Dealing with Markup Semantics

Silvio Peroni
Department of Computer Science
University of Bologna
speroni@cs.unibo.it

Aldo Gangemi
Institute of Cognitive Sciences and Technology
CNR
aldo.gangemi@cnr.it

Fabio Vitali
Department of Computer Science
University of Bologna
fabio@cs.unibo.it

ABSTRACT

The correct interpretation of markup semantics is necessary for the semantic interpretation of linguistic expressions that use markup in their structuring and for enabling sophisticated operation on markup documents, such as semantic validation, multi-format document conversion and searching on heterogeneous digital libraries. The semantics of XML-based markup languages is usually provided informally, for example through textual descriptions in the specification of the language. While the syntax of XML-based languages is entirely machine-readable, its semantics is obscure for machines. Semantic Web technologies can be useful for filling the gap between the well-defined syntax of a language and the informal specification of its semantics. In this paper we show how to integrate LMM, an OWL vocabulary that represents some core semiotic notions, with EARMARK, a model for the specification of semantic and structural characteristics of markup languages, in order to provide a better understanding of the semantics of markup.

Categories and Subject Descriptors

I.7.2 [**Document And Text Processing**]: Document Preparation—*Markup languages*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation—*Representation languages*

General Terms

Languages

Keywords

EARMARK, LMM, linguistic act, markup semantics

1. INTRODUCTION

The advent of the Semantic Web (and social web) has induced a shift of meaning for some terms that are traditionally associated with markup languages. Originally, the act of *marking up* was strictly associated with document markup, where the term "tag" was used to refer to *markup elements*: syntactic items representing the building blocks of a document structure. While in the original definition markup "tells us something about [the text or content of a *document*]" [6], in the Semantic Web the term "markup" is sometimes used to identify any data added to a *resource* with the intention to semantically describe it (as well as "metadata" or "resource description"). Because of this recent re-drawing of the markup meaning, the term "tag" has also drastically changed its definition to "a non-hierarchical keyword or term assigned to a piece of information (such as an Internet bookmark, digital image, or computer file)"[1].

Partially because of this shift of meaning – that brought, as first consequence, the fact of having two different (and often unrelated) visions of the Web: the *Web of documents* and the *Web of data* – the Semantic Web has not considered in detail the issue of *markup semantics (*e.g., what is the meaning of a markup element *title* contained in a document *d*?), concentrating all its efforts in dealing with *semantic markup* (e.g., the resource *r* has the string "Semantic enhancement of document markup" as title) [17].

However, markup semantics is a very well-known and relevant issue for markup languages and consequently for digital libraries. Nowadays, a large amount of content stored in digital libraries is encoded with XML. XML, as any markup (meta-)language, provides a machine-readable mechanism for defining document structure, by associating labels to fragments of text and/or other markup. This association has a particular meaning, since each markup element asserts something about its content. What is asserted by the markup is not an issue of the markup itself. In fact, one of the goals of markup meta-languages is to avoid imposing any particular semantics: they express mere syntactic labels on the text, leaving the implicit semantics of the markup to the interpretation of humans or tools programmed by a human. Of course, a lot of markup languages, such as HTML, TEI and DocBook, are accompanied by natural language descriptions of their markup, but those descriptions are not machine-readable; in other words, there is no formal mechanism to embed markup semantics within markup language schemas.

Previous works [17] [18] [20] pointed out some clear advantages in having a mechanisms to define a machine-readable semantics of markup languages: enabling parsers to perform both syntactic and *semantic validation* of document markup; *inferring facts* from documents automatically by means of inference systems and reasoners; simplifying the *federation*,

[1] http://en.wikipedia.org/wiki/Tag_%28metadata%29

I-SEMANTICS 2011, 7th Int. Conf. on Semantic Systems, Sept. 7-9, 2011, Graz, Austria

conversion and *translation* of documents marked up with different and non-interoperable markup vocabularies; allowing users to *query* upon the structure of the document considering its semantics; creating *visualizations* of documents by considering the semantics of their structure rather than the specific vocabulary in which they are marked up; increasing the accessibility of documents' content, even in the case of *tag abuse* [9], i.e., "using markup languages construction in ways other than intended by the language designer"; promoting a more flexible *software design* for those applications that use markup languages, guaranteeing a better *maintainability* even when markup language schemas evolve.

For instance, it could be interesting to query documents for specific XML structures (e.g., all data tables in a collection of scientific papers written by a specific author, regardless of the fact that they were marked up with different vocabularies), or verifying semantic constraints of XML elements regardless of their position within the document (e.g., that all instances of speech fragments as transcribed in a parliamentary debate document is correctly assigned to the correct person that purportedly made the speech).

Although XML semantics might be directly addressed by the Semantic Web in order to gather the above-mentioned advantages, the Semantic Web community has always considered XML only as a serialization language for RDF or OWL, or as a way to encode relational data to be subsequently extracted and expressed in RDF. However, these two usages depart from the original goal of XML, i.e. to provide a mechanism for marking up digital documents (books, papers, messages, etc.). Consequently, it is often the case that e.g. relational data in XML encode both domain and document semantics; in such cases, extracting semantics from markup by means of bulk recipes generates semantic issues, because the dataset and/or ontologies obtained from that extraction will be unreliable (due to the usually conflicting data/text implicit semantics). A case study of this heterogeneity is the translation of FAO FIGIS document management schemata[2], which generates an ontology describing real world entities as well as documents, provenance, interfaces, versioning data, etc.

Of course, eRDF[3] and RDFa [1] may be valid choices for associating – and extracting by means of GRDDL [5] applications – formal semantics with arbitrary text fragments, and to markup elements within documents. Although they are very helpful for annotating documents and adding semantic information about markup elements and their content, their use is possible only by adding new attributes or, worse, new elements, therefore changing the document structure. The problem here is that the need of modifying the document structure is not easily suitable for domains, for example within organizations that deal with administrative or juridical documents, which must always preserve their structure as it is.

In this paper we introduce a proposal for defining markup semantics by using *EARMARK*, a markup meta-language based on Semantic Web technologies [8] [7], paired with *LMM* [16], a modular vocabulary to talk about textual semantics based on semiotical theories.

The *Extremely Annotational RDF Markup* (EARMARK) is at the same time a markup meta-language, that can express both the syntax and the semantics of markup as OWL assertions, and an ontology of markup that make explicit the implicit assumptions of markup languages (and, in particular, of the hierarchy of XML-based languages), providing a finer specification of the properties of markup, up to and including the possibility of toggling on and off the strict hierarchy of XML instantiations. It is important to stress that EARMARK does not prevent document authors from using RDFa and eRDF; rather, they can be used jointly.

Using EARMARK with LMM, it becomes possible to express and assess facts, constraints and rules about the markup structure as well as about the inherent semantics of the markup elements themselves, and about the semantics of the content of the document.

The purpose of this paper is to extend EARMARK with a particular module of LMM, used for describing *linguistic acts*, in order to represent the role that markup (as in XML and/or HTML markup) has in the semantics of expressions used in documents.

The paper is organized as follows: after illustrating in Section 2 some significant works in this area, we introduce EARMARK in Section 3 and, in Section 4, its extension, based on LMM, for associating formal semantics to markup elements. Then, in Section 5, we present two different use cases in which the markup semantics described by EARMARK+LMM is relevant and useful for addressing the related issues. We finally conclude the paper with a description of future developments about our work (Section 6) and some final considerations (Section 7).

2. RELATED WORKS

There is a large literature about semantics applied to markup. One of the first attempts for describing a formal markup semantics is introduced in [4]. The basic idea of the authors is to point out how users apply markup: through it, they make inferences about the document structures and the text those structures contain. According to the authors, "the meaning of markup is the set of inferences it licenses". The general framework they developed to associate semantics to markup and to make inferences on it needs some representation of the document (containing markup), a *sentence skeleton* for each item of the markup language we are considering in order to associate a meaning, and a set of (categorized) predicates and rules for allowing inferences. In this work, all the examples are illustrated using Prolog both for the representation of the nodes and for defining/inferring semantics using predicates and rules.

Focusing on the best-known meta-markup language, XML, in [18] the authors discuss problems characterizing schema languages for XML, from DTD to XMLSchema: those languages only permit a clear definition of the language syntax, and some of them (RelaxNG, XMLSchema) allow the declaration of a simple semantics on the datatypes, and little more. Although annotations can be specified for XMLSchema structures, there is no predefined semantics associated to them. Everything else concerning semantics – the meaning of an element, the relationships among items, etc. – is not expressible in a machine-readable format through those schema languages. The authors propose the BECHAMEL Project as a candidate solution to express markup semantics. As the authors explain in [17], BECHAMEL allows one to associate semantics with markup by adding new hierarchies to the original structure of the document. Using

[2]http://www.fao.org/fi/figis/devcon/diXionary/index.html
[3]http://www.egeneva.ch/w3c-RDF-ResourceDescriptionFramework/

these additional hierarchies, one can define the meaning of the elements and properties that cannot be expressed using the schema languages alone.

A different approach is used in [19]. The authors developed a framework to associate semantics with any XML document D in a three-step process:

1. defining an OWL ontology O to express all the meanings they want to use;
2. writing a set of rules R in a specific XML language to associate those meanings to a set of elements D;
3. through a XSLT transformation, processing D using O and R, so obtaining a new semantically-enriched XML document.

Similarly to the previous one, other works, such as [14] [10] [21], propose a general process that, starting from an XMLSchema S, an XML document D (written according to S) and an ontology O (that can be generated starting from S), allows to convert all the data in D, described by XML elements and attributes, into appropriate RDF instances consistent with O.

Finally, the approach introduced in [12] and [13] does not provide a formal machine-readable specification for defining markup semantics, but it is useful when human interpretation is needed in structuring a document. The authors describe *Intertextual Semantics*, a mechanism to associate meaning with markup elements and attributes of a schema as natural language constructs; this happens by associating a pre-text and a post-text with each of them. When the vocabulary of a schema is used correctly, the markup content is combined with the pre-text and post-text descriptions to make a correct natural language text that describes the entire information contained in a document. The difference between the common natural language documentation and Intertextual Semantics is that in the latter the meaning of a markup item is dynamically added when writing a document, and, as a consequence, can be read sequentially in the document editor itself.

3. EARMARK

EARMARK (Extremely Annotational RDF Markup) [8] [7] is a different approach to meta-markup based on ontologies and Semantic Web technologies. The basic idea is to model EARMARK documents as collections of addressable text fragments, and to associate such text content with OWL assertions that describe structural features as well as semantic properties of (parts of) that content. As a result, EARMARK allows not only documents with single hierarchies (as with XML), but also multiple overlapping hierarchies where the textual content within the markup items belongs to some hierarchies but not to others. Moreover, EARMARK makes it possible to add semantic annotations to the content through assertions that may overlap with existing ones.

Our Java-based implementation[4] strictly follows what is defined in the EARMARK ontology[5] that specifies classes and properties as summarized in Fig. 1. The core classes of our model describe three disjoint base concepts: *docuverses*, *ranges* and *markup items*.

[4]http://earmark.sourceforge.net
[5]http://www.essepuntato.it/2008/12/earmark

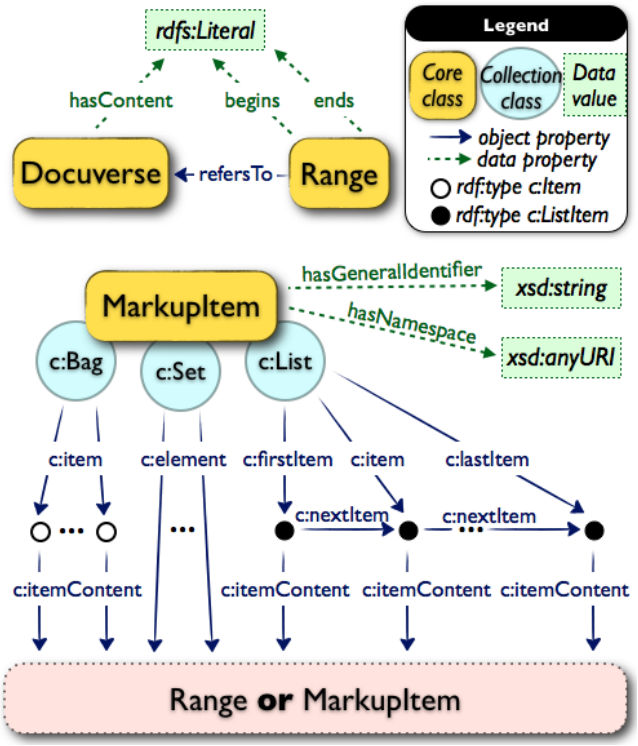

Figure 1: A graphical representation of the EARMARK ontology. The prefixes *rdfs*, *xsd* and *c* refer respectively to RDF Schema, XML Schema, and to an imported ontology used for handling collections.

The textual content of an EARMARK document is conceptually separated from its annotations, and is referred to through the *Docuverse* class. The individuals of this class represent the objects of discourse, i.e. all the containers of text from an EARMARK document. Any individual of the *Docuverse* class – commonly called a *docuverse* (lowercase to distinguish it from the class) – specifies its actual content through the property *hasContent*.

We define the class *Range* for any text lying between two locations of a docuverse. A *range*, i.e, an individual of the class *Range*, is defined by a starting and an ending location (any literal) of a specific docuverse through the functional properties *begins*, *ends* and *refersTo* respectively.

The class *MarkupItem* is the superclass defining artefacts to be interpreted as markup (such as elements and attributes). A *markupitem* individual is a collection[6] (*c:Set*, *c:Bag* and *c:List*, where the latter is a subclass of the second one and all of them are subclasses of *c:Collection*) of individuals belonging to the classes *MarkupItem* and *Range*. Through these collections it is possible:

- to define a markup item as a set of other markup items and ranges by using the property *c:element*;
- to define a markup item as a bag of items (defined by individuals belonging to the class *c:Item*), each of

[6]In the following descriptions the prefix *c* to indicate entities taken from an imported ontology used for handling collections, available at http://swan.mindinformatics.org/spec/1.2/collections.html.

them containing a markup item or a range, by using the properties *c:item* and *c:itemContent* respectively;

- to define a markup item as a list of items (defined by individuals belonging to the class *c:ListItem*), each of them containing a markup item or a range, in which we can also specify a particular order among the items themselves by using the property *c:nextItem*.

A *markupitem* might also have a name, specified in the functional property *hasGeneralIdentifier*[7], and a namespace specified using the functional property *hasNamespace*.

All the three core classes are specialized in other subclasses for giving more specific information about EARMARK instances. First of all, the class *Docuverse* is specialized into either a *StringDocuverse* (the content specified through *hasContent* is a string) or into an *URIDocuverse* (the actual content is located at the URL specified in *hasContent*), that are disjoint. Specialized subclasses of *Range* (*PointerRange* and *XPathRange*) are defined to cope with plain-text and XML docuverses with different addressing schemes. *MarkupItem* is specialized in three disjointed sub- classes: *Element*, *Attribute* and *Comment*.

In order to understand how EARMARK is used to describe markup hierarchies, let us to introduce an XML excerpt, using TEI fragmentation to express overlapping elements upon the string "Fabio says that overlhappens":

```
<p>
  <agent>Fabio</agent> says that
  <noun xml:id="e1" next="e2">overl</noun>
  <verb>
    h<noun xml:id="e2">ap</noun>pens
  </verb>
</p>
```

Here, the two elements *noun* represent the same element fragmented and overlapping with part of the textual content of *verb*, i.e., the characters "ap". The EARMARK translation of it is the following:

```
@prefix : <http://www.essepuntato.it/2008/12/
    earmark#> .
@prefix c: <http://swan.mindinformatics.org/
    ontologies/1.2/collections/> .
@prefix ex: <http://www.example.com/> .
ex:doc :hasContent "Fabio says that
    overlhappens" .
ex:r0-5 a :PointerRange ; :refersTo ex:doc
  ; :begins "0"^^xsd:integer
  ; :ends "5"^^xsd:integer .
ex:r5-16 a :PointerRange ; :refersTo ex:doc
  ; :begins "5"^^xsd:integer
  ; :ends "16"^^xsd:integer .
ex:r16-21 a :PointerRange ; :refersTo ex:doc
  ; :begins "16"^^xsd:integer
  ; :ends "21"^^xsd:integer .
ex:r21-28 a :PointerRange ; :refersTo ex:doc
  ; :begins "21"^^xsd:integer
  ; :ends "28"^^xsd:integer .
ex:r22-24 a :PointerRange ; :refersTo ex:doc
  ; :begins "22"^^xsd:integer
  ; :ends "24"^^xsd:integer .
ex:p a :Element ; :hasGeneralIdentifier "p"
  ; c:firstItem [ c:itemContent ex:agent
  ; c:nextItem [ c:itemContent ex:r5-16
  ; c:nextItem [ c:itemContent ex:noun
  ; c:nextItem [ c:itemContent ex:verb ]]]] .
ex:agent a :Element
  ; :hasGeneralIdentifier "agent"
  ; c:firstItem [ c:itemContent ex:r0-5 ] .
ex:noun a :Element
  ; :hasGeneralIdentifier "noun"
  ; c:firstItem [ c:itemContent ex:r16-21
  ; c:nextItem [ c:itemContent ex:r22-24 ]] .
ex:verb a :Element
  ; :hasGeneralIdentifier "verb"
  ; c:firstItem [ c:itemContent ex:r21-28 ] .
```

[7] *General identifier* was the SGML term for the local name of the markup item, e.g., "p" for markup element " ... ".

4. LINGUISTIC ACTS AS SEMANTIC ENHANCEMENT OF MARKUP ELEMENTS

EARMARK is suitable for expressing markup semantics straightforwardly. However, we want to associate coherent semantics to markup items following precise and theoretically-founded principles, which makes our application interoperable across different vocabularies used e.g. in digital libraries.

As a matter of fact, different existing vocabularies tackle the representation of terms vs. meanings vs. things in general, and this is not only true for XML markup languages, but also for semantic web ontologies such as SKOS, FRBR, CIDOC, OWL-WordNet, LIR, LMF, etc. Unfortunately, each of them has a particular approach depending on the original requirements they were designed for (thesauri encoding, media item representation, standardizing digital library vocabularies, lexicon or (multi-)linguality representation, etc.), so that aligning all or part of them for a specific use is a difficult operation, specially when we consider the domain of document structures, where arbitrary representations lead to different realizations for the user, to lack of interoperability, and lock markup semantics into islands. A viable solution to get around this problem is to align existing vocabularies to a more general and comprehensive vocabulary focused on semiotic notions.

We adopt the *linguistic act*[8] (*LA*) ontology design pattern, based on the *Linguistic Meta-Model* (*LMM*) [16]. It provides a semiotic-cognitive representation of linguistic knowledge. The general idea beyond it is to handle the representation of different knowledge sources developed according to different (and even implicit) semiotic theories, putting each of them in the context of the *semiotic triangle* [15] and some related semiotic notions, as shown in Fig. 2.

The pattern *linguistic act* is defined through an OWL ontology that implements the basic ideas of semiotics:

- *References*: any individual, set of individuals, or fact from the world we are describing. They can have interpretations (meanings) and can be denoted by information entities. For example: *Fabio*, *the set of Fabio's relatives*, or *the fact that Fabio is a professor*;
- *Meanings*: any (meta-level) object that explains something, or is intended by something, such as linguistic definitions, topic descriptions, lexical entries, thesaurus concepts, logical concepts or relations, etc. They can be "interpretants" for information entities, and "conceptualizations" for individuals and facts. For example, concepts such as *person*, *paragraph*, *having a role*;
- *Information entities*: any symbol that has a meaning, or denotes one or more references. They can be natural

[8] http://ontologydesignpatterns.org/cp/owl/semiotics.owl

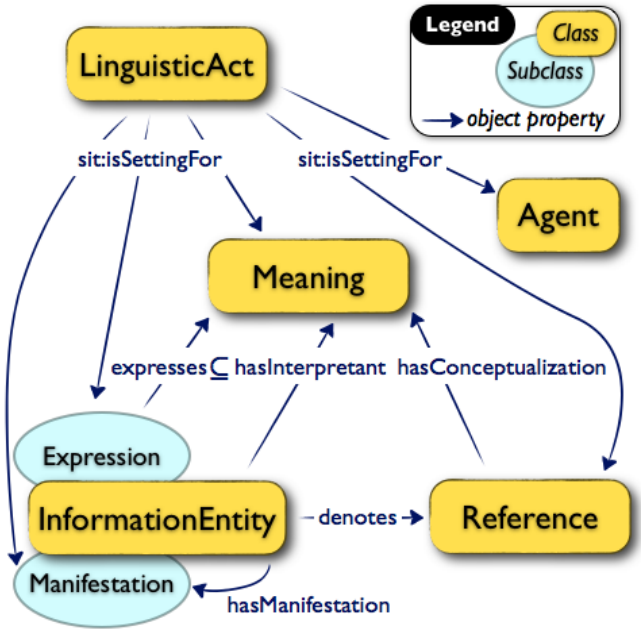

Figure 2: A diagram summarizing the ontology pattern *linguistic act*.

language terms, sentences or texts, symbols in formal languages, icons, or whatever can be used as a vehicle for communication – for example: the string "Fabio", the markup elements *p*, *agent*, *noun* and *verb*. They have at least one meaning and can denote references. Moreover, each information entity can be an *expression* (e.g., the string "Fabio") realized in one or more *manifestations* (e.g., the string "Fabio" contained in a particular XML file stored on somebody's hard drive) having the same interpretation.

- *Linguistic acts*: any communicative situation including information entities, agents, meanings, references, and a possible spatio-temporal context (i.e. when and/or where the act has been performed). For example, dialogs, taggings, writings.

Considering these premises, EARMARK markup items are specific kinds of expressions expressing a particular meaning, usually assigned implicitly by the author of a schema or a markup, which are used to denote local objects (e.g., their content, according to the definition of a markup object) and/or social entities (e.g., persons, places, events, etc.).

For example, in the XML example introduced in Section 3 we have different semantic blocks: firstly, the element *agent* expresses the meaning of "agent" (i.e., as the resource defined by DBPedia[9]) and denotes a specific person (i.e., the person, using FOAF, is known as "Fabio Vitali"), while the element *p* must be interpreted as a paragraph (i.e. a specific document structure according to the DOCO ontology[10]) and denotes the string "Fabio says that overlhappens" (rather than the corresponding concept). This in a way differs from the XML syntactical structure in which the element *p* contains the elements *agent*, *noun* and *verb* – that themselves express/denote/contain the other meanings/references.

In LA-EARMARK, we can describe both the rigid syntactical structure, as described in Section 3, as well as its semantical connotation:

```
@prefix ar: <http://www.ontologydesignpatterns.
    org/cp/owl/agentrole.owl#> .
@prefix la: <http://www.ontologydesignpatterns.
    org/cp/owl/semiotics.owl#> .
@prefix sit: <http://www.ontologydesignpatterns
    .org/cp/owl/situation.owl#> .
@prefix doco: <http://purl.org/spar/doco/> .
@prefix dbpr: <http://dbpedia.org/resource/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
ex:r0-28 a :PointerRange ; :refersTo ex:doc
  ; :begins "0"^^xsd:integer
  ; :ends "28"^^xsd:integer .
ex:p la:expresses doco:Paragraph
  ; la:denotes ex:r0-28 .
ex:agent la:expresses
   dbpr:Agent , doco:TextChunk
  ; la:denotes ex:fv , ex:r0-5 .
ex:fv a foaf:Person
  ; foaf:givenName "Fabio"
  ; foaf:familyName "Vitali" .
ex:markupAuthor a ar:Agent
  ; ar:hasRole [ a ar:Role
    ; rdfs:label "markup author" ] .
[] a la:LinguisticAct
  ; rdfs:comment "marking a paragraph up"
  ; sit:isSettingFor ex:p , ex:r0-28
    , doco:Paragraph , ex:markupAuthor .
[] a la:LinguisticAct ; rdfs:comment "marking
    text up"
  ; sit:isSettingFor ex:agent , ex:r0-5
    , doco:TextChunk , ex:markupAuthor .
[] a la:LinguisticAct
  ; rdfs:comment "markup element as instance"
  ; sit:isSettingFor ex:agent
    , ex:fv , dbpr:Agent . ...
```

5. USING LA-EARMARK IN REAL-CASE SCENARIOS

The examples introduced in the previous sections explain how it is possible to describe markup hierarchies – and therefore their semantics – upon those markup items. In the next sub-sections we show the advantages of using LA-EARMARK in two different use cases, previously highlighted in [17]: querying documents marked up with the same implicit semantics but marked up with different vocabularies that share the same implicit semantics and the semantic validation of markup items.

5.1 Searches on heterogeneous digital libraries

Digital libraries about journal research articles use to actually store their documents' content using specific XML formats, e.g. the common TEI, DocBook, or other less common vocabularies developed expressly for a specific collection. Clearly, the more digital libraries we consider, the more non-interoperable formats we will find, although they express, more or less, the same kinds of documents and, consequently, document semantics. In fact, paragraph, sections (implicitly or explicitly labelled as abstract, introduction, results, discussion, related works, conclusions, acknowledgements, bibliography, etc.), figures, tables, formulas are a little but significative part of the elements that we will find

[9]http://dbpedia.org/resource/Agent
[10]http://purl.org/spar/doco

in the markup of journal papers, regardless of the actual vocabulary used.

In this scenario of heterogeneous formats expressing homogeneous content, looking throughout a number of digital libraries for particular document fragments, such as "All the tables that are part of the results sections of articles written by Silvio Peroni", can be approached only by addressing each digital library with a query specific of the vocabulary used, and then merging the results. Obviously, the implicit (shared) semantics of the query must be implemented in each digital library in a (different) explicit way, for example by using tools for mapping the query into each specific markup structure. This means requiring a particular *ad hoc* and non-interoperable mechanism for each format of each digital library.

Expressing semantics of elements in a journal article by considering a shared model may help for increasing interoperability, but it is not enough, because the different formats will still be a substantial problem. For example, being a *section* presenting *results* in a particular research article may be expressed differently depending on the format used: `<div class="section.results">`, `<section id="results">`, `<sec class="results">`, `<results>`, etc.

Expressing journal articles in LA-EARMARK – obtained, for instance, by translating the original XML documents via GRDDL – allows to specify the semantics of markup elements according to some formal model, without attention to the specific markup vocabulary[11]:

```
@prefix deo: <http://purl.org/spar/deo/>
ex:div a :Element ; :hasGeneralIdentifier "div"
  ; c:firstItem [ c:itemContent ex:classAttr ]
  ; la:expresses doco:Section , deo:Results .
ex:results a :Element
  ; :hasGeneralIdentifier "results"
  ; la:expresses doco:Section , deo:Results .
```

As shown in the previous excerpt, both *ex:div* and *ex:results* elements express the same semantics even if their names differ: they are syntactically different (their content models differ), but semantically equivalent.

Enabling digital libraries to express each LA-EARMARK document as a named graph, with all the document metadata referring to it, allows to query more than one digital library at the same time by using a single SPARQL 1.1 query [11]. For instance, a plausible SPARQL query for the abovementioned request – "All the tables that are part of results sections of the article written by Silvio Peroni" – is:

```
SELECT ?table WHERE {
  GRAPH ?doc {
    ?table a :Element ; la:expresses doco:Table
      ; (^c:itemContent/^c:item)+
        [ a :Element
        ; la:expresses
          doco:Section , deo:Results ] } .
  ?doc dc:creator "Silvio Peroni" }
```

5.2 Validation of "Markup sensibility"

Sometimes it is not possible to understand whether a particular markup element that is valid at the syntactical and structural level is also valid at the semantic level, i.e., the level that Bauman described as *markup sensibility*: "Does a construct make sense, e.g., a proposition or an assertion?" [3]. A clear example of this difficulty can be found with heavily interlinked documents that make systematic references to precise concepts in their content.

For instance, Akoma Ntoso [2] 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. 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 and GRDDL.

This markup language is defined by means of a very complex XML Schema document, that defines the vocabulary and the content models of markup items. Although that schema is enough to guarantee the validity of a document from a pure syntactical point of view, there are semantic connections that are useful to verify but cannot by simply using a schema language. Let us introduce an Akoma Ntoso excerpt to clarify the point:

```
<akomaNtoso>
  <meta> ... <references source="#fv">
      <TLCPerson id="fv"
        href="/ontology/it/person/FabioVitali">
      <TLCPerson id="smith"
        href="/ontology/uk/person/JohnSmith">
      <TLCRole id="mineconomy"
        href="/ontology/role/government/
            MinisterOfEconomy"> ...
    </references> ... </meta>
  <body> ...
    <speech id="sp1" by="#smith" as="#
        mineconomy">
      <p>Honorable Members of the Parliament,
          ...</p>
    </speech> ...
  </body>
</akomaNtoso>
```

The elements *TLCPerson* and *TLCRole*, introduced within the metadata block (element *meta*) of the document, are used for specifying the presence, in the document in which it is defined, of two particular ontological entities, respectively a person and a role, according to a specific underlying ontology. Wherever these elements are referred to by a markup element by means of its identifier (as expressed in the attribute *id*), what really is referred to are the ontological individuals that are specified by the attribute *href*. For instance, within the body of the document, the element *speech* is used to mark up the transcription of a speech performed by the person *John Smith* (attribute *by*) who is temporarily playing the particular role of Minister of the Economy (attribute *as*). Moreover, the attribution of all the metadata concerning the speech transcription is an editorial activity, rather than authorial, made specifically by an agent identified through the attribute *source* of the element *reference*. For self-containment, the attributes *by* and *as* do not refer directly to the ontological concepts associated to John Smith and the Minister of the Economy, but to an intermediate jumping station, i.e., the elements TLCPerson and TLCRole in the metadata block.

[11] The prefix *deo* refers to an ontology for the characterisation of the major rhetorical elements of a document (e.g., a research article), such as the introduction part, the evaluation section, the conclusions and so on.

Although it is a fundamental requirement of the language, the syntactic validation through XML Schema of the document does not provide sufficient information to understand whether an Akoma Ntoso document is really correct and coherent, because it cannot prove the sensibleness of markup. In the preceding example, we also need to check:

- the validity of the elements *TLCPerson* and *TLCRole* as reflection of the consistence of people and role individuals within an underlying ontology, particularly by checking whether each individual can really be a person (or a role) without provoking an inconsistency with other classes the individual may belong to;
- the validity of the element *speech* as markup denoting a particular speech event that involves only and at least a person as speaker. Moreover, because it reflects a speech, it must contain some text.
- the fact that the person John Smith was, at the moment of the speech, either the Minister of Economy or acting as a authorized delegate through a track of explicit delegations starting from the current minister.

The XML Schema language is not able to express these kinds of constraints. In fact, naive or inexpert metadata authors could very well generate documents that are syntactically and structurally valid, possibly even apparently correct from a semantic point of view, but fundamentally incoherent. For instance, a common misconception is to confuse persons and roles, as in the following (syntactically valid but ontologically incorrect) example,

```
<speech id="sp1" by="#mineconomy">
  <p>Honorable Members of
    the Parliament, ...</p></speech>
```

The LA-EARMARK translation of the above fragment, that includes its semantical description, is the following[12]:

```
@prefix akomantoso: </ontology/entity/> .
</ontology/uk/person/JohnSmith>
  a akomantoso:Person .
[] a la:LinguisticAct
  ; sit:isSettingFor
     <smith> , akomantoso:Person
    , </ontology/uk/person/JohnSmith> .
<sp_1> a :Element
  ; :hasGeneralIdentifier "speech"
  ; la:expresses akomantoso:Speech
  ; la:denotes _:aSpeechEvent , _:p .
_:aSpeechEvent a akomantoso:SpeechEvent
  ; akomantoso:hasSpeaker
    </ontology/uk/person/JohnSmith> .
_:p a :Element ; :hasGeneralIdentifier "p" .
[] a la:LinguisticAct
  ; sit:isSettingFor
     <sp_1> , _:aSpeechEvent
    , </ontology/uk/person/JohnSmith>
    , akomantoso:Speech .
[] a la:LinguisticAct
  ; sit:isSettingFor <sp_1> , _:p
    , akomantoso:Speech .
```

LA-EARMARK allows to check the sensibility of markup precisely, by defining semantic constraints as ontological axioms, taking into account both classes and properties defined in LA-EARMARK and in the underlying ontology behind Akoma Ntoso. Inasmuch as such semantic constraints can be defined as axioms adhering to or in contrast with axioms of the underlying ontologies, they can be directly applied to reasonings even in open world frameworks such as OWL.

For example, a plausible ontological constraint (written in Manchester Syntax) for all the markup elements *speech* is:

```
(Element that hasGeneralIdentifier
  value "speech") SubClassOf
(sit:hasSetting only
  (la:LinguisticAct that
    sit:isSettingFor exactly 1
      (Element and la:InformationEntity) and
    sit:isSettingFor exactly 1
      (Range and la:Reference) and
    sit:isSettingFor value akomantoso:Speech)
  or
  (la:LinguisticAct that
    sit:isSettingFor exactly 1
      (Element and la:InformationEntity) and
    sit:isSettingFor exactly 1
      ((akomantoso:SpeechEvent and
        la:Reference) that
          akomantoso:hasSpeaker some
            akomantoso:Person) and
      sit:isSettingFor
        value akomantoso:Speech))
```

This specification would be able to capture ontological errors in the actual Akoma Ntoso document such as the one presented previously, where the author of the speech is specified as a role rather than a person.

6. FUTURE WORKS

We are now working on the development of a tool for assisting users in the definition of markup semantics of a given schema using LA-EARMARK features. Considering markup items of a schema as information entities involved in linguistic acts, the LA-EARMARK assistant (shown in the mockup in Fig. 3) simplifies the creation of an ontology for the description of markup semantics, through a user-friendly interface that allows to use existing models, and to create new ontological entities by means of appropriate constructs. Using the defined ontology, local occurrences of each markup item described in the schema within a document can be automatically enhanced with its semantic definition.

Having semantic definitions associated to each markup item, we plan to develop a plugin for the semantic validation of markup documents under an ontology defining markup semantics, and another plugin for visualizing parts of a document according to the particular semantics of markup elements contained therein.

7. CONCLUSIONS

Complementary to existing Semantic Web research work, which typically aims at studying uses and applications of *semantic markup* (i.e., defining relations within or among resources), in this paper we have addressed the issue of *markup semantics*: the formal definition of meanings of markup elements, besides the syntactical structure of a markup document. We have described a model for defining markup semantics called LA-EARMARK, based on the EARMARK ontology, and a markup language that enables the definition of complex documents using Semantic Web technologies and tools. The model reuses a design pattern for describing *lin-*

[12]The prefix *akomantoso* is associated to the minimal *glue* ontology within the XML document itself that connects markup structures to legal concepts according to the model explained in [2].

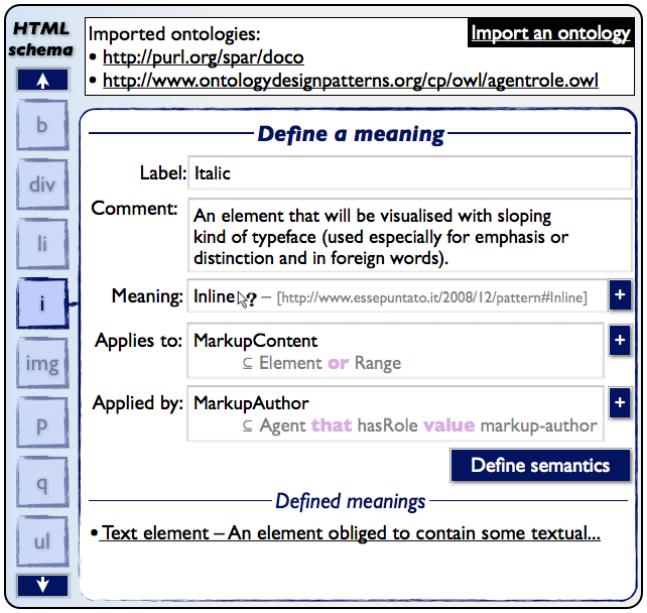

Figure 3: A mockup of the LA-EARMARK assistant.

guistic acts, the core fragment of LMM – an OWL ontology that encodes a semiotic-cognitive representation of linguistic knowledge, with the scope of abridging informal and formal semantics. LA-EARMARK has been used in two different real-case scenarios, also indicated previously in other scholarly works as applications of markup semantics.

8. REFERENCES

[1] Adida, B., Birbeck, M., McCarron, S., Pemberton, S. (2008). RDFa in XHTML: Syntax and processing. W3C Recommendation, 14 October 2008. World Wide Web Consortium.

[2] Barabucci, G., Cervone, L., Palmirani, M., Peroni, S., Vitali, F. (2009). Multi-layer markup and ontological structures in Akoma Ntoso. In Proceeding of the International Workshop on AI approaches to the complexity of legal systems II (AICOL-II). Rotterdam, The Netherlands.

[3] Bauman, S. (2010). The 4 "Levels" of XML Rectitude. Presented as poster in Balisage: The Markup Conference 2010. Montréal, Canada. http://bauman.zapto.org/~syd/temp/XML_rectitude.pdf

[4] C. M. Sperberg-McQueen, Claus Huitfeldt, and Allen Renear. (2000). Meaning and interpretation of markup. In Markup Languages: Theory & Practice, 2 (3), MIT Press, pp. 215-234.

[5] Connolly, D. (2007). Gleaning Resource Descriptions from Dialects of Languages (GRDDL). W3C Recommendation, 11 September 2007. World Wide Web Consortium.

[6] Coombs, J. H.,Renear A. H., DeRose, S. J. (1987). Markup Systems and the Future of Scholarly Text Processing. Communications of the ACM 30, pp. 933-947.

[7] Di Iorio, A., Peroni, S., Vitali, F. (2011). A Semantic Web Approach To Everyday Overlapping Markup. To be published in Journal of the American Society for Information Science and Technology. Wiley.

[8] Di Iorio, A., Peroni, S., Vitali, F. (2011). Using Semantic Web technologies for analysis and validation of structural markup. To be published in International Journal of Web Engineering and Technologies. Inderscience Publisher.

[9] Dubin, D. (2003). Object mapping for markup semantics. In Proceedings of Extreme Markup 2003. Montréal, Canada.

[10] Garcia, R., Celma, O. (2005) Semantic Integration and Retrieval of Multimedia Metadata. In Proceedings of the 5th International Workshop on Knowledge Markup and Semantic Annotation (SemAnnot 2005). Galway, Ireland.

[11] Garlik, S. H., Seaborne, A. (2010). SPARQL 1.1 Query Language. W3C Working Draft, 14 October 2010. World Wide Web Consortium.

[12] Marcoux, Y. (2006). A natural-language approach to modeling: Why is some XML so difficult to write? In Proceedings of the Extreme Markup Languages 2006. Montreal, Canada.

[13] Marcoux, Y., Rizkallah, E. (2009). Intertextual semantics: A semantics for information design. Journal of the American Society for Information Science and Technology, 60 (9), pp. 1895-1906.

[14] Nuzzolese, A., Gangemi, A., Presutti, V. (2010). Gathering Lexical Linked Data and Knowledge Patterns from FrameNet. In Proceedings of The Sixth International Conference on Knowledge Capture (K-CAP 2011). Banff, Canada.

[15] Peirce, C. S. (1958). Collected Papers of Charles Sanders Peirce. MIT Press, Cambridge, Massachusetts.

[16] Picca, D., Gliozzo, A., Gangemi, A. (2008). LMM: an OWL-DL MetaModel to Represent Heterogeneous Lexical Knowledge. In Proceedings of the 6th Language Resource and Evaluation Conference. Marrakech, Morocco.

[17] Renear, A., Dubin, D., Sperberg-McQueen, C. M. (2002). Towards a Semantics for XML Markup. In the Proceedings of the 2002 ACM Symposium on Document Engineering. McLean, Virginia.

[18] Renear, A., Dubin, D., Sperberg-McQueen, C. M., Huitfeldt, C. (2003). XML Semantics and Digital Libraries. In the Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries. Houston, Texas.

[19] Simons, G. F., Lewis, W. D., Farrar, S. O., Langendoen, D. T., Fitzsimons, B., Gonzalez, H. (2004). The semantics of markup: mapping legacy markup schemas to a common semantics. In Proceedings of the Workshop on NLP and XML (NLPXML-2004). Barcelona, Spain.

[20] Sperberg-McQueen, C. M., Marcoux, Y., Huitfeldt, C. (2009). Two representations of the semantics of TEI Lite. In Proceedings of Digital Humanities 2010. London, UK.

[21] Van Deursen, D., Poppe, C., Martens, G., Mannens, E., Van de Walle, R. (2008). XML to RDF Conversion: a Generic Approach. In Proceedings of the 4th International Conference on Automated solutions for Cross Media Content and Multi-channel Distribution (AXMEDIS 08). Florence, Italy.