Semantic Web 7 (2016) 167–181
DOI 10.3233/SW-150177
IOS Press

The Document Components Ontology (DoCO)

Editor(s): Oscar Corcho, Universidad Politécnica de Madrid, Spain
Solicited review(s): Francesco Ronzano, Universitat Pompeu Fabra, Barcelona, Spain; Almudena Ruiz Iniesta, Universidad Politécnica de Madrid, Spain; one anonymous reviewer

Alexandru Constantin [a], Silvio Peroni [b,c,*], Steve Pettifer [d], David Shotton [e] and Fabio Vitali [b]

[a] *École Polytechnique Fédérale de Lausanne, PFL IC IIF LSIR, BC 159 (Bâtiment BC), Station 14, CH-1015 Lausanne, Switzerland*
E-mail: alex.constantin@epfl.ch

[b] *Department of Computer Science and Engineering, University of Bologna, Mura Anteo Zamboni 7, 40127 Bologna (BO), Italy*
E-mails: silvio.peroni@unibo.it, fabio.vitali@unibo.it

[c] *Semantic Technology Laboratory, Institute of Cognitive Sciences and Technologies, National Research Council, Via Nomentana 56, 00161 Rome (RM), Italy*

[d] *School of Computer Science, University of Manchester, Kilburn Building, M13 9PL Manchester, United Kingdom*
E-mail: steve.pettifer@manchester.ac.uk

[e] *Oxford e-Research Centre, University of Oxford, 7 Keble Road, OX1 3QG Oxford, United Kingdom*
E-mail: david.shotton@oerc.ox.ac.uk

Abstract. The availability in machine-readable form of descriptions of the structure of documents, as well as of the document discourse (e.g. the scientific discourse within scholarly articles), is crucial for facilitating semantic publishing and the overall comprehension of documents by both users and machines. In this paper we introduce *DoCO*, the *Document Components Ontology*, an OWL 2 DL ontology that provides a general-purpose structured vocabulary of document elements to describe both structural and rhetorical document components in RDF. In addition to giving a formal description of the ontology, this paper showcases its utility in practice in a variety of our own applications and other activities of the Semantic Publishing community that rely on DoCO to annotate and retrieve document components of scholarly articles.

Keywords: DEO, DoCO, PDFX, SPAR ontologies, Utopia Documents, document components, rhetoric, structural patterns

1. Introduction

One of the most important criteria for the evaluation of a scientific contribution is the coherent organisation of the textual narrative that describes it, most often published as a scientific article or book. In most academic disciplines, such writings have well-established models of organisation and rhetorical structure, to which scholars and contributors generally abide. These expectations are promoted by academic publishers, who ask for standardised models in the submissions they receive, constructed to efficiently describe the content's organisation in logical sections. Such models not only express the expected structure of the article or book, but facilitate the detection of omissions, redundancies or incorrect sequences. Unfortunately, the number of distinct vocabularies adopted by publishers to describe these requirements is quite large, expressed in bespoke document type definitions (DTDs). There is thus a need to integrate these different languages into a single, unifying framework that may be used for all content, regardless of provenance and scientific context. For instance, a recent report by Beck [3] explains the requirements for an XML vocabulary of scientific journals to be acceptable for inclusion in PubMed Central[1].

*Corresponding author. E-mail: silvio.peroni@unibo.it.

[1] PubMed Central: http://www.ncbi.nlm.nih.gov/pmc/.

Several studies exist that discuss models and theories for describing the structural, rhetorical and argumentative functions of texts. Such detailed descriptions in machine-readable form (e.g. [31]) have become a necessity for high-volume data access and comprehension both by humans and machines [8,10]. It is also a strict requirement for the complex process of semantic publishing [36,37]. Being able to simplify and automate the time-consuming process of annotating structural and rhetorical behaviours of document components (such as identifying front/body/back matters, Abstract, Results, etc.) may be instrumental in providing a number of services to publishers, open archives, and scientists themselves. For instance, the correct identification of structural patterns in academic documents could be used to generate lists and summaries automatically (e.g., tables of contents, lists of figures), to render the content in a web browser, or to provide full-scale converters between different component vocabularies, readily usable by delivery and publication platforms.

This paper describes *DoCO* – the *Document Components Ontology*, an OWL 2 DL ontology that provides a general-purpose structured vocabulary of document elements, that is one of the principal ontologies within the SPAR (Semantic Publishing and Referencing) Ontologies (http://www.sparontologies.net), a suite of orthogonal and complementary ontology modules for creating comprehensive machine-readable RDF metadata for all aspects of semantic publishing and referencing. DoCO has been designed as a general unifying ontological framework for describing different aspects related to the content of scientific and other scholarly texts. Its primary goal has been to improve the interoperability and shareability of academic documents (and related services) when multiple formats are actually used for their storage. In the following sections, both the structural and the rhetorical foundations of DoCO are presented, along with hybrid structures that describe components in terms of their complementary structural and rhetorical behaviour. The utility of the ontology in practice is then illustrated by showcasing a variety of our own applications that rely on DoCO to annotate and retrieve document components of scholarly articles, and by introducing other activities of the Semantic Publishing community that directly use or promote DoCO as a comprehensive ontology for modelling document components in RDF.

The rest of this paper is organised as follows. In Section 2 we discuss some relevant work about models describing document components. In Section 3 we give an overview of DoCO, presenting its foundations and formal characterisation to describe the organisation of documents according to both structural patterns and rhetoric structures. In Section 4 we illustrate how DoCO is presently being used for annotation and document component retrieval, two high-value tasks in literature management and analysis. Finally, in Section 5 we present further development planned for the near future.

2. Related works

2.1. Semantic Publishing and Referencing ontologies

In the past, several groups have proposed (Semantic Web) models, such as RDFS vocabularies and OWL ontologies, to describe particular aspects of the publishing domain, these being mainly concerned with the description of the metadata of bibliographic resources (e.g., DCTerms[2], PRISM[3] and BIBO[4]). One of the first attempts to address the description of the whole publishing domain is the introduction of the Semantic Publishing and Referencing (SPAR) ontologies[5]. SPAR is a suite of orthogonal and complementary OWL 2 ontologies that enable all aspects of the publishing process to be described in machine-readable metadata statements, encoded using RDF.

The original suite of SPAR ontologies comprises eight distinct modules. The following is a brief description of seven of these, while the last one, DoCO, is appropriately discussed in Section 3:

1. The *FRBR-aligned Bibliographic Ontology* (*FaBiO*)[6] [29] is an ontology for describing entities that are published or potentially publishable (e.g., journal articles, conference papers, books), and that contain or are referred to by bibliographic references;
2. The *Citation Typing Ontology* (*CiTO*)[7] [29] is an ontology that enables characterization of the nature or type of citations, both factually and rhetorically;

[2] DC Terms: http://purl.org/dc/terms.
[3] PRISM: http://www.prismstandard.org/resources/mod_prism.html.
[4] BIBO: http://purl.org/ontology/bibo/.
[5] Semantic Publishing and Referencing ontologies: http://www.sparontologies.net.
[6] FaBiO: http://purl.org/spar/fabio.
[7] CiTO: http://purl.org/spar/cito.

3. The *Bibliographic Reference Ontology* (*BiRO*)[8] [12] is an ontology used to define bibliographic records, bibliographic references, and their compilation into bibliographic collections and bibliographic lists, respectively;
4. The *Citation Counting and Context Characterisation Ontology* (*C4O*)[9] [12] is an ontology that permits the number of in-text citations of a cited source to be recorded, together with their textual citation contexts, along with the number of citations a cited entity has received globally on a particular date;
5. The *Publishing Roles Ontology* (*PRO*)[10] [30] is an ontology for the characterisation of the roles of agents – people, corporate bodies and computational agents in the publication process. These agents can be, e.g. authors, editors, reviewers, publishers or librarians;
6. The *Publishing Status Ontology* (*PSO*)[11] [30] is an ontology designed to characterise the publication status of documents at each stage of the publishing process (draft, submitted, under review, etc.);
7. The *Publishing Workflow Ontology* (*PWO*)[12] [18] is a simple ontology for describing the steps in the workflow associated with the publication of a document or other publication entity.

The above seven ontologies, along with the Document Components Ontology (DoCO), form the original set of SPAR ontologies. This set has more recently been extended with four other complementary ontologies that extend the coverage of the possible description of the publishing domain. These are as follows:

- The *Scholarly Contributions and Roles Ontology* (*SCoRO*)[13] – an ontology based on PRO for describing the contributions that may be made, and the roles that may be held by a person with respect to a journal article or other publication (e.g. the role of author, data manager, article guarantor or illustrator);
- The *Funding, Research Administration and Projects Ontology* (*FRAPO*)[14] is an ontology for describing the administrative information of research projects, e.g., grant applications, funding bodies, project partners, etc.;
- The *DataCite Ontology*[15] is an ontology that enables the metadata properties of the *DataCite Metadata Schema Specification*[16] (i.e., a list of metadata properties for the accurate and consistent identification of a resource for citation and retrieval purposes) to be described in RDF;
- The *Bibliometric Data Ontology* (*BiDO*)[17] [25], is a modular ontology that allows the description of numerical and categorial bibliometric data (e.g., journal impact factor, author h-index, categories describing research careers) in RDF.

Still being actively maintained and expanded, the SPAR ontologies have drawn the attention of the Semantic Publishing community, as a reference point for standardising entity descriptions and fostering interoperability between services – as discussed in Section 4.

2.2. Existing models describing document components

To the best of our knowledge, the first concrete attempt at describing document components by means of Semantic Web technologies is the *Semantically Annotated LaTeX* (*SALT*) project[18] [20,21]. SALT includes a set of ontologies for the description of the semantic organisation of documents according to three different layers: the structural layer (*Document Ontology*), describing sentences, paragraphs, figures, and the like; the rhetorical layer (*Rhetorical Ontology*), describing logical entities such as background knowledge, claims and evidence; and the annotation layer (*Annotation Ontology*) to link rhetorical characterisations with structural components.

Similar to the above, the *SWAN biomedical discourse ontology* [7] is a set of complementary OWL 2 DL ontologies that describe the discourse of scientific papers, with particular regard to the biomedical domain. The *Discourse elements ontology*[19] that forms

[8] BiRO: http://purl.org/spar/biro.
[9] C4O: http://purl.org/spar/c4o.
[10] PRO: http://purl.org/spar/pro.
[11] PSO: http://purl.org/spar/pso.
[12] PWO: http://purl.org/spar/pwo.
[13] SCoRO: http://purl.org/spar/scoro.
[14] FRAPO: http://purl.org/cerif/frapo.
[15] DataCite Ontology: http://purl.org/spar/datacite.
[16] DataCite schema: http://schema.datacite.org.
[17] BiDO: http://purl.org/spar/bido.
[18] Currently the SALT ontologies are not available at their original URLs, but we are informed that they will in future be hosted at http://nlp.uni-passau.de/vocab/salt. However, one can find the earliest versions of those ontologies at Linked Open Vocabularies (http://lov.okfn.org).
[19] The SWAN Discourse Elements Ontology: http://purl.org/swan/2.0/discourse-elements/.

part of SWAN allows one to characterise the parts of a text referring to claims, hypotheses, research questions and statements, while the relations among these and other document elements are defined in the *Discourse relationships ontology*[20] [5].

In [4], Ciccarese and Groza introduce the *Ontology of Rhetorical Blocks* (*ORB*)[21]. ORB is a model to describe large blocks of text (e.g., sections) in a rhetorical way, by capturing their logical roles within the whole scientific discourse of an article. In particular, the ontology defines seven different rhetorical blocks: one describing the front matter of the article (i.e., *orb:Head*), four blocks describing the major divisions of the body text (i.e., *orb:Introduction*, *orb:Methods*, *orb:Results*, and *orb:Discussion*), and two blocks referring to the back matter (i.e., *orb:Acknowledgements* and *orb:References*).

A detailed review and analysis of other RDF/OWL vocabularies and ontologies targeting the description of document components in terms of argumentative elements is presented by Schneider et al. in [35].

Other non-OWL proposals describing the structures that may be used in documents also exist. An example is the *Medium-Grained structure* [11] devised by the W3C Scientific Discourse Task Force, which offers a medium-grained description (hypothesis, objects of study, direct representation of measurements, etc.) of the rhetorical components of a document.

From a more syntactical point of view, Tannier et al. [38] associate each (XML) element in a document with one of three different categories: *hard elements* – elements that are commonly used to structure the document content in different blocks and usually interrupt the linearity of a text, such as paragraphs and sections; *soft elements* – elements that identify significant text fragments and are transparent while reading the text, such as emphasis and links; and *jump elements* – elements that are logically detached from the surrounding text, and that give access to related information, such as footnotes and comments.

Zou et al. [41] make Tannier et al.'s classification more extreme, defining only two categories of document elements: *inline* (those that do not introduce horizontal breaks) and *line-break* (those that do).

Finally, several XML vocabularies, which have been developed in the past years and which are currently used by scholarly publishers (e.g., the Elsevier Journal Article DTD[22], DocBook [40] and JATS [24]), define the most frequent structural components, such as sections, paragraphs, figures, tables, and the like. However, the same component is often expressed by different elements (e.g., a paragraph can be expressed using the elements *p*, *para*, or *par*) depending on the particular language in consideration.

Even if each of the aforementioned works proposes to model document components according to a particular perspective (e.g., structural vs. rhetorical, minimalistic vs. all-inclusive), a generic model harmonising all these aspects is still missing. DoCO is our tentative attempt to cover all these different perspectives, since it is an OWL model for describing all the extrinsic and intrinsic characterisations of document components.

3. Document components

There is an intrinsic difficulty in defining certain document components as purely rhetorical or purely structural. Even a well-known, easily identifiable component such as the paragraph cannot be considered as being strictly structural (i.e., carrying only a syntactic function), since it intrinsically carries rhetoric as well, through its natural language sentences. Paragraphs therefore have more than a syntactic function.

However, document markup languages often define a paragraph as a pure structural component, without any reference to its rhetorical function:

- "A paragraph is typically a run of phrasing content that forms a block of text with one or more sentences" [22];
- "Paragraphs in DocBook may contain almost all inlines and most block elements" [40][23].

The above definitions emphasise the structural connotation of the paragraph, that it "forms a block of text" or that it "contains" other elements, and this connotation is amplified by our direct experience as readers. It is the structural aspect that readily stands out in a book or webpage and that helps us, as readers, to distinguish a paragraph from the surrounding text. Yet this is insufficient for describing this element in

[20] The SWAN Discourse Relationships Ontology: http://purl.org/swan/2.0/discourse-relationships/.

[21] ORB – the Ontology of Rhetorical Blocks: http://purl.org/orb/.

[22] Elsevier XML DTDs and transport schemas: http://www.elsevier.com/author-schemas/elsevier-xml-dtds-and-transport-schemas.

[23] The words *inline* and *block* in these list items do not refer to the structural pattern theory introduced in the following section, although some sort of overlapping exist.

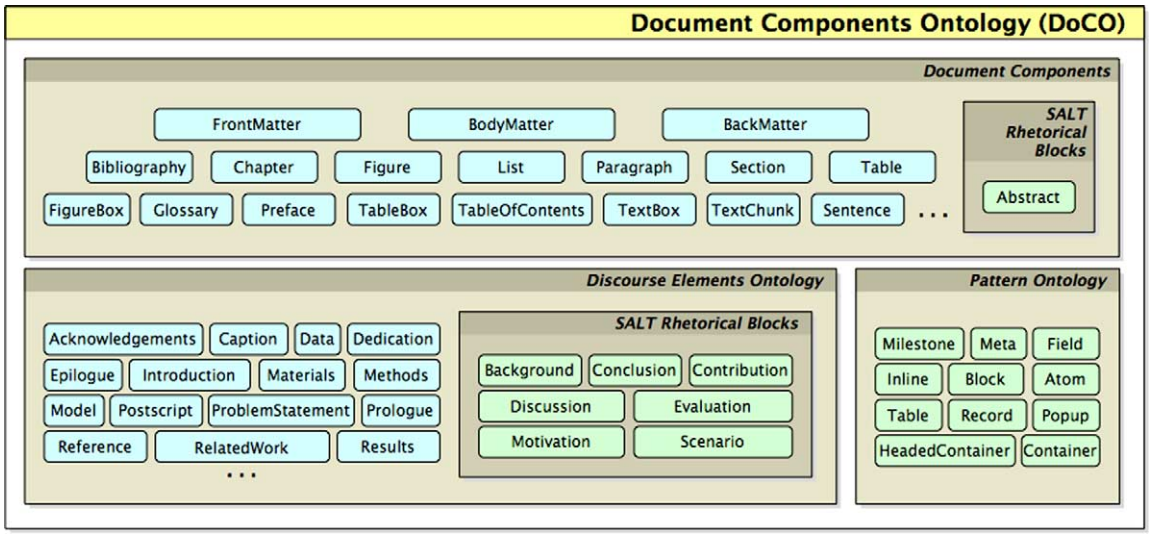

Fig. 1. Diagram describing the composition and the classes of the *Document Components Ontology* (*DoCO*). Note that only 22 of the 31 DEO classes are shown. For a full list of all the DEO classes and their definitions, see the ontology itself at http://purl.org/spar/deo.

its entirety. For instance, what is missed is the characterisation of a paragraph as a "self-contained unit of a discourse in writing dealing with a particular point or idea"[24], which mainly concerns the rhetorical nature of the paragraph rather than its structural/syntactical organisation as introduced by the aforementioned definitions.

The *Document Components Ontology* (*DoCO*) that we detail below has been developed so as to bring together the purely structural characterisations of document elements and their purely rhetorical connotations.

The creation of DoCO was undertaken by studying different corpora of documents (mainly scientific literature and web documents on different topics) and publishers' guidelines, from two perspectives – the structural and the rhetorical – as was also done by past works on document patterns [13–15]. We also undertook some informal interviews with researchers in different fields and with academic publishers, in order to gather as much information as possible about document components and their use. In addition, when developing DoCO and all its imported ontologies, we followed all the best practices already adopted in [5] and [6], which are directly inspired by the OBO Foundry Principles[25]. In particular, our ontologies:

- are open for use by all;
- possess a unique identifier space (namespace);
- are published in distinct successive versions;
- have clearly specified and delineated content;
- are orthogonal to other SPAR ontologies;
- include textual definitions for all terms;
- use relationships (object and data properties) that are unambiguously defined;
- strive to be well documented;
- are meant to serve a plurality of independent users;
- have been developed collaboratively.

DoCO imports the *Pattern Ontology* that describes structural patterns [14], and the *Discourse Element Ontology* (*DEO*)[26], which was developed with DoCO and describes rhetorical components. Additionally, it also defines hybrid classes describing elements that are both structural and rhetorical in nature, such as *paragraph*, *section* or *list*. A diagram describing the composition and classes of DoCO is shown in Figure 1.

In the next subsections we briefly introduce our theory of structural patterns as described in [14], and the rhetorical components that usually appear in scholarly articles, which represent the theoretical underpinnings of DoCO. Then, we introduce some of the document components of DoCO relevant for the description of scientific articles. We provide their formal definitions using DL formulas.

[24] Wikipedia article about "Paragraph": http://en.wikipedia.org/wiki/Paragraph.

[25] OBO Foundry Principles: http://www.obofoundry.org/crit.shtml.

[26] Discourse Elements Ontology: http://purl.org/spar/deo.

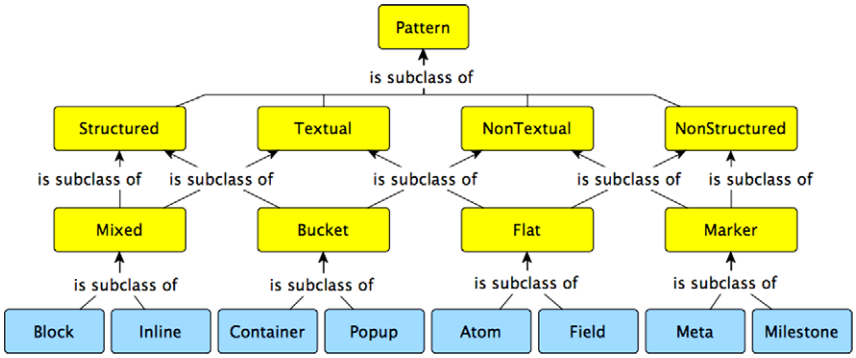

Fig. 2. A Graffoo diagram [17] showing the eight concrete patterns for document structures (bottom classes, in blue) and their relationships to high-level and abstract patterns (top classes, in yellow). (Color figure online)

3.1. Structural foundation: structural patterns

We have been investigating patterns of textual documents to understand how their structure can be segmented into atomic components that can be addressed independently and manipulated for different purposes. Instead of defining a large number of complex and diversified structures, in [13] we proposed a small number of structural *patterns* that are sufficient to express what most users need, characterised by two main aspects:

- *orthogonality* – each pattern needs to have a unique and specific purpose, fitting a specific context;
- *specificity* – each pattern can be used only in specific locations (e.g., within other patterns).

These patterns for textual documents were fully described in [14] and modelled as an OWL ontology called *Pattern Ontology*[27], which is summarised in Figure 2.

All the patterns are defined in terms of two main kinds of entities, themselves characterised by two different properties[28]: the possibility of containing text (*po:Textual*) or not (*po:NonTextual*, disjoint with the previous one), and the possibility of being organised in substructures (*po:Structured*) or not (*po:NonStructured*, disjoint with the previous one). These basic properties are thus combined in order to obtain four different disjoint classes describing entities that (A) contain both text and substructures (*po:Mixed*), (B) contain substructures but do not contain text (*po:Bucket*), (C) contain text but do not contain substructures (*po:Flat*), (D) do not contain text, nor substructures (*po:Marker*). Each of these four classes is a superclass to two other disjoint subclasses that collectively define the eight concrete patterns that can be used to characterise structures in text. A special case is that of the pattern *po:Container*, which is further split into three more specialised subunits, *po:Table*, *po:Record* and *po:HeadedContainer*.

These patterns are briefly introduced in Table 1. They facilitate the creation of unambiguous, manageable and well-structured documents. The regularity of pattern-based documents (defined by means of markup languages such as DocBook or LaTeX) then makes it possible to perform complex operations easily, even when knowing very little about the documents' markup vocabulary. This in turn enables designers to implement more reliable and efficient tools [14], make hypotheses regarding the meanings of document fragments [15], identify special cases, and study global properties of sets of documents [13].

3.2. Rhetorical foundation: discourse elements

The pure rhetorical characterisation of document components is not necessarily linked to the structural organisation that a scholarly article may have. For example, some scientific journals (such as the Journal of Web Semantics[29]) require their articles to follow a particular rhetorical segmentation, in order to identify explicitly what the meaningful parts are from a scientific point of view – e.g. Introduction, Background, Evalu-

[27] Pattern Ontology: http://www.essepuntato.it/2008/12/pattern.

[28] All prefixes are declared in http://www.essepuntato.it/2014/doco/prefixes.

[29] Journal of Web Semantics Guide for Authors: http://www.elsevier.com/journals/journal-of-web-semantics/1570-8268/guide-for-authors.

Table 1
Eight (plus three) structural patterns for descriptive documents

Pattern	Description	Example
po:Atom	Any simple box of text, without internal substructures, that is allowed in a mixed content structure but not in a container.	The various parts composing a free-text bibliographic reference of an article (title, source, etc.)
po:Block	Any container of text and other substructures except for (even recursively) other block elements.	A paragraph, a cell in a table
po:Container	Any container of a sequence of other substructures that does not directly contain text.	The body part of the article, a floating box containing a figure
po:Field	Any simple box of text, without internal substructures that is allowed in a container but not in a mixed content structure.	An e-mail address of an author specified in the front matter of an article
po:Inline	Any entity containing text and other substructures, including (even recursively) other inline elements.	An emphasis, an hyper-textual link
po:Meta	Any content-less structure (but data could be specified in attributes) that is allowed in a container but not in a mixed content structure.	A marker identifying the corresponding author of an article
po:Milestone	Any content-less structure (but data could be specified in attributes) that is allowed in a mixed content structure but not in a container.	A picture inserted in the body of the article
po:Popup	Any structure that, while still not allowing text content inside itself, is nonetheless found in a mixed content context and interrupts but does not break the main flow of the text.	A footnote, a comment
po:HeadedContainer (subtype of po:Container)	Any container starting with a head of one or more block elements. The pattern is usually employed to represent nested hierarchical elements as well as their headings.	A section or subsection of the article with its heading
po:Record (subtype of po:Container)	Any container that does not allow substructures to repeat themselves internally. The pattern is meant to represent database records with their variety of (non-repeatable) fields.	The set containing the metadata concerning the authors of the article (first name, family name, address, affiliation list, email, etc.)
po:Table (subtype of po:Container)	Any container that allows a repetition of homogeneous substructures. The pattern is meant to represent a table of a database with its content of multiple similarly structured records.	A table (as a sequence of ordered rows) or a list (as a sequence of ordered items) inserted in the body of the article

ation, Materials, Methods and Conclusion. These parts usually, but not necessarily, correspond to the coarse structural parts of the article – its sections. Whilst the background is usually woven together with the introduction, it may also be presented as a separate section, or indeed may substitute for the introduction entirely.

The characterisations of these purely rhetorical components, which are not always linked explicitly to a particular structure, are defined in the *Discourse Element Ontology* (*DEO*). DEO was developed according to the same principles followed for the creation of DoCO, i.e., by studying different corpora of scientific literature on different topics and publishers' guidelines. It provides a structured vocabulary for rhetorical elements within documents, enabling these to be described in RDF. The main class of this ontology is *deo:DiscourseElement*, which describes all those elements of a document that carry out a rhetorical function. All the remaining rhetorical behaviours are modelled as subclasses of this class. DEO reuses some of the rhetorical blocks from the SALT Rhetorical Ontology and extends them by introducing 24 additional classes, as partially shown in Figure 1, including the following eight:

- *deo:Reference*, which specifies a connection either to a specific part of the document or to another publication. In written text, numbered superscripts standing for footnotes, items in a table of contents, and items describing entities in a reference section, can be modelled as individuals of this class;
- *deo:BibliographicReference*, a subclass of the *deo:Reference* that describes references to other publications, such as journal articles, books, book chapters or websites; such references are often contained in a footnote or a bibliographic reference list;
- *deo:Caption*, that defines the text accompanying another item (e.g., the legend describing a picture);
- *deo:Introduction*, the initial description that states the purpose and goals of the subsequent text;

- *deo:Materials*, that documents the specific materials used in the described work;
- *deo:Methods*, that documents the methods used in the work (may be combined with a description of the materials used);
- *deo:Results*, that describes a report of the specific findings of an investigation;
- *deo:RelatedWork*, that describes a critical review of current knowledge by specific reference to other relevant works, both in terms of substantive findings and theoretical and methodological contributions within a domain of study;
- *deo:FutureWork*, a proposal for new investigations to be undertaken in order to continue and advance the work described in the publication.

Note that it is still possible to apply two different rhetorical characterisations to the same block of text. For instance, in journal articles it is common to have a section entitled "Materials and Methods", which can be characterised rhetorically by using both the classes *deo:Methods* and *deo:Materials*.

3.3. Hybrid structures within DoCO

In this subsection, we introduce those classes of DoCO that bring together both the purely structural elements of a document (i.e., the structural patterns introduced in Section 3.1) and generic rhetorical characterisations (i.e., the rhetorical components recounted in Section 3.2). We focus particularly on the structures that usually define the main components of scientific papers[30].

The class *Sentence* describes all those expressions in natural language forming single grammatical units. Usually, in written text, a sentence is terminated by major punctuation, such as a full stop, a colon, a semicolon, etc. It is defined in DoCO as follows:

```
Sentence ⊑ deo:DiscourseElement ⊓ po:Inline
```

A *paragraph* is a self-contained unit of discourse that deals with a particular point or idea, structured in one or more sentences. In written text, the start of a paragraph is indicated by beginning on a new line, which may be indented or separated by a small vertical space from the preceding paragraph. In DoCO, the class *Paragraph* is disjoint with *Sentence* and is modelled as follows[31]:

```
Paragraph ⊑
  deo:DiscourseElement ⊓ po:Block ⊓
  ∃po:contains.Sentence
```

A *footnote* is a particular structure that permits the author to make a comment or to cite another publication in support of the text, or both. A footnote is normally flagged by a superscript marker (e.g., a number) immediately following the portion of text to which it relates. For convenience of reading, the text of the footnote is usually printed at the bottom of the page or at the end of a text. The DoCO class *Footnote* is disjoint with the previous classes and is defined as follows[32]:

```
Footnote ⊑
  deo:DiscourseElement ⊓
  (po:Container ⊔ po:Popup)
```

A *table* is a set of data arranged in cells within rows and columns. From a pure structural pattern perspective, the element identifying the whole structure is organised according to the pattern *po:Table*, while those elements identifying the rows are always containers. The DoCO class *Table*[33] is disjoint with the previous classes and is defined as follows:

[30] As already mentioned, DoCO contains more classes than those described here in the text, to enable description of other kinds of bibliographic entities, such as books and poems, in addition to scientific articles. For a full list, see the ontology itself at http://purl.org/spar/doco.

[31] In this and the following description logic excerpts, we use some properties that are defined in imported ontologies. In particular, *po:contains*, and its inverse *po:isContainedBy*, are object properties defined in the Pattern Ontology that allow us to specify explicitly direct containment (i.e., parent-child) relations among pattern-based elements (in particular, those having type *po:Structured*). In DoCO, these two properties are defined as subproperties of *dcterms:hasPart* and *dcterms:isPartOf* respectively, which are used to express generic containment (i.e., ancestor-descendant) relations. Note that even if it is not explicitly stated in DoCO, we consider these DC Terms object properties to be transitive.

[32] Potentially there exist two different ways of organising footnotes, since their structural semantics can depend on the particular (markup) language we use to express it, as discussed in [15]. The first, is a container-based behaviour, as adopted by JATS [24], that allows one to specify footnotes (through the element *ft*) by using an element that is totally separated from the main text from which it is referenced (usually through XML attributes). The alternative is a popup-based behaviour, as used in LaTeX (by using the marker *footnote*{}), where a paragraph can be abruptly interrupted by one or more paragraphs specified in a footnote.

[33] Any table in DoCO is described as a po:Table that contains at least one po:Container, without referring explicitly to its rows, columns and cells. In the current version of DoCO, the explicit formalisation of these finer-grained elements was purposely avoided.

```
Table ⊑
  deo:DiscourseElement ⊓ po:Table ⊓
  ∃po:contains.po:Container
```

A *figure* is a communication object comprising one or more graphics, drawings, images, or other visual representations. In DoCO, it is disjoint with the previous classes and is modelled as a non-structured element without textual content, as introduced in the following definition:

```
Figure ⊑
  deo:DiscourseElement ⊓ (po:Milestone ⊔ po:Meta)
```

Commonly, in scientific publications, figures and tables are placed in captioned boxes (i.e., a *po:Container* containing a caption). The class *CaptionedBox* is disjoint with the previous classes and is defined as follows:

```
CaptionedBox ⊑
  deo:DiscourseElement ⊓ po:Container ⊓
  ∃dcterms:hasPart.deo:Caption
```

Captioned boxes can be used to define a space within a document that contains either a figure (i.e., *FigureBox*) or a table (i.e., *TableBox*) and its caption. These two classes are mutually disjoint and are defined respectively as follows:

```
FigureBox ⊑
  CaptionedBox ⊓ ∃dcterms:hasPart.Figure

TableBox ⊑
  CaptionedBox ⊓ ∃dcterms:hasPart.Table
```

A *list* is an enumeration of items, which may be paragraphs, author names, bibliographic references, etc., often delimited by distinct graphical symbols, either inline with the article text, or following a uniform spatial alignment. In DoCO, the class *List* is disjoint with the previous classes and is defined as follows:

```
List ⊑
  deo:DiscourseElement ⊓ po:Table ⊓
  ∃po:contains.po:Pattern ⊓
  ∀po:contains.((po:Container ⊓ ¬ (po:Table ⊔
    po:HeadedContainer)) ⊔ po:Field ⊔ po:Block)
```

This class is particularly useful to describe other, more specific kinds of lists: table of contents, list of figures, list of tables, etc. In particular, the class *BibliographicReferenceList* describes a list, usually within a bibliography, of all the references within the citing document that refer to articles, books, chapters, websites or similar publications. It is defined in DoCO as follows:

```
BibliographicReferenceList ≡
  List ⊓ ∀po:contains.deo:BibliographicReference
```

All above textual or graphical constructs are usually contained within broader elements that aim to describe the overall organisation of the document structure. First, we have the *front matter*, i.e., the initial principal part of a document, usually containing self-referential metadata. Although in a book it can be quite extensive, in a journal article the front matter is normally restricted to the title, authors and the authors' affiliation details, although the latter may alternatively be included in a footnote or in the back matter. The DoCO class *FrontMatter* is disjoint with the previous classes and is defined as follows:

```
FrontMatter ⊑
  deo:DiscourseElement ⊓ po:Container ⊓
  ∀po:isContainedBy.(¬ (BodyMatter ⊔ BackMatter))
```

Following the front matter, the *body matter* describes the central principal part of a document, that contains the core discourse of the work. The class *BodyMatter* is disjoint with the previous classes and is defined as follows:

```
BodyMatter ⊑
  deo:DiscourseElement ⊓ po:Container ⊓
  ∀po:isContainedBy.(¬ (FrontMatter ⊔ BackMatter))
```

The *back matter* is the final principal part of a document, usually comprising the bibliography, index, appendices, etc. Disjoint to both the previous classes, it is defined as follows:

```
BackMatter ⊑
  deo:DiscourseElement ⊓ po:Container ⊓
  ∀po:isContainedBy.(¬ (FrontMatter ⊔ BodyMatter))
```

The aforementioned elements are composed of other textual structures used for a coarse-grained and hierarchical organisation of text, such as *chapters* and *sections*. Both the classes *Chapter* and *Section* describe entities used for logically dividing the text, organised in paragraphs and possibly other (sub)sections, numbered and/or titled. While chapters and sections may contain (sub)sections, they cannot contain any other chapter. They are mutually disjoint and also disjoint with the previous classes, and are defined in DoCO as follows:

```
Chapter ⊑
  deo:DiscourseElement ⊓ po:HeadedContainer ⊓
  ∃po:contains.(Paragraph ⊔ Section) ⊓
  ∀po:contains.(¬ Chapter)

Section ⊑
  deo:DiscourseElement ⊓ po:HeadedContainer ⊓
  ∃po:contains.(Paragraph ⊔ Section) ⊓
  ∀po:contains.(¬ Chapter)
```

Articles normally, and even chapters sometimes, have particular kinds of sections that have a particular structural and rhetorical function, such as the *bibliography* or the *abstract*. The former contains a list of bibliographic references, and the related DoCO class *Bibliography* is defined as follows:

```
Bibliography ⊑
 (Section ⊔ Chapter) ⊓
 ∃dcterms:hasPart.BibliographicReference
```

The latter kind of section/chapter, defined by the class *sro:Abstract* imported from the SALT Rhetorical Ontology, describes a brief summary of a bibliographic entity, the purpose of which is to help the reader quickly ascertain the publication's purpose and points of focus. In DoCO, it is disjoint with *Bibliography* and defined as follows:

```
sro:Abstract ⊑
  (Section ⊔ Chapter) ⊓
  ∃dcterms:isPartOf.(FrontMatter ⊔ BodyMatter)
```

Sections and other high-level constructs such as chapters, captioned boxes or the document itself, can be introduced by a *title*. The DoCO class *Title* was introduced to describe a word, phrase or sentence that precedes and indicates the subject of a document or a document component. It is disjoint with the previous classes and is defined as follows:

```
Title ⊑
  deo:DiscourseElement ⊓ (po:Block ⊔ po:Field) ⊓
  ∃po:isContainedByAsHeader.po:HeadedContainer
```

Starting from the above definition, it is then easy to describe particular kinds of titles, such as *section titles* or *chapter titles* modelled as the title being part of a particular section/chapter:

```
SectionTitle ⊑
  Title ⊓ ∃po:isContainedByAsHeader.Section

ChapterTitle ⊑
  Title ⊓ ∃po:isContainedByAsHeader.Chapter
```

The following excerpt, written in Turtle [32], is an example of how DoCO may be used to describe some of the components characterising this article:

```
:paper a fabio:JournalArticle ;
  po:contains
    :front-matter , :body-matter , :back-matter ;
  co:firstItem [ co:itemContent :front-matter ;
    co:nextItem [ co:itemContent :body-matter ;
      co:nextItem [
        co:itemContent:back-matter ] ] ] .

:front-matter a doco:FrontMatter ;
  po:contains :title , :abstract ;
  co:firstItem [ co:itemContent :title ;
    co:nextItem [ co:itemContent :abstract ] ] .

:title a doco:Title ;
  c4o:hasContent
   "The Document Components Ontology (DoCO)" .

:abstract a sro:Abstract ;
  c4o:hasContent
   "The availability... scholarly articles." .

:body-matter a doco:BodyMatter ;
  po:contains :section-introduction ,
    :section-related-work , ... ;
  co:firstItem [
    co:itemContent :section-introduction ;
    co:nextItem [
      co:itemContent :section-related-work ;
      co:nextItem ... ] ] .

:section-introduction
  a doco:Section , deo:Introduction ;
  po:containsAsHeader
    :section-introduction-title ;
  po:contains :paragraph-1 , :paragraph-2 , ... ;
  co:firstItem [
    co:itemContent :section-introduction-title ;
    co:nextItem [ co:itemContent :paragraph-1 ;
      co:nextItem [ co:itemContent :paragraph-2 ;
        co:nextItem ... ] ] ] .

:paragraph-1 a doco:Paragraph ;
  po:contains :sentence-1 , :sentence-2 , ... ;
  co:firstItem [ co:itemContent: sentence-1 ;
    co:nextItem [ co:itemContent: sentence-2 ;
      co:nextItem ... ] ] .

:sentence-1 a doco:Sentence ;
  c4o:hasContent
   "One of ... scientific article or book." .
...
```

The main container (i.e., the paper) is described through FaBiO [29], while the order among the various components has been described by means of the Collections Ontology (CO)[34] [5]. The actual textual content of each component has been specified through the C4O property *c4o:hasContent* [12].

A more detailed version of this example, describing the paper in RDF according to DoCO, is available in [28].

4. Adoption and uses of DoCO

This section represents an evaluation of the uses of DoCO, made by listing its adoption in different application scenarios involving the works of different research groups. In particular, we discuss some relevant applications of DoCO in tools and algorithms for the annotation and processing of scholarly articles developed in the past years by two of our research groups, one at the University of Bologna, and another at the University of Manchester. In addition, at the end of this section, we briefly list other external works that con-

[34] The Collections Ontology: http://purl.org/co.

cretely use DoCO for different purposes within the Semantic Publishing community.

4.1. Processing scholarly articles: PDFX

PDFX[35] [8,9] is a rule-based system for analysing scientific publications in PDF form and recovering their fine-grained logical and rhetorical structures. Its analysis result is stored in an XML format that describes the document's organisation over logical units, and also links it to geometrical typesetting markers in the original PDF, such as column or page breaks. As of version 1.9, PDFX can differentiate 19 different element types. These types, given in Table 2, cover the principal parts of a typical research article.

The identified elements are ultimately stored in an XML file with a tag hierarchy that closely follows the ANSI/NISO Journal Article Tag Suite standard (JATS) [24]. The semi-structured nature of the XML serves as a quick and convenient access route to any of the article's components.

A "class" attribute has been added to each XML element in order to facilitate interoperability with other services. This attribute is derived from the tag given to an element in the identification stage, and is set in accordance with DoCO. This procedure facilitates aligning the structure recognition output of PDFX with the inputs that other text processing pipelines expect, thus adding a valuable metadata layer to the original publication. A multitude of different-purpose workflows can treat the PDF-to-DoCO-compliant-XML conversion as a pre-processing step, greatly widening their application domain in terms of accepted input.

4.2. Enhancing scholarly articles: Utopia Documents

Utopia Documents[36] [1] is a PDF-reader designed to improve the user's experience of reading scholarly papers (particularly in the domain of the Life Sciences) by linking the article and its contents to online resources.

DoCO provides a disciplined way for PDFX and Utopia Documents to interoperate. In particular, for any visualised PDF document, Utopia Documents runs the PDFX service in the background, using information about identified structural elements to provide additional user functionality. DoCO is used as a mechanism for tagging the output of PDFX and other Utopia Documents plugins in an interchangeable way; thus if plugins want to exchange tables/figures and references, they use DoCO annotations. Additionally, third-party plugins that are used for text mining can use the tagged structure to tune their behaviour as they pass through the document (e.g., some algorithms may want to include/exclude certain sections, or to become more or less sensitive, or to include/exclude captions or references during processing). For example, the mention of a particular gene or protein in the Introduction or Discussion sections of a paper is likely to have a very different meaning to the mention of it in the Materials and Methods section, where it is likely to be an "ingredient".

Table 2
The rhetorical element types that PDFX can differentiate

Front matter	Body matter	Back matter/others
Title	Body text	Bibliographic item
Author	(Sub)section	URI
Abstract	(Sub)section heading	Email
Author Footnote	Image	Side note
	Table	Header/Footer
	Caption	Page number
	Figure/Table reference	
	Bibliographic reference (in-text citation)	
	Labelled formula	

Utopia Documents works as follows. When a user opens an article, Utopia Documents uses PDFX to analyse the document's structure. DoCO *FrontMatter* features are used as search terms to identify the article in various online databases and tools, allowing Utopia Documents to display data such as Article Level or Alternative metrics, and to find entries in databases that cite the article as a whole. In the article's body, regions identified by PDFX and tagged as instances of *Image* or *Table* are converted into interactive objects allowing the user to browse the article by figures, or to export the data from tables in actionable numerical form. In the back matter, bibliographic references (i.e., *BibliographicReference* objects) are identified and linked to their in-text citation positions in the PDF document, enabling users to see the full bibliographic references of articles being cited at a particular location within the text, without the need to scroll to the reference section.

4.3. Retrieving structures from XML sources

Although the most frequently occurring structural components of documents are expressed in most XML

[35] The PDFX web service: http://pdfx.cs.man.ac.uk/.
[36] Utopia Documents – http://getutopia.com.

vocabularies used by scholarly publishers – e.g., the Elsevier Journal Article DTD, DocBook and JATS – they are often expressed by different elements. For instance, the element *para* in DocBook and the element *p* in JATS refer to the same concept of one of a set of vertically-organised containers of text often called *paragraph*. Starting from these bases, the services previously mentioned, such as table of contents generation or in-browser rendering, would need to be developed according to the peculiarities of each individual markup language. DoCO represents a generic model by which the semantics of any structural XML tag could be retrieved automatically, circumventing the need to write bespoke parsers for each encountered format.

In making steps towards addressing this issue, we have recently used DoCO as a theoretical base for the development of an ontology-aware algorithm to retrieve the meaning of markup structures in XML article sources [15], without explicitly looking either at the particular markup language used, or the actual content of the document. The algorithm was developed by starting from the actual specification of DoCO classes, and then tuned according to other statistical and topological principles (e.g. the frequency of markup elements, their position within the document, etc.)[37]. The final goal of the algorithm is to associate a particular DoCO class to each markup element used in these documents.

We performed a preliminary test (fully described in [15]) on a dataset consisting of 117 scientific papers encoded in DocBook and published between 2008 and 2011 in the Balisage Series Conferences[38]. The documents vary a lot in their internal structure and size: from 3 Kbytes to 160 Kbytes, with an average size of about 60 Kbytes. We compared the outcomes of the algorithm with a hand-crafted gold standard created by studying the XML vocabulary originally used to mark up the documents, and by associating each of its elements with one or more DoCO structures[39]. The overall results of this test were encouraging, since the overall values of precision and recall were quite high (0.887 and 0.890, respectively).

We are currently extending the algorithm in order to try to recognise additional DoCO components such as *Introduction*, *RelatedWork*, *Methods*, *Evaluation*, and *Conclusion*. For this, we are collecting a more comprehensive document test set of XML sources that will include articles from the PubMed Central Open Access Subset[40] and from Elsevier's Science Direct[41].

4.4. Community uptake

In addition to our work described in the previous sections, we list here some of the most important activities within the Semantic Publishing community that work with or reference DoCO, according to a bipartite classification: works that use DoCO for internal project goals, and works that discuss its use for modelling document components.

4.4.1. Adoptions of DoCO as part of existing works

Biotea The Biotea project [19] aims to convert scholarly documents into self-describing machine-readable formats on the basis of several ontologies developed for the publishing domain. As a first step, the authors processed all the XML sources contained in the PubMed Central Open Access Subset and converted them into RDF. DoCO was used to represent textual portions of the paper such as sections, paragraphs, figures and tables, and to link these portions to cited material.

Alighieri's Convivio Trying to develop mechanisms to represent the knowledge in the notes of Dante Alighieri's essay named *Convivio*, Bartalesi et al. [2] described a preliminary study to convert such notes (expressed in XML format) into RDF. Along the same lines as the Biotea project, the authors chose to use several ontologies to model the various aspects involved in the conversion, including DoCO to represent portions of the *Convivio*'s structure.

SLOR In [26,27], the authors introduce a tool that allows any researcher to create an open repository of research-relevant objects by adding semantic links between them according to specific RDF vocabularies and OWL ontologies. This repository, called *Semantic Linkages Open Repository* (*SLOR*), uses DoCO as

[37] The algorithm (fully introduced in [15]) is neither an intelligent nor an adaptive algorithm, but rather a prescriptive one that uses the logical characterisations of DoCO components as a basis to identify them in documents through an iterative process.

[38] Balisage Conference Series: http://www.balisage.net – all the data gathered during the test are available at http://www.essepuntato.it/2013/doco/test.

[39] We acknowledge that this analysis was subjective and solely based on our understanding of the semantics of the element, its definition schema and its documentation.

[40] PubMed Central Open Access Subset: http://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/.

[41] Science Direct: http://www.sciencedirect.com.

one of the main ontologies for the description of possible structural and taxonomical relationships between scholarly works.

4.4.2. Use of DoCO for modelling documents
Reviewing ontologies for scholarly documents In their work [34], Ruiz-Iniesta and Corcho review several ontologies according to three different contexts: document structure, scientific discourse and citations. As an outcome of their analysis, the authors recommend using DoCO for describing document structures, and one of its imported ontologies, DEO, for describing the majority of rhetorical elements.

HuCit *HuCit* is a light-weight ontology for the description of citation data (with a particular focus on the Humanities). In [33], its authors acknowledge the classes *doco:BibliographicReferenceList* and *deo:BibliographicReference* as components of one of the first RDF-based models to describe bibliographic references in scholarly articles.

Mathematical knowledge In his review article [23], Lange analyses which ontologies could be used to represent mathematical knowledge in form of RDF data. He includes a description of DoCO as a comprehensive way to represent structure and rhetoric of components in mathematical literature and publications.

ParlBench *ParlBench* [39] is an RDF benchmark that models digitally-published parliamentary proceedings and related actors, e.g., parliament members and political parties, from the Dutch legislation. DoCO is cited as one of the vocabularies that can be used to describe generic components of parliamentary documents.

5. Conclusions

In this paper we introduced *DoCO*, the *Document Components Ontology*. DoCO is currently one of the most widely used ontologies for the description of document components. It allows one to query, for example, all the bibliographic references cited in *Materials* sections of articles, or to retrieve all the sentences containing citations. Its viability as well as its usefulness have been demonstrated through its adoption by different research groups, some of which have been mentioned in Section 4.

Technically speaking, DoCO is a model that provides a general structured vocabulary of document components, based on our previous work on document patterns [14] and other existing works on the rhetorical characterisation of documents, such as [20,21]. DoCO was developed in order to be used in a complementary way with other ontologies describing different aspects of the publishing domain and scientific discourse. It can, for example, be used in conjunction CiTO to identify the specific sections, paragraphs, figures or tables to which a citation relates, instead of citing the paper as a whole. It can likewise be used with the SALT Rhetorical Ontology to explicitly characterise sentences or pieces of text as carrying a particular argumentative function.

In particular, in this article we formally described the DoCO components that most commonly appear within scientific articles, such as paragraphs, figures, tables, sections, chapters, references, front/body/back matters, and the like. In addition, we described tools and methods that use DoCO for different purposes, such as annotating PDF documents or retrieving the intended semantics of components of scholarly articles.

As future work, building from the encouraging results we obtained from our tests described in Section 4.3, we plan to refine the heuristics used in the algorithm for automated document component analysis, so as to increase the precision and recall for each element relative to the gold standard. We plan to extend the set of DoCO structures handled, to enable automated identification of other significant document components such as mathematical formulas, block quotes and front matter metadata (authors, affiliations, e-mail addresses for corresponding authors, etc.).

An initial mapping between DoCO and DocBook is already described in [15]. We plan to add additional mappings, for example to JATS metadata elements, in the near future.

In addition, we are working on extending the current implementation of PDFX in order to identify other document components, including those which are purely rhetorical (e.g., methods, materials, experiment, data, result, evaluation, discussion). All these components will have adequate DoCO annotations in the XML conversion outputs. Another future planned development for PDFX will concern the automatic conversion of all the structures retrieved and declared in the XML outputs into RDF according to DoCO and other relevant models, such as EARMARK [16] and SALT [21].

Acknowledgements

We would like to thank Angelo Di Iorio and Francesco Poggi for their support and contribution for

the development of the structural pattern theory summarised in Section 3.1, and for the algorithm for retrieving the structural characterisation of document components introduced in Section 4.3.

References

[1] T.K. Attwood, D.B. Kell, P. McDermott, J. Marsh, S.R. Pettifer and D. Thorne, Utopia documents: linking scholarly literature with research data, *Bioinformatics* **26**(18) (2010), i568–i574, Open Access (OA) at doi:10.1093/bioinformatics/btq383.

[2] V. Bartalesi, E. Locuratolo, L. Versienti and C. Meghini, A preliminary study on the semantic representation of the notes to Dante Alighieri's Convivio, in: *Proc. of the 2013 Workshop on Collaborative Annotations in Shared Environments: Metadata, Vocabularies and Techniques in the Digital Humanities (DH-CASE 2013)*, 2013, doi:10.1145/2517978.2517983.

[3] J. Beck, Report from the Field: PubMed central, an XML-based archive of life sciences journal articles, in: *Proc. of the International Symposium on XML for the Long Haul: Issues in the Long-term Preservation of XML*, 2010, OA at doi:10.4242/BalisageVol6.Beck01.

[4] P. Ciccarese and T. Groza, Ontology of Rhetorical Blocks (ORB), Editor's Draft, 5 June 2011, World Wide Web Consortium, 2011, OA at http://www.w3.org/2001/sw/hcls/notes/orb/ (last accessed 23/01/2015).

[5] P. Ciccarese and S. Peroni, The collections ontology: creating and handling collections in OWL 2 DL frameworks, *Semantic Web – Interoperability, Usability, Applicability* **5**(6) (2014), 515–529, doi:10.3233/SW-130121, OA preprint at http://www.researchgate.net/publication/256803144_The_Collections_Ontology_creating_and_handling_collections_in_OWL_2_DL_frameworks/file/60b7d523c704559261.pdf (last accessed 23/01/2015).

[6] P. Ciccarese, D. Shotton, S. Peroni and T. Clark, CiTO+SWAN: the Web Semantics of bibliographic references, citations, evidence and discourse relationships, *Semantic Web – Interoperability, Usability, Applicability* **5**(4) (2014), 295–311, doi:10.3233/SW-130098, OA preprint at http://speroni.web.cs.unibo.it/publications/ciccarese-2014-cito-swan-semantics.pdf (last accessed 04/02/2015).

[7] P. Ciccarese, E. Wu, G. Wong, M. Ocana, J. Kinoshita, A. Ruttenberg and T. Clark, The SWAN biomedical discourse ontology, *Journal of Biomedical Informatics* **41**(5) (2008), 739–751, OA at doi:10.1016/j.jbi.2008.04.010.

[8] A. Constantin, Automatic structure and keyphrase analysis of scientific publications, PhD thesis, The University of Manchester, UK, 2014, OA at https://www.escholar.manchester.ac.uk/uk-ac-man-scw:230124 (last accessed 04/02/2015).

[9] A. Constantin, S. Pettifer and A. Voronkov, PDFX: fully-automated PDF-to-XML conversion of scientific literature, in: *Proc. of the 2013 ACM Symposium on Document Engineering*, 2013, pp. 177–180, OA at doi:10.1145/2494266.2494271.

[10] A. De Waard, From proteins to fairytales: directions in semantic publishing, *IEEE Intelligent Systems* **25**(2) (2010), 83–88, doi:10.1109/MIS.2010.49.

[11] A. De Waard, Medium-grained document structure, 2010, OA at http://www.w3.org/wiki/HCLSIG/SWANSIOC/Actions/RhetoricalStructure/models/medium (last accessed 23/01/2015).

[12] A. Di Iorio, A.G. Nuzzolese, S. Peroni, D. Shotton and F. Vitali, Describing bibliographic references in RDF, in: *Proc. of 4th Workshop on Semantic Publishing*, CEUR Workshop Proceedings, Vol. 1155, 2014, OA at http://ceur-ws.org/Vol-1155/paper-05.pdf (last accessed 23/01/2015).

[13] A. Di Iorio, S. Peroni, F. Poggi and F. Vitali, A first approach to the automatic recognition of structural patterns in XML documents, in: *Proc. of the 2012 ACM Symposium on Document Engineering*, 2012, pp. 85–94, doi:10.1145/2361354.2361374, OA preprint at https://www.researchgate.net/profile/Silvio_Peroni/publication/256767265_A_first_approach_to_the_automatic_recognition_of_structural_patterns_in_XML_documents/links/00b49523c22a94d1fb000000.pdf.

[14] A. Di Iorio, S. Peroni, F. Poggi and F. Vitali, Dealing with structural patterns of XML documents, *Journal of the American Society for Information Science and Technology* **65**(9) (2014), 1884–1900, doi:10.1002/asi.23088, OA preprint at http://www.researchgate.net/publication/256803055_Dealing_with_structural_patterns_of_XML_documents/file/e0b49523c71f11b310.pdf (last accessed 23/01/2015).

[15] A. Di Iorio, S. Peroni, F. Poggi, F. Vitali and D. Shotton, Recognising document components in XML-based academic articles, in: *Proc. of the 2013 ACM Symposium on Document Engineering*, 2013, pp. 181–184, doi:10.1145/2494266.2494319, OA preprint at http://www.researchgate.net/publication/256795281_Recognising_document_components_in_XML-based_academic_articles/file/e0b49523c51c3f134c.pdf (last accessed 23/01/2015).

[16] A. Di Iorio, S. Peroni and F. Vitali, A Semantic Web approach to everyday overlapping markup, *Journal of the American Society for Information Science and Technology* **62**(9) 2011 1696–1716, in: Proc. of the 2013 ACM Symposium on Document Engineering: 181–184, doi:10.1002/asi.21591, OA preprint at http://palindrom.es/phd/wp-content/uploads/2010/07/jasist_earmark.pdf (last accessed 23/01/2015).

[17] R. Falco, A. Gangemi, S. Peroni and F. Vitali, Modelling OWL ontologies with Graffoo, in: *ESWC 2014 Satellite Events – Revised Selected Papers*, Lecture Notes in Computer Science, Springer, Berlin, Germany, 2014, pp. 320–325, doi:10.1007/978-3-319-11955-7_42, OA preprint at http://2014.eswc-conferences.org/sites/default/files/eswc2014pd_submission_114.pdf (last accessed 23/01/2015).

[18] A. Gangemi, S. Peroni, D. Shotton and F. Vitali, A pattern-based ontology for describing publishing workflows, in: *Proc. of the 5th International Workshop on Ontology and Semantic Web Patterns*, CEUR Workshop Proceedings, Vol. 1302, 2014, OA at http://ceur-ws.org/Vol-1302/paper1.pdf (last accessed 23/01/2015).

[19] L. Garcia Castro, C. McLaughlin and A. Garcia, Biotea: RDFizing PubMed Central in support for the paper as an interface to the Web of Data, *Journal of Biomedical Semantics* **4**(Suppl 1) (2013), S5, OA at doi:10.1186/2041-1480-4-S1-S5.

[20] T. Groza, S. Handschuh, K. Moller and S. Decker, SALT – Semantically Annotated LaTeX for scientific publications, in: *Proc. of the 4th European Semantic Web Conference*, 2007, pp. 518–532, doi:10.1007/978-3-540-72667-8_37, OA preprint at http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.116.5093&rep=rep1&type=pdf (last accessed 23/01/2015).

[21] T. Groza, K. Moller, S. Handschuh, D. Trif and S. Decker, SALT: weaving the claim web, in: *Proc. of the 6th In-*

ternational Semantic Web Conference, 2007, pp. 197–210. doi:10.1007/978-3-540-76298-0_15, OA preprint at http://siegfried-handschuh.net/pub/2007/salt_iswc2007.pdf (last accessed 23/01/2015).

[22] I. Hickson, HTML5: a vocabulary and associated APIs for HTML and XHTML, W3C Candidate Recommendation 29 April 2014, World Wide Web Consortium, 2011, OA at http://www.w3.org/TR/html5/ (last accessed 23/01/2015).

[23] C. Lange, Ontologies and languages for representing mathematical knowledge on the Semantic Web, *Semantic Web – Interoperability, Usability, Applicability* **4**(2) (2013), 119–158, doi:10.3233/SW-2012-0059.

[24] National Information Standards Organization, JATS: Journal Article Tag Suite, American National Standard No. ANSI/NISO Z39.96-2012, 9 August 2012, 2012, OA at http://www.niso.org/apps/group_public/download.php/10591/z39.96-2012.pdf (last accessed 23/01/2015).

[25] F. Osborne, S. Peroni and E. Motta, Clustering citation distributions for semantic categorization and citation prediction, in: *Proc. of the 4th Workshop on Linked Science (LISC 2014)*, CEUR Workshop Proceedings, Vol. 1282, 2014, OA at http://ceur-ws.org/Vol-1282/lisc2014_submission_9.pdf (last accessed 23/01/2015).

[26] S. Parinov, Open repository of semantic linkages, in: *Proc. of the 11th International Conference on Current Research Information Systems*, 2012, pp. 33–42.

[27] S. Parinov and M. Kogalovsky, Semantic linkages in research information systems as a new data source for scientometric studies, *Scientometrics* **98**(2) (2014), 927–943, OA preprint at http://sparinov.socionet.ru/files/parinov_kogalovsky-sem-link-data.doc (last accessed 23/01/2015).

[28] S. Peroni, Partial example of use of DoCO, figshare, 2015, OA at doi:10.6084/m9.figshare.1289776 (last accessed 23/01/2015).

[29] S. Peroni and D. Shotton, FaBiO and CiTO: ontologies for describing bibliographic resources and citations, *Web Semantics: Science, Services and Agents on the World Wide Web* **17** (2012), 33–43, doi:10.1016/j.websem.2012.08.001, OA preprint at http://smtp.websemanticsjournal.org/index.php/ps/article/viewFile/324/324 (last accessed 23/01/2015).

[30] S. Peroni, D. Shotton and F. Vitali, Scholarly publishing and linked data: describing roles, statuses, temporal and contextual extents, in: *Proc. of the 8th International Conference on Semantic Systems*, 2012, pp. 9–16, OA at doi:10.1145/2362499.2362502.

[31] S. Pettifer, P. McDermott, J. Marsh, D. Thorne, A. Villeger and T.K. Attwood, Ceci n'est pas un hamburger: modelling and representing the scholarly article, *Learned Publishing* **24**(3) (2011), 207–220, OA at doi:10.1087/20110309.

[32] E. Prud'hommeaux and G. Carothers, Turtle – Terse RDF Triple Language, W3C Recommendation, 25 February 2014, 2014, OA at World Wide Web Consortium, http://www.w3.org/TR/turtle/ (last accessed 23/01/2015).

[33] M. Romanello and M. Pasin, Citations and annotations in classics: old problems and new perspectives, in: *Proc. of the 2013 Workshop on Collaborative Annotations in Shared Environments: Metadata, Vocabularies and Techniques in the Digital Humanities*, 2013, doi:10.1145/2517978.2517981, AO preprint at http://phd.mr56k.info/files/romanello-pasin_dhcase2013.pdf (last accessed 23/01/2015).

[34] A. Ruiz-Iniesta and O. Corcho, A review of ontologies for describing scholarly and scientific documents, in: *Proc. of 4th Workshop on Semantic Publishing (SePublica 2014)*, Aachen, Germany, CEUR Workshop Proceedings, 2014, CEUR-WS.org, OA at http://ceur-ws.org/Vol-1155/paper-07.pdf (last accessed 23/01/2015).

[35] J. Schneider, T. Groza and A. Passant, A review of argumentation for the Social Semantic Web, *Semantic Web – Interoperability, Usability, Applicability* **4**(2) (2013), 159–218, OA at doi:10.3233/SW-2012-0073.

[36] D. Shotton, Semantic Publishing: the coming revolution in scientific journal publishing, *Learned Publishing* **22**(2) (2009), 85–94, OA at doi:10.1087/2009202.

[37] D. Shotton, K. Portwin, G. Klyne and A. Miles, Adventures in Semantic Publishing: exemplar semantic enhancements of a research article, *PLoS Computational Biology* **5**(4) (2009), e1000361, OA at doi:10.1371/journal.pcbi.1000361.

[38] X. Tannier, J. Girardot and M. Mathieu, Classifying XML tags through "reading contexts", in: *Proc. of the 2005 ACM Symposium on Document Engineering*, 2005, pp. 143–145, doi:10.1145/1096601.1096638, OA preprint at http://perso.limsi.fr/xtannier/Publications/files/Tannier_DocEng05.ps.gz (last accessed 23/01/2015).

[39] T. Tarasova and M. Marx, ParlBench: a SPARQL benchmark for electronic publishing applications, in: *ESWC 2013 Satellite Events – Revised Selected Papers*, 2013, pp. 5–21, doi:10.1007/978-3-642-41242-4_2, OA preprint at http://ceur-ws.org/Vol-981/BeRSys2013paper2.pdf (last accessed 23/01/2015).

[40] N. Walsh, DocBook 5: the definitive guide, O'Really Media, Sebastopol, CA, USA, Version 1.0.3., 2010, ISBN: 0596805029.

[41] J. Zou, D. Le and G.R. Thoma, Structure and content analysis for HTML medical articles: a hidden Markov model approach, in: *Proc. of the 2007 ACM Symposium on Document Engineering*, 2007, pp. 199–201, doi:10.1145/1284420.1284468, OA preprint at http://www.lhncbc.nlm.nih.gov/files/archive/pub2007082.pdf (last accessed 23/01/2015).