launched in 2003 with all legislation published since 2002. In the following years, earlier legislation, treaties and other types of regulations were made available through the portal as well. In several respects, the features of the wetten.nl portal are symptomatic for the way in which the Dutch government communicates information to its citizens in the Netherlands: it looks *fancy* and costs a *tonne*, but is *not flexible*. Although current versions of regulations are available in XML, they are stripped of essential information, such as the version date of the document. Wetten.nl presents regulations as books with hyperlinks; the position of an article within the running text of a regulation is the only context provided. Given the highly networked structure of legislation, this traditional restricted presentation is suboptimal: potential alternative ways of serialising one or more regulation texts (e.g. by topic) are discarded. This is not only a potential problem for businesses and citizens trying to understand the norms applying to their case, it is problematic for the civil servants and government organisations that have to apply these norms as well.

This paper describes our efforts to publish the contents of wetten.nl as *5-star open data*: i.e. to extract, aggregate, reconstruct and enrich the datasource underlying the wetten.nl portal using publicly accessible webservices, and publish it both as CEN MetaLex[8], as Linked Data, and in a format suitable for social network analysis. By design, this conversion is independent of the language and XML format in which regulations are published. This conversion is the first large scale effort to transform an existing legacy legislative XML format to MetaLex.

Section 2 describes the requirements and use cases for the information that will be published through the MetaLex Document Server (MDS). Section 3 describes the current situation in the Netherlands in more detail. Section 4 intro-

[4] See http://bit.ly/aND1Rj.

[5] See http://www.legislation.gov.uk and http://bit.ly/cV2MRu for a discussion of its features.

[6] See http://www.w3.org/DesignIssues/LinkedData.html.

[7] Literally 'legislation.nl', see http://wetten.overheid.nl.

[8] CEN MetaLex is published as CEN Workshop Agreement, CWA 15710, see ftp://ftp.cen.eu/CEN/Sectors/List/ICT/CWAs/CWA15710-2010-MetaLex2.pdf

duces MetaLex, and is followed by a description of our methodology in section 5. Section 6 describes the results, followed by a discussion.

2 Use Cases and Requirements

We identify four stakeholders when it concerns the interpretation of legal texts: *citizens*, *businesses*, *legal professionals* and *government bodies*. *Citizens* are expected to 'know the law'. Governments have a duty to make the law known to their citizens. Even though citizens may not be interested or able to understand legal language [6, 5], they must at least be offered the opportunity to know their rights and duties. *Businesses* have a vested interest in complying to regulations as the (financial) risks of not complying are high, and governments have the means to check for compliance through audits and obligatory reports. They therefore need to be kept up-to-date with respect to new or changed regulations they are subject to.

Legal *professionals* not only need to be kept up-to-date, but they frequently require access to non-current versions of regulations when dealing with cases that emerged prior to the latest change (retroactive enactment is quite seldom). Even for legal professionals, the texts of regulations are not self explanatory, and they consult a wide variety of additional sources to interpret the law. Examples are the official motivation of the legislator, case law, notes provided by other experts, journals, and reports of parliamentary hearings.

Government bodies enact, enforce, implement and execute regulations. Law is a large interconnected, and therefore interdependent network of norms. Understanding and guiding the effects of new proposed legislation is one of the primary concerns of the legislator. Currently, legislative drafting largely depends on the expertise of civil servants, their access to books and legal search engines. In the Netherlands and Switzerland, no specific editing environment is currently available: Laws are drafted by editing and sending regular Word documents around. Secondly, executive agencies have internal business processes that need to align with all potentially applicable versions of the law. Lastly, government organisations are increasingly required to share information amongst themselves. However, organisations form different and sometimes incompatible speech communities. The term 'income' means something different for determining social benefits as it does for taxation. Legislation (and in particular its structure) forms the ideal 'coat rack' for knowledge management and interchange between government bodies.[9]

2.1 Use Cases

Each of our stakeholders has benefited from the increased transparency offered by web-based search engines. However, their interests and needs go beyond simple search. Businesses are increasingly aware of the importance of streamlining

[9] With thanks to Hans Overbeek of ICTU for the metaphor.

their internal operations. The market of business process management suites is expected to grow to \$3.4 billion globally by 2014.[10]. This provides opportunities for more fine grained *regulatory compliance* management: business processes that are potentially affected by regulations can be identified by explicitly linking them to applicable norms. Legal professionals working at businesses and government bodies need to *annotate* parts of legal texts with interpretations and guidelines, and share them with their colleagues. *Provenance* information is essential for determining the motivation of a legislation: what parliamentary hearings and led to the current version of an article? Annotation and provenance are a key requirement in the current modernisation of the legislative drafting system of the Swiss Federal Chancellery [10].

Regulations are not *integrated*, different types of 'law' are issued by different government bodies. National government issues legislation, judges produce case law, and municipalities issue local regulations. Regulations are published on different websites, a situation that misrepresents the dependencies between them.

It is hard to *consistently interpret the meaning of concepts.* Regulations contain both 'hyperlink' style references to parts of text, as well as '*imports*' that import the meaning of a term from another regulation. Furthermore, the meaning of a term can be *scoped* to a particular part of the regulation, such as a chapter or article [12]. To complicate matters, regulations contain so-called 'deeming provisions' that, within a specified scope, assign the label of one concept to another concept. For instance, the provision "for the purposes of this chapter, a house boat is deemed to be a house" allows the legislative drafter to use the term 'house' to refer both to houses and house boats. Although legislative drafters are very careful to be specific about their choice of words, not all concepts used are properly introduced.

Not all parts of a regulation are equally *important.* That is, it is often the case that a small number of articles hold for the majority of cases regulated by a law, while the rest deal with more specific cases and exceptions. Furthermore, although related articles are often grouped within a chapter, this grouping does not cross the borders of a single law. Even though articles in distinct laws may be more closely related. The Dutch Immigration and Naturalisation Service (IND) has to deal with highly dynamic legislation. Knowing what parts of a law matter most to them, as well as the dependency between articles and their internal business processes (cf. regulatory compliance) is key in their ability to advice the ministry on the possibilities and difficulties in amending existing immigration laws.

The IND has a hard time dealing with all different *versions* of legislation, caused by dynamic legislation and lengthy procedures. Legislation follows an intricate versioning scheme [3, 4, 7, 11]: enactment, efficacy, publication and repeal dates all interact. This information may even be part of other regulations, e.g. in the Netherlands efficacy of regulations is typically described by Royal Decree. Legislation is frequently modified at the *sentence* level: e.g. a modifying law will

[10] Gartner Inc.: "Forecast: Enterprise Software Markets, Worldwide, 2009-2014, 4Q10 Update, December 2010

replace the second sentence of article X. Finally, references between legal texts can point to a specific version of a regulation, as well as to the 'current' version.

Key in these use cases is the ability to refer to parts of legal texts. It requires persistent *identifiers* for every element of a legal text. These identifiers should be dereferencable to the element they describe, or a description of the element's metadata. It is furthermore a feature if these identifiers are *transparent* and follow a prescribed *naming convention*. This allows third parties to construct valid identifiers without having to first query a name service.

To support *versioning* of legal texts, references, and metadata, requires identifiers that reflect its different versions. The various parts of a text should be versioned *independently*, allowing for transitory regimes. Furthermore, the versioning mechanism should distinguish between a regulation text as it exists at a particular time, and the regulation 'as such'. A likely solution is the adoption of the distinctions made by the IFLA FRBR [15]:[11] the *work* as a "distinguishable intellectual or artistic creation" (e.g. the constitution);the *expression* as the "intellectual or artistic form that a work takes each time it is realised" (e.g. "The Constitution of July 15th, 2008"); the *manifestation* as the "physical embodiment of an expression of a work" (e.g. a PDF version of "The Constitution of July 15th, 2008"); and the *item* as a "single exemplar of a manifestation" (e.g. the PDF version of "The Constitution of July 15th, 2008" residing on my USB stick).

Metadata and annotations should be traceable to the most detailed part of a text, as well as to its version, when needed. The same requirement holds for references between texts, allowing for fine-grained analysis of interdependencies between texts. Current regulation search portals are developed from the perspective of the issuing government body, and are jurisdiction specific. The document server should provide a publishing platform that is independent of language, region and jurisdiction.

3 Wetten.nl and the Basiswettenbestand

Wetten.nl is part of a larger Overheid.nl (government.nl) website that provides access to a wide range of government information, both legal (national regulations, local regulations, permits, and publications, official national publications and disciplinary rulings) and general information such as information about the structure of goverment, addresses of government bodies, and a link to the Dutch open data catalog.[12] Amongst these, the wetten.nl portal is one of the oldest.

Users can perform a full text search through the titles and text of all regulations of the Kingdom of the Netherlands. They can search for a specific article, as well as for the version of a text holding at a specified date. Wetten.nl also supports deeplinks, but is not very consistent about it. For instance, both:

[11] IFLA: International Federation of Library Associations and Institutions. FRBR: Functional Requirements for Bibliographic Records. See http://archive.ifla.org/VII/s13/frbr/frbr1.htm for the exact definitions.

[12] See http://data.overheid.nl, a CKAN installation currently containing 40 datasets.

http://wetten.overheid.nl/cgi-bin/deeplink/law1/bwbid=BWBR0005416/article=6/date=2005-01-14

and

http://wetten.overheid.nl/BWBR0005416/TitelII698946/HoofdstukII/Artikel16/geldigheidsdatum_14-01-2005

point to article 6 of the Municipal law, as it was valid on January 14th, 2005.[13] These deeplinks can be considered to be permanent URIs of work level (without date) or expression level (with date) identifiers. Unfortunately, they are not always predictable (cf. the '698946' in the second URI), nor stable, nor part of a government standard.

The string 'BWBR0005416' is the opaque *BWB identifier* (BWB-ID) of the regulation. The Basiswettenbestand (BWB) is the content management system for all Dutch regulations that underlies the Wetten.nl portal. An 'R' following 'BWB' indicates that the document is a regulation, a 'V' indicates a treaty ('verdrag'). The 7-digit number does not carry a specific meaning. The opaqueness of the BWB identifier is unfortunate, but hard to avoid, as the title of a regulation may change over time and cannot be used. An index of all BWB identifiers, with basic attributes such as official and abbreviated titles, enactment and publication dates, retroactivity, etc. is available as a zipped XML dump.[14] Alternatively, a SOAP service allows retrieval of the same information for individual regulations. Unfortunately, the date of the latest change to a regulation is not really the date of the latest modification, but of the latest update of the regulation in the CMS. The two dates often coincide, but not all civil servants work weekends.

The BWB uses its own XML format for storing regulations. BWB XML provides elements for structure as well as annotation elements for capturing version history. It does not separate structural elements (e.g. 'article' or 'chapter'), presentation-type elements (e.g. 'emphasis', 'paragraph') and content-type elements (e.g. 'law', 'treaty'). The text of regulations is contained within meaningless presentation-type elements ('al' for alinea) rather than as separate sentences. The schema does not allow for intermixing with any third-party elements or attributes, ruling out obvious extensions such as RDFa.[15] Finally, the REST web service for obtaining the BWB XML representation of regulations only provides the *latest* version of an entire regulation.[16] The XML document returned is stripped of all version history, and does not even contain the version date of the text itself.

The BWB-ID forms the basis of the *Juriconnect* standard for referring to parts of regulations.[17] The standard describes a procedure for constructing unique

[13] Note that 'geldigheidsdatum' is the validity date.

[14] See http://www.overheid.nl/help/wr/deeplinks.

[15] RDFa: RDF attributes for use in XHTML, see http://www.w3.org/TR/rdfa-syntax/. RDF is the Resource Description Framework, see http://www.w3.org/standards/techs/rdf.

[16] See http://bit.ly/kdTniY and e.g. http://bit.ly/mQTWwo for a BWB XML version of the Municipal law.

[17] Juriconnect is a consortium of government bodies, legal publishers and academia, see http://www.juriconnect.nl.

identifiers from the structure of BWB XML documents. BWB XML documents use these identifiers to specify citations between regulations. For instance, the Juriconnect identifier of article 16 of the Municipal law, valid on January 14, 2005 is:

```
1.0:c:BWBR0005416&artikel=16&g=2005-01-14
```

Juriconnect does not prescribe a method by which the identifier should be used inside the XML of regulations or referring text: BWB XML elements do not carry Juriconnect identifiers. Neither does the standard specify whether an expression-level reference without validity date points to the latest, or current version, nor does it specify what should be returned for a work-level reference. Furthermore, the standard does not describe a method for dereferencing an identifier to the actual text of (part of) a regulation.

The wetten.nl portal meets most, if not all requirements of the pre open-data day and age. However, more demanding use of the content underlying the portal is not straightforward. The content service is crippled by incomplete information (the version date of retrieved documents, version history), limited functionality (no time travel) and identifiers in a non-standard format. The following section introduces the CEN MetaLex format for representing the text of legal resources, after which section 5 describes our method for republishing the regulations of wetten.nl as MetaLex and Linked Data.

4 CEN MetaLex

CEN MetaLex[18] is a jurisdiction independent XML standard for representing, publishing and interchanging the structure of legal resources. It is the result of a 10 year standardisation project in which multiple European government organisations, publishers and academic partners participated. MetaLex was initially designed by [1] as a syntactic grounding for building elaborate knowledge-based services.[19] At the time, the BWB XML was a proprietary format, and the Dutch government was still in negotiation with legal publishers about freely distributing its self-created content on the web. CEN MetaLex combines the original MetaLex with (primarily) insights from the Italian Norme in Rete project,[20], the Akoma Ntoso legal XML standard of African parliaments[21], the Austrian government and LexDania.[22] Amongst others, adopting CEN MetaLex allows the use of generic legislative drafting tools, rather than only jurisdiction (and often vendor) specific solutions.

[18] See http://www.metalex.eu and

[19] See legacy.metalex.eu for more information.

[20] Norme in Rete: laws online portal for the Italian government. The portal itself is no longer available.

[21] See http://www.akomantoso.org/.

[22] LexDania is the XML format behind the Danish ministerial regulations portal, see http://www.ministerialtidende.dk/.

MetaLex elements are purely *structural*. Syntactic elements (structure) are strictly distinct from the meaning of elements by specifying for each element its name and its *content model* [16]. What this essentially does, is paving the way for a purely semantic description of the types of content of elements in an XML document.

The standard prescribes the *existence* of a *naming convention* for minting URI-based identifiers for all structural elements of a legal document [2]. Names should be guessable from identifying features (attributes, context) of elements, described in the metadata. Names must exist for each of the FRBR levels, and a standard GRDDL[23] transformation for producing an RDF graph of the identifying metadata. MetaLex explicitly encourages the use of *RDFa* attributes on its elements, and provides special metadata-elements for serialising additional RDF triples that cannot be expressed on structural elements themselves. MetaLex includes an *ontology*, which defines the different FRBR levels in the context of legislation, and an event model for legislative modifications.[24]

Elements defined by the MetaLex schema can be *refined* via XML Schema to the jurisdiction specific elements of legacy legal XML formats such as BWB XML, LexDania and CHLexML.[25]. These generic elements are: root as the root of every MetaLex document; hcontainer and container for titled and untitled parts of a text; block elements for textual content, and inline for elements that occur in running texts. The htitle block element is used to specify the title of a hcontainer, the cite inline element carries a reference to another element; milestone elements for fixed-position, but content-less inline elements such as page breaks; mcontainer and meta elements for listing additional RDFa metadata inside the body of hcontainer and mcontainer elements. MetaLex is agnostic to non-conflicting third-party XML elements and attributes in block and inline elements, such as HTML markup for rendering tables.

Although most of its predecessors were implemented at enterprise scale, the MetaLex language itself has never been applied to a realistically large corpus. Although the language holds the promise of flexible interchange of legal texts, government institutions are slow movers. This is part of the challenge; does MetaLex live up to its promise as generic schema for legal texts, and does its commitment to Semantic Web standards provide substantial added value to government goals? The following section describes the methodology and vocabularies used for transforming legacy XML to MetaLex, and producing metadata descriptions in RDF. Section 6 discusses the results.

[23] GRDDL: Gleaning Resource Descriptions from Dialiects of Languages, see http://www.w3.org/TR/grddl/.

[24] See http://www.metalex.eu. The legislation.co.uk portal has adopted the MetaLex event model for representing modifications, but uses the standard FRBR ontology for indicating levels.

[25] CHLexML was designed as an XML standard for the representation of Swiss legal texts, see http://www.svri.ch/CHLexML/CHLexML_Reference_1.0.pdf.

5 Conversion and Publication

The transformation of legacy XML to MetaLex and RDF is implemented in the MetaLex converter, an open source Python script available from GitHub.[26] Conversion occurs in four stages: *mapping* legacy elements to MetaLex elements, minting *identifiers* for newly created elements, *producing metadata* for these elements, and *serialising* to appropriate formats. In this section, we briefly discuss how each of these is implemented in the MetaLex converter.

For the transformation of BWB XML files, the converter is sequentially fed with all BWB XML files and identifiers listed in the BWB ID index. Version information, citation titles and other medatada is retrieved through via a custom build scraper of the information pages on the wetten.nl website.[27] The information pages provide more elaborate and reliable information about regulations than can be obtained through the web service, such as the entire version history and types of modification of each law.

5.1 Mapping Legacy Elements to CEN MetaLex

The MetaLex schema is designed to be independent of jurisdiction, which means that it should be possible to map each legacy XML element to a MetaLex element in an unambiguous fashion. For the BWB to MetaLex translation, element mappings were obtained semi-automatically from the BWB DTD. Elements allowed to contain #PCDATA are mapped onto block or inline elements, where inline elements only occur inside the definition of blocks. All hcontainer elements allow a title-element, while container elements are only allowed to contain the block elements identified earlier.

Based on a mapping table, the converter traverses the DOM[28] tree of the source document, and synchronously builds a DOM tree for the target document. There are three special cases for which the converter has to make additional repairs. Sometimes 'obvious' candidates for the MetaLex hcontainer element do not fit the MetaLex schema as the source schema allows them to contain text, e.g. the artikel element in BWB XML. Secondly, footnotes are typically present as block or container-type and occur inside other blocks. The converter moves these to the parent container element of the containing block. On some occasions, inline-type source elements appear directly underneath container-type elements. Their target elements are wrapped inside an extra block element to ensure MetaLex compliance.

Attributes on source elements are passed to the identifier and metadata generators. Target elements receive five standard attributes: name, with the value of the target element name (for MetaLex compliance), class, with the value of

[26] See http://github.com/RinkeHoekstra/metalex-converter

[27] See e.g. http://wetten.overheid.nl/BWBR0005416/geldigheidsdatum_14-01-2005/informatie

[28] DOM: XML Document Object Model, see http://www.w3.org/TR/REC-DOM-Level-1/

the source element name (for custom CSS rendering), xml:lang, a language tag (if specified on the source element, or one of its parents), id, with an item-level identifier, relative to the xml:base of the document, and about, with an expression-level identifier.

5.2 Minting Identifiers

For every element in the document we create transparent URL-like URIs for the *work*, *expression* and *manifestation* level, and two opaque URIs for the *expression* and *item* level in the FRBR specification.

We use a naming scheme that is based on the URIs used at legislation.gov.uk, with slight adaptations to allow for the Dutch situation.[29] Juriconnect references in the source BWB XML are automatically translated to this naming scheme:

```
{API-URL}/{document-identifier}
    [(/{hcontainer-class}/{index}]*[/{block-class}/{block-id}]*
     [/{authority}][/{extent}][/{lang}][/{version}])|
     (/{opaque-version-hash})]
    [/{repr}]
```

The *API URL* is the first part of all URIs, and the URL at which the URI resolver resides. Examples are http://legislation.gov.uk and http://doc.metalex.eu, for obvious reasons we use the latter. This part is followed by a *document identifier*, a work-level identifier of the entire legal text. Different countries may have different forms for this identifier, e.g. the Dutch portal uses opaque BWB identifiers, while the UK portal uses {type}/{year}/{number} as document identifier. These two components are followed either by a transparent reflection of the hierarchical structure of the XML document or an opaque hash of the contents of the element.

Hierarchical *work* URIs consist of a path from root node to current element. For hcontainer elements we use its class, i.e. the source elements' name, combined with its official index. [30] For block elements, we use its class combined with a generated index based on the position of the element amongst all children of its parent. The third part of the hierarchical URI consists of an optional indication of the *authority* (issuer) and *extent* (jurisdiction) of the text. Several European member states, such as the UK, have lower governments that can alter or implement specific parts of national regulations. For *expression*-level identification, the work URI is followed by an optional language tag, and the version identifier: the date at which this version became official. Manifestation URIs follow the same conventions as those of 'document URIs' in the legislation.gov.uk portal. Item level identifiers are required by the MetaLex standard, but cannot be generated in any meaningful way. We have therefore chosen to use randomly generated character strings as item identifiers, combined with an empty xml:base.

The *opaque version URI* is needed to distinguish different versions of a text. The current webservice does not provide access to all versions of regulations

[29] See http://www.legislation.gov.uk/developer/uris

[30] Note that a single combination of class and index already provides a locally unique identifier within the legislation, i.e. the relative identifier 'chapter/1/article/1' is identical to 'article/1'. This does not hold for elements below the article level [1].

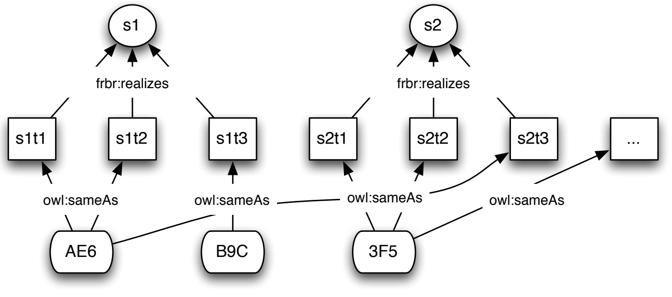

Fig. 1. The benefit of opaque URIs for versioning legal texts

(only to the latest), let alone at a level of granularity lower than entire regulations. We therefore need some way of constructing a version history by regularly checking for new versions, and comparing them to those we looked at before. By including a unique SHA1 hash of the textual content of an XML element in the opaque URI, and simultaneously maintaining a link between the opaque URI and the transparent identifier, different expressions of a work can be automatically distinguished through time. This is needed to work around issues with identifiers based on numbers: the insertion of a new element can change the position (and therefore the identifier) of other elements without a change in the content of the elements. How to find out how the new identifiers correspond to the old ones?

Consider two sentences with work-level URIs *s1* and *s2* (see Figure 1). At time *t1*, these sentences are respectively realised by the transparent expression-level URIs *s1t1* and *s2t1*, and by the opaque version URIs *AE6* and *3F5*. The two identifier-types hold for the same XML element, and are therefore considered to be semantically equivalent, hence the owl:sameAs relation. At *t2* the sentences undergo no changes, sentences *s1* and *s2* are realised respectively by *s1t2* and *s2t2*, and again by *AE6* and *3F5*. At *t3*, however, a new text is inserted as sentence before the old version of the first sentence: *s1* is now realised by *s1t3* and *B9C*. Consequently *s2* is now realised by *s2t3* of which the hash is the same as that of *s1t2*: *AE6*.

By this method, globally persistent URIs of every element in a legal text can be consistently generated for both current and future versions of the text. By simultaneously generating an opaque and a transparent expression level URI, identification of these text versions does not have to rely on numbering.

5.3 Producing Metadata

The MetaLex converter produces three types of metadata. First, *legacy* metadata from attributes in the source XML is directly translated to RDF triples with an expression URI as subject, the literal attribute value as object, and an RDF property with the source attribute's name as predicate. Second, metadata describing the *structural* and *identity* relations between elements. This includes typing

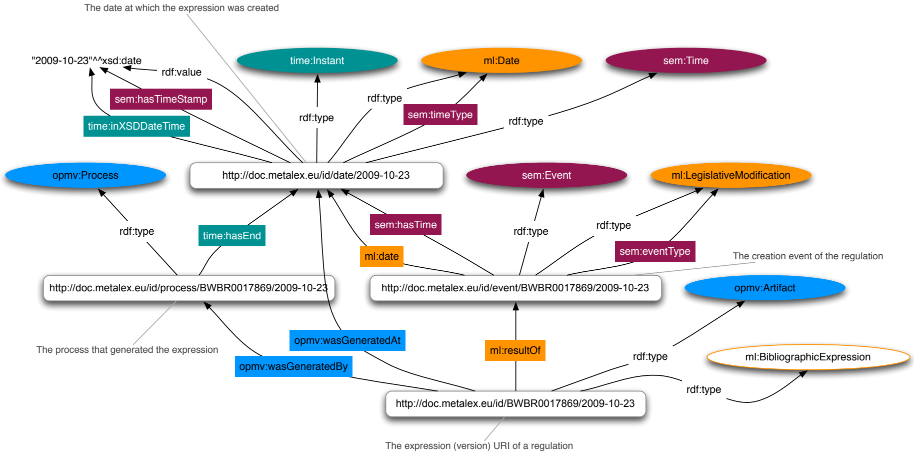
The date at which the expression was created
"2009-10-23"^^xsd:date
rdf:value
time:Instant
ml:Date
sem:Time
sem:hasTimeStamp
rdf:type
sem:timeType
rdf:type
time:inXSDDateTime
rdf:type
opmv:Process
http://doc.metalex.eu/id/date/2009-10-23
sem:Event
ml:LegislativeModification
sem:hasTime
rdf:type
rdf:type
time:hasEnd
ml:date
sem:eventType
The creation event of the regulation
http://doc.metalex.eu/id/process/BWBR0017869/2009-10-23
http://doc.metalex.eu/id/event/BWBR0017869/2009-10-23
opmv:Artifact
opmv:wasGeneratedAt
The process that generated the expression
ml:resultOf
rdf:type
ml:BibliographicExpression
opmv:wasGeneratedBy
rdf:type
http://doc.metalex.eu/id/BWBR0017869/2009-10-23
The expression (version) URI of a regulation

Fig. 2. Event model of the MDS

resources according to the MetaLex ontology, e.g. as ml:BibliographicExpression, creating ml:realizes relations between expressions and works, owl:sameAs relations between opaque and transparent expression URIs, ml:cites relations between citing and cited resources, and ml:partOf relations between expressions and parent elements in the XML. For each expression, we generate additional links to manifestations in RDF, XML and HTML, using rdfs:isDefinedBy, foaf:page and foaf:homePage properties, respectively. The official title, abbreviation and publication date of regulations are respectively represented using the dcterms:title, dcterms:alternative and dct:valid properties.

Events and Processes Event information plays a central role in determining what version of a regulation was valid when. Processes capture essential provenance information needed for the interpretation of regulations. Traditional methods for assigning validity intervals to parts of regulations use multiple attributes to indicate e.g. enactment, publication, and efficacy dates [4]. Making explicit which events and modifying processes contributed to an expression of a regulation provides for a more flexible and extensible model. Especially since multiple different timestamps for the same element can be grouped together via the opaque URIs described in the preceding section.

The MDS uses the MetaLex ontology for legislative modification events, the Simple Event Model (SEM) of [8] (used in the eCulture domain) and the W3C Time Ontology [9] for an abstract description of events and event types, and the Open Provenance Model Vocabulary (OPMV) of [14] for describing processes and provenance information.[31] As depicted in Figure 2, these vocabularies can be combined in a compatible fashion, allowing for maximal reuse of event and process descriptions by third parties that may not necessarily commit to the MetaLex ontology.

[31] See http://www.w3.org/TR/owl-time and http://openprovenance.org.

5.4 Serialization

The MetaLex converter supports three formats for serialising a legal text to a manifestation. First of all, it can produce the MetaLex format itself. The converter does not produce HTML since this can be easily obtained from the XML version. MetaLex can be viewed in a browser by linking a CSS stylesheet.[32] Secondly, generated RDF metadata can be serialised as inline RDFa attributes on meta tags. This is a very verbose format similar to N-triples (one element per triple), and it is often preferable to serialise the RDF as separate files using Turtle syntax,[33] unless the use case requires the representation of a legal text to be self contained. If required, the converter can automatically upload RDF to a triple store through either the Sesame API, or SPARQL updates.[34]

During conversion, citations are stored in a separate graph, linking citing resources at the level of articles (rather than at the level of inline elements carrying the reference) to cited resources. This graph can be exported to a '.net' network file, for further analysis in social network software tools such as Pajek and Gephi.[35] The MetaLex converter script optionally generates a network file containing citations of all regulations converted in the same batch.

5.5 Publication

The result of this procedure can be published through the MetaLex Document Server (MDS).[36] The MDS is essentially a Python wrapper for a SPARQL endpoint for RDF metadata, and a file-based store for MetaLex documents and network files. It follows the Cool URIs specification,[37] and implements HTTP-based redirects for work- and expression level URIs to corresponding manifestations based on the HTTP accept header. Requests for an HTML mime-type are redirected to a the Marbles[38] HTML rendering of a Symmetric Concise Bounded Description (CBD) of the RDF resource.[39]. Similarly, requests for RDF content return the SCBD itself; supported formats are RDF/XML and N3/Turtle. A request for XML will return the MetaLex of an XML snippet corresponding to the requested element. For work level identifiers MDS will only return RDF.

The MDS provides two convenience methods for retrieving manifestations of a regulation. Appending '/latest' to a work URI will redirect to the latest expression present in the triple store. Appending an arbitrary ISO date will return the last expression published before that date if no direct match is available. Lastly, the MDS offers a simple search interface for finding regulations based on the title and version date.

[32] See http://www.w3.org/Style/CSS/ and http://doc.metalex.eu/static/css/metalex.css

[33] See http://www.w3.org/TR/turtle/

[34] See http://openrdf.org and http://www.w3.org/TR/sparql11-update/.

[35] See http://pajek.imfm.si/doku.php and http://gephi.org respectively.

[36] See http://doc.metalex.eu for the server, and http://github.com/RinkeHoekstra/metalex-web-converter for the sources.

[37] See http://www.w3.org/TR/cooluris/.

[38] See http://marbles.sourceforge.net/.

[39] See http://www.w3.org/Submission/CBD/.

Table 1. Conversion performance for 300 randomly selected regulations.

	Number	%		Number	%
Substitutions[42]			**Corrections**		
container	22312	29 %	artikel	2525	72 %
hcontainer	3730	5 %	divisie	519	15 %
htitle	3730	5 %	colspec	289	8 %
block	34325	44 %	illustratie	54	2 %
inline	13527	17 %	*others*	99	3 %
Total	77624		*Total*	3486	
			Total no. of regulations	300	
			Revoked regulations	109	30 %
			Correction %		4 %

6 Conclusion and Results

We ran the MetaLex conversion script on all regulations available through the wetten.nl portal in May 2011, resulting in a total of 27.687 versions of regulations being converted, roughly 1 GB in size for BWB XML, and 2.27 GB as MetaLex.[40] The size increase is primarily due to the length and number of identifiers in MetaLex, which aren't present in BWB XML. The generated Turtle files comprise 9.9 GB, and contain 87.9 million triples. At this moment, the MDS runs on a 32GB Dell PowerEdge and a 4Store triple store.[41] New and modified regulations are published almost every other day, which means that the number of regulation versions accumulates with time: currently 28.752 versions and 100M triples (August 2011).

We evaluated the ability of MetaLex to map onto the BWB XML by running the converter on 300 randomly selected BWB identifiers; results are presented in Table 1. The artikel element accounts for 72% of all corrections, and corresponds to 68% of all htitle substitutions (5 % of total). This means that only a very small part of BWB XML does not directly fit onto the MetaLex schema. We have conducted a similar exercise on a single example of a CHLexML document and results were comparable; the main cause for incompatibility is the restriction in MetaLex that hcontainer elements are not allowed to contain block elements.

6.1 Meeting Requirements

Publishing identifiers and metadata of regulations in RDF meets the minimal requirements for facilitating *regulatory compliance* and *annotation.* Third parties can freely and transparently annotate regulations with specialised vocabularies and business rules. Our versioning scheme allows these annotations to be fine-grained and stable through time. For instance, annotating an opaque expression URI ensures that the annotation remains valid until the text of the expression changes, rather than when the official 'version' changes. Versioning and *time*

[40] The actual number of regulations available at a single time is typically a bit lower. The conversion was done in several batches, and several modified regulations were published in the meantime.

[41] See http://4store.org.

travel is possible through the combination of SEM and the MetaLex ontology on the one hand, and opaque and transparent expression URIs on the other. Adoption of the OPMV vocabulary for expressing *provenance*, allows the construction of elaborate provenance trails, potentially referring even to pre-publication processes in the legislative drafting workflow.

Together with the Dutch Finance Ministry we started a pilot, based on [12], to detect both the definitions and scope of concepts as well as implicitly introduced concepts (nouns and noun phrases) in the domain of inheritance tax. All concepts are linked to both to the Cornetto Wordnet thesaurus in RDF and the relevant elements in law.[43] Although the scope of concepts can be made explicit by using namespaces or suffixes, ensuring *concept consistency* by resolving the scope to a set of elements internal and external to the law is an open issue. Furthermore, the inheritance tax law alone contains 1255 concepts in 72k triples, which will put a further strain on our hardware if concept extraction is let loose on other regulations.

Although we have shown that ingestion of a large corpus of legacy XML is feasible, other regulatory datasets need to be investigated to ensure the genericness of the approach. In particular, the transformation of different types of regulations, such as municipal regulations and EU directives, will contribute to the *integration* of regulations about similar topics. Conceptual annotation of legal sources will certainly improve the integration of these sources across the borders of government organisations.

Social network analysis of reference structures in legal texts allows us to determine certain properties of articles, such as the number of incoming citations (in degree), and the role of an article in connecting other articles (betweenness centrality). We conducted a small experiment at the immigration service (IND) to determine whether these measures corresponded to their intuitions of which articles are most *important* in immigration law: the in degree proved to be an almost perfect match to the most important articles, while betweenness centrality corresponded to articles in regulations that translated legislation to guidelines and procedures for civil servants at IND. Network analysis tools such as Gephi also provide nice visualisations of these reference networks, where closely related articles are grouped together, indicating *themes* in legislation. Gephi can even simulate how citations change through time. Indeed, all this is not new technology, but until now it has been beyond the grasp of civil servants in agencies such as the IND.

The MetaLex Document Server is an important step in opening up Dutch regulations for advanced analysis and semantic annotation. We described a procedure for incrementally rebuilding metadata and version information not made available by publicly accessible regulatory content services. Although the MDS and conversion script has not yet been used for converting other types of regulations, it was designed to be generically applicable to a wide range of legal XML formats by adopting the CEN MetaLex standard. We have furthermore gathered evidence that MetaLex is indeed able to represent and augment legal

[43] See http://ckan.net/package/cornetto,

resources expressed in legacy XML. Perhaps most importantly, we have made a couple of people at the IND and the Dutch Ministry of Finance rather enthusiastic about the combination of legal information, network analysis, and Semantic Web technology.

References

1. Boer, A., Hoekstra, R., Winkels, R.: METALex: Legislation in XML. In: Bench-Capon, T., et al. (eds.) Jurix 2002: The Fifteenth Annual Conference. pp. 1–10. FAIA, IOS Press, (2002)
2. Boer, A.: MetaLex Naming Conventions and the Semantic Web. In: Governatori, G. (ed.) Jurix 2009: The Twenty-Second Annual Conference. IOS Press, (Dec 2009)
3. Boer, A., Winkels, R., van Engers, T., de Maat, E.: A Content Management System based on an Event-based Model of Version Management Information in Legislation. In: Gordon, T. (ed.) Jurix 2004: The Seventeenth Annual Conference. pp. 19–28. IOS Press, (2004)
4. Boer, A., Hoekstra, R., Winkels, R., van Engers, T., Breebaart, M.: Time and Versions in METALex XML. In: Proceeding of the Workshop on Legislative XML. Kobaek Strand (2004)
5. Dick, J.P.: Conceptual Retrieval and Case Law. In: Proceedings of the International Conference on Artificial Intelligence & Law (ICAIL). pp. 106–115 (1987)
6. Fillmore, C.J.: The Case for Case. In: Bach, E., Harms, R.T. (eds.) Universals in Linguistic Theory. Holt, Rinehart and Winston (1968)
7. Gangemi, A., Pisanelli, D., Steve, G.: A formal ontology framework to represent norm dynamics. In: Winkels, R., Hoekstra, R. (eds.) Proceedings of the Second International Workshop on Legal Ontologies (LEGONT) (2001)
8. van Hage, W., Malaisé, V., Segers, R., Hollink, L.: Design and Use of the Simple Event Model (SEM). Journal of Web Semantics, to appear (2011)
9. Hobbs, J., Pan, F.: An Ontology of Time for the Semantic Web. Transactions on Asian Language Processing (TALIP): Special issue on Temporal Information Processing 3(1), 66–85 (2004)
10. Hoekstra, R.: Modernisation of the KAV system - A second opinion study on technology and implementation. Tech. rep., University of Amsterdam (2011)
11. Klarman, S., Hoekstra, R., Bron, M.: Versions and Applicability of Concept Definitions in Legal Ontologies. In: Clark, K., Patel-Schneider, P.F. (eds.) Proceedings of OWLED 2008 DC. Washington, DC (metro) (Apr 2008)
12. de Maat, E., Winkels, R.: Automatic Classification of Sentences in Dutch Laws. In: Jurix 2008: The 21st Annual Conference. IOS Press (Dec 2008)
13. de Maat, E., Winkels, R., van Engers, T.: Automated Detection of Reference Structures in Law. In: van Engers, T.M. (ed.) Jurix 2006: The Nineteenth Annual Conference. IOS Press (Dec 2006)
14. Moreau, L., et al.: The Open Provenance Model core specification (v1.1). Future Generation Computer Systems, in press (2010)
15. Saur, K.: Functional requirements for bibliographic records. IFLA Section on Cataloguing 19 (1998)
16. Vitali, F., Iorio, A., Gubellini, D.: Design patterns for document substructures. In: Extreme Markup 2005 Conference. Montreal (2005)