Multi-layer markup and ontological structures in Akoma Ntoso

Gioele Barabucci, Luca Cervone, Monica Palmirani,
Silvio Peroni, and Fabio Vitali

Università degli studi di Bologna

Abstract. The XML documents that represent legal resources contain information and legal knowledge that belong to many distinct conceptual layers. This paper shows how the Akoma Ntoso standard keeps these layers well separated while providing ontological structures on top of them. Additionally, this paper illustrates how Akoma Ntoso allows multiple interpretations, provided by different agents, over the same set of texts and concepts and how current semantic technologies can use these interpretations to reason on the underlying legal texts.

1 Introduction

Akoma Ntoso is an open legal XML standard for parliamentary, legislative and judiciary documents. Promoted by the Kenya Unit of the United Nations Department for Economics and Social Affairs (UN/DESA) in 2004, it means "Linked Hearts": a symbol used by the Akan people of West Africa to represent understanding and agreement. Originally meant for African Countries, it is now promoted also in Latin America, Asia and various European countries.

Akoma Ntoso describes structures for legal documents using a vocabulary of common structures based on XML, references to legal documents across countries using a common naming convention based on URIs, and a systematic set of legal metadata values using an ontologically sound approach compatible with OWL [18] and GRDDL [7].

Akoma Ntoso aims at being extensible for the individual needs of any country, preserving the legal digital resources over time (even long spans of time, in decades and centuries even), guaranteeing legal principles, and favoring trust by means of authoritative versions, legal copies, etc.

Akoma Ntoso has been designed so that XML documents can be managed in any step of the legislative or judiciary life cycle (for instance, in the publishing phase) without any modification of the content published by the body empowered by law to endorse it. Additionally, long term preservation of Akoma Ntoso documents must be possible even without access to the extensive original documentation.

The information added by Akoma Ntoso (the markup) can be seen as distributed over multiple layers, each layer addressing a specific problem: the text layer provides a faithful representation of the original content of the legal text,

the structure layer provides a hierarchical organization of the parts present in the text layers, the metadata layer associates information from the underlying layers with ontological information so that semantic tools can apply inference rules (like simple rules of description logic) or perform advanced reasoning using logic frameworks like defeasible logic or argumentation [9].

This paper is organized as follows. Section 2 illustrates the layered architecture of Akoma Ntoso documents. Section 3 describes the lower layers of the Akoma Ntoso document architecture where the legal text is described and the hooks used by upper layers are added. Section 4 shows how authoritative and non-authoritative legal knowledge (metadata) can be codified in an Akoma Ntoso document. Finally, Section 5 illustrates how current semantic tools can use the generic Akoma Ntoso ontology system to reason over legal documents.

Note: throughout this paper we will use the terms *legal text* and *legal resource* for legal texts as endorsed by an official authority; we will use the terms *legal document* or *legal XML document* for their computer representation; we will use the terms *legal content* or *normative content* for pointing out the normative meaning of the text

2 The Akoma Ntoso document architecture

The role of Akoma Ntoso is to mark up legal and legislative texts so that the legal knowledge and the legal structure of the text can be *understood* by current and future computer applications. This means that legal texts form the basis on which Akoma Ntoso documents exist. Akoma Ntoso documents add information on top of the actual text (*decorate the text* in the markup lingo). The added information can itself be seen as composed of different stratified layers (text, structure, metadata, etc) [15]. Akoma Ntoso clearly separates the legal text from these different levels of information but still allows higher layers to reference the underlying layers, thus building knowledge on top of other knowledge, with the content of the legal text acting as the base knowledge. This architecture allows different actors with different field experiences (e.g. experts in legal drafting, experts of document classification, experts in legal-knowledge engineering, experts in normative references, etc.) to mark up independent parts of the document.

As with any technology that deals with legal resources, Akoma Ntoso has been designed to work on the original text without changing to it. Words, punctuation marks and other typographical symbols are left untouched by the mark up process that transforms a plain-text legal document into an Akoma Ntoso-compliant XML document.

Additionally, Akoma Ntoso maintains a strict separation between *data* and *metadata* and provides an unambiguous definition of them as well as an operational distinction in authoriality: as such, data is any information that has been created or at least approved by the relevant legal author (for example the whole of the text of an act), while metadata is any information that was not present in the original version of the document as it was approved by the relevant legal author but was added editorially in a later moment of the production process

(e.g., the issue number of the official gazette or, even, the page numbers in the printed version of the same act).

The distinction between data and metadata is not only a theoretical distinction, since the actual layers of markup in Akoma Ntoso, *text*, *structure* and *metadata*, are based on it.

Textual markup identifies, within the content of the legal documents, fragments that have a precise legal or referential meaning, e.g., concepts such as "this piece of text is a date", "this piece of text is a legal reference" or "this piece of text contains the name of a party of the trial". Structural markup identifies and organize the parts of the content that divide it into containers, and especially hierarchical containers: "this piece of text is an article" or "this piece of text is the title of an act", "this piece of text is the background section of a judgment", *etc.* Metadata markup adds knowledge generated by an interpretation of the legal text performed by an human or mechanical agent: "the phrase *the pre-existing Acts* refers to *Act 32 of 1989* and *Act 2 of 1990*", or "the person cited in the minutes as *Mr. Gidisu* is really *Mr. Joe Kwashie Gisidu, the only member of the current Ghanaian parliament with that name and elected to a seat in it since 2000*".

Moreover, when the typographical styles are used to represent the semantic role of of piece of text, this role is captured in the metadata section so that the specific meaning of that typographical style is not lost.

There are different positions in the literature on which category is the most appropriate for structural data: textual or metadata. In our vision it belongs to a third, separate category. Actually the structure represents the wish of the author to organize the legal text in a certain order and therefore it is authoritative matter. If an annex is in the third level inside a hierarchical nested list of exhibits of an act (e.g. Table 1 of the Annex A.1) it embeds a legal message that belongs to the theory of law and in particular, a message on the hierarchy of the legal sources. What is finally voted on by a parliament in the assembly is a specific organization of the text that incorporates, through the text structure, a precise legal message (e.g. the article 50 is in this position because there is a linguistic and legal message to deploy).

The analysis of the textual and structural information is quite straightforward and its results are rarely disputed. On the contrary, the analysis of more advanced concepts found in the legal text requires some experience and it is easy for different sources to disagree on the generated interpretation. For this reason Akoma Ntoso documents have exactly one textual and one structural layer in each documents while interpretation of the advanced concepts are store as metadata, and Akoma Ntoso allows in the same document multiple metadata layers, each providing an interpretation by a different source.

Finally, each interpretation added by a specific actor can be linked to a particular ontology of legal concepts (e.g. date of enter into force as modeled in LKIF-core ontology or High Court of South Africa as modeled in the ontology about the judiciary system in Africa) for connecting to the ABox assertions, typically described in the XML document, from the general TBox proprieties, axioms and relationships defined into the core or domain ontology.

3 The authorial layers

Any Akoma Ntoso document is based on a legal resource that has been endorsed by an authority empowered by law: an act approved by a parliament, a decree issued by a ministry, a judgment entered by a court. Fidelity to the approved text is, thus, of primary importance; the data layers of Akoma Ntoso have been designed so that it is possible to markup a received legal text while preserving all the information contained in it and changing its content in no way.

The documents Akoma Ntoso deals with are legal resources whose significance is given by the fact that they have the power to influence citizens' life. Legal texts must, thus, be handled with extreme care and all the measures should be taken to make sure that the technological tools employed to manipulate the texts do not change or interfere with their intended meaning.

In Akoma Ntoso, legal documents are created by enclosing parts of the legal text in XML tags. No pieces of the legal text are discarded, even those that could be generated by an application (e.g. the article numbers in an act). The resulting documents are thus augmented versions of the authentic text; the approved text can be retrieved by simply removing all the XML tags.

The Akoma Ntoso markup process strives to preserve the legal validity of the text as endorsed by the official authority, without adding any additional content to the text. Obviously, the mere act of marking up a sentence involves an act of interpretation or annotation and thus cannot be considered perfectly neutral. However, the kind of markup done at the Akoma Ntoso data layers is almost objective (e.g, some automatic parsers have been developed [14]) and is rarely subject to disputes. For this reason Akoma Ntoso documents are designed to contain only one interpretation of the text and structure layers.

3.1 Text

The text layer, the first of the data layers, is the XML representation of the legal text. It is the base layer on top of which all the other layers are developed. As discussed in the previous sections, the text layer only adds XML tags to the existing text.

```
<speech id="sub2-spe02" >
    <p>Mr. Opare-Hammond:
    Mr. Speaker, page 4 has a repetition of the numbers
    25 to 47. I do not know whether it was a typographical
    error or from the printing. It needs to be corrected.</p>
</speech>
```

Fig. 1. An example of markup at the *text* layer in a debate of Ghana

The loose structure adopted by Akoma Ntoso, technically called *mixed content*, contrasts with the more rigid approach used by the first generation of XML standards for legal documents (e.g. EnAct or Formex [5]) which favored an organized structure where each legal text was treated like a database record. While

database-oriented documents are easier to process in a computer environment, mixed content documents are a better choice when working with already-existing texts, a situation that happens with real-life legal resources and that will keep on happening in the foreseeable future.

3.2 Structure

The structure layer, the second of the data layers, gives a role to the blocks and fragments of the text: anonymous blocks thus become articles or clauses or simple blocks. These associations organize the text in a (often hierarchical) structure that is used to organize the content, provide a reasonable interpretation and even a base for the addresses used in the interpretation of the legal text in the metadata layer, both of the current document and of other documents. Akoma Ntoso defines a set of common section names (for acts, they include part, chapter, tome, book, paragraph, article, clause) but does not impose a rigid structure on the way they can be combined so that they can be used differently in different law systems.

```
<subsection id="sec4-sub2">
    <num>(2)</num>
    <content>
    <p>The appointment of an honorary game warden-</p>
    <list id="sec4-sub2-lst1">
        <item id="sec4-sub2-itma">
            <num>(a)</num>
            <p>shall be notified in the Gazette;</p>
        </item>
        <item id="sec4-sub2-itmb">
            <num>(b)</num>
            <p>shall be effective unless sooner revoked by the
            Director, for a period of three years; and</p>
        </item>
        [...]
```

Fig. 2. An example of markup at the *structure* layer in an act

4 The editorial layer

The Akoma Ntoso metadata layer is a collection of pieces of legal knowledge that can be added onto a legal text as the personal interpretation of the written text, for example the analysis of the reasoning being performed by the judge while writing a judgment or the explicit consequences of the text of an amendment over an act. These pieces of legal knowledge by their nature are often subjective and dependent on one's interpretation. Instead of forcing a single interpretation for each legal document, Akoma Ntoso allows multiple, and even contrasting,

interpretations to be expressed in the same document, and associates a different actor to each of them. These interpretations of the underlying text form the groundwork upon which semantic technologies can make inferences (as discussed in Section 5).

4.1 Different kinds of information

The metadata layer allows agents to provide different kinds of information over the legal text. The following are examples of the information that can be added to Akoma Ntoso.

Reference disambiguation the `references` section links pieces of text to ontological entities. The usefulness of this information is twofold. First, conflicts between ambiguous phrases are resolved: for instance, in a sentence of a speech the text "Speaker" may be related to the TLCRole `/ontology/roles/gha/speaker` while in another sentence the same text "Speaker" may refer to a specific TLCPerson that is in charge as speaker in the specific time when the debate was held: `/ontology/persons/gha/parliament/JohnSmith`. Additionally, different

```
<references>
    <TLCRole id="speaker" showAs="Speaker"
        href="/ontology/roles/gha/speaker"/>
    <TLCPerson id="smith" showAs="Mr. Smith"
        href="/ontology/persons/gha/parliament/JohnSmith"/>
</references>
[...]
<speech id="sub2-spe01" by="smith" as="speaker">
    <from>Mr. Speaker</from>
    <p>Order! Order! Hon. Members, we shall take item 2 -- Correction
        of Votes and Proceedings. Page 1... page 4</p>
</speech>
```

Fig. 3. Disambiguation of different meanings of the word "Speaker"

spellings found in the text are consolidated in a single entity; in a court judgment, phrases like "Ms. Poliey", "Judge Poliey" and "Her Honour" can all be linked to the same TLCPerson `/ontology/Person/Poliey.1954`.

Legal analysis the `analysis` section provides information about many legal aspects that can be inferred by a legal expert when interpreting the text. An example is the interpretation of the effects of an amendment in an amendment act.

```
<textualMod type="substitution" id="am5">
    <source href="#sec4-sub1-itma"/>
    <destination href="/ke/act/1997-08-22/3/main#sec34-sub2-itma"/>
    <old href="#mod10-qtd1"/>
    <new href="#mod10-qtd2"/>
</textualMod>
```

Fig. 4. An example of amendment analysis

Another example is the identification of the role played by citations of precedents in the judgment argumentation of a judge (e.g. the application of a rule of law of a precedent, overriding a previous rule, etc.)

```
<judicial>
    <result type="deny"/>
    <supports id="jdc01">
        <source href="#ref01"/>
        <destination href="/gb/judgement/1829/QB273/eng@/main.xml"/>
    </supports>
</judicial>
```

Fig. 5. An example of judicial analysis on the outcome of a trial

Work identification the `identification` section classifies the document using the FRBR [11] conceptual model. This classification is used to inform the semantic tools that the document is the manifestation (in the FRBR sense) of a certain abstract work, so that they can distinguish between different versions of the same work. A more detailed account of FRBR usage in Akoma Ntoso can be found in [13] .

In addition to the shown kinds of metadata, there are other types of metadata currently defined (e.g. `lifecycle` and `workflow` for tracking the events affecting the document) and other are being added as Akoma Ntoso extends to its reach to more and more types of analysis of the legal text.

4.2 Multiple interpretations

All the information gathered in the metadata layer is derived from the legal text (using the data layers) though subjective reasoning. Many different interpretation can arise over the same legal text from different legal experts. Take, for instance, the following sentence: "*the subsection 3 of the section 42 states a modification of the section 44 of the same act*". Two different actors SO (someone) and SB (somebody) may disagree on the interpretation of that sentence: SO sees this modification as an authentic interpretation, SB sees it as a derogation. From a the legal point of view, the two types of modification produce different effects: the authentic interpretation is applied *ex-tunc* (since the beginning), while the exception is a derogation of a norm under some conditional. Akoma Ntoso allows both interpretations to coexists in the same document, even if they are in contrast.

The Akoma Ntoso XML representation of these two different interpretations would be as show in Figure 6 on page 8

```
<analysis source="#bungeni">
    <activeModifications>
        <meaningMod type="authenticInterpretation" id="am1" refersTo="so">
            <source href="#sec42-sub3"/>
            <destination href="#sec44"/>
        </meaningMod>
    </activeModifications>
</analysis>
<analysis source="#cirsfid">
    <activeModifications>
        <scopeMod type="exceptionOfScope" id="am2" refersTo="sb">
            <source href="#sec42-sub3"/>
            <destination href="#sec44"/>
        </scopeMod>
    </activeModifications>
</analysis>
```

Fig. 6. XML representation of two different interpretations of an amendment

5 Semantic technologies and reasoning on Akoma Ntoso documents

Currently, there are interesting developments in the area of legal knowledge representation and manipulation. Akoma Ntoso documents, with their rich metadata layer, can serve as the basis upon which various tools can work on. For example representations expressed at the metadata layer can be used to generate a legal ontology to be used by legal rule modeling technologies like RuleML [8] or the more specialized LKIF [10].

Akoma Ntoso documents are not tied to a particular semantic technology. The current format is very loose and permits the conversion of information into more specific formats (like RDF [6] or OWL [18]). This strategy warrants that semantic technologies of the future decades will be able to convert Akoma Ntoso documents into their own format without going through what, by then, will be seen as ancient formats.

5.1 Everything is semantically anonymous

Akoma Ntoso implicitly defines an ontological structure for representing metadata that is grounded by what we call *Top Level Classes* (TLC). Here we use the word *implicitly* because, on purpose, there is not no implemented, exhaustive and shared ontology that defines those classes and the relation among them: what exists is a sort of guideline that allows user (especially producers) of Akoma Ntoso documents to develop their own ontology according to their particular needs.

TLCs, even if represented by a clear label and a particular URI, have neither formal (logically defined) nor informal (written in natural language) semantics. The meaning beyond the text of a label of a TLC X does not give implicitly a meaning to X: it is just a label. Technically speaking, we do not define classes of a particular ontology but just URIs and labels that can be used to express particular classes defined in a separate ontology.

This assumption is needed to allow a great degree of flexibility in what can be expressed in the metadata layer of Akoma Ntoso documents, in order to adapt any legal document to any ontological representation of concepts. It is the duty of a third party, possibly the document creator, possibly its users, to associate a clear and formal semantics to each class using a specific formalism (e.g. OWL). This semantical anonymity is an important feature that allows Akoma Ntoso to maintain documents understandable and consumable independently from the passing of time: future toolmakers ("*The 'future toolmaker' is 10 years old now.*" [2]) will have clues about the intended meaning of a marker even in the unfortunate case the formal ontology is no longer available.

Akoma Ntoso makes ten different and disjoint TLCs available to document creators in order to identify individual entities present in the document: Person (`/ontology/Person`), Organization (`/ontology/Organization`), Concept (`/ontology/Concept`), Object (`/ontology/Object`), Event (`/ontology/Event`), Place (`/ontology/Place`), Process (`/ontology/Process`), Role (`/ontology/Role`), Term (`/ontology/Term`), Reference (`/ontology/Reference`).

Using these TLCs and the canonical FRBR classes *Work*, *Expression*, *Manifestation* and *Item*, we can make complex assertions on Akoma Ntoso documents. It must be underlined that Akoma Ntoso does not aim at describing neither objective facts nor personal opinions about such facts according to the author of the document: rather it allows to express an interpretation that is due, in a precise moment, to a particular actor working on the statements that can be found in the published legal text.

A fundamental step towards being able to reason over an Akoma Ntoso document is to have a mechanism for describing items (actors, legal documents, properties, concepts, etc) that are involved in the assertions (making assertions or being the subject of an assertion). Additionally, we need to describe how each of them is related to some TLC. Only then we will be able to assert facts about an Akoma Ntoso document, implicitly producing a data model for its semantic data.

5.2 Ontology URI naming conversion

All the items in an Akoma Ntoso document can belong to a particular TLC simply specifying an URI that must follow a particular naming convention [1]. The following example shows a list of URI, all pointing to the same entity representing a judge.

```
ontology/Person/lewanika
ontology/Person/judges/lewanika
ontology/Person/za/judges/lewanika
```

Fig. 7. Different URIs for the same entity

Even if a human can interpret them in some ways depending on the particular interpretation used, from an ontological point of view these three URIs say the same thing: the instance identified by them belongs to the top level class

Person. Moreover: they identify exactly the same resource in the Akoma Ntoso ontology (regardless of the ontology).

The Akoma Ntoso naming convention contains few precise rules:

- the last fragment of the URI (`lewanika` in this above example) is the identifier for the instance we are talking about;
- the first two fragments of the URI (`/ontology/<TLC name>`) specifies the TLC the instance belong to;
- the middle URI fragments (`za/judges/`), when they are present, provide evocative information for the human reader and for the systems that cannot use the underlying ontology used by the document. They have in fact a clear interpretation from a human perspective, e.g., that Lewanika is a South African person holding the role of judge. Since Akoma Ntoso does not force any given set of properties on the top level classes, we must assume that the number, order and correct values of such fragments are not given by the language but by the author of the document only.

Considering the implicit semantics beyond URIs, we can query a particular Akoma Ntoso manifestation using XPath [4], without knowing the particular ontological structure that a particular user has developed. Let us take into consideration the following excerpt:

```
<akomaNtoso xmlns="http://www.akomantoso.org/1.0">
    <references source="#cirsfid">
        <TLCOrganization id="parliament"
            showAs="Parliament of Kenya"
            href="/ontology/Organization/ke/parliament" />
        <TLCOrganization id="cirfid" showAs="CIRSFID"
            href="/ontology/Organization/cirfid" />
        <TLCPerson id="fv" showAs="Fabio Vitali"
            href="/ontology/Person/fv" />
        <TLCRole id="author" showAs="Author"
            href="/ontology/Roles/author" />
        <TLCRole id="jurist" showAs="Jurist"
            href="/ontology/Roles/jurist" />
        <TLCRole id="editor" showAs="Editor"
            href="/ontology/Roles/editor" />
    </references>
    [...]
```

Fig. 8. `references` section for a Kenyan act

Through XPath, we can easily query a lot of semantic data there exist in the document. Let us show some example queries for the above excerpt:

- looking for all the roles,
  `//references/element()[matches(@href,'/Roles/')]`;
- looking for all the Kenyan organizations,
  `//references/element()[matches(@href,'/Organization/.*/?ke/')]`.

This shows that, through the naming convention used by Akoma Ntoso, it is possible to make complex and sophisticated semantic queries on the documents even without knowing the underlying ontology model.

5.3 Transformation into a proper semantic data model format

In order to be able to carry out even more queries on Akoma Ntoso documents, it is necessary to transform the documents using a complete ontological model, which means binding the abstract Akoma Ntoso classes and instances to more concrete representations that the current semantic tools can work with. We do not want to specify a particular format to use: we want to allow users to choose their favorite format and tools, a choice that the users will take on the base of current standards, their knowledge and other technical constraints.

An example of concrete data model format that fits the current technology scenario is RDF [6]. If we wanted to convert an XML document into a set of RDF statements, we could use a GRDDL transformation. GRDDL (Gleaning Resource Descriptions from Dialects of Languages) [7] is a way to glean assertions from XML documents. It is a W3C Recommendation that explains how to extract semantic data from XML documents using a combination of one or more XSLT stylesheets [12], in order to obtain a new document containing those data expressed by RDF statements. Note that this particular mechanism it is also suggested in the current CEN Metalex proposal [3]:

> If metadata is not available as RDFa[1], it must be systematically translatable from the custom format to RDF. The translation from a proprietary metadata format to RDF must be publicly available following the Gleaning Resource Descriptions from Dialects of Languages (GRDDL) specification.

In order to use GRDDL, we only need to specify in the Akoma Ntoso document the GRDDL namespace and the reference(s) to the XSLT file(s) we use to perform the extraction. These declarations are added as attributes of the document's root element as shown in the following piece of code:

```
<akomaNtoso xmlns="http://www.akomantoso.org/1.0"
            xmlns:grddl="http://www.w3.org/2003/g/data-view#"
            grddl:transformation="xslt/fromAkomaNtosoToRDF.xsl">
[...]
</akomaNtoso>
```

Fig. 9. GRDDL attributes for an Akoma Ntoso document

Adding those declarations to an Akoma Ntoso document, we inform the document readers (both humans and computer tools) that it is possible to glean all the assertions embedded in the document itself using the XSLT stylesheet specified by the attribute `transformation` to transform the document into an RDF document.

After a first analysis, and even if they are defined in a well-known and appreciated standard, we could conclude that RDF statements are not enough to fit the needs of Akoma Ntoso assertions: while RDF define assertions as subject-predicate-object triples of resources, Akoma Ntoso uses an octuple approach for

[1] RDFa is technique used to embed RDF statements in XHTML documents

asserting facts. In fact, the general schema employed by Akoma Ntoso documents to express assertions is:

> the **author of a manifestation** asserts on the **manifestation date** that **the author of the corresponding expression** asserts on the **expression date** in a particular **context** that **subject** does **predicate** on **object**.

Let us show a concrete example. Considering the following excerpts from an Akoma Ntoso document as additional data of the code introduced in 5.2:

```
[...]
<analysis source="#cirfid">
    <textualMod type="substitution" id="am5">
        <source href="#art4-cla1-itma" />
        <destination
            href="/ke/act/1997-08-22/3/eng/main#art34-cla2-itma" />
        <old href="#mod10-qtd1" />
        <new href="#mod10-qtd2" />
    </textualMod>
[...]
<FRBRExpression>
    <FRBRthis value="/ke/act/2003-12-10/8/eng@" />
    <FRBRuri value="/ke/act/2003-12-10/8/eng@" />
    <FRBRdate date="2003-23-10" name="Generation" />
    <FRBRauthor href="#parliament" as="#author" />
</FRBRExpression>
<FRBRManifestation>
    <FRBRthis value="/ke/act/2003-12-10/8/eng.akn" />
    <FRBRuri value="/ke/act/2003-12-10/8/eng.akn" />
    <FRBRdate date="2007-07-27" name="Generation" />
    <FRBRauthor href="#cirfid" as="#editor" />
    <FRBRauthor href="#fv" as="#editor" />
</FRBRManifestation>
[...]
```

Fig. 10. Analysis and FRBR records for an Akoma Ntoso document

One of the assertion contained in the above extract is "*CIRSFID and Fabio Vitali assert on 27th July, 2007 that the Parliament of Kenya asserts on 10th December, 2003 in the corresponding expression that the fifth text modification is of type substitution*". As we have just said in 5.1, this means to use the octuple *(marker, marking_time, expression_author, content_time, context, subject, predicate, object)*, that in this instance is *(CIRSFID and Fabio Vitali, 2007-07-27, Parliament of Kenya, 2003-12-10, expression, fifth text modification, is of type, substitution)*.

Contrarily to what we said about RDF expression power to define Akoma Ntoso metadata, we can use a sort of trick to express these octuples through RDF statements by reification. Using the reification mechanism, we can associate a particular identifier to any RDF statement and then we can give it the role of subject or object in other statements. In Figure 11 on page 13 we show how the octuple above introduced can be represented through reified RDF statements (in the figure the oval shapes represent RDF resources, while the values

between quotation marks are strings and all the resources expressed by relative URI refer to the manifestation we are taking into consideration).

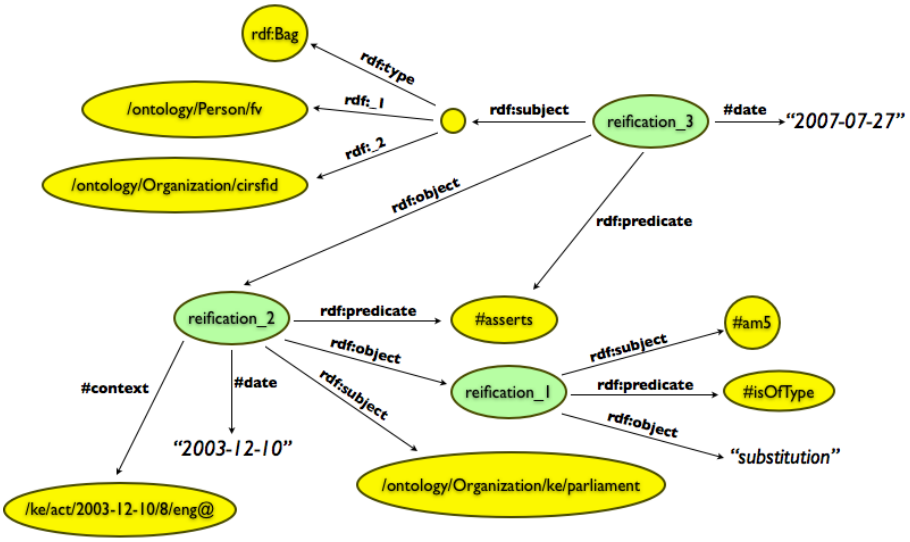

Fig. 11. A possible RDF representation of the octuple *(CIRSFID and Fabio Vitali, 2007-07-27, Parliament of Kenya, 2003-12-10, expression, fifth text modification, is of type, substitution)*

It is clear that there is not a unique way to express these reifications: it depends on how the XSLT stylesheet specified through GRDDL extract those data.

5.4 No default formal ontology

To have an RDF model of the semantic data on an Akoma Ntoso document could be useful, for instance, to share these data among different organizations using a common format. Moreover, one of the most appreciable advantages in using Semantic Web technologies, such as RDF and OWL, is the possibility to infer new data in an automatic way, for example processing the original data through reasoners.

The choice to maintain Akoma Ntoso free from any semantic structure constraint was born out of the desire to guarantee that any user could independently decide which particular ontological model they want to use and which data in an Akoma Ntoso document they have interest in. Independence from a particular technology is a key point for documents that are supposed to last, without modifications, for decades.

In any case, is it easy to convert an Akoma Ntoso document in one of the formats currently available. For example, if we consider the FRBR class implicitly defined within the Akoma Ntoso schema, we could generate other assertions that link FRBR instances gleaned from a document to the OWL-implemented FRBR

ontology, simply using an additional XSLT stylesheet specified through GRDDL. Let us take as example the following extract from an Akoma Ntoso document:

```
<FRBRExpression>
    <FRBRthis value="/ke/act/2003-12-10/8/eng@"/>
    <FRBRuri value="/ke/act/2003-12-10/8/eng@"/>
    <FRBRdate date="2003-12-10" name="Generation"/>
    <FRBRauthor href="#parliament" as="#author"/>
</FRBRExpression>
<FRBRManifestation>
    <FRBRuri value="/ke/act/2003-12-10/8/eng.akn"/>
[...]
<TLCOrganization id="parliament" showAs="Parliament of Kenya"
    href="/ontology/Organization/ke/parliament"/>
<TLCRole id="author" showAs="Author"
    href="/ontology/Roles/author"/>
```

Fig. 12. Statements in an Akoma Ntoso document

Extracting the RDF statements concerning FRBR metadata from this code using GRDDL means to obtain a document such as the following[2]

```
@prefix za: </ke/act/2003-12-10/8/> .
@prefix an: </ontology/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

za:eng@
    a an:FRBRExpression ;
    an:hasDate "2003-12-10"^^xsd:date ;
    an:hasAuthor an:Organization/ke/parliament .

an:Organization/ke/parliament a an:Roles/author .
```

Fig. 13. RDF extracted from the statements in Figure 12 on page 14

As you can see, these are just plain RDF statements that are not related to any logical structure. To address particular demands (such as reasoning, data sharing and so on) we need to associate those instances, classes and properties to well-defined ontologies. In the next code example we show how an additional XSLT stylesheet in the previous GRDDL process could add new ontological data concerning the FRBR ontology:

```
@prefix frbr: <http://purl.org/vocab/frbr/core#> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .

an:FRBRExpression owl:sameAs frbr:Expression .

za:eng@ frbr:realizer an:person/Organization/ke/parliament .

an:hasDate owl:sameAs dc:date .
```

Fig. 14. Other RDF statements extracted from Figure 12 on page 14

[2] All the RDF examples will be illustrated in Turtle syntax.

Through the last GRDDL step, we are associating particular semantics to all the instance extracted from the Akoma Ntoso document.

Note that, even if it offers the possibility to define semantics in order to infer automatically new data, OWL is not able to make reified RDF-like statement. To represent in OWL the Akoma Ntoso octuple statement illustrated previously, we have to use particular approaches to define complex ontological structures. Many different techniques can be used. One of the most elegant is the one that was born during the development of the NeOn project [17]: the Ontology Design Patterns [16]. The Ontology Design Patterns are a set of guidelines that help the design process of ontologies by adopting best practices. The following examples will use and briefly introduce some of those patterns; a proper description of each used pattern is out of the scope of our paper.

To demonstrate the effectiveness of this theory for handling complex OWL assertions such as the octuple, we use the n-ary participation pattern. Using it, we can describe events happening in particular moments that involve one or more entities. The aim of this pattern is to warrant the formalization in OWL of the RDF reification scenarios presented above. A graphical representation of how to express in OWL the semantics beyond the octuple is shown in the following figure.

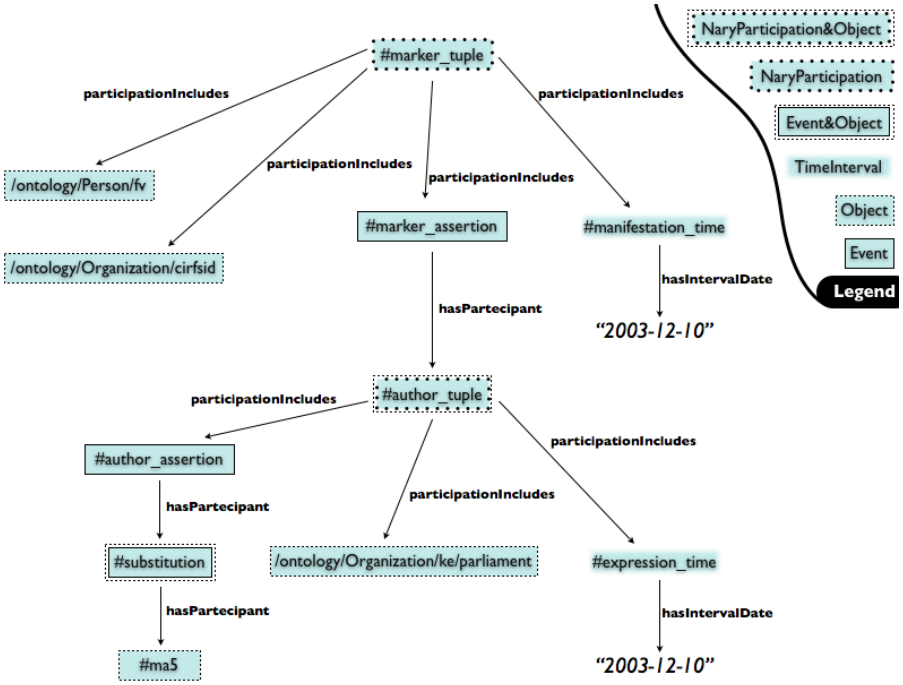

Fig. 15. A possible Akoma Ntoso data extraction into a OWL pattern-based format

Even if both RDF and OWL offer different ways to define formal semantics for Akoma Ntoso data, they are not used as standard methods to specify

those additional statements. It is true that there exist guidelines, such as the Ontology Design Patterns, that we could use to emulate complex scenarios. But, definitely, there is not a shared and shareable stable data model to represent every statement, such as the Akoma Ntoso octuple. For this reason, it is preferable to embed these data within XML documents in order to extract the data model when it is needed so that it is possible to adapt to the particular needs that are faced in that moment.

Moreover, there is another reason to use XML as format for Akoma Ntoso semantic data: the readability. Even if RDF and OWL have been developed to define data models, they have been also thought to be machine-readable. For this reason, their simplest serialization is not particularly comprehensible and since legal documents are primarily written by human beings for human beings themselves, it is clear that expressing such information in an ad-hoc XML format ends up being much more understandable by humans than a complex data model expressed in OWL.

The final goal of Akoma Ntoso is to create legal documents easily understandable and completely free from any unneeded constraint. For the reasons illustrated above, we think XML is the best format in which to represent these kind of documents.

6 Conclusions

Akoma Ntoso has been designed as a format for legal documents that must be read and understood for decades and at the same time be useful to computer reasoners. In order to balance clearness, fidelity to the authentic legal text, interoperability and usability with semantical tools, Akoma Ntoso made some clear choices. In this paper we showed how these choices fit the stated goals: using XML as the base mark-up format and having clearly separated layers allow documents to be preserved without modifications to the endorsed texts. Additionally, multiple agents can provide their own interpretation of certain legal aspects of the given legal text. Moreover, computer reasoners can extract semantic information from Akoma Ntoso document and reason over them.

The approach used by Akoma Ntoso allows the development of systems that use more sophisticated formal logic modeling framework, like non-monotonic or non-deductive logics in order to apply sophisticated legal reasoning theories, more suitable for the complex legal domain, filling the gap between all the semantic web layers while preserving interdependency and expressiveness.

7 Acknowledgments

The authors would like to thank Flavio Zeni (Kenya Unit of UN/DESA) and all of his team, for supporting the Akoma Ntoso project and for permitting this research project. We would also like to thank Alexander Boer and Enrico Francesconi for their precious contributions on the definition of the URI syntax.

References

1. Akoma Ntoso: Naming Conventions, http://www.akomantoso.org/release-notes/akoma-ntoso-1.0-schema/naming-conventions-1
2. Akoma Ntoso: Potential users, http://www.akomantoso.org/akoma-ntoso-in-detail/users
3. Alexander Boer et al.: Cen metalex workshop agreement (2009-08-28 proposal) (August 2009), http://www.metalex.eu/WA/proposal
4. Boag, S., Berglund, A., Kay, M., Robie, J., SimÃ©on, J., FernÃ¡ndez, M.F., Chamberlin, D.: XML Path Language (XPath) 2.0. Recommendation, W3C (Jan 2007), http://www.w3.org/TR/2007/REC-xpath20-20070123/. Latest version available at http://www.w3.org/TR/xpath20
5. C. Lupo et al.: General XML format(s) for legal sources. Estrella European Project IST-2004-027655 (2007)
6. Carroll, J.J., Klyne, G.: Resource Description Framework (RDF): Concepts and Abstract Syntax. Recommendation, W3C (Feb 2004), http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/. Latest version available at http://www.w3.org/TR/rdf-concepts/
7. Connolly, D.: Gleaning Resource Descriptions from Dialects of Languages (GRDDL). Recommendation, W3C (Sep 2007), http://www.w3.org/TR/2007/REC-grddl-20070911/. Latest version available at http://www.w3.org/TR/grddl/
8. David Hirtle et al.: Schema Specification of RuleML 0.91 (August 2006), http://ruleml.org/0.91/
9. Gordon, T.F., Governatori, G., Rotolo, A.: Rules and Norms: Requirements for Rule Interchange Languages in the Legal Domain. In: Proceeding of RuleML 2009. pp. 282–296 (2009)
10. Gordon, T.F.: Constructing Legal Arguments with Rules in the Legal Knowledge Interchange Format (LKIF). Computable Models of the Law: Languages, Dialogues, Games, Ontologies pp. 162–184 (2008)
11. IFLA Study Group on the Functional Requirements for Bibliographic Records: Functional requirements for bibliographic records: final report. K . G. Saur (1998)
12. Kay, M.: XSL Transformations (XSLT) Version 2.0. Recommendation, W3C (Jan 2007), http://www.w3.org/TR/2007/REC-xslt20-20070123/. Latest version available at http://www.w3.org/TR/xslt20
13. de Oliveira Lima, A., Palmirani, M., Vitali, F.: Moving in the Time: An Ontology for Identifying Legal Resources. Computable Models of the Law, Languages, Dialogues, Games, Ontologies pp. 71–85 (2008)
14. Palmirani, M., Benigni, F.: Norma-System: A Legal Information System for Managing Time. In: V Legislative XML Workshop. pp. 205–224 (2007)
15. Palmirani, M., Contissa, G., Rubino, R.: Fill the gap in the legal knowledge modelling. In: Proceeding of RuleML 2009. pp. 305–314 (2009)
16. Presutti, V., Gangemi, A.: Content ontology design patterns as practical building blocks for web ontologies. In: ER2008. Barcelona, Spain. (2008)
17. Valentina Presutti et al.: A Library of Ontology Design Patterns. NeOn project deliverable D2.5.1. (2008)
18. W3C OWL Working Group: OWL 2 Web Ontology Language Document Overview. Recommendation, W3C (Nov 2009), http://www.w3.org/TR/2009/REC-owl2-overview-20091027/. Latest version available at http://www.w3.org/TR/owl2-overview/