



Introduction to Akoma Ntoso

Prof. Fabio Vitali
University of Bologna





Purpose of this talk

- To introduce the audience to the on-going UNDESA project called **AKOMA NTOSO**.
- Sponsored by the UN Department of Economic and Social Affairs, **AKOMA NTOSO** is an on-going initiative to develop
 - ~ XML-based data formats
 - ~ Naming mechanisms
 - ~ Ontologies for metadata

for describing and managing legislative documents and Parliamentary workflow documentation needs in Africa



Next: Strengthening...





Strengthening...

“Strengthening Parliamentary Information System in Africa”

- In 2004 the United Nation Department for Economic and Social Affairs (UN/DESA) started a project
 - ~ To allow Parliaments in Africa to use current technologies in their daily workflows
 - ~ To improve citizens’ access to legislation and dialogue between electors and elected
 - ~ To provide a positive feedback mechanism that allow positive experience in one Parliament to be reflected back in initiatives in other Parliaments across Africa





AKOMA NTOSO

- **Architecture for Knowledge-Oriented Management of African Normative Texts using Open Standards and Ontologies.**
 - ~ A disciplined approach to designing document classes in XML
 - ~ A systematic mechanism for referencing documents within and across country borders based on URIs
 - ~ A ontological sound approach to designing metadata and relationships between different documents and different versions of documents.





Name and symbol



- “Akoma Ntoso” means Linked hearts
- It is the symbol used by the Akan people of West Africa to represent understanding and agreement.
- Likewise, AKOMA NTOSO represents the common standards that provide open access to parliamentary documentation and allow Parliaments to exchange information more efficiently.





The objectives of Akoma Ntoso

- Define a common format for documents of African parliamentary activities
 - ~ Primary Legislation – covering the lifecycle of a piece of legislation
 - ~ Parliamentary Debates
 - ~ Amendment lists
 - ~ Committee briefs
 - ~ Journals
- Define a model for tools
 - ~ Generation of documents
 - ~ Presentation of documents
 - ~ Access to documents
 - ~ Description of documents
- Easy to implement, easy to adopt and deploy, interoperable at all levels.





Designing Akoma Ntoso

- Building on successes and errors of NormeInRete (the corresponding Italian project started in 1999 and still going on strongly)
 - ~ Simpler, less prone to unimportant, small exceptions and more open to new changes and evolutions
 - ~ Support for multilinguism without renouncing to understandability and interoperability
 - ~ Support for simple implementations without renouncing to complexity and sophistication
- Available in DTD and XML Schema from the start. Soon in Relax NG, too





Users of Akoma Ntoso (1)

- The legislator wants to
 - ~ Retrieve bills and acts, and create new ones with appropriate references.
 - ~ Be able to follow the lifecycle of bills and acts, and access point-in-time consolidation of amended acts.
- The legal drafter wants to
 - ~ Rapidly prepare documents for discussion in the chambers and committees
 - ~ Rapidly publish stable forms of acts and bills in official publication channels
 - ~ Rapidly generate point-in-time consolidation of amended acts and new versions of bills





Users of Akoma Ntoso (2)

- The toolmaker wants to
 - ~ Easily create sophisticated computer tools to provide sophisticated services with acts and bills:
 - ~ Print, display, search, catalog, analyze, compare
- The citizen wants to
 - ~ Regardless of the fact he/she is a judge, a lawyer, a public employee, an ordinary citizen
 - ~ Easily access acts and bills, both in the current form and in point-in-time consolidation in the past
 - ~ Make use of explicit references among acts and other official documents





Users of Akoma Ntoso (3)

- The future toolmaker wants to
 - ~ Now: play with his school friends.
 - ~ In 10-15 years: be able to create new and incredibly sophisticated tools on Akoma Ntoso documents guaranteeing long duration of the project and the documents.
 - ~ Regardless of the overall poor state of documentation and of the evolution in time of the standard
 - ~ Simply through a visual analysis of existing Akoma Ntoso documents





Tools for Akoma Ntoso (1)

- The editor
 - ~ As an interface to activate, control and verify the automatic conversion tool
 - ~ As a tool to manually mark-up a document provided in a different format.
 - ~ As an application for direct insertion of both text and markup, starting off an empty document.
- The converter
 - ~ As a tool to convert into AKOMA NTOSO the documents that are still produced traditionally using normal Word Processors
 - ~ As a tool to convert into AKOMA NTOSO all legacy documents, including already approved bills and acts.
 - ~ The converter is based on the idea of semi-automatic conversion, examining the typographical and textual regularities of the document, and inferring a structural or semantic role for each text fragment. Experiences with European laws show that many fundamental structures of a legislative document can in fact be determined automatically with great precision.





Tools for Akoma Ntoso (2)

- Post-editing tools
 - ~ content and structure validators
 - ~ References validators
 - ~ Metadata validators
 - ~ Document management system, with search engines, hypertext functionalities, XSLT support and versioning facilities.
 - ~ XSLT stylesheets for visualizations of individual documents
- Other non official tools. E.g.:
 - ~ A text analysis tool for managing textual differences and evolutions in the drafting of laws
 - ~ A metadata engine to allow for semantic analysis of laws and even comparison with other countries.
 - ~ A versioning engine that automatically builds the any version of any act that has been modified in time.
 - ~ A publishing system that allows a private company to print on paper the acts and some comments on them from jurists.





Design Issues (1)

- Simple data model
 - ~ to facilitate usage and understanding
 - ~ relying on all the relevant W3C and ISO standards
- Long term feasibility and evolution (backward and forward)
 - ~ To support documents being drafted now and already drafted and enacted a long time ago.
 - ~ to support lifespan of the project and the documents in the tens and possibly hundreds of years
- Self explanation and tool support
 - ~ Documents need to be able to provide all information for their use and meaning through a simple examination, even without the aid of specialized software.
 - ~ Tools can be created with ease to provide automatic and semi-automatic aid to data markup and document description, yet manual markup or fine tuning to be considered a possible option for emergencies.





Design Issues (2)

- Extensibility
 - ~ It must be possible to allow local customizations of the Akoma Ntoso model
 - ~ It must be possible to extend the reach of the language towards more countries, more document types, larger vocabularies of fragment qualification
- Document exchange and homogeneity
 - ~ Documents produced by different tools and individuals need to be, as much as possible, identical
 - ~ Documents produced by hand and by tools need to be, as much as possible, identical
- Multiple uses
 - ~ Display on PC Screen, display on cell phone, display on Braille terminal, print on paper, print on paper *with a different paper size*, cataloguing, searching, workflow management (during drafting **and** active lifecycle), automatic consolidation, textual analysis, semantic analysis, provision analysis, cross-country comparison, synchronized translation, etc.





A taste of Akoma Ntoso (1)



- Overall structure

```
<akomantoso xmlns="http://www.akomantoso.org/1.0">
  <bill>
    <meta>
      METADATA ELEMENTS HERE
    </meta>
    <preface>
      TEXT HERE
    </preface>
    <preamble>
      TEXT HERE
    </preamble>
    <clauses>
      HIERARCHY OF PARTS HERE
    </clauses>
    <attachments>
      REFERENCES HERE
    </attachments>
  </bill>
</akomantoso>
```





A taste of Akoma Ntoso (2)

- Clauses

```
<chapter id="chap02">
  <num>CHAPTER 2</num>
  <title>TRADITIONAL COMMUNITIES AND TRADITIONAL COUNCILS</title>
  <article id="art02">
    <num>2.</num>
    <title>Recognition of traditional communities</title>
    <clause id="art02-cla01">
      <num>(1)</num>
      <p>A community may be recognised as...</p>
    </clause>
    <clause id="art02-cla02">
      <num>(2)</num>
      <p>The Premier of a province may, ... </p>
    </clause>
  </article>
</chapter>
```





A taste of Akoma Ntoso (3)

• Metadata

```
<meta>
  <descriptor>
    <publication date="2003-09-04" name="GG25437"
      showAs="Government Gazette # 25437" />
    <enactmentDate date="2003-09-04" />
    <editors>
      <editor id="ed01" date="2005-09-17" name="FV" />
      <editor id="ed02" date="2006-01-12" name="AB" />
    </editors>
    <uri href="http://www.parliament.gov.za/bill/2003/76_0904_en.xml"/>
  </descriptor>
  <lifecycle>
    <event id="evn1" source="org01" date="2005-08-29" type="Generation"/>
    <event id="evn2" source="pas01" date="2005-12-31" type="Repeal" />
  </lifecycle>
  <references>
    <Original href="/bill/2003/76_0904_en.xml" id="org01"/>
    <PassiveRef href="/act/2005/97_1231_en.xml" id="pas01"/>
  </references>
</meta>
```





Presentation, Structure, Semantics

- There is interplay between presentation (how the information looks), structure (how the information is organized, e.g. in document parts) and semantics (what the information is/represents).
 - ~ Semantic markup is the richest and most complete, and implies both structural role and expected presentation.
 - ~ Structural markup needs to be added for those document parts (e.g., hierarchies, preambles, etc.) that have no specific semantic role; it implies the expected presentation.
 - ~ Presentation markup: purely typographical aspects of text - if the semantic/structural model is incomplete, or if the element represents a one-off situation if there just an aesthetics reason for a specific presentation choice.
- For residual presentation problems Akoma Ntoso has a full presentational model.
 - ~ XHTML + CSS, for full and sophisticated presentation effects.





Descriptive vs. prescriptive approach

- **Descriptive schemas:** a very loose set of constraints providing a full vocabulary of elements and little or no check on their presence and order. They are meant to:
 - ~ Describe a set of documents with allowable many exceptions to the basic rule.
 - ~ Describe an existing (and thus non-modifiable) set of documents
 - ~ Describe a set of documents created by a higher authority than the XML coder.
- **Prescriptive schemas:** a more restricted set of constraints providing the same full vocabulary plus tight checks on presence and order. They are meant to:
 - ~ Impose adherence to drafting guidelines, and reject incompliant documents
 - ~ Impose homogeneity on the work of multiple different authors
 - ~ Allow applications to expect certain characteristic of the documents to be present
- Akoma Ntoso provides a two-tiered level of documents allowing the full potentiality of both to be expressed





Patterns and best practices

- Abstraction and distillation of past experiences solving design problems.
 - ~ Large vocabulary, few content models (5)
 - ~ Single root for simple processing
 - ~ Well-known vocabulary (HTML + CSS) for residual presentation needs
 - ~ Generic elements for semantically-heavy elements not explicitly listed in the vocabulary





The Document Architecture



- Akoma Ntoso describes two different but connected families of document formats
 - ~ The **General Schema**: One vocabulary and minimal set of constraints that all documents must comply to.
 - ~ □□□ **Detailed Schemas**: A set of stricter schemas. They provide more constraints over the same vocabulary to enforce the rules of specific national Parliaments.
- All documents satisfying one of the Detailed Schemas must also satisfy the General Schema.
- The General Schema is absolutely descriptive, and is meant to be applied consistently across all adopting countries.
- The Detailed schemas are country-specific, and can be more prescriptive, as long as the legal drafting office can impose forms and structures to the Parliament itself.





The General Schema

- The Akoma Ntoso General Schema is a single document handling all managed document types
 - ~ Based on previous successful experience for describing norms (the Italian standard Norme In Rete)
 - ~ Simple to implement and manage
 - ~ Fully and precisely descriptive
 - ~ Fully qualified to XML Namespaces
 - ~ Available in both DTD and XML Schema format
 - ~ Based on design patterns and best practices
 - ~ Fully documented
 - ~ Downloadable from <http://www.akomantoso.org/>





Conformance to Akoma Ntoso

- All tools **MUST** adhere to the General Schema in full. Tools **MAY** also choose to adhere to one or more of the Detailed Schemas, depending on locality and requirements of users.
- All tools **MUST** adhere to the Metadata Core Set and to the Document-Specific Metadata Set. Tools **MAY** adhere to the Metadata Local Set. Nonetheless, all tools **MUST** to ignore and maintain all metadata elements they do not understand.
- Tools **MAY** adopt the proposed stylesheets, but are in no requirement to do so. Tools can in fact require no stylesheet (if a tool is not meant for display, for instance) or provide their own.





Named resources in Akoma Ntoso

- All documents being managed within Akoma Ntoso are resources. They include:
 - ~ Original documents as created by the original authors (e.g., acts, bills, hansards, etc.)
 - ~ Attachments to original documents (and attachments to attachments, if any)
 - ~ Derived documents (such as point-in-time consolidated documents created on the fly)
 - ~ Navigation documents (such as the list of all acts emanated in 2005, or the home page of the publishing site, etc.)
- All resources have names that intelligibly describe their nature, their status, and their relationship to other resources.
- Names are URIs that are meant to be at the same time directly accessible and permanent (non-modifiable in time).





Naming convention in Akoma Ntoso

- The naming schema is still being finalized. But the final schema will provide unambiguous information about:
 - ~ The country publishing the resource
 - ~ The type of document
 - ~ Its status (original, derived, repealed, etc.)
 - ~ Version number and dates, where appropriate
 - ~ Emanating body





Metadata in Akoma Ntoso

- We define metadata all editorial content (i.e. content added by the editorial process out of Parliament rooms)
- Vice versa, all actual content of the document needs to have a place in the appropriate content sections.
 - ~ **Descriptors:** i.e., a set of meta information providing info about the document and its publication and edition details, including its official promulgation date, its official network address, and so on.
 - ~ **Lifecycle:** the lifecycle element provides information about the events that the document has undergone, and references to the documents that have caused these events..
 - ~ **Notes:** the text of the editorial notes that might be produced to comment and expand the actual text of the document. Note references inside the text point to notes contained here.
 - ~ **Proprietary:** this subsection allows any additional metadata to be specified in any order and vocabulary.
- The development of the meta section is not finished yet. Support for Dublin Core and Eurovoc-Africa is granted.





Conclusions

- Akoma Ntoso is a non proprietary data format for legislative data. It is open, application independent, and very rich and precise.
- It proposes a declarative and non-procedural approach allowing relevant information to be stored in the document with the content, rather than in the system or in the application.
- The vocabulary is rich and sophisticated, and derives from past experiences with Italian norms
- The application to the different African Parliamentary Systems will show that Akoma Ntoso can fruitfully be used for most situations where structured legislative acts are necessary.

