ALMA MATER STUDIORUM – UNIVERSITÀ DEGLI STUDI DI BOLOGNA
FACOLTÀ DI SCIENZE MATEMATICHE FISICHE E NATURALI
Dipartimento di Informatica: Scienza e Ingegneria

Corso di Laurea Specialistica in Informatica

Semantic Enrichment of Scientific Documents with Semantic Lenses – Developing methodologies, tools and prototypes for their concrete use

Relatore:
Chiar.mo Prof. Fabio Vitali
Correlatore:
Dott. Silvio Peroni

Candidato:
Jacopo Zingoni

ANNO ACCADEMICO 2011-2012
Sessione II

A Marta e Stefano,
i miei amatissimi ed esemplari genitori,
straordinari, dolcissimi e sempre pronti a tutto.
A Giuliana ed a Tosca,
indimenticabili nonne, sempre vicine al mio cuore,
delle cui lezioni spero di poter fare sempre tesoro.

Grazie per tutto l'appoggio, per ogni parola di incoraggiamento.
E soprattutto grazie per tutto l'amore che mi avete sempre donato.

Indice

Introduzione

Con questa dissertazione di tesi miro ad illustrare i risultati della mia ricerca nel campo del Semantic Publishing, consistenti nello sviluppo di un insieme di metodologie, strumenti e prototipi, uniti allo studio di un caso d'uso concreto, finalizzati all'applicazione ed alla focalizzazione di Lenti Semantiche (*Semantic Lenses*): un efficace modello per l'arricchimento semantico di documenti scientifici [PSV12a].

Il Semantic Publishing è un approccio rivoluzionario nel modo in cui i documenti scientifici, come ad esempio degli articoli di ricerca, possono essere letti, usati, indirizzati e diventare oggetto di nuovi modi di interazione.

Ma in cosa consiste di preciso il Semantic Publishing? Per definirlo con le stesse parole del suo principale proponente:

"Definisco come Semantic Publishing tutto ciò che aumenta la resa del significato di un articolo scientifico pubblicato, che ne facilita la sua scoperta in modo automatizzato, che consente di collegarlo ad articoli semanticamente correlati, che fornisce accesso a dati presenti nell'articolo, in modo azionabile, oppure che facilita l'integrazione di dati fra diversi documenti. Fra le altre cose, richiede l'arricchimento dell'articolo con metadati appropriati, comprensibili, analizzabili e processabili automaticamente, in modo da consentire un miglioramento della verificabilità delle informazioni presenti nella pubblicazione, ed al fine di provvedere ad un loro riassunto automatico, o la loro scoperta automatica da parte di altri agenti." [SKM09]

Il lavoro che ho svolto e che tratterò nelle pagine seguenti è quindi un contributo completo al campo del Semantic Publishing. Innanzitutto è un modo di mostrare la fattibilità ed i vantaggi del modello delle Lenti Semantiche ai fini di un appropriato arricchimento con metadati, tramite la proposta di una metodologia dettagliata per il raggiungimento di questo obiettivo. È una indicazione di una possibile via per risolvere le sfide che si potrebbero incontrare lungo questo percorso, sviluppando gli appropriati strumenti e le soluzioni praticabili. Ed è una dimostrazione pratica di alcune delle nuove interazioni ed opportunità rese possibili da un prototipo di interfaccia generato a partire da un documento scientifico arricchito tramite l'appropriata annotazione delle lenti su di esso.

La mia dimostrazione si basa appunto sulle Lenti Semantiche, che sono un modello per l'arricchimento semantico di documenti scientifici, accompagnato da un insieme di tecnologie raccomandate per la sua implementazione. È importante osservare come l'arricchimento di un tradizionale articolo scientifico non sia una operazione monodimensionale, in quanto, al di là del mero atto di aggiungere asserzioni semanticamente precise riguardo il contenuto testuale, sono coinvolte in essa molte altre sfaccettature. Tutti questi aspetti che coesistono simultaneamente in un articolo possono essere definiti nella concreta manifestazione della semantica di un documento tramite l'applicazione di specifici filtri, in grado di enfatizzare un preciso insieme di informazioni significative su un dominio piuttosto che su un altro: dalla struttura retorica, all'intento di una citazione bibliografica, o fino ad un modello che definisca esplicitamente le tesi all'interno di una argomentazione. Immaginiamo di poter essere in grado di scegliere fra una specifica collezione di lenti semantiche, ognuna di essa in grado di mettere a fuoco l'oggetto della nostra osservazione in un modo differente, mettendo in evidenza un preciso sottoinsieme di qualità e significati rispetto al resto del documento.

Nel contemplare un sistema del genere, ci sono due ovvie operazioni da portare a termine per renderlo pienamente funzionale. La prima è l'**applicazione** dei metadati associati ad una di queste specifiche lenti semantiche sull'articolo. L'altra è la **focalizzazione**, da parte del lettore, di una delle lenti selezionate sull'articolo stesso, in modo da favorire l'emergere dell'insieme di significati legati al sottoinsieme selezionato, e consentire a questi di venire alla luce, possibilmente in un modo interattivo.

Ho scelto di espandere la mia indagine oltre lo studio di una metodologia teorica, verso lo sviluppo di strumenti adeguati in grado di assistere nell'uso delle Lenti, ed ho infine optato per testare l'intero concetto di Lenti Semantiche mettendo questi principi in azione: Per prima cosa, **applicando** concretamente alcune delle lenti proposte su un documento (dopo aver sviluppato gli strumenti per farlo in modo appropriato), esaminandone poi i risultati ed infine mostrando alcune delle possibili applicazioni ed interazioni risultanti dalla **focalizzazione** di queste lenti tramite un prototipo di interfaccia.

Di conseguenza, ho selezionato un articolo conosciuto come oggetto dei miei test. La scelta è ricaduta sulla versione HTML di "*Ontologies are us*" di Peter Mika [Mik07] (un lavoro molto importante sulle *folksonomie*, ontologie emergenti da contesti sociali), che ho convertito nel formato EARMARK [PV09] per ragioni implementative, cosa che mi ha consentito di sfruttarne le peculiarità nella gestione dell'*overlapping markup* [DPV11a]. Dopo aver selezionato l'appropriato insieme di tecnologie web e di ontologie, in accordo con i suggerimenti del modello delle Lenti Semantiche, ho studiato una metodologia generale di tipo **bottom-up** – **SLM** o "***S**emantic **L**enses **M**ethodology*" – focalizzata sull'**applicazione** di quattro specifiche lenti semantiche (*Strutturale, Retorica, Citazionale* ed *Argomentativa*). Ho successivamente proseguito il mio lavoro con l'annotazione, sul documento bersaglio, dei metadati appropriati relativi a queste lenti, tramite statement RDF, prima sviluppando un package Java – **SLAM** or "***S**emantic **L**enses **A**pplication and **M**anipulation*" – che mi consentisse di effettuare le operazioni richieste dalla metodologia in modo adeguato. SLAM offre funzionalità aggiuntive rispetto alle API Java di EARMARK[1] sulla base delle quali è stato costruito, e mira ad essere la prima base per la costruzione di un insieme di strumenti che possano essere riutilizzabili da chiunque abbia interesse a replicare o estendere la metodologia che suggerisco.

Dopodiché, ho provveduto a scrivere le annotazioni stesse, non manualmente, ma tramite una serie di istruzioni processate dalla sopracitata implementazione Java, emulando l'attività autoriale (e co-autoriale) dell'arricchimento documentale nell'ottica di mantenerne la correttezza semantica, finalizzando le decisioni in tal senso allo scopo di tradurre il significato percepito dal contenuto in modo da aderire il più possibile sia alla metodologia proposta che ai requisiti del modello delle Lenti Semantiche. Infine, dopo aver analizzato i risultati del lavoro, nonché i vari possibili vantaggi che possono essere ottenuti tramite l'arricchimento di un documento tramite Lenti Semantiche, ho generato un primo prototipo di una interfaccia basata su una pagina HTML, arricchita con JQuery[2], generata tramite Java. Il prototipo di questa UI – **TAL** o "***T**hrough **A** **L**ens*" – consente all'utente di effettuare alcune semplici attività di **focalizzazione** nell'ambito intra-documentale, e mostra la loro utilità in uno scenario concreto.

Questa dissertazione è così strutturata: Nella sezione 2 introdurrò il dominio generale e l'ambito di ricerca in cui questo lavoro si colloca, assieme alle nozioni di Semantic Web e Semantic Publishing, sotto una prospettiva generale e storica, contestualizzando le scoperte scientifiche nello stesso ambito ed altri lavori correlati. Nella sezione 3 discuterò il contesto tecnologico per questa dissertazione, e fornirò una breve rassegna delle tecnologie e delle ontologie accessorie a questo lavoro. Nella sezione 4 sarà presente una esposizione molto più dettagliata sul modello delle Lenti Semantiche e sulle ontologie ad esse correlate.

Segue la sezione 5 con i dettagli della metodologia SLM che ho ricercato e scelto di adottare per svolgere questa prova finale. Le sezioni 6 e 7 conterranno, rispettivamente, informazioni e documentazioni sul design, sullo sviluppo e sull'implementazione di SLAM e di TAL. Nella sezione 8 osservo alcuni dei risultati relativi all'applicazione concreta tramite SLAM su [Mik07], nonché la generazione ed user-test del TAL generato a partire da questi risultati, ottenendo così un caso di studio concreto per l'uso delle lenti e per

[1] S. Peroni, EARMARK API: http://earmark.sourceforge.net/

[2] JQuery: http://jquery.com/

l’applicazione della metodologia e degli strumenti precedentemente illustrati. In essa discuto i risultati di questa attività di applicazione, presento dati statistici raccolti durante questo intero progetto e riassumo l’esperienza ottenuta con questa attività, osservando infine i risultati dei test utente eseguiti su TAL. Un breve riassunto delle opportunità di sviluppo future troverà spazio assieme alle conclusioni finali.

1 Introduction and Aims of this Work – Applications for Semantic Lenses

With this thesis dissertation I aim to illustrate the results of my research in the field of Semantic Publishing, consisting in the development of a set of methodologies, tools and prototypes, accompanied by a case study, for the application and focusing of Semantic Lenses [PSV12a] as an effective mean to semantically enrich a scholarly paper.

Semantic Publishing is a revolutionary approach in the way scientific documents, such as research articles, can be read, perused, reached and interacted with. But what are the characteristics of Semantic Publishing? Allow me to define it in the very own words of his first proponent:

"I define semantic publishing as anything that enhances the meaning of a published journal article, facilitates its automated discovery, enables its linking to semantically related articles, provides access to data within the article in actionable form, or facilitates integration of data between papers. Among other things, it involves enriching the article with appropriate metadata that are amenable to automated processing and analysis, allowing enhanced verifiability of published information and providing the capacity for automated discovery and summarization." [SKM09]

The work I have done and I am going to show in the following pages is then, according to this definition, a full contribution to the field of Semantic Publishing. It is an exposition on the feasibility and the advantages of the Semantic Lenses model for appropriate metadata enrichment, together with the proposal of a detailed methodology for their application. It is a path to overcome the challenges we are likely to encounter in this process, by developing the appropriate tools and solutions. And it is a showcase for the new interactions and knowledge discovery opportunities enabled through a basic UI prototype generated from the appropriate annotations of semantic lens on an enriched paper.

I am basing my demonstration on Semantic Lenses, a model for the semantic enhancement of scientific papers, accompanied by a set of suggested technologies for its implementation [PSV12a]. It is important to observe that the enhancement of a traditional scientific article is not a straightforward operation, as there are many aspects involved besides the mere act of making semantically precise statements about its content. All these different facets that coexist simultaneously within an article can be defined in the semantic rendering of the paper by applying specific filters emphasizing a specific set of meaningful information, which might be about its rhetorical structure, or the purpose of a citation, or a way to explicitly define the claims of an argumentation. Imagine then being able to choose within a set of *semantic lenses*, each one allowing the user to focus the object of his observation in a different way, magnifying a selected subset, with its qualities and meanings, rather than others.

In envisioning such a system, there are two obvious operations involved to make it fully functional. One is the **application** of the metadata associated with a specific semantic lens over the article. Then there is the **focusing,** by the reader**,** of a selected lens over the article, making the chosen set of metadata emerge and putting it in the forefront, possibly in an interactive way.

I have chosen to expand my investigation from a theoretical methodology to the development of adequate tools to assist in the use of Lenses, and I also opted to field-test the whole Semantic Lenses concept by putting these principles into action: first by concretely **applying** some of the proposed lenses on a document (and developing the appropriate means to do so), then by examining the results and finally by showing some of the possible applications and interactions resulting from the **focusing** of those applied lenses.

Consequently, I have selected a known paper as the object for of my tests, which is the HTML version of Peter Mika's "*Ontologies are us*" [Mik07] (a very important work on *folksonomies*, ontologies emerging from social contexts), which I converted into the EARMARK format [PV09] for implementation purposes, allowing me to use its peculiarities for handling overlapping markup [DPV11a].

After choosing the appropriate set of web technologies and ontologies according to those suggested by the definition of Semantic Lenses, I studied a general **bottom-up** methodology – **SLM** or "***S****emantic* ***L****enses* ***M****ethodology*" – in order to specify a way to concretely apply four specific semantic lenses (*Document Structure, Rhetoric Organization, Citation Network and Argumentation*).

I then proceeded to annotate the appropriate metadata for those lenses on the whole document through RDF statements, first by developing an adequate Java package – **SLAM** or "***S****emantic* ***L****enses* ***A****pplication and* ***M****anipulation*" – which allowed me to perform the operations required. SLAM offers extended functionalities for the EARMARK Java API[3] on which it has been built on, aiming to be the first foundation to create a set of tools which will then be re-usable by anyone willing to replicate or improve the methodology I suggested.

After that, I went on by writing the annotations themselves as a series of instructions to be processed by said Java implementation, emulating the

[3] S. Peroni, EARMARK API: http://earmark.sourceforge.net/

authorial and co-authorial task of enriching this document in a semantically correct way. In order to reach this goal, my final decisions on how to best translate the perceived meaning of the content were based on finding a way adhering both to the methodology I proposed and to the requirements of the Semantic Lenses approach. Finally, after analyzing several possible advantages that might be brought with the enrichment of a document through Semantic Lenses, I created a basic prototype of an HTML-based, JQuery[4] enhanced, Java-generated interface – **TAL** or "***Through A Lens***" – capable of performing some basic focusing of Semantic Lenses and some of their possible intra-document applications, showing their usefulness in a real-case scenario.

This document is structured as follows: In section 2 I will introduce the general domain of the problem that this work addresses, as well as the notions of Semantic Web and Semantic Publishing in general and in an historical perspective, contextualizing scientific advancement facing the same issues and other related works. In section 3 I will discuss the technological context for this dissertation, and give a brief review of the technologies and onthologies used in this work. In section 4 there will be a much more detailed explanation of what Semantic Lenses are, together with their related vocabularies. Section 5 follows with the details of the methodology I have researched and chosen to adopt for this thesis. Section 6 and 7 contain, respectively, information and documentation about the design, development and implementation of SLAM and TAL.

In section 8 I observe on the results of testing SLAM and TAL over [Mik07], obtaining a concrete case study for the effectiveness of lenses and applying the methodology I previously detailed. I discuss the results for the application of lenses, present statistical data collected for the whole project, I summarize the experience obtained from this activity and I observe on the outcomes of test executed on TAL. A short summary of future opportunities for development and of the advantages of a widespread and methodical adoption of Semantic Lenses (as a methodology and a set of technologies), is located, together with the final conclusions, in section 9.

[4] JQuery: http://jquery.com/

2 Scientific Context and Related Works

2.1 General introduction to Semantic Web and Semantic Publishing

Words, in all their beauty and heterogeneity, are the fundamental language unit through which human beings communicate. But, as it often happens, the little, primal things we usually take for granted are very far from being the simplest notions to wrap our minds around. As John Locke wisely put, words are not just "*regular marks of agreed notions*"[5], but "*in truth are no more but the voluntary and unsteady signs of (men's) own ideas.*" [1] And indeed, to keep quoting him, *"So difficult it is to show the various meanings and imperfections of words, when we have nothing else but words to do it with"* [1] – an excellent summary of our everyday quest to correctly comprehend ideas, experience and intentions being communicated by others.
Consider the simple act of saying out loud something as simple as "*Good morning!*" - If your interlocutor is feeling especially witty it might reply: *"Do you wish me a good morning, or mean that it is a good morning whether I want it or not; or that you feel good this morning; or that it is a morning to be good on?"*[6]
Being able to extract meaning (correctly, if possible) from communications received is part of what information science is all about. This daunting but often unconscious daily task our mind is so adept at performing becomes harder and more deliberate when we consider a written text, especially one debating on a complex subject whose author we might not be familiar with, as it can be the case for a scholarly scientific publication. If we envision the act of examining a piece of written text, there are many different approaches we can take, in as many different contexts, to extract significance from it. Indeed, "*meaning is not embedded within words, but rather triggered by them*" [DeW10], and takes shape according to the aspects we are considering the most important at the moment of our examination. And we do this not just once, but many times, for all those different contexts part of natural language, until in our minds we are satisfied with a multi-dimensional representation of the information we processed.
Although this may seem overly complicated at first, in practice it is a process which we often apply not just for the interpretation of written text, but for

[5] J. Locke (1689); *An Essay Concerning Human Understanding.*
[6] J.R.R. Tolkien (1937); *The Hobbit.*

everyday human interactions with reality, from grand ideas to the most mundane of items: We can relate to something as widespread as a modern Smartphone in many ways – thinking about it as a medium for communication, a recording device, a mobile entertainment system, a storage for important contacts, a keepsake of memories, a manufactured technological object, a consumer good, a status symbol, or even a badge of affiliation, and so on…

"In other words, this meaning is not contained within the words themselves, but in the minds of the participants" [DeW10]. To better clarify this concept, let us contemplate this very piece of text. We might consider its structure, and say that it is a section of text inside a document, made of paragraphs, organized in sentences, which might be modeled as inline components of the text, intermixed with some internal references to other part of this document. But we can also dwell on the rhetorical aspect of this text. We might say that this is an introduction for the contents of this document, with parts where a problem requiring a solution is stated and some other parts where the author is explaining the motivation behind this authorial effort. And since we are thinking about authors, one might be interested in knowing more about this topic, perhaps in discovering which people played which part in creating this document. And what about whole the document itself, or the data behind it? We might be interested in gathering information on the research context that originated this document, or to find if this is the only Manifestation of an authorial Work, or its possible publication status.

Then again, switching back to the text on these pages, we might notice that some of the sentences (visually characterized in a different ways than others) appear to be quotes and citations. Why are these other authors quoted, and what was the purpose behind each of these citations? In short, how are these citations handled by the author? They might represent a foundation for the expansion of a discourse, they might be called in as examples or as a source of background information, or they might be supporting evidence for a thesis. Speaking of which, the reader will at some point focus on the actual meaning of this text. What are the claims being made by the author? We might inspect the argumentation model used to state and assert these claims, and try to parse between the sentences that constitute what is being argued and those that, for instance, make up the evidence sustaining these assertions, or those logical warrants that acts as a bridge between those two. Finally, to obtain anything useful from a written communication, the recipient must be able to assign some actual meaning to the words themselves, to understand if they refer to specific entities or definitions, and to relate with those "unsteady ideas" that the author had originally in mind.

And here we are again, right at the heart of our problem, but "*Coming back to where you started is not the same as never leaving.*"[7].

Thus we are now more familiar with one of the challenges that the multi-faceted discipline of *Semantic Publishing* [SKM09] is trying to tackle, with a combined effort aimed at improving the effectiveness of written communication (especially scholarly journal articles), enhancing the meaning of a published scientific document by providing a large quantity of information about it as machine-readable metadata, facilitating its automated discovery, querying or integration [Sho09].

It should also be clearer why the enhancement of a traditional scientific article is not a straightforward operation, and, we should have a first glimpse on how many different semantically relevant aspects coexist within a (scientific) document, like facets of a gem, even if so far I have sketched them only informally. These aspects are all subtly interlinked and yet each one is adding its specific contribution to our final understanding of the overall meaning of the document.

[7] T. Pratchett (2004); *A Hat full of Sky*.

2.2 Semantic web, Semantic Publishing and the Enrichment of knowledge

The idea of semantic publishing is but the latest addition to a long-standing prolific cooperation between web technologies aimed at content classification and distribution, scientific research in general and publishing activities; and it is one dating back to the inception of the web itself [BCL94], and now growing even more quickly with the widespread popularity of xml-based languages and technologies, (such as DocBook or XHTML), online paid content distribution systems, mobile e-reader, wireless connections, and, most importantly, with the growth of *Semantic Web.*

Semantic web, as envisioned by Tim Berners-Lee more than 10 years ago [Ber01], was described as a way to bring structure to the meaningful content of web pages, by extending the traditional web (and not substituting it) in a way that could allow newer, better, machine-readable definitions of information and their meaning. This idea soon had an explosive growth, both in popularity and in different definitions, backed by the swift development and evolution of the technologies behind it, such as RDF, OWL or SPARQL, to the point that we have dozens of different way to describe this evolution. In general, the semantic web initiative aims to represent web content in a form that is more easily machine-processable [AH04], by building a web of data with common formats for integration and combination of said data, and defining formal languages and technologies to record the meaning of this data, allowing the user to leap seamlessly from a set of information to another[8].

Berners-Lee himself gave two other interesting definitions on what is becoming the Semantic Web, first observing that it resembles a Giant Global Graph [Ber07] on which data of all kind, whether social or scientific, are connected in meaningful relationships, discoverable and re-usable, allowing us *"to break free of the* [single] *document layer".* He also argued that a great number of Semantic Web patterns have a fractal nature [BK08] , much like human language and the way we already classify knowledge, and strongly advocated the development of web systems made of overlapping specialized communities of all size, interlinked (with the help of properly designed ontologies) both to other communities and made of sub-communities themselves.

The last few years saw the great success of one of the main Semantic Web initiatives, Linking Open Data. At its core, LOD is a collaborative community project, sponsored by the W3C, whose aim is to extend the traditional web by

[8] W3C Website (2012), Introduction to the Semantic Web Activity: http://www.w3.org/2001/sw/

encouraging the publication of open, RDF-enriched sets of machine-readable data in a way adhering to four basic principles [Ber06], together with a standard set of technologies and best practices. Here are those principles as of their latest formulation [HB11]:

1. Use URIs as names for things.
2. Use HTTP URIs, so that people can look up those names.
3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).
4. Include links to other URIs, so that they can discover more things.

The goal is to use the Web to connect data that were not previously linked, to do so in a structured, typed way, and to lower the barriers between these datasets, so that their meaning is explicitly defined. Or, to quote [HB11], *"In summary, Linked Data is simply about using the Web to create typed links between data from different sources"*. The success of Linked Data as an application of the general architecture of the World Wide Web to the task of sharing structured data on a global scale can be best summarized by the fact that the amount of data involved almost doubled every year, from the already impressive 6,7 billions of RDF triples of July 2009 to the 32 billion triples as of September 2011. With the resulting web of data based on standards and a common data model, it becomes possible to implement applications capable to operate on this interlinked data graph.

There are a lot of analogies with the classic Web, like having data-level links connecting data from different sources into a single global space, much like it is done in the World Wide Web. And, just like the "traditional" WWW, data is self-describing, anyone can publish data on the LOD. However, this web of Linked Open Data is based on standards for the identification, retrieval and representation of data. This opens up the chance to use general purpose standardized data browsers to explore the whole giant global space, and, considering that well-structured data from different sources is linked in a typed way, all kinds of data fusion from different sources become possible, and operations such as queries can be done on this aggregated data. Not only that, but data sources can be discovered at runtime by crawlers simply following the data-level link, allowing for a far greater depth in delivering answers.

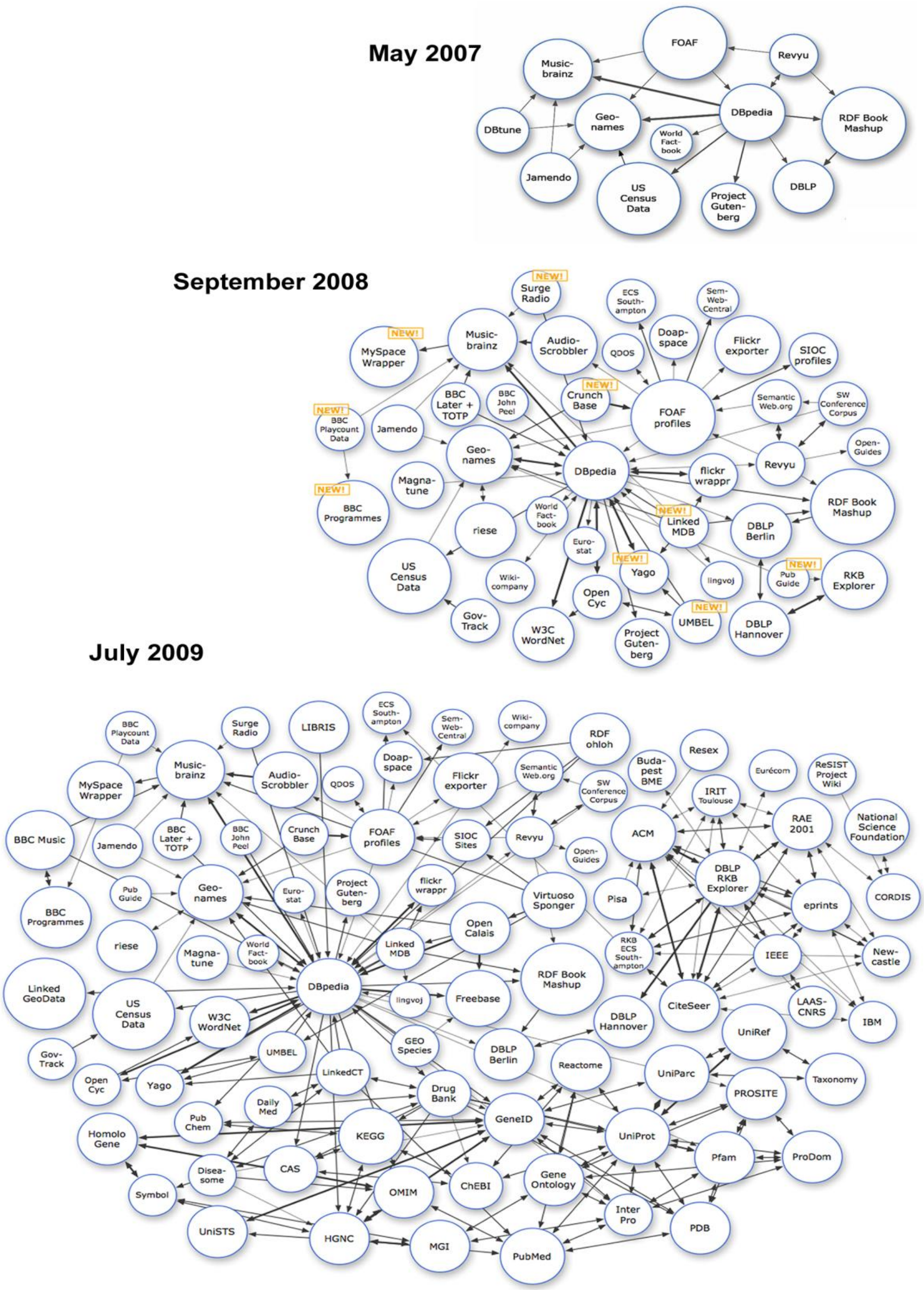


Fig. 1 - The expansion of the Semantic Web - 2007-2009

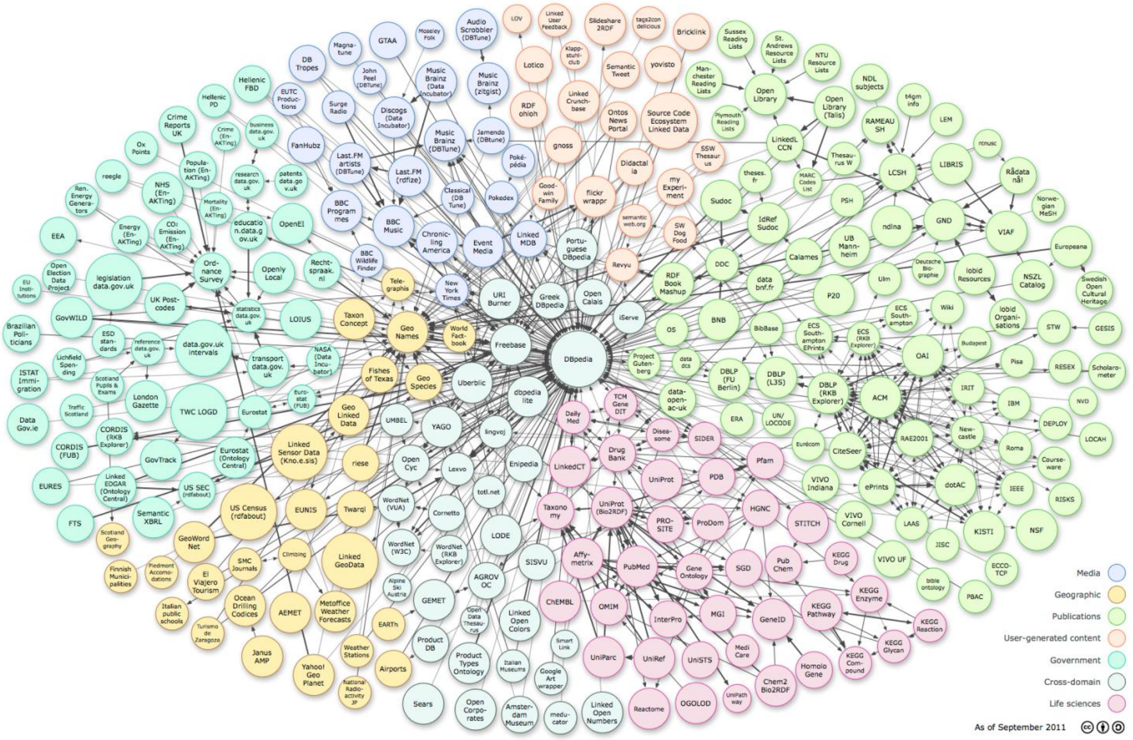


Fig. 2 - The Semantic Web, as of September 2011

It is in this context that David Shotton suggested the opportunity for a *Semantic Publishing "Revolution"* in 2009 [Sho09], whose main idea we already hinted at. We know all too well that scientific innovation is based not only in hypothesis formulation, experimentation, interpretation of data and publication, but also on finding, understanding, re-using, discussing and possibly challenging the results of previous research; discovering ways to improve the effectiveness of this process is tantamount to the betterment of the research output on its whole.

A large part of this scientific production is in the form of scholarly papers published by academic journals. Even considering just those publications, and not the various conference proceedings or complete books, the magnitude of the numbers involved is a testament to the determined progress of humanity in the field of knowledge: It has been esteemed that, in 2006 alone, more than 1,350,000 articles were peer reviewed and published, roughly 154 per hour [BRL09]. In 2011, the number of publications in just the field of health and medicine recorded by the US National library of Medicine amounts to more than 828 thousands[9]. As it is, there is a widening gap between our ability to generate data and knowledge, and our ability to retrieve it and link it. The

[9] United States National Library of Medicine and Health
Publication per year data available here:
http://www.nlm.nih.gov/bsd/medline_cit_counts_yr_pub.html

Semantic Web and all its "children" initiatives are part of an effort to resize this gap.

The idea behind Semantic Publishing is to use the most recent developments in web technologies for the semantic enhancement of scholarly journal articles, in a process that would involve contributions from publishers', editor's and authors', aiming to assist the publication of data and meaningful metadata related to the article, as well as providing means to explore them and interactive access to content [Sho09]. All these enhancements could then increase the value of the improved articles, making them easier to understand, allowing for a better emergence of knowledge, making datasets interactive and allowing for the development of potential secondary services (called "ecosystem services") for the integration of said enhanced information between several articles, or between the articles and other parts of the LOD, for example by having named entities automatically linked to the appropriate ontology.

Semantic Publishing could then merge the already existing advantages of systemic online article publication, which are similar to those of traditional web, where documents are designed mainly to be used by people, with the advantages of LOD, so that the readers could benefit from quicker, more complete and more practical access to meaningful and reliable information, while possibly discovering and exploring other related data seamlessly.

Shotton describes both the current state of on-line journal publishing, including its shortcomings, and his prefigured state of the art for Semantic Publishing, listing a wide amount of possible course of actions that could be taken by the stakeholders, such as the semantic mark up of text, providing structured digital abstract, allowing for interaction on media and data, and so on… He also underlines [Sho09] the different contributions that could be made by the different stakeholders, according to their roles (publishers, editors, and authors should all be involved, but in different part of the process). In that paper he also defines principles and guidelines for future semantic publishing activities.

Leading the way in practice as well as in theory, an exemplar application of Semantic Publishing as a semantic enhancement of an existing article by Shotton *et al* had just been published at that time [SKM09], and it showed a concrete implementation for several intra-document and inter-document interactive applications (such as *data fusion* with other data sources, tooltips for citation in context and citation typing, or the highlighting of semantically relevant terms), as well as theorizing many other advantages and uses, ultimately changing the perception of how can an article be better read and understood just with the proper application of existing web technologies, according to the belief that much could be done to make the data contained **within** a research article more readily accessible.

The authors of [SKM09] chose an approach which has been an inspiration for mine own, as they selected an existing article [RRF08], to serve as a target for their concrete examples and as a reference platform for the new functionalities they suggest. The result of their work is available for online consultation and interaction [10].

The features showcased as functional enhancements to the article in [SKM09] are heterogeneous, and comprehend many interesting data fusion experiments or actionable data interfaces, but the part most related to this dissertation is the one detailing several ways to "Add value to text". These includes, to give a non comprehensive list: The highlighting of named entities and their linking to external information sources (such as appropriate ontologies), citations in context with tooltips, tag cloud and tag tree on the entities, document statistics, citation typing analysis, enhancement of links and machine-readable metadata with RDF.

The authors also commented on the *"needs to approach research publications and research datasets with different presuppositional spectacles",* acknowledging the importance of having tools to emphasize one aspect rather than the other, and first advanced the idea behind Semantic Lenses. Many of the suggestions in this article, like the support for a structural markup of greater granularity, as well as the already mentioned integration with a citation typing ontology, will be fundamental for the development of Semantic Lenses, which I will explore better in section 4.

[10] – Enhanced version of [RRF08] by [SKM09]. DOI: http://dx.doi.org/10.1371/journal.pntd.0000228.x001

2.3 Other Related Works

Obviously, the idea of semantic enhancement for scientific papers or journal articles predates the formal definition of Semantic Publishing, even though most of this other works focused on a specialized aspect of it.

For example, the interest in explicitly defining the rhetoric structure of a scientific publication has been there for a while, as exemplified by De Waard *et al* in [DBK06], where the authors made the compelling argument that a scientific article is very much an exercise in rhetoric having the main objective of persuading readers of the validity of a particular claim. The authors lamented that, despite the advent of computer-centered ways of creating and accessing scientific knowledge the format of an article has remained mostly static. Their answer was the development "*of a more appropriate model for research publications to structure scientific articles*" [DBK06], based on a rhetorical structure which they identify as ubiquitous in scholarly articles.

This model, developed for usage in a computerized environment, relies on authors explicitly marking up the rhetorical and argumentational structures of their findings and claims during the authoring/editing process, then making these metadata available to a search engine. The goal was to allow for the creation of well defined *lines of reasoning* within a text, and between texts, to present an user with a network of linked claims: some of these ideas were further developed in within the concept Semantic Lenses. The model proposed by [DBK06] was based on three elements, namely a rhetorical schema with the definition of the logical order and the rhetorical role of the document sections, an analysis of the argumentation structure of the paper, and the identification of data and entities within the documents.

De Waard's interest in rhetorical analysis of papers and Semantic Publishing technologies did not abate with the passing of time, and her recent "*Directions in Semantic Publishing*" [DeW10] makes for a most compelling read, as well as a magnificent summary of the state semantic enrichment at the time of its publication. Expanding the subject of her discourse from the simple enhancement of entities, something that is being done by several tools, like Pubmed [11], [DeW10] makes a persuasive case in favor of statements (in form of subject-predicate-object triples) as the most complete way to provide machine-readable access to pertinent facts, then observes that we should not limit ourselves to simple statements as the only way to transmit meaningful scientific knowledge, arguing that the main method for this communication is

[11] United States National Library of Medicine and Health, Pubmed: http://www.ncbi.nlm.nih.gov/pubmed/

scientific discourse, reinforcing her previous claim that scientific articles are akin to *"stories that persuade with data"*, as well as endorsing the effort to develop and connect scholarly publications to the LOD space.

In [DBC09] these talking points evolve into HypER – Hypothesis, Entities, Relationships: the proposal to design a system where specific scientific claims are connected, through sequences of meaningful relationships, to experimental evidence. Once again, at the center of this work lies the fact that knowledge representation focuses on scientific discourse as a *"rhetorical activity"*, and that tools and modeling processes should take this consideration into proper account. When comparing this approach with others based solely on isolated triples, there is a considerable shift in assigning the epistemic value of sentences to the explicit characterization of author intent, consequently implying a shift of the conceptualization of text towards the rhetorical discourse. The main intent of HypER is thus summarized into changing the focus of the reading comprehension from the subject studied back to the author's rhetorical, pragmatical and argumentative intent.

Another must-read recent paper is the one by Pettifer *et al* [PMM11], on modeling and representing the scholarly article. In this, one of the initial focuses is once again the consideration that our ability to generate data is surpassing by far our capacity to harness its potential and to retrieve and reuse it effectively, and we are losing track of what we know, adding to the costs of research the effort of rediscovering our own knowledge (even if this not just a problem of modern age). The authors ponder on the conflicting needs of publishing, especially the conflicting nature of having documents that must serve both as platforms for the human reader, as well as being "good" at delivering and hosting machine-readable metadata. As it is painfully obvious, the two possible recipients require very different languages, structures, and have very specific and different needs.

The authors examines the pros and cons of several data formats in light of these considerations, and try (perhaps a little unconvincingly) to deflate some of the arguments against PDF. It also explains very well the foundation of FRBR – The Functional Requirements for Bibliographic Records, which is a model introducing several levels for the classification of a Bibliographic Entity. A Work is realized into one or more Expressions, which are then embodied as one or more Manifestations, of which exist one or more concrete Items.

The final purpose of [PMM11] is to introduce Utopia Documents, a software approach to mediate between the underlying semantics of an enhanced article and its representation as a PDF document, striving to combine all the interactivity of a Web page with the advantages of the PDF. Once again, the emphasis is on actionable data, recognition of content and features, and automatic linking of citations and entities.

Another approach is the one made by SALT – Semantically Annotated LaTeX [GHK07], an authoring framework where authors can embed semantic annotations on LaTeX documents for the enrichment of scientific publications. The framework features three different ontologies, one for capturing the structure of the document, one for the externalization of the rhetorical and argumentational content and one for linking the argumentation to the structure and to the document itself. It is also available for use outside the LaTeX environment, but that is still its main area of application, and that's where an annotation module is being developed. In the specific, SALT permits the enrichment of the document as an activity concurrent to its writing, giving the author ways to express formal descriptions of claims, supports and rhetorical relations as part of the writing process. However, the final result is an enriched PDF document with slightly less features when compared to Utopia, and its scope is more limited than that of Semantic Lenses, as the main usage environment at the base of its design remains LaTeX, with all its *peculiar* characteristics.

As I mentioned ontologies, there are many addressing several of the problem areas. While those that are used in this work will be better described in section 3 and 4, some are more than worthy at least of a passing mention. Aside from the already mentioned FRBR [IFL98] (which has the advantage of not being tied to a specific metadata schema or implementation), there is also BIBO, the Bibliographic Ontology [DG09], able to describe bibliographic entities and their aggregations. DOAP [Dum04], the Description Of A Project ontology is a vocabulary for the description of software development research projects.

The interest in Semantic Publishing technologies by the stakeholders, especially publisher, has grown quite a lot in the last years. As an example, allow me to mention the Elsevier Grand Challenge (2009) for Knowledge Enhancement in the Life Sciences, a competition between proposals "to improve the way scientific information is communicated and used". Participants were required to submit descriptions and prototypes of tools "to improve the interpretation and identification of meaning in online journals and text databases relating to the life sciences". The competition, which offered a total of $50,000 as prize money (35,000 for the first prize), was won by Reflect [POJ09] which is an impressive research tool designed to be an augmented browser for life scientists. Reflect is able to identify relevant entities like proteins and genes and to generate pop-up windows with related contextual information, together with additional links to those entities as defined in other ontologies. Its architecture is focused on text-mining the content of the articles and then comparing the results to its internal synonym dictionary for automatic entity recognition, which performs quite accurately in its field. While it provides with different useful disambiguation tools and tagging features, its design is quite different from the approach of semantically enrich articles by embedding relevant meta-information in it, and it focuses mainly on the speed of the recognition, as well as being very dependent on the maintenance of synonym list. Still, it's an excellent example of how many different practical approaches could be taken to create viewpoints tailored for the need of researchers, with systematic emergence of meaning, and quick and easy access to more detailed information.

Another recent sign in this direction is the birth of two new important conferences dedicated to Semantic Web. In 2011, the 10th International Semantic Web conference hosted the Linked Sciences workshop, a full day event with discussion about new ways for publishing, linking, sharing and analyzing scientific resources like articles and data, while the 8th Extend Semantic Web Conference inaugurated SePublica, the first formal event entirely dedicated to Semantic Publishing, a workshop where several papers were presented, and the best one of them was awarded a prize.

3 Technologies and onthologies

3.1 RDF and Ontologies in General

As described in Sections 1 and 2, a Semantic Publishing activity such as this one is part of the broader range of works that fall under the domain of "*Semantic Web*". I will start my brief review of the technological context for this work by quickly introducing some of the basic concepts behind most of the technologies presented and used in this demonstration.

Let's start with Ontologies. An Ontology, at least in computer and information science terms, is an agreed upon formal specification of knowledge, consisting in a set of concepts and properties within a domain. Or, to put it another way, an ontology is the expression of a shared consensus on a way to explicitly represent the meaning of terms in vocabularies and the relationships between those terms. [Gru07].

An Ontology is designed to give users a set of non-ambiguous, formal methods with which to model a certain domain of knowledge (or discourse). Ontology components are typically member of one of these primitive groups:

- **Classes (or sets)**, which are the concepts defined in the ontologies. Each class will of course have its individual members
- **Properties (or attributes)**, which are the characteristics or the parameters defining or refining the meaning an object can have. There can be two main categories of properties: Object Properties, defining meaningful links between individuals, and Data Properties, defining meaningful links between and individual and a (typed) dataset.
- **Relationships (or relations),** which are the way classes or individuals can be related to each other, for example hierarchy relationships or membership relationships.
- **Restrictions,** formal requirements that must be met and verified.
- **Axioms,** introductory assertions used to associate class and properties with some specification on their characteristics, or to give logical information about classes and properties which is held to be true in the model the Ontology uses in its knowledge domain

Ontologies are often referred to as "vocabularies". According to the W3C: "*There is no clear division between what is referred to as "vocabularies" and "ontologies". The trend is to use the word "ontology" for more complex, and possibly quite formal* *collection of terms, whereas "vocabulary" is used when such strict formalism is not necessarily used or only in a very loose sense.*"[12]

Ontologies can be defined in many different languages. For the time being, the reference one (for Semantic Web) is the Web Ontology Language 2 - OWL 2 [W3C09][HKP09], a specification by W3C in three different versions, each one corresponding to a different level of expressiveness. These are, in order of expressiveness: OWL 2 Lite, OWL 2 DL, and OWL 2 Full. Being a language aimed at the definition of Ontologies, OWL is not especially relevant to this work, as Ontology definition is not part of the scope of my activity: Semantic Lenses are meant to operate with already existing, well designed and widely tested ontologies.

More important within this demonstration is the W3C Resource Description Format – RDF [KC04]. This is a family of specifications aimed at modeling data representation and interchange over the Web, and is the basis over which OWL is built. More specifically it can be divided between RDFS (RDF Schema), which is a schema language used to define RDF itself, one allowing some basic ontology definition, and the RDF model itself. The first not being relevant to this work, I will illustrate the basic concepts of the latter, as the metadata specified by the Semantic Lenses, and the one I will embed to semantically enrich [Mik07], are based on the RDF model, and the whole enhanced version of [Mik07] will be an RDF document.

A fundamental technology for the Semantic Web, the RDF model mimics and extends the basic linking structure of the traditional Web by using URIs to identify relationships between subjects and objects, as well using those to indentify the two ends of a link. RDF is thus made out of **statements** that hinge on *subject-predicate-object* ***triples***.

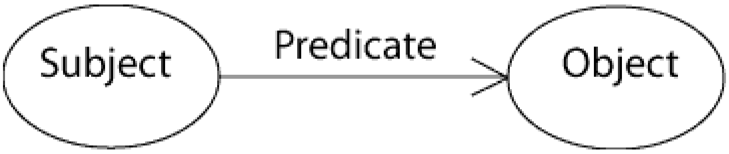


Fig. 3 - The structure of the basic RDF triple

RDF Resources, which are "things" identified by URIs, are the main building blocks of RDF, and can be either subjects or objects of statements, but the object of a statement can also be a simple data type, known as *literal,* which can be a string or a number. Properties are a special kind of Resources, and are those used to describe relationships between resources – a predicate is a property inserted in a statement. Types can be represent by a resource, and can

[12] Definition of an Ontology by the W3C: http://www.w3.org/standards/semanticweb/ontology

be assigned by the rdf:type property to another resource. RDF supports blank nodes and several types of containers and collections.

It is important to realize that RDF is a graph-based model. A set of RDF statements forms a graph, connecting Resources subjects of statements to their objects by the way of URI identified properties. It is also important to note that identical URIs in different graphs refer to the same resource. This is because RDF is especially designed for representing information that is machine-readable and thought to be processed by applications, such as metadata about documents, and one of the aims of RDF is to provide a framework to express this information so that it can be exchanged and processed between different sources and agents without loss or alteration of meaning [MM04]. To do so, an RDF model graph can be linearized to a list of textual statements, in several languages – such operation does not produce unique results: a graph can be correctly linearized in more than one way. We'll quickly introduce two of the most relevant ones.

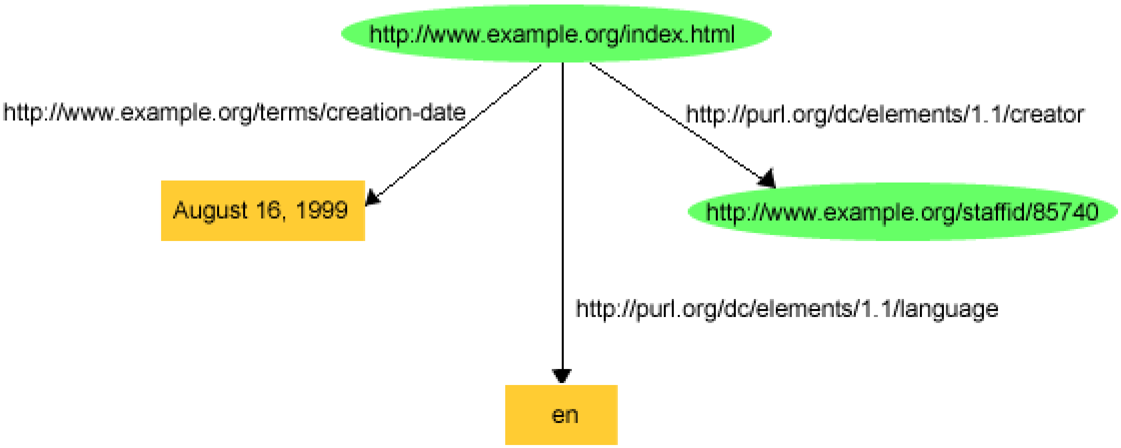

http://www.example.org/index.html
http://www.example.org/terms/creation-date
http://purl.org/dc/elements/1.1/creator
August 16, 1999
http://www.example.org/staffid/85740
http://purl.org/dc/elements/1.1/language
en

Fig. 4 - An example RDF Graph

3.1.1 RDF/XML Syntax Linearization

RDF/XML [Bec04] is the basic syntax for the linearization of an RDF Model, and is a way to generate triples in textual form for statements part of an RDF graph, and represent this triples in an XML compatible format, (together with namespaces). It is quite useful for machine accessibility, but it is somewhat overly verbose in term of human readability.

The Description of a resource collects all the statements having that resource as subject, and the resource is identified by the `rdf:about` attribute. If the object of a predicate is a resource, it can be identified with the `rdf:resource` attribute An `rdf:type` can be specified as a standalone property, and typed literals can be declared using the `rdf:datatype` attribute associated to a property.

The linearization of our example graph:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
            xmlns:dc="http://purl.org/dc/elements/1.1/"
            xmlns:exterms="http://www.example.org/terms/">
<rdf:Description
rdf:about="http://www.example.org/index.html">
       <exterms:creation-date>August 16,
1999</exterms:creation-date>
       <dc:language>en</dc:language>
       <dc:creator
rdf:resource="http://www.example.org/staffid/85740"/>
  </rdf:Description>
```

3.1.2 Terse RDF Triple Language – TURTLE

A Turtle [BBP12] document allows writing down an RDF graph in a compact and natural text form, with abbreviations for common usage patterns and datatypes. It is a non-XML format derived from the N-TRIPLES format, and is an RDF-compatible subset of the Notation-3 language. It is less verbose than RDF/XML, and a reasonable mix of machine-readability (easy to parse) and human readability, although its syntax might be a little trick at first.

I'll give a very short introduction to its syntax. @prefix can be used to declare a named shorthand which can then be combined to a local part of the text to obtain a complete fragment, and it is not limited just to XML namespaces. Comments are preceded by the hash sign #.

A statement can be written as:

`<subjectURI> <predicateURI> <objectURI> (or "literal")`

Subject of triples can be repeated by using a `";"` semicolon, to have a list of triples varying only in subject written in a shorter way. Both subject and predicate can be repeated in a similar fashion by using the `","` comma.

The rdf:type property can be declared with the use of `"a"`, as in `; a <typeURI>`

Literal datatypes can be specified by postponing an `^^nameofdatatype` after the literal.

The linearization of our example graph:

```
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-
ns#>.
@prefix dc:      <http://purl.org/dc/elements/1.1/#>.
@prefix exterms: <http://www.example.org/terms/>.
<http://www.example.org/index.html>
    exterms:creation-date "August 16, 1999";
    dc:language "en";
    dc:creator <http://www.example.org/staffid/85740>.
```

3.2 EARMARK – Extremely Annotational RDF Markup

EARMARK (Extreme Annotational RDF Markup) [PV09] is an ontological model designed to combine in a single document both the embedded markup, which can define the structure of the document (like XML and its derivatives), together with annotations and statements over resources (like RDF), with the aim to have all the advantages of both technologies available at the same time within a single model. [DPV11a]

With the EARMARK ontological approach for meta-markup the user can explicitly make structural assertions of markup, describing the structure of a document in a way suitable for the semantic Web.

The model is as well able to express semantic assertions about the document, its content, or the relationships between its components. This allows a very straightforward and powerful integration of the syntactic markup (like HTML) with the semantics of the content document (like RDF), allowing to combine the qualities of both traditional Web and semantic Web in a single format. Not only that, EARMARK also allows for a perfect integration with ontologies aimed at explicating the semantic meaning of syntactic markup (e.g.: Pattern Ontology) as well as being able to support *overlapping markup* in a way that is seamless and very easy to handle, without any absurd workarounds. [DPV11a]

In short, EARMARK is a way to "*bring full RDF expressiveness to document markup (or, equivalently, to provide full fragment addressing to RDF assertions)*" [PV09].

The founding idea for EARMARK is to model documents as collection of addressable text fragments, identified by *Ranges* over text collections called *Docuverses*, and then to associate said text content with assertions that describe syntactic and structural features (such as the equivalent of a paragraph element), via *MarkupItems,* or to define semantic enhancement for the content or part of it. As a result EARMARK allows to represent not just documents with single hierarchies, but also ones with multiple overlapping hierarchies, as well as annotations on the content through assertions that can overlap with the ones already present. [DPV11a] [DPV11b]

A brief list of the features of EARMARK would comprehend:

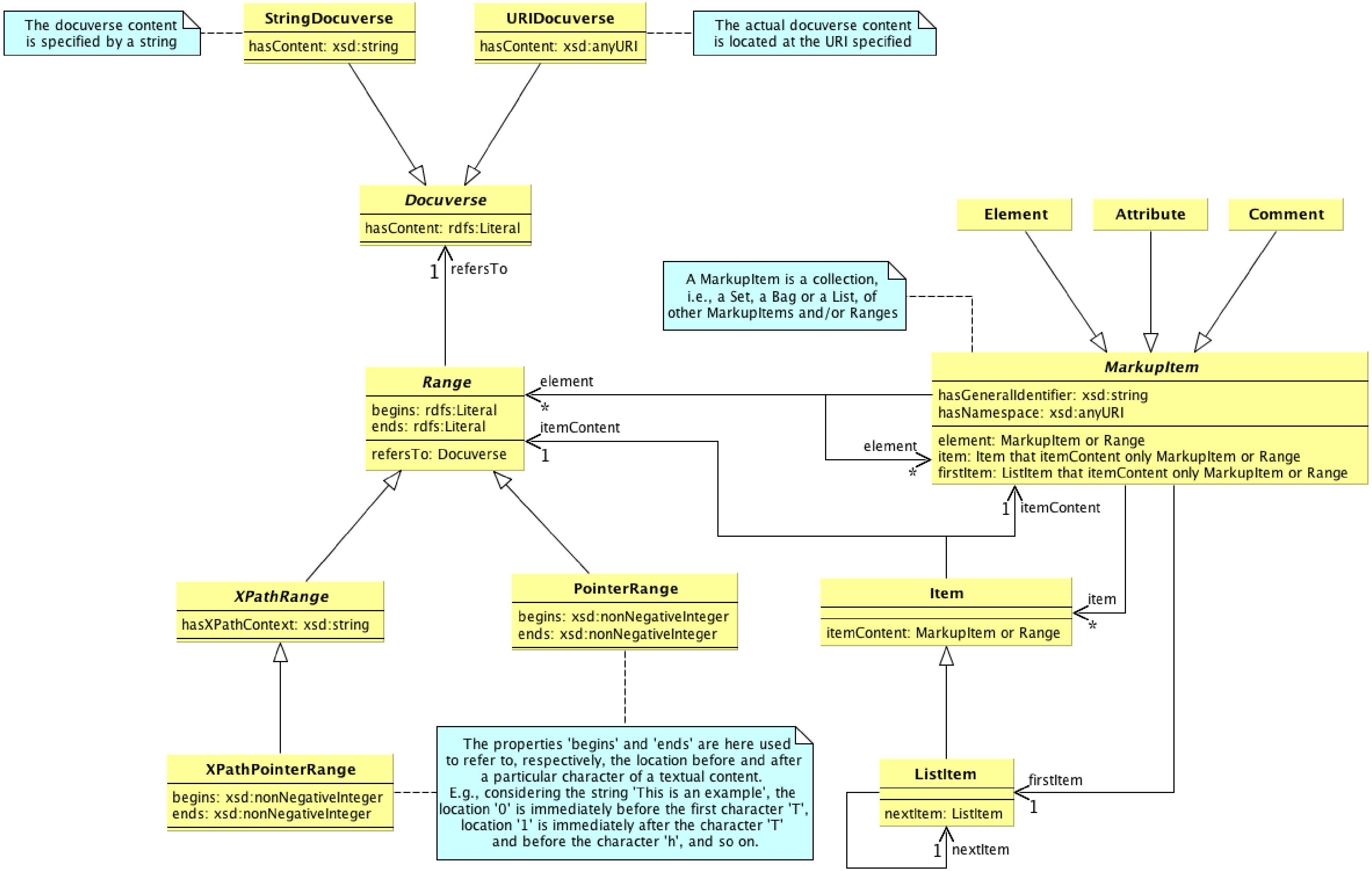
- The possibility to express any kind of arbitrarily complex assertion over documents (be them text, XHTML or XML) without any restriction to the overall structure of the assertions, with support for hierarchies and graphs, either according to the document order or independently
- The ability to convert any embedded semantic markup in RDF triples, and to externalize them

- Being a model completely compatible with RDF, allowing several types of linearization
- The capacity to express out-of-order and repeated uses of the same text fragment, a property originally unique to EARMARK and very rare among any markup embedded in documents.
- The possibility to handle easily overlapping markup, and the compatibility to the full XPointer W3C standard.
- Being able to produce a model over which ontology properties can be verified and validated with reasoners, including consistency of semantic assertions against OWL ontologies.

The EARMARK software package also has a full featured set of JAVA API, as well as a very useful HTML to EARMARK converter, which I used to port [Mik07] into EARMARK.

I am now going to introduce Earmark and its core ontology [Per08], the structure of its model, and give a short overview on how to use it to express properties over elements and text, as well as showing how it solves the problem of overlapping markup.

The core EARMARK model itself, being an ontological one, is distinguished with an OWL document specifying classes, properties and relationships. We distinguish between *ghost classes*, the ones defining the general concepts for the model, and *shell classes*, which are those actually used to instance individual instances of EARMARK components.



The docuverse content is specified by a string

The actual docuverse content is located at the URI specified

A MarkupItem is a collection, i.e., a Set, a Bag or a List, of other MarkupItems and/or Ranges

The properties 'begins' and 'ends' are here used to refer to, respectively, the location before and after a particular character of a textual content. E.g., considering the string 'This is an example', the location '0' is immediately before the first character 'T', location '1' is immediately after the character 'T' and before the character 'h', and so on.

Fig. 5 - The Architecture of EARMARK

3.2.1 Ghost Classes

EARMARK's ghost classes are used to describe its three basic disjointed concepts – *Docuverses, Ranges and Markup Items.* [DPV11a]

- **Docuverses** identify the textual content of an EARMARK document, which is kept apart from ALL annotations on it, regardless of their nature. This textual content is referred to through the Docuverse class, and individual of this class represent the containers of text in an EARMARK document. For example, if we consider a traditional XML document, there might be a Docuverse for the all the textual content of elements, another for the content of all attributes, and another one for all comments. Instanced individual of the Docuverse class specify their content with the property "*hasContent*", which has the content as object.
- **Ranges** are the way for an EARMARK document to identify fragments of text within a Docuverse. The class Range is thus defined for any text lying between two locations of a Docuverse. An instance of the Range class is instanced by the definition of a starting and an ending location within a specific Docuverse, which is referred by the property "*refersTo*". The two main properties for Ranges are "*begins*" and "*ends*", which refer to a literal object indicating the starting and ending points for a range. It is interesting to note that there are no order restrictions over the *begins* and *ends* properties, so it is very well possible to define ranges that either follow or reverse the order of the Docuverse they refer to. For instance, if we consider a Docuverse with hasContent containing the string "*bats*" I can either refer to it if the begins location (0) is lower than the ends location (4), and obtain it in document order, or reverse it by simply having begins = 4 and ends = 0, thus obtaining "*stab*"
- **MarkupItems** are those syntactic artifacts allowing us to define the traditional document markup, such as Elements, Attributes and Comments. An instanced MarkupItem individual is a Collection (Set, Bag or List) of individuals belonging to the classes *MarkupItem* ***and*** *Range.* Through these collections EARMARK specifies that a markup item can be a set, bag or list for other markup items, text fragments as identified by ranges, or a mixture of both, by using properties like "element", "item" and "itemContent" (according to the collection used). By doing so, it becomes possible to define elements with nested elements or attributes, as well as mixed content models, as well as overlapping markups or even other complex, multi-hierarchy structures (such as graphs). Beside the mandatory URI, it is possible to define both

a general name for an instance of MarkupItem, with the property "*hasGeneralIdentifier*", as well as a namespace with "*hasNameSpace*".

3.2.2 Shell Classes

While the ghost classes presented so far give us an abstraction of the EARMARK conceptual model, there is the need to specialize it, specifying concrete definition of the ghost classes. Thus we have several shell classes as subclasses of ghost classes, applying specific restrictions to them and being the ones whose instances can be concretely declared. [DPV11a]

- **A Docuverse is limited to be either a StringDocuverse or an URIDocuverse**. The difference is simple: A StringDocuverse is a docuverse where the actual content is a string included in the document, while an URIDocuverse has its content located at the URI specified.
- **A Range can be either a:**
  - **PointerRange,** which is a range defined by counting single characters over a docuverse. In this case, the value of the properties "begins" and "ends" must be non negative integers that identify the position in the character stream. The index 0 refers to the location just before the last character, while the value n refers to location just after the n-th character. PointerRanges on the same Docuverse having the same starting and ending points are the same range.
  - **XPathRange** is a range defined by considering a context within a Docuverse with an XPath expression, identified by the property value of "hasXPathContext".
  - **XPathPointerRange** Is a subclass of XpathRange where the value of the properties "begins" and "ends" must be a non negative integer identifying the position in the character stream selected by the PointerRange.
- **MarkupItem is specialized in three disjointed sub-classes – Element, Attribute or Comment.** This is done in order to allow for a more specialized and precise characterization of the usual traditional markup items, which usually fall under one of these categories.

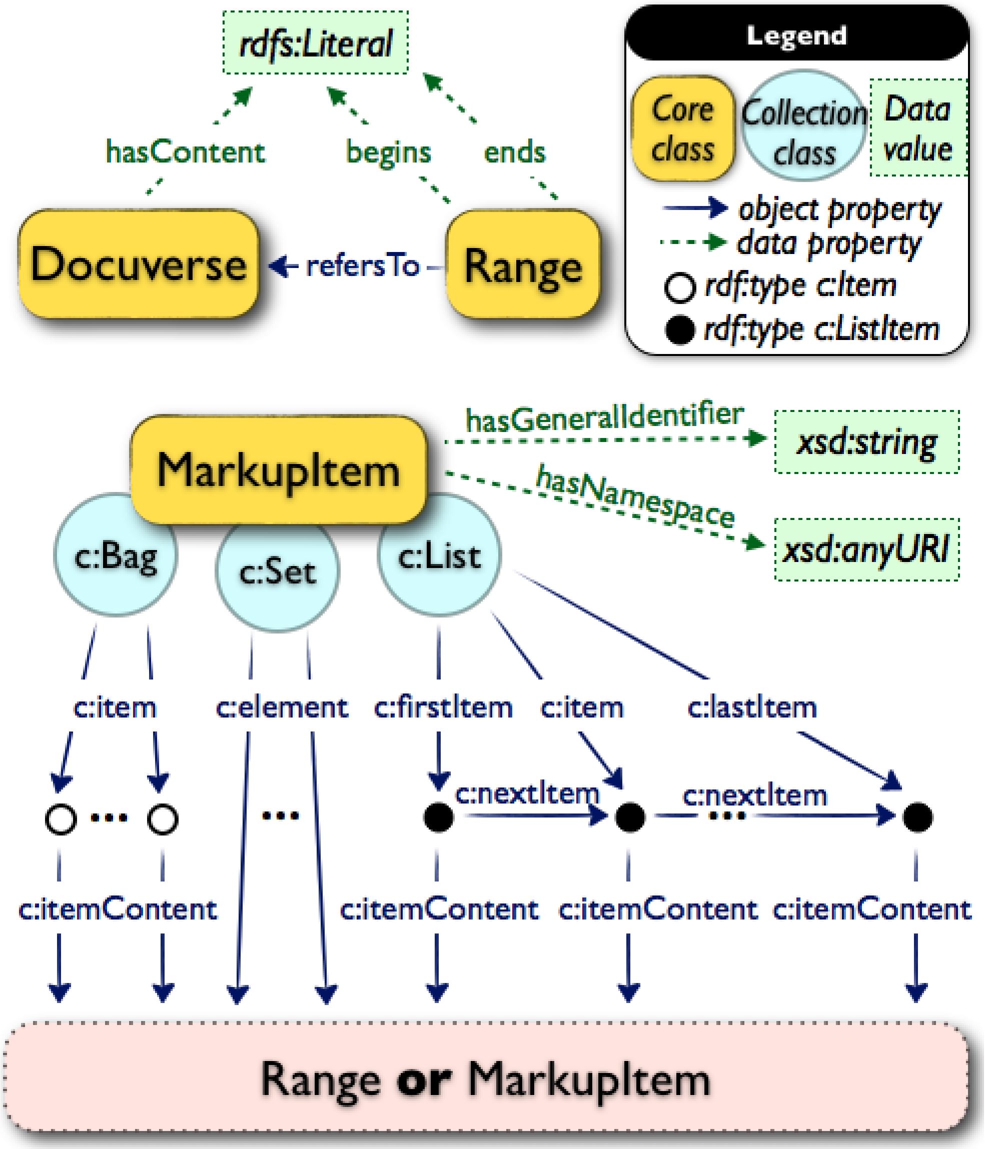


Fig. 6 – Relationships between the main EARMARK Components

3.2.3 Handling Overlapping Markup with EARMARK

EARMARK relies on a sub-ontology, the EARMARK Overlapping Ontlogy [Per11], to model overlapping scenarios on EARMARK documents. Different types of overlap exists, depending on the subset of items involved, so different approaches are needed to correctly detect the problem and deal with it. There is an especially clear distinction between overlapping ranges and overlapping markup items. [DPV11a]

Overlapping ranges, are, by definition, two ranges that refer to the same docuverse, and so that at least one of the locations, and so that at least one of the locations of a range is contained with the interval of the other one. There can be a total overlap, where both locations of a range are contained within another, or just partial overlaps.

Let's make an example to clarify this problem: Suppose that we have ranges A, B and C, and let's say that range A begins at "0" and ends "10", B begins "5" and ends at"14" and C begins at "2" and ends at "8" then we have that A and B are partially overlapping, while C is totally overlapped by both.

There is also the case of overlapping markup items, which can happen in one of the following three different situations. Let us consider two markup elements, X and Y:

- **Overlap by range:** In this case, X contains a range *rX* that overlaps with another range *rY* contained by Y.
- **Overlap by content hierarchy:** Both X and Y contain the same range R
- **Overlap by markup hierarchy:** Both X and Y contain the same markup item Z

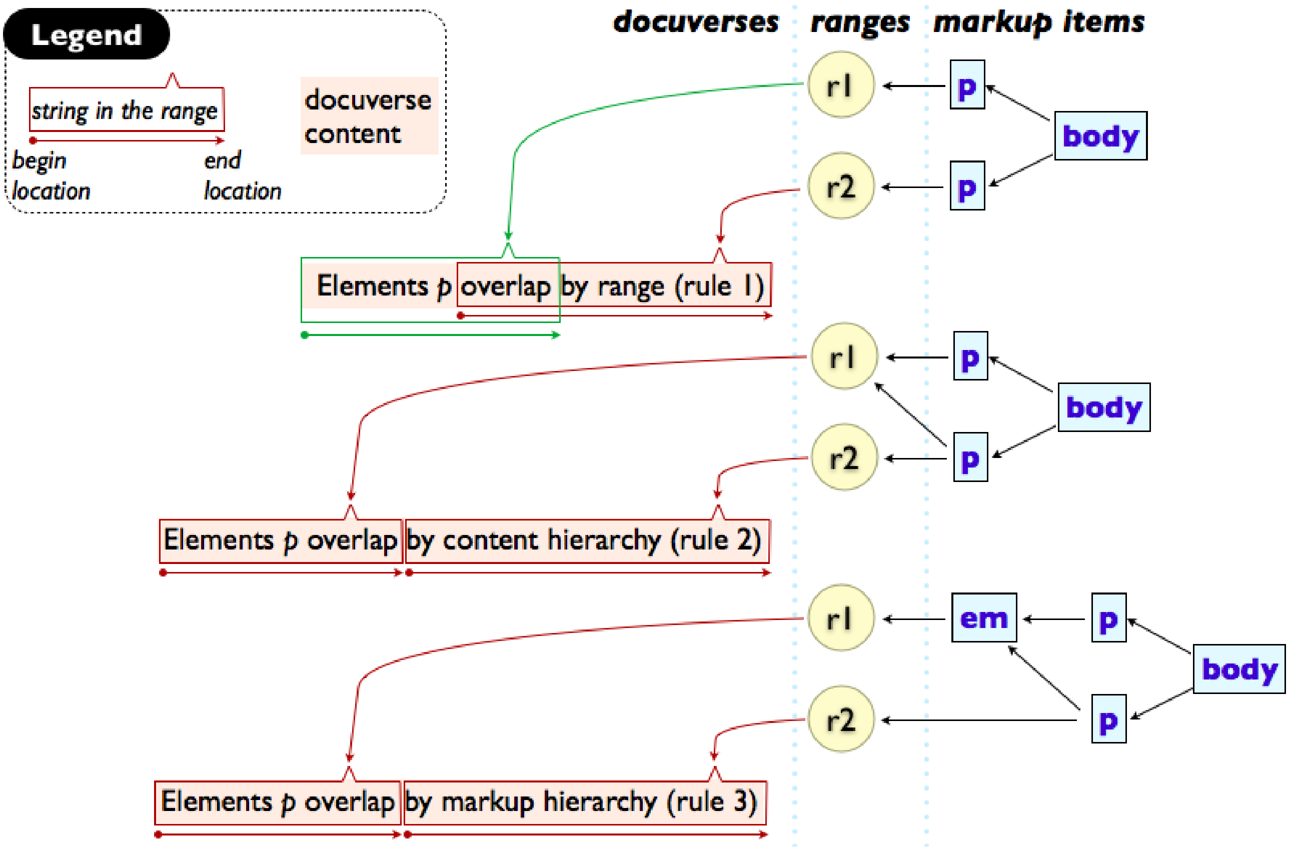

Legend
string in the range
begin location
end location
docuverse content
docuverses
ranges
markup items
r1
r2
p
body
em
Elements p overlap by range (rule 1)
Elements p overlap by content hierarchy (rule 2)
Elements p overlap by markup hierarchy (rule 3)

Fig. 7 - Examples of Overlapping Markup

Let us conclude this quick overview by giving a brief example of EARMARK in action, together with some overlapping markup, inspired from [DPV11b]. Let us consider a stanza from Dante's Inferno of the Divine Comedy:

"E 'l duca lui: "Caron, non ti crucciare:
vuolsi così colà dove si puote
ciò che si vuole, e più non dimandare"[13].

Now, if we wanted to model the stanza both specifying both the structure of the verses and the dialogue, we would have an overlap, since the dialogue starts in the middle of the first verse and ends with the last one. Indeed, a naïve and **INCORRECT** XML interpretation could go as following:

```
<stanza>
 <verse>E 'l duca lui: <dialogue>"Caron, non ti
crucciare:</verse>
 <verse>vuolsi così colà dove si puote </verse>
 <verse>ciò che si vuole, e più non
dimandare".</verse></dialogue>
</stanza>
```

[13] D. Alighieri (1304?-1307?); *La Divina Commedia, Inferno, Canto Terzo.*

As stated before, this is an incorrect and invalid XML, but it serves to exemplify the idea of what we would like to do. Fortunately, here comes EARMARK to the rescue. In Turtle notation, the following EARMARK snippet represents the concepts we tried to apply above:

```
@prefix inf: <http://divina.commedia.it/Inferno/>
inf:doc hasContent "E 'l duca lui: 'Caron, non ti
crucciare: vuolsi così colà dove si puote ciò che si vuole,
e più non dimandare'."

inf:r-0-42 a PointerRange ; refersTo inf:doc
; begins "0"^^xsd:integer ; ends "42"^^xsd:integer .
inf:r-42-73 a PointerRange ; refersTo inf:doc
; begins "42"^^xsd:integer ; ends "73"^^xsd:integer .
inf:r-73-111 a PointerRange ; refersTo inf:doc
; begins "73"^^xsd:integer ; ends "111"^^xsd:integer .
inf:r-16-111 a PointerRange ; refersTo inf:doc
; begins "16"^^xsd:integer ; ends "111"^^xsd:integer .

inf:stanza a Element ; hasGeneralIdentifier "stanza"
; c:firstItem [ c:itemContent inf:verse1
; c:nextItem [ c:itemContent inf:dialogue
; c:nextItem [ c:itemContent inf:verse2
; c:nextItem [ c:itemContent inf:verse3 ]]]] .

inf:verse1 a Element ; hasGeneralIdentifier "verse"
; c:firstItem [ c:itemContent inf:r-0-42 ] .
inf:verse2 a Element ; hasGeneralIdentifier "verse"
; c:firstItem [ c:itemContent inf:r-42-73 ] .
inf:verse3 a Element ; hasGeneralIdentifier "verse"
; c:firstItem [ c:itemContent inf:r-73-111 ] .

inf:dialogue a Element ; hasGeneralIdentifier "dialogue"
; c:firstItem [ c:itemContent inf:r-73-111 ] .
```

3.3 Linguistic Acts Ontology

In order to give user ways to correctly interpret markup semantics, this project will also make use of the Linguistic Acts [Gan07]. It is the result of an integration between LMM [PGG08] and EARMARK, as introduced by [PGV11], whose purpose is to act as a mean to express clear semantics about meta-markup. I will be mostly using its "`expresses`" property.

The main inspiration behind the Linguistic Acts is the consideration that while the syntax of XML-based languages is machine-readable, its semantics is not explicitly defined, so it is meaningless for machines. The authors of [PGV11] consequently resolved to use Semantic Web Technologies to fill the gap between the well defined syntax and the informal specification of its semantics, by integrating LMM, an OWL vocabulary representing some basic semiotic notions, with EARMARK, which we have already presented some pages before.

The origin of the problem is that the evolution in the importance of markup as a way to provide metadata (resource descriptions and relationships) led the Semantic Web effort mostly to concentrate on dealing with *semantic markup* (e.g. the resource r has the string s as title) but at the same time skirting around the issue of *markup semantics* (e.g. what is the meaning of a markup element p contained in resource r?).

We also have to consider that avoiding imposing any specific semantics along with their syntax is among the design aims of markup meta-languages. Take for example XML: it does express simple syntactic labels on the text, leaving the semantics of the markup to the interpretation of humans or tools appropriately instructed, because it is deliberately designed to do so.

However, it would be extremely important to have a mechanism to define machine-readable semantics for markup languages, for a lot of reasons: parsers could perform semantic validation on the document markup, as well as a simple syntactic one, reasoners could infer new assertions from documents, and documents could be queried over the markup semantics and so on… Notably, being able to correlate machine-readable semantics for the markup of a document is also a fundamental activity for semantic publishing, and also very important in the context of explicitly defining the structure of a semantically enriched paper. So, by "*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 markup elements themselves, and about the semantics of the content of the document*" [PGV11]

We have already discussed the advantages of EARMARK and observed on how it makes feasible to express markup semantics quite simply and in a straight-forward way. But to associate coherent semantics to markup items it is advisable to follow precise and theoretically founded principles of semiotics, making the applications of them interoperable. As a solution, [PGV11] proposes to adopt the Linguistic Act ontology design patter, based on LMM, as a mean to provide semiotic-cognitive representation of linguistic knowledge.

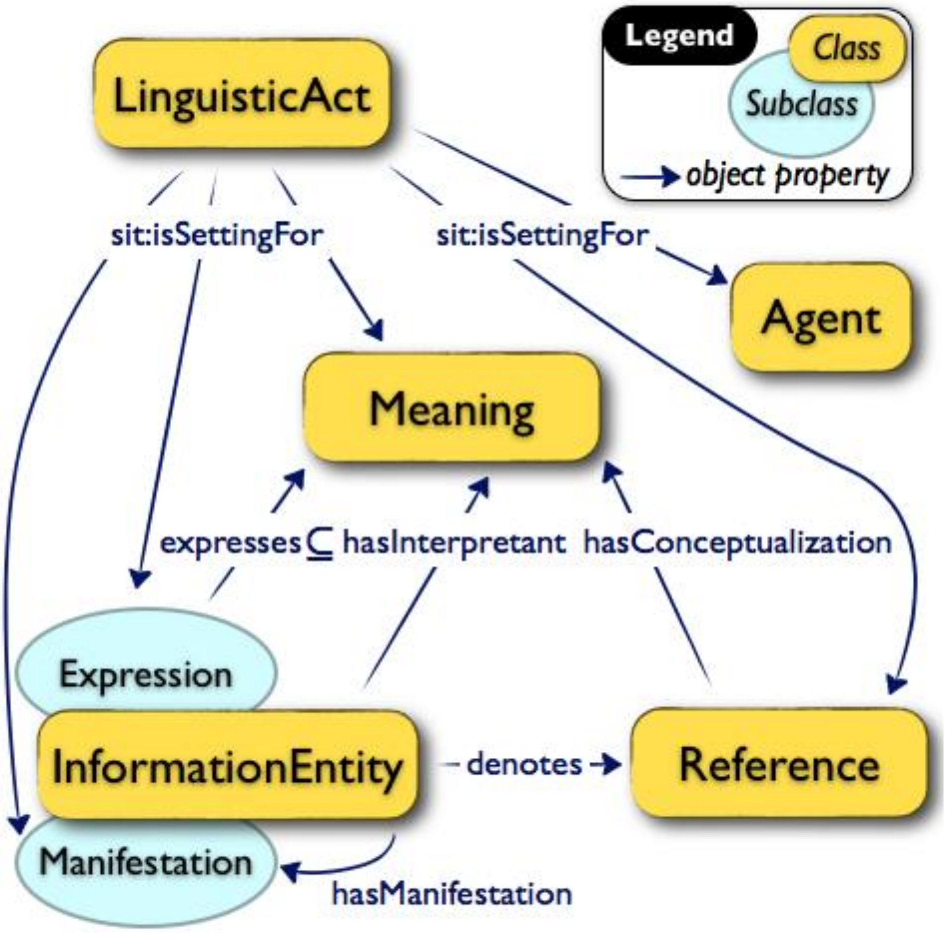


Fig. 8 - The Architecture of the Linguistic Acts Ontology

The main idea behind it is to be able to handle the representation of knowledge from different sources according to different theories, putting each of them in the context of the *semiotic triangle* and some related semiotic notions. These are as follow:

- **References** are any individual or set of individuals, or fact from the world being described
- **Meanings** are any object explaining something or being intended by something, such as definitions, topic descriptions, concepts, etc.

- **Information Entities** are any symbol that have a meaning or denotes one or more References
- **Linguistic Acts** are any communicative situation including Information Entities, Agents, Meanings, References and a possible spatial-temporal context.

Given these premises, Markup Items in EARMARK are specific kinds of Expressions expressing a particular Meaning, assigned by the author of a schema, used to denote local objects or social entities.

Focusing more on the aims of this dissertation, the "expresses" property is used to identify a relation between an Expression and a Meaning. The intuition for 'meaning' is intended to be very broad, as there are a lot of different approaches to meaning characterization and modeling:

For example, let us consider the word "*beehive*" – in all these cases, some aspect of meaning is involved [PGG08]:

- Beehive means "a structure in which bees are kept, typically in the form of a dome or box." (Oxford dictionary)

- 'Beehive' is a synonym in noun synset 09218159 "beehive|hive" (WordNet)

- 'the term Beehive can be interpreted as the fact of 'being a beehive', i.e. a relation that holds for concepts such as Bee, Honey, Hosting, etc.'

- 'the text of Italian apiculture regulation expresses a rule by which beehives should be kept at least one kilometer away from inhabited areas'

- 'the term Beehive expresses the concept Beehive'

- ''Beehive' for apiculturists does not express the same meaning as for, say, fishermen'

- 'Your meaning of 'Beautiful' does not seem to fit mine'

- ''Beehive' is formally interpreted as the set of all beehives'

- 'from the term 'Beehive', we can build a vector space of statistically significant co-occurring terms in the documents that contain it'

As the examples suggest, the **"meaning of meaning"** is dependent on the background approach/theory that one assumes. One can hardly make a summary of the too many approaches and theories of meaning, therefore this relation is maybe the most controversial and difficult to explain; However, the usefulness of having a 'semantic abstraction' in modeling information objects is so high (e.g. for the semantic web, interoperability, reengineering, etc.), that [PGV11] accepted to tackle this challenging task. It also anticipates some of the possible solutions on how to explicitly specify semantics of markup elements, which we will explore further in the sections about the Pattern Ontology (4.4) and about the Document Structure Semantic Lens (4.3.4, 5.5).

3.4 Spar Area Ontologies

In recent years, a cohesive effort has been made to develop, merge together and rationalize a set of Ontologies allowing to cover with metadata all aspects of semantic publishing. This effort resulted in the development of SPAR – The Semantic Publishing and Referencing Ontologies suite [Sho10a]. This is a an integrated ecosystem of independent and reusable orthogonal and complementary ontology modules, usable for creating comprehensive machine-readable RDF metadata on semantic publishing and referencing: these can be used either individually or in conjunction with each other, according to the user needs.

There are 8 main component ontologies in SPAR, each organized a named following the flower diagram shown below. Each is encoded in OWL 2.0 DL [W3C09], and together they provide the ability to describe a bibliographic entity in all of its aspect, from its inception to citations, from bibliographic records to the component parts: the aim is to be able to cover all aspects of a scholarly publication process, and to enhance its semantics. As such, these Ontologies represent an invaluable asset for the development and the use of Semantic Lenses.

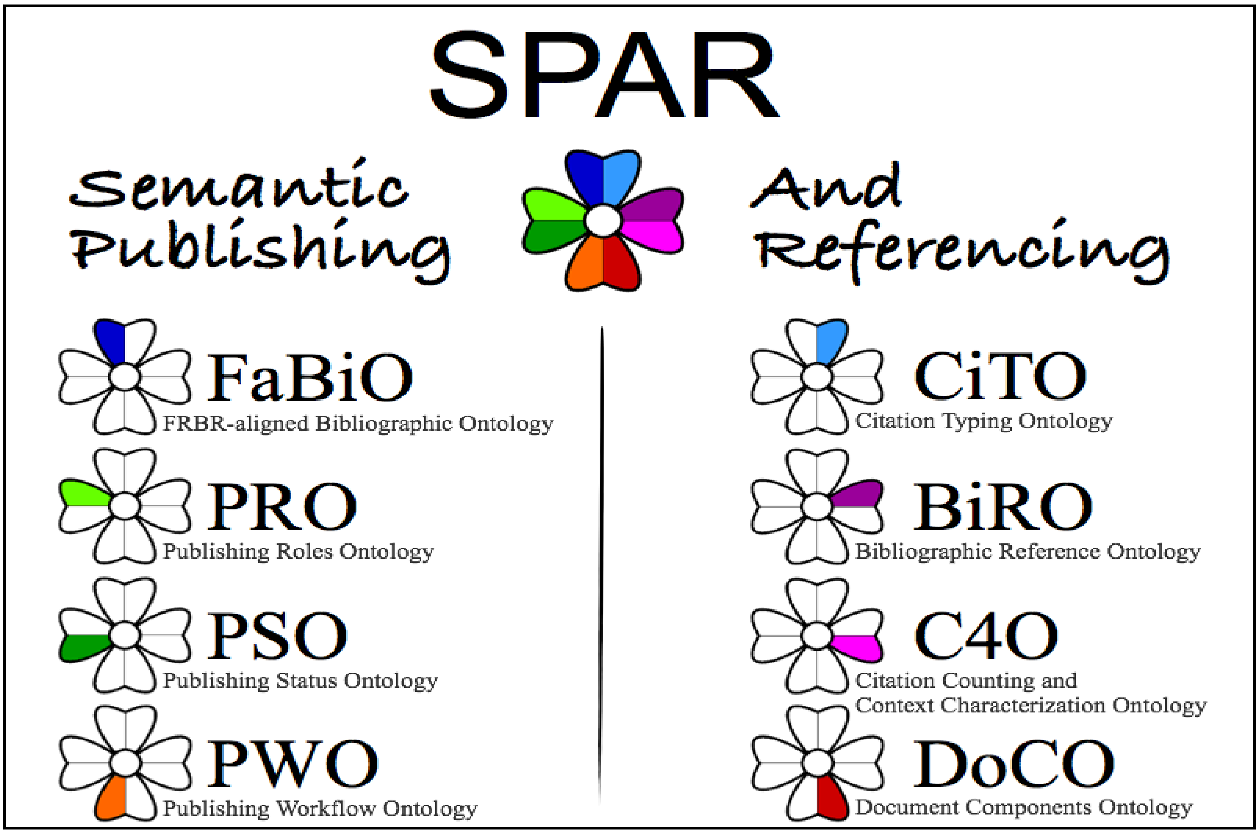

SPAR
Semantic Publishing
And Referencing
FaBiO
FRBR-aligned Bibliographic Ontology
PRO
Publishing Roles Ontology
PSO
Publishing Status Ontology
PWO
Publishing Workflow Ontology
CiTO
Citation Typing Ontology
BiRO
Bibliographic Reference Ontology
C4O
Citation Counting and Context Characterization Ontology
DoCO
Document Components Ontology

Fig. 9 - Summary of SPAR Ontologies

All eight SPAR ontologies – FaBiO, CITO, PRO, BiRO, PSO, C40, PWO and DoCO – are available for inspection comment and use.
As we will further detail in section 4, a good deal of these Ontologies are the ideal choice for some Semantic Lenses Layers, such as FaBiO and BiRO for the publication context lens, DoCO for the rhetoric lens and CiTO for the citation lens. As such, those modules extensively used in this work will be discussed more in depth further in the text, but for the sake of completeness we are going to give a cursory overview of all SPAR modules in this section, explaining their usefulness, their composition and scope.

Some of the modules of the SPAR suite expand and re-use, where appropriate, other popular Ontologies and classification models, such as FOAF (Friend of a Friend) to describe individuals, or the FRBR (Functional Requirements for Bibliographic Records) classification model. Since its inception in 2009 (detailed in [Sho10b]), CiTO has also been the subject of an important process of harmonization with the SWAN (Semantic Web Application in Neuromedicine) Scientific Discourse Module and Swan Collection Module, resulting in the current integration [CSP12]
The architecture of the SPAR suite of Ontologies is quite straightforward, and easy to summarize in a scheme:

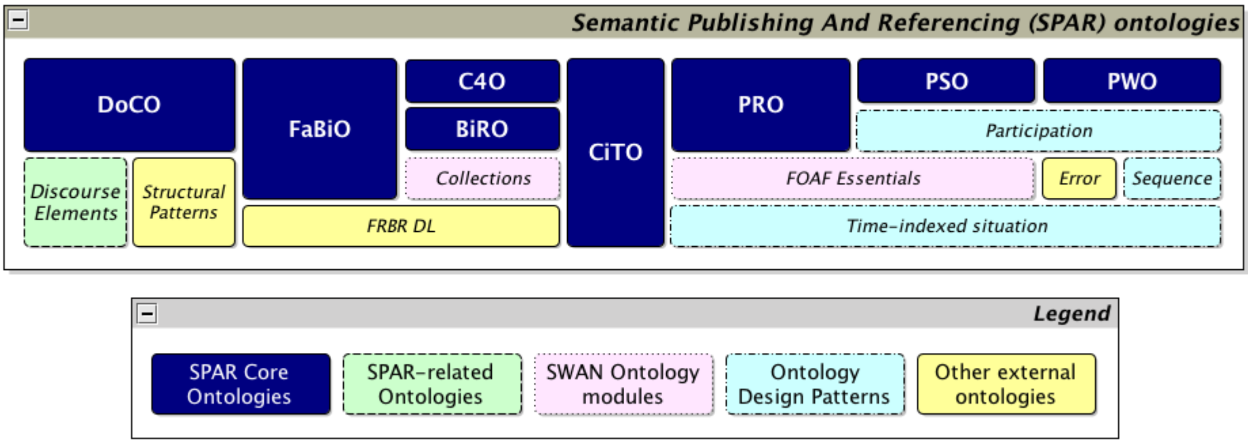


Fig. 10 - The Architecture of SPAR

A very simple summary of the roles and peculiarities of each ontology is to follow. Some Ontologies modules, like Structural Patterns, DEO, DoCO, and CITO will be described more in depth further in the text:

FaBiO, the *FRBR-aligned Bibliographic Ontology*

URL: http://purl.org/spar/fabio
SVN: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/FaBiO

FaBiO, the FRBR-aligned Bibliographic Ontology, is an ontology for recording and publishing on the Semantic Web descriptions of entities that are published or potentially publishable, and that contain or are referred to by bibliographic references, or entities used to define such bibliographic references. It extends FRBR with formal object properties to describe relations across the FRBR objects part of the bibliographic universe, such as Works, Expressions, Manifestations and Items.

FaBiO entities are designed to provide an extensive set of publication types, aiming to cover primarily textual publications such as books, magazines, newspapers and journals, and items of their content such as poems and journal articles. However, they also include other types, such as datasets, computer algorithms, experimental protocols, formal specifications and vocabularies, legal records, governmental papers, technical and commercial reports and similar publications, and also bibliographies, reference lists, library catalogues and similar collections.

FaBiO imports the FRBR Core ontology, and it extends the FRBR data model by the provision of new properties, aiming to extend the FRBR data model by linking Works and Manifestations (with the properties fabio:hasManifestation and fabio:isManifestationOf), Works and Items (fabio:hasPortrayal and fabio:isPortrayedBy), and Expressions and Items (fabio:hasRepresentation and fabio:isRepresentedBy). Its properties and its structure make FaBiO one of the best tools to represent the publication context lens

CiTO, the *Citation Typing Ontology*

URL: http://purl.org/spar/cito
SVN: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/CiTO

The Citation Typing Ontology (CiTO) is an ontology whose purpose is to enable characterization of the nature or type of citations, both factually and rhetorically. It allows much more than simply asserting in RDF than citations exists, but also encourages to define the factual or rhetorical nature of the citation, and the reasons behind it.

This ontology contains the object property cito:cites and its sub-properties, like cito:updates or cito:obtainsBackgroundFrom, which are to be used to better

characterize the semantics of a citation in a bibliographic entity. It also contains and its inverse property cito:isCitedBy, from the original Citation Typing Ontology, (CiTO v1.6), and all the sub-properties are present in their inverted form as well.
It is a fundamental ontology in the application of the Citation Lens, and it will be described in more details in section 4.7

BiRO, the *Bibliographic Reference Ontology*

URL: http://purl.org/spar/biro
SVN: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/BiRO

BiRO, the Bibliographic Reference Ontology, is an ontology structured according to the FRBR model to define bibliographic records (as subclasses of frbr:Work) and bibliographic references (as subclasses of frbr:Expression), and their compilation into bibliographic collections and bibliographic lists, respectively. It imports both the FRBR Core Ontology and the SWAN Collections Ontology (to allow for the description of ordered lists) and it provides a logical system for relating an individual bibliographic reference, such as appears in the reference list of a published article (which may lack the title of the cited article, the full names of the listed authors, or indeed the full list of authors):

1. to the full bibliographic record for that cited article, which in addition to missing reference fields may also include the name of the publisher, and the ISSN or ISBN of the publication;
2. to collections of bibliographic records, such as library catalogues; and
3. to bibliographic lists, such as reference lists.

It is designed to be a necessary part of a complete bibliographic reference system, and it can be used in conjunction with FaBiO to apply the Publication Context Lens on an entity.

C4O, the *Citation Counting and Context Characterization Ontology*

URL: http://purl.org/spar/c4o
SVN: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/C4O

C4O, the Citation Counting and Context Characterization Ontology (C4O) allows the characterization of bibliographic citations in terms of their number and their context.

It imports and extends BiRO, (thus indirectly importing FRBR Core and SWAN Collections), and it aims to provide the ontological structures required to allow the recording of the number of in-text citations of a cited source (i.e. the number of in-text reference pointers to a single reference in the citing article's reference list), and also the number of citations a cited entity has received globally, as determined by a bibliographic information resource such as Google Scholar, Scopus or Web of Knowledge on a particular date.
Moreover, it enables ontological descriptions of the context within the citing document in which an in-text reference pointer appears, and permits that context to be related to relevant textual passages in the cited document.

DoCO, the *Document Components Ontology*

URL: http://purl.org/spar/doco
SVN: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/DoCO

DoCO, the Document Components Ontology, is designed to provide a structured vocabulary written in OWL 2 DL of document components, both structural (e.g. block, inline, paragraph, section, chapter) and rhetorical (e.g. introduction, discussion, acknowledgements, reference list, figure, appendix), defined by the imported Discourse Elements Ontology (DEO). As such, it allows the description in RDF of these components and of documents composed by them. Given its important role in the application of the Rhetoric Lens, both DOCO and DEO will discussed in further detail in section 4.5 and 4.6

PRO, the *Publishing Roles Ontology*

URL: http://purl.org/spar/pro
SVN: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/PRO

PRO, the Publishing Roles Ontology, is an ontology written in OWL 2 DL for the characterization of the roles of agents in the publication process, whether they are people, corporate bodies or computational agents. It allows to specify how an agent has a role relating to a bibliographic entity, and it permits the recording of time/date information about the period of time during which that role is held.

Because it is based on the Time-indexed situation ontology pattern, it is easy to extend the set of specified roles, simply by adding new individuals to the class pro:Role.

PSO, the *Publishing Status Ontology*

URL: http://purl.org/spar/pso
SVN: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/PSO

PSO, the Publishing Status Ontology, is an ontology written in OWL 2 DL for characterizing the publication status of a document or of other bibliographic entities at each of the various stages in the publishing process (e.g. draft, submitted, under review, rejected, accepted for publication, proof, published, Version of Record, catalogued, archived).
Because it is based on the Time-indexed situation ontology pattern, it is easy to extend the set of specified statuses, simply by adding new individuals to the class pso:Status.

PWO, the *Publishing Workflow Ontology*

URL: http://purl.org/spar/pwo
SVN: http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/PWO

PWO, the Publishing Workflow Ontology, is an ontology written in OWL 2 DL for describing the steps in the workflow associated with the publication of a document or other publication entity (e.g. being written, under review, XML capture, page design, publication to the Web).
It is based on the Time-indexed situation pattern to describe workflows steps and on the Sequence pattern to define their order.

3.5 Toulmin Argument Model

Before concluding this section and moving onwards to the discussion over Semantic Lenses and the methodology in sections 4 and 5, another important introduction is in order. We have already mentioned the opportunity and the importance of modeling with appropriate markup the argumentative structure of a scientific document and of its contents as one of the significant aspects worthy of being better highlighted within Semantically Enhanced papers. We are going to delve deeper in this facet while exploring the Argumentation Model Lens in section 4.3.7 and the ontology related to it, AMO [VP11], in section 4.8, but it is important to introduce the basis of the Argumentation Model we are going to use.

Among the many possible argument model descriptions, Stephen Toulmin's Model detailed in [Tou59] is one of the seminal work in the field, in which he suggests several answers about Argumentation Theory and develops a structural model of "practical arguments" by which rhetorical arguments can be analyzed, focusing on the justificatory scope of argumentation. He observed that effective, well formed and realistic arguments typically consist of six interlinked, explicitly denoted components.

Believing that "*logic is generalized jurisprudence*" [Tou59], and criticizing the over-simplification of classical syllogism and similar model imply, Toulmin observes that *"Many of the current problems in the logical tradition spring from adopting the analytic paradigm-argument as a standard by comparison with which all other arguments can be criticized. But analyticity is one thing, formal validity is another, and neither of these is an universal criterion of necessity, still less of the soundness of arguments"* and overturns the classic inferential model of theoretical arguments, and arguing that reasoning is a process of testing and improving over already existing ideas, an act which requires that practical arguments should declare a claim of interest, and then provide justification for it. *"There must be an initial stage at which the charge or claim is clearly stated, a subsequent phase in which evidence is set out or testimony given in support of the charge or the claim, leading on to the final stage at which a verdict is given, and the sentence or other judicial act issuing from the verdict is pronounced"*.

Toulmin starts from a simple three elements model organized as follows. When structuring an argument, we make an assertion, which is our claim. Then we are being challenged to identify the justification behind our claim, and finally, we are to answer how we go from said justification to our claim (in a sense, we have to justify how we step from evidence to claim). At this moment, the model corresponds to the simple structure of **CLAIM – DATA – WARRANT.**

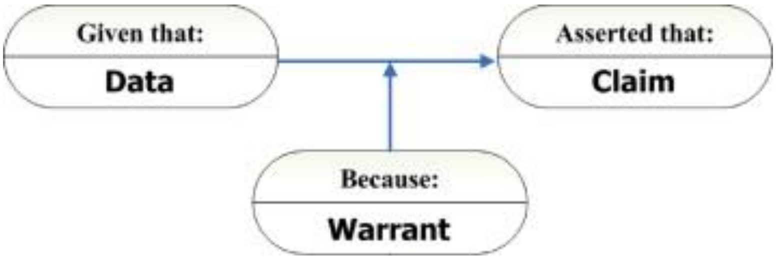


Fig. 11 - The core components of Toulmin's Model

However, this is only the most basic type of reasoning admitted as a valid argumentation by Toulmin. Toulmin's full model comprehends all the following elements:

- **Evidence** (or Data): The facts or the data used as grounds to prove the argument. It is important that the grounds themselves are not challenged, or, if they are, then they should be at least the resulting claim of another properly built practical argument.
- **Claim:** The assertion or the thesis being argued and proponed.
- **Warrant:** The general, hypothetical (and quite often, implicit or very concise) statement used as logical connectors between the Evidence and the Claim. The Warrants are the crucial link between evidence and claim, and as such an argument is only as strong as its weakest warrant.
- **Qualifier:** Statements that limit the strength of the argument or statements that specify under which conditions the argument holds true. They can be reservations, modal qualifiers, probability statements or assertion on significance.
- **Rebuttal:** Counter-arguments indicating situations where the general argumentation is not considered true. They are anticipated and expected exceptions to the Claim.
- **Backing:** Statements which serve to provide additional support to the warrant, like other arguments proving that the warrants are true.

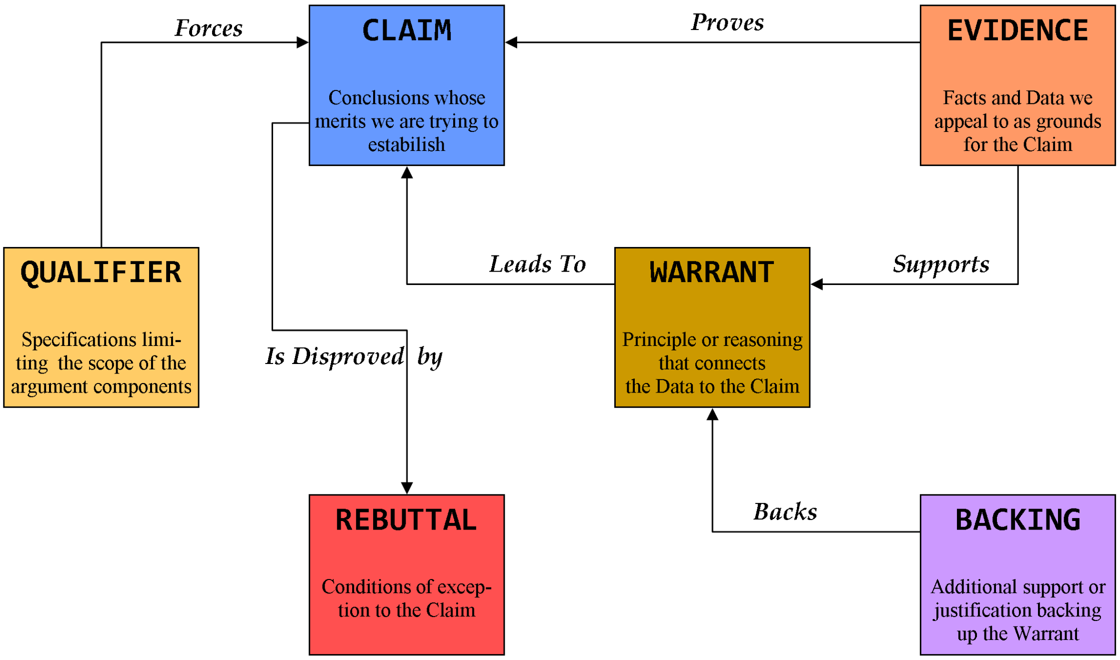

Forces
CLAIM
Conclusions whose merits we are trying to estabilish
Proves
EVIDENCE
Facts and Data we appeal to as grounds for the Claim
QUALIFIER
Specifications limi-ting the scope of the argument components
Leads To
WARRANT
Principle or reasoning that connects the Data to the Claim
Supports
Is Disproved by
REBUTTAL
Conditions of excep-tion to the Claim
Backs
BACKING
Additional support or justification backing up the Warrant

Fig. 12 – The overall Architecture of Toulmin's Argument Model

It is very important to observe that Toulmin's model does not aim to provide any kind of judgment on the truthfulness of a claim, or on the correctness of the contents an argument components. It is not used to determine if an argument corresponds to the truth, whatever that might be in the case, but it is used to validate if an argument is well structured and thus if it could POSSIBLY be true, or if, on the contrary, it does not even have the chance to stand on its own feet.

It should also be important to remember that an argument correctly written according to the model reveals both its limits and its strengths, as it should be: No argument should strive to apply further than it is meant to. This is because, in step with the jurisprudence similarity, arguments are not simply expressed as absolutes, but rather expressed in a way that lets the reader know how far to take the reasoning, and at which conditions it should apply.

Finally, Toulmin's Argument Model closely resembles a fractal, as all components, with the exception of claims (at least usually), can be (and often are) the results of other arguments, and so on. It also does support fully circular argumentations.

Here's a parting example from [Mik07], right from the start of its 4th section, with the pieces colored and identified according to their roles.

[Qualifier] In absence of a golden standard, evaluating the results of ontology learning or ontology mapping is a difficult task:[/Qualifier] [Claim] inevitably, it requires consulting the community or communities whose conceptualizations are being learned or mapped.[/Claim] [Evidence] In order to evaluate our results, we have thus approached in email 61 researchers active in the Semantic Web domain, [/Evidence] [Qualifier] most of whom are members of the ISWC community and many of them are in the graph-theoretical core of the community.[7] [/Qualifier] [Evidence] The single question we asked was *In terms of the associations between the concepts, which ontology of Semantic Web related concepts do you consider more accurate?* [/Evidence] [Rebuttal] Lacking a yardstick, there is no principled correct answer to this question that we expected to receive. [/Rebuttal] [Warrant] Instead, we were interested to find out if there is a majority opinion emerging as an answer and if yes, which of the two ontologies (produced by the two different methods) would that majority accept as more accurate. [/Warrant]

4 The Semantic Lenses Model

4.1 Introduction to Semantic Lenses

As we already discussed in Sections 1 and 2, Semantic Publishing is the use of Web Technologies (especially those related to Semantic Web) to enrich a published scientific document, such as a scholarly journal article, thus aiming to enable several important features such as the ability to define a formal representation for the meaning of the paper and of its content, the enabling of its linking to other semantically related content (which could be discovered at runtime), the possibility to facilitate the automatic discovery of both the paper and its metadata within the Linked Open Data initiative, the provision of actionable and interactive data and data fusion between different sources. We had also discussed the importance and the interest for Semantic Publishing within the scientific publishing domain, as exemplified by initiatives like the Elsevier Grand Challenge or SePublica.

As already illustrated, the enhancement of a traditional scientific publication by the use of RDF annotations to serve as either semantic markup or to convey the meaning of markup semantics is not a simple, straightforward operation, as it is not just limited to making specific statements about its content or about named entities within the text. The scope of Semantic Lenses is to be able to give the user the possibility to choose within a set of different views over which he could **focus** on a specific aspect of the document, enhancing its understanding of the subject and facilitating the emergence of meaning. Of course, for the user to be able to do so, Semantic Lenses have first to be properly and methodically **applied** as metadata to the scientific document in question, associating the markup of each semantic lenses within the proper parts of the article.

I had already informally introduced some of the several aspects that can characterize a paper besides its mere textual content, such as its rhetorical or argumentative structure, or the use and context for the citations and data included therein. I am going to review them a little more at length, within a general context and in natural language, in section 4.2, before introducing and discussing their formalization as part of the Semantic Lenses Stack in section 4.3, where I will be detailing each of the 8 Semantic Lenses defined, both in scope as well in technologies suggest for its implementation, and each will be supplemented by short examples.

After that, considering that my final aim for this section of the dissertation is to illustrate the methodology that I adopted for annotating 4 specific Semantic Lenses (*Structure, Rhetoric, Citation, Argumentation*) on [Mik07], I will proceed to introduce in more detail the specific Ontologies used to do so, and they will be reviewed from section 4.5 to section 4.9.

4.2 Facets of a Document – Different outlooks, aspects and layers.

In section 2.1, I had introduced briefly how many different discernible and relevant aspect coexist within a scientific paper such as scholarly journal article, and observed how they all contribute to the final interpretation and understanding of its meaning, the one that is created within the mind of the reader, almost like all the complex and intricate gears and cogs within a clockwork device all have to interact for it to function correctly.

However, they are far more than simple components, meaningless in themselves without the others. In practice, while the mental image that is the end result of our comprehension of written communication is dependent on all these aspects and their meaning within the plain text itself, these facets do not lose importance when considered alone with just the main textual content, unlike the gears of the previous metaphor. A more correct similitude would then be one to atlases and geographical maps. Consider the map of a continent – Besides the basic contours of the lay of the land, there are so many information and data that could be of interest in describing the area: the political layout, geographic information about the altitude or about the type of terrain, satellite view, average climate, temperature and weather patterns, economical indicators and most important imports and exports, administrative organizations, main routes of transportations, etc., and yet there's only so much we can put on a single map before it gets too cluttered to be understood, becoming meaningless. Often we will have to settle to for a mixed map highlighting a bit of the data deemed more important or searched more often. But, if we want, all those other specialized maps are there ready to be consulted – traditionally they were available as separate entities (much like literary criticism on a text was a separate document), but now there's plenty of tools to have them show up as interactive layers, ready to be applied or removed at the user convenience (for a notable example, see the US National Atlas Mapmaker [14]).

The idea behind Semantic Lenses (and semantic publishing in general) is to enable, in the future, users browsing and researching scientific knowledge stored in enriched scholarly paper to do likewise, with all aspects and layers that could be part of a scientific publication, instead of geographical maps. That being clearly asserted, I will now proceed to list the most relevant of these possible meaningful context layers. Of course, some might be considered

[14] US National Atlas Mapmaker tool, available at: http://nationalatlas.gov/mapmaker

important more frequently than others, (it all depends on what kind of information you are after), and some might be a characterization of the whole document (like data on the publishing process and status of a work), as opposed to meta-data about specific parts of the document (like the rhetorical attributes for a certain block of text). These aspects thus include:

- The context behind the origin of a publication, including the motivation, the general field (and possible related keywords), the background, the source of research grants, the sponsors, the institutions involved in the research
- The people involved in authoring, editing and publishing a document, and in general information about the contributors, their roles, their affiliation or background, and detailed information about which specific contributions did they make to a paper, or which parts were authored or reviewed by each person.
- The status of the publication, and data on the publishing process of a paper, including information about its inclusion in journals, conference proceedings, books, annuals and so forth.
- The structure of a paper and its organizations in specific components (from chapters to paragraphs, from tables to inline components)
- The already mentioned rhetorical denotation of the paper components, and the overall rhetorical organization of discourse within a document
- The citations and the quotes within a paper, their role, scope and purpose, within the citing Work. Their characteristics, the denotation of the section of the document in which they are cited, their relationship to it, or between the authors
- The argumentation model, as we have seen in section 3.5, including the claims or thesis made by the authors, the data and warrants associated to them, the conditions under which these assertions hold and which sub arguments are called in support (or in rebuttal) to which part of the text.
- The semantics of the text itself, phrase by phrase, entity by entity, assertion by assertion.

As I will illustrate in the following section, to each of this informally defined facets, a correspondent Semantic Lens has been defined, so that, when all the lenses are taken together, it will be possible to have a complete and semantically sound description of a scientific publication.

4.3 Semantic Lenses in Detail

As already stated, the semantics of a written document, especially a scientific one, could be defined by applying appropriate meta-data markup for different perspectives on the document itself or its content. Any of these different view could be considered as an independent Semantic Lens **applied** to the document, which then could be brought into the spotlight by the reader of a paper, with the act of focusing the lens highlighting a chosen facet.

In the previous section I have just listed and described informally eight different aspects for the complete characterization of a scientific paper. Here is a list where to each of these eight aspects is associated to one of the Semantic Lenses formally proposed in [PSV12a] and [PVZ12].

1. **Research Context Lens:** This lens covers the background from which the publication originated, including the nature and the field of the research described in the paper, the motivations, the sources of funding and possible sponsors, the nature and the details of the grants, the administrative process behind it, the institutions involved in the research, and so on.
2. **Contribution and Roles Lens:** This lens provides information about the individuals involved in the authorship of the semantically enhanced paper, and delivers meta-data on the people who had any peculiar authorship role with the publication, and which were its contributions to the Work.
3. **Publication Context Lens:** It is the lens including all the data about the publication status of the document, and all information related to the event, the publication or the journal to which the paper is associated and has appeared (or is expected to appear). It is also the correct place to provide links associating the document to the other papers sharing the same publication context, e.g. listing the references for other papers published within the same volume or presented at the same conference.
4. **Document Structure Lens:** Unlike the previous three, this is the first lens that involves a description not just related to the document as a whole, but to its specific contents. In particular, the structure Lens aims to describe the paper’s structural markup semantics for its most components (e.g. by linking specific markup elements to the role of blocks, inline elements, containers, tables, etc.) and to hold information about the way its component are arranged, presented and organized.

5. **Rhetoric Organization Lens:** This lens contributes metadata about the identification and the organization of the rhetorical components of the document, storing information about both the rhetorical discourse and the rhetorical structure of the document. Thus it can assist the reader both by identifying the rhetoric hierarchy of a certain component (e.g. this markup item is a paragraph, this other denotes a section, this one is a title, and so on) and its role within the overall discourse, e.g. by branding the component as an Introduction, a Quotation, some Data, a Discussion, etc.
6. **Citation Network Lens:** As we had already anticipated, this is one of the lenses that are most relevant to scientific research in the perspective of both inter-document and intra-document interactions. This lens provides all the metadata related to the citations part of the document, citation by citation. Each can be associated to information about its purpose, linked to its target document (which could ideally be another enriched paper, or at least be reachable within the LOD), and in short it allows the annotation of semantics relevant to the reasons behind every individual reference within a paper, potentially allowing to build a citation network (both within the paper and at an higher, inter-document level).
7. **Argumentation Lens:** Within this lens it is stored the argumentation model of the semantically enriched scientific paper. This lens allows to define and markup argumentations within the text, and to denote their inner structure, identifying specific components, such as claims, data, warrants, and so on, according to Toulmin's Argument Model introduced in section 3.5. markup items, structural or rhetoric components, or specific pieces of text are consequently assigned a role (if relevant) within this model of the argumentative structure of the paper.
8. **Textual Semantics Lens:** Finally, in this lens we reach the most content specific layer for the Semantic Lens model. The final goal of a (scientific) paper is to express findings or concepts that have a specific (scientific) and precise value. This lens serves to highlight the actual meaning of a piece of text itself, entity by entity and statement by statement. While communication of meaning is often designed for human recipients, this lens aims to apply definitions and statements to the text itself as a way to express semantic markup about its content. What could be done is strongly dependent on the actual content and topic of the paper, but some activities are surely within this level, e.g. linking named entities to appropriate domain specific ontologies.

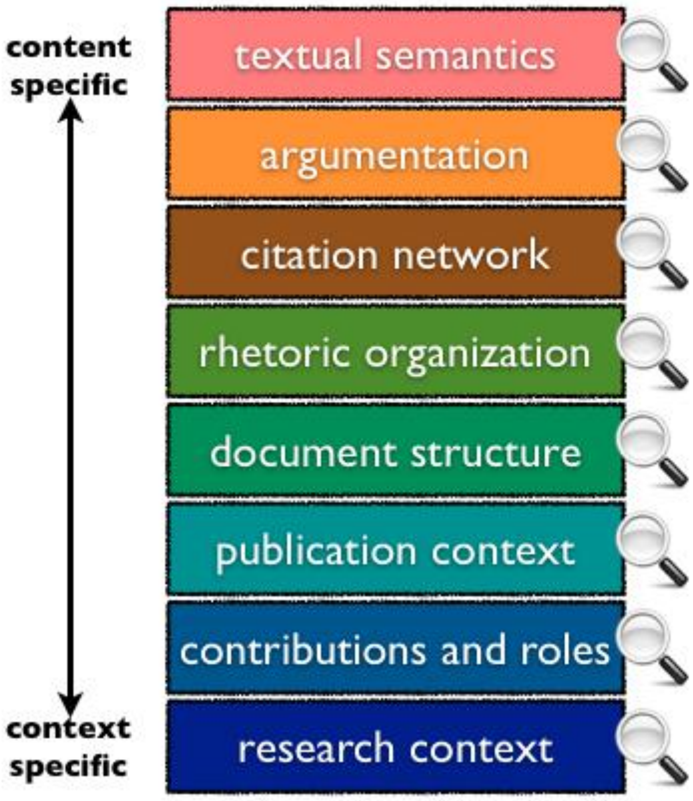


Fig. 13 - The Semantic Lenses Stack

I am now going over each Lens in more depth an detail, explaining the best technologies (mostly in the form of ontologies) suggested for their concrete application, and providing some appropriate examples over [Mik07] for each one, inspired by [PVZ12].

4.3.1 Research Context Lens

Writing a scientific paper is usually the ending stage of a long and complex collaborative process, consisting in undertaking several research activities, ranging from experimentation to data gathering, from background research to analysis. These activities usually involve many people and organizations or institutions, and they also require appropriate funding to be successfully completed. Describing all parties involved is the task of the Research Context Lens. While several other existing ontologies are available, like VIVO and DOAP, in order to describe the contextual environment that made possible writing an enriched it is suggested to use **FRAPO**, the Funding, Research Administration and Projects Ontology , part of the SPAR [Sho10a] suite of ontologies detailed in section 3.2. The following sample excerpt, targeted at [Mik07], specifies the Vrjie University Amsterdam as a University (line 2) that awarded a Ph.D scholarships in 2004 (line 4) to fund the investigation that led to the aforementioned paper (line 10).

```
1.    :research-context {
2.    :vua a frapo:University ;
3.         foaf:name “VU University Amsterdam” ;
4.         frapo:awards [
5.              a frapo:Grant ;
6.              rdfs:label “Ph.D. Scholarship 2004” ;
7.              frapo:funds :investigation ] .
8.    :investigation a frapo:Investigation ;
9.         # Mika's paper
10.        frapo:hasOutput :ontologies-are-us }
```

4.3.2 Contributions and Roles Lens

There are several roles that people can have within research projects part of the context from which the paper originates, as well as there are a variety of roles and several levels of contributions for the authorship of scientific document. The Contributions and Roles Lens deals with the individuals claiming authorship on the paper and with what specific contributions each made

This aspect of semantic description is provided by SCoRO (the Scholarly Contributions and Roles Ontology) and its imported ontology PRO (the Publishing Roles Ontology) [PSV12b], both introduced in section 3.4, which can be used to identify the roles (e.g. being affiliate with VU University Amsterdam during the realization of that paper – lines 6 to 10) and contributions within the context of a paper. (e.g. In this case Peter Mika was the only person writing the paper [Mik07] – lines 11 to 16)

```
:c-and-r a lens:ContributionsAndRolesLens .
:c-and-r {
:mika a foaf:Person ;
      foaf:name “Peter Mika” ;
      pro:holdsRoleInTime
            [
                  a scoro:OrganizationalRole ;
                  pro:withRole scoro:affiliate ;
                  pro:relatesToOrganization :vua ;
                  pro:relatesToDocument :ontologies-are-us ] ,
      scoro:makesContribution [
            a scoro:ContributionSituation ;
            scoro:withContribution scoro:writes-paper ;
            scoro:withContributionEffort
                  scoro:solo-effort ;
            scoro:relatesToEntity :ontologies-are-us ] }
```

4.3.3 The Publication Context Lens

This third context-specific Lens documents the context in which a scientific document is written and published, and its importance is especially in explaining how the paper is grouped with other documents and publications. It allows, for example, to know which book, journal or annual contains the article, or to which conference or workshop it could be associated, allowing connections to other related scientific pieces, sharing the same context. As such, it is a lens very much aimed at inter-document applications.

There are two widely used ontologies for the descriptions of bibliographic entities, which are the already mentioned BIBO and FRBR, but their model do not really respond to requirements for this lens. However, as we already mentioned, the fact that FRBR is not tied to a specific metadata schema or implementation turns to our advantage: Two ontologies were developed within SPAR (see again section 3.4 for more details) for the purpose of asserting metadata on the publication context. It is thus possible to describe it using FaBiO, the FRBR-aligned Bibliographic Ontology [PS12] and BiRO, the Bibliographic Reference Ontology, specifying the journal in which the paper was published (lines 4 to 12) and the list of its references to other related documents (lines 13 to the end):

```
1. :publication-context a lens:PublicationContextLens .
2. :publication-context {
3. # The textual realization of the paper
4. :version-of-record a fabio:JournalArticle ;
5. frbr:realisationOf :ontologies-are-us ;
6. frbr:partOf [ a fabio:JournalIssue ;
7.   prism:issueIdentifier "1" ;
8.   frbr:partOf [ a fabio:JournalVolume ;
9.        prism:volume "5" ;
10.  frbr:partOf [ a fabio:Journal ;
11.       dcterms:title "Web Semantics: Science, Services
12.            and Agents on the World Wide Web" ] ] ] ;
13.  frbr:part [ a biro:ReferenceList ;
14.       co:element [ biro:references
15.  <http://dx.doi.org/10.1007/978-3-540-24571-1_2> ] ,
16.  … ] }
```

4.3.4 The Document Structure Lens

As I had already anticipated, this is the first lens focusing more on the content of the document rather than its context or the document as a whole. This lens aims to provide basic information about the markup semantics of the structure of the document, and denote the role of an element within the structural organization of the paper.

Usually, the structure of a textual (scientific) document is expressed through the use of markup languages such as XML (XHTML, DocBook…) or LaTeX, which have plenty of constructs available to describe a tree-like hierarchy of the content structure. But within the Semantic Web domain, it would preferable to have the document represented as an ontology that describes the markup structures, possibly in OWL. So if EARMARK (which we already introduced in section 3.2) is used to represent a document, Lenses are able to support a far wider range of hierarchies, as well as overlapping markup.

In any case, what really matters within this facet is the possibility to assign a determined and specific structural role to relevant elements of the hierarchy. To do so, the Semantic Lenses approach recommends the use of the Patterns Ontology [DFP08], which I will present in more detail later within this section. The authors of [DPP12] have identified eleven structural patterns, and most complex and different structural components can be assigned a role within one of these, as they have proven to be sufficient to explicitly characterize the structure of most documents, especially scientific papers, and they are mostly independent from the underlying document format itself.

Thus, with the use of the Pattern Ontology, in combination with EARMARK I have been able to assign specific structural semantics to markup elements, such an element `<h2>` expressing the concept of being a block of text (lines 3-8), or the `<div>` element containing it being a container (lines 9-18), as shown in the following excerpt of the Structure Lens for [Mik07]:

```
:structure a lens:StructureLens .
:structure {
:div a earmark:Element ; # Container of the text
   la:expresses pattern:Container ;
   earmark:hasGeneralIdentifier “div” ;
   c:firstItem [ c:itemContent … ; c:nextItem [
   c:itemContent :h-sec-2 ; … c:nextItem [ …
   c:itemContent :p4-sec-2 … ] ] ] .
:h-sec-2 a earmark:Element ; # Title of Sec 2
   la:expresses pattern:Block ;
   earmark:hasGeneralIdentifier “h2” ;
   c:firstItem [ c:itemContent :r-h-sec-2 ] .
   # The title text node
   # “A tripartite model of ontologies”
:r-h-sec-2 a earmark:PointerRange …
:p4-sec-2 a earmark:Element ; # Sec 2, Par 4
   la:expresses pattern:Block ;
   earmark:hasGeneralIdentifier “p” … }
```

Both the application of the Structure Lens and the Pattern Ontology will be better explained further on in the document.

4.3.5 The Rhetoric Organization Lens

Rising in the Semantic Lenses stack as we get nearer to the aspects more related to content, the next Lens we encounter is the Rhetoric Organization one, which I had defined as the one tasked of describing the organization of the rhetorical components of the document, storing information about both the rhetorical discourse and the rhetorical structure of the document.

As anticipated, this is a twofold task: On one side it is possible to describe both the rhetoric meaning of a component within the structure of a document, by identifying, for instance, a markup item <p> with a Paragraph, or a specific <div> item with a Table Box. On the other hand, there is also the need to denote the role of a component within the rhetorical organization of discourse, and thus this Lens makes it possible to assert that said Paragraph can also be seen as a Summary or a Discussion, while that figure box might contain Data or a Caption.

Such rhetoric characterization of markup structures can be specified through DoCO [SP11a], the Document Components Ontology, and DEO [SP11b], the Discourse Elements Ontology, both part of the SPAR suite of ontologies [Sho10a] already introduced in section 3.4. The following example adapted from the application of this lens on [Mik07] expresses that the elements div, h2 and p introduced in the previous excerpt represent, respectively, the front matter of the paper (line 3), a section title (line 4), and a paragraph introducing some background assets (lines 5-6):

```
1.   :rhetoric a lens:RhetoricLens .
2.   :rhetoric {
3.   :div la:expresses doco:FrontMatter .
4.   :h-sec-2 la:expresses doco:SectionTitle .
5.   :p4-sec-2 la:expresses doco:Paragraph ,
6.        deo:Background . # etc. }
```

Both the application of the Rhetoric Lens on [Mik07], DOCO and DEO will be better explained further on in the document.

4.3.6 The Citation Network Lens

The measuring of citations between research papers is widely acknowledged as a very important metric in evaluating the impact and the productivity of scientists and of research projects. At the moment, citation metrics just take in account the simple fact that one paper cites another, but there is a substantial difference between citing a paper because it is being considered an important source or a seminal work within a field, or citing another paper to disprove its findings. Of course, future citation network metrics on the impact of a scientific document could greatly benefit if it could be possible to take into account the reasons for the citation of a source within a scientific publication: if that information could be readily and unequivocally available, it would seem a reasonable consequence, in measuring the impact factor, to weight differently citations according to the motivation behind them. The Citation Network Lens is designed to offer us the tools to encode answers to this problem and to formalize why a paper was cited in a certain context or within a certain document, and with which purpose. As such, this is one of the lenses that offer perhaps the widest possibility for the research community, in terms of inter-document semantics.

However, possible interactions at the intra-document level that might be enabled by this Lens should not be underestimated, as it can immediately offer a quick characterization of the relationship of a document to other known works within his field, as well as other possibilities that we will explore further in the document.

As it is, I can state that a document takes part to a citation network with its cited documents, by taking into account the reasons behind each individual citation in the text – e.g. a document could be cited to express qualification of or disagreement with the ideas presented in the cited paper – which may significantly affect the evaluation of a citation network itself.

For instance, the analysis of the content of [Mik07], like in the 4th paragraph of the 2nd section of the paper (e.g. :p4-sec-2), I encountered several citations to other works that are introduced for a particular reason (in this specific case to express qualification of or disagreement with the ideas presented in the cited papers). Using CiTO, the Citation Typing Ontology [PS12], it could be in theory possible to provide descriptions of the factual or rhetorical nature of the citations, as shown in the following example, where paper #5 is used as an authority (lines 6-7), and as source of conclusions (lines 8-9), while paper #3 is used as a source of background information (lines 11-12) and is corrected by [Mik07] (lines 13-14):

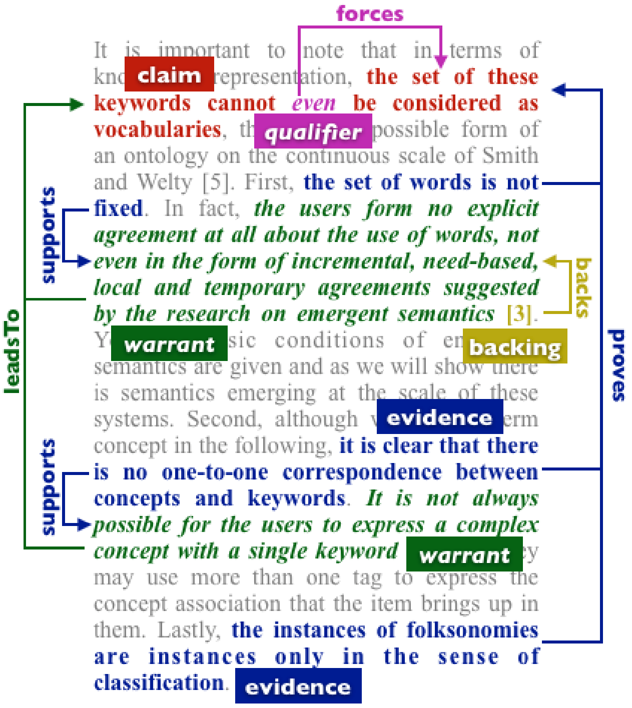
```
1. :citation a lens:CitationLens .
2. :citation {
3. # Sec 2, Par 4
4. :ontologies-are-us
5. # citation to [5]
6.   cito:citesAsAuthority
7.   <http://dx.doi.org/10.1002/047084289X> ;
8.   cito:usesConclusionsFrom
9.   <http://dx.doi.org/10.1002/047084289X> ;
10.# citation to [3]
11.   cito:obtainsBackgroundFrom
12.   <http://dx.doi.org/10.1007/978-3-540-24571-1_2> ;
13.   cito:corrects ,
14.   <http://dx.doi.org/10.1007/978-3-540-24571-1_2> ;
}
```

Both the application of the Citation Lens on [Mik07], and CITO [SP09] will be further explored later on in the document.

4.3.7 The Argumentation Lens

Getting even more closer to content-specific semantics, we have the argumentation organization and structure of the paper. As already explained, this is another crucial facet for both the understanding and the summarization of a scientific paper's contents. The role of a scientific document is to propose hypothesis and corroborate them with relevant evidence, explaining why this evidence fits the ideas suggested by the researchers.

This can be modeled with several argumentation theories, and one of the most useful and widely acknowledged one is Toulmin's Argument Model already introduced in section 3.5, as its suggested model of data, claims and warrant (together with rebuttals, qualifiers and backings) fits quite perfectly most scientific argumentations. Consequently, the Argumentation Lens purpose is to define argumentations within the document, and to provide information about their structure, their components (identifying the single components, such as claims, data, warrants, and so on) as well modeling the relationships between the argumentations, all in accordance with Toulmin's Argument Model and its fractal organization of argumentations. There are some ontologies able to model an argumentation within a paper, such as the SALT ontology mentioned in the related works section, but the suggested choice to express an Argumentation lens over a paper is AMO, the Argument Model Ontology, an ontology which implements in OWL Toulmin's model of argumentation, designed with in mind its application within the field of Semantic Publishing.



It is important to note that in terms of kn representation, the set of these keywords cannot *even* be considered as vocabularies, th possible form of an ontology on the continuous scale of Smith and Welty [5]. First, the set of words is not fixed. In fact, *the users form no explicit agreement at all about the use of words, not even in the form of incremental, need-based, local and temporary agreements suggested by the research on emergent semantics* [3]. Y sic conditions of e semantics are given and as we will show there is semantics emerging at the scale of these systems. Second, although erm concept in the following, it is clear that there is no one-to-one correspondence between concepts and keywords. *It is not always possible for the users to express a complex concept with a single keyword* y may use more than one tag to express the concept association that the item brings up in them. Lastly, the instances of folksonomies are instances only in the sense of classification.

Fig. 14 - Example for the Argumentation Lens, from [Mik07]

The image in the previous page and the following excerpts show a preliminary and “simplified” (in order to offer a better summarization) application of the Argumentation Lens to the already discussed fragment of [Mik07]. This is the argument organization of the third paragraph of Section 2 in Mika's paper:

```
:argumentation a lens:ArgumentationLens .
:argumentation {
:argument a amo:Argument ;
     # the set of these … vocabularies
      amo:hasClaim :r-claim-p4 ;
     # even
     amo:hasQualifier :r-qualifier-p4 ;
     amo:hasEvidence
          # the set of words is not fixed
          :r-evidence-1-p4 ,
          # it is clear that … and keywords
          :r-evidence-2-p4 ,
          # the instances of … classification
          :r-evidence-3-p4 ;
     amo:hasWarrant
          # the users from no … semantics
          :r-warrant-1-p4 ,
          # it is not always … single keyword
          :r-warrant-2-p4 ;
     amo:hasBacking
          # "Emergent Semantics Principles and Issues"
     <http://dx.doi.org/10.1007/978-3-540-24571-1_2> .
:r-qualifier-p4 amo:forces :r-claim-p4 .
:r-evidence-1-p4 amo:proves :r-claim-p4 ;
     amo:supports :r-warrant-1-p4 .
:r-warrant-1-p4 amo:leadsTo :r-claim-p4 .
<http://dx.doi.org/10.1007/978-3-540-24571-1_2>
     amo:backs :r-warrant-1-p4 .
:r-evidence-2-p4 amo:proves :r-claim-p4 ;
     amo:supports :r-warrant-2-p4 .
:r-warrant-2-p4 amo:leadsTo :r-claim-p4 .
:r-evidence-3-p4 amo:proves :r-claim-p4 }
```

Both the application of the Argumentation Lens on [Mik07], and AMO will be explored in better details further on in the document.

4.3.8 The Textual Semantics Lens

We conclude the Semantic Lenses stack with the most content specific of the lenses, addressing the most content specific layer, the one dealing with the literal meaning of statements, assertions and words (as named entities perhaps parts of Ontologies). The Textual Semantics Lens analyzes the final formal meaning of ideas, definitions and relationships expressed in natural language.

For example, the formal description of a claim needs to be expressed in such a way as to represent as faithfully as possible the meaning of the entities involved in the claim itself.

As each document usually provides content that is very domain-specific, there is no universal classification of knowledge in the form of an Ontology suggested for the implementation of this lens. Since it is impossible to provide an encompassing ontology to express this lens, it is rather suggested to choose those most apt to serve our purposes within that knowledge domain. However, In some cases, the claim of an argument can be encoded through using a simple model, e.g. DBPedia [BLK09], as shown in the following excerpt, while in other more appropriate specific ontologies exist.

```
:semantics a lens:SemanticsLens .
:semantics {
:my-keywords a dbpedia:Set_(mathematics) ,
    [ a owl:Class ;
          owl:complementOf dbpedia:Vocabulary ] }
```

4.5 Structural Patterns and the Pattern Ontology

In introducing the definition of the *Document Structure* Semantic Lens, I had already mentioned that its purpose is to provide assertions that enable the association of a predetermined and unambiguous structural role to relevant (markup) elements of the document hierarchy.

To do so, we mentioned using Structural Patterns and the Pattern Ontology [DFP08], as introduced in [DPV11c] and refined in [DPP12]. In order to better understand the proposed methodology for the application of the Structure Lens and the motivation behind my implementation choices, a quick overview of the conceptual architecture of the Structural Patterns model is in order, and I will provide it in this section.

Patterns have been first suggested and developed in [DPV11c], as a way to allow EARMARK (see section EM) to explicitly express structural assertions on syntactic markup structures and the adherence to content model constraints for a document hierarchy as represented in an EARMARK document. In general, patterns are first and foremost a meta-level theory for the description of document structures and of the requirements of use for the structural markup, which has been then formalized in the OWL Pattern ontology.

The fundamental intuition is that, regardless of the many different possible vocabularies that can be used to express the overall syntactic structure of a document, like DocBook or XHTML, all these share some well-established patterns, thus creating meta-structures (like containers, blocks, inline elements, meta information placeholders, etc.) which are recurrent and persisting over the whole spectrum of these languages, and as such could be researched and used to generate a more general and schema-independent description of a document's building blocks. By doing so, not only we do gain an improved understanding of what are the fundamental structural components of document, but we can identify underlying mechanisms over which it will be possible to work and de-structure, re-structure or simply analyze documents and their markup semantics (within the structural facet) even without being tied to a specific schema or presentational rendering (such as a stylesheet).

So, instead of focusing in an effort to catalogue all the aspects of a domain, from the most widely used to the most particular cases, Structural Patterns [DPP12] approaches the issue in a minimalistic way, aiming to create as few classes as necessary to represent all possible persisting conceptualizations common to most document and their components, in order to segment the structure into atomic components that can then be manipulated independently and reflowed, re-constructed or de-constructed within different contexts for different purposes.

A simpler model eases documents' processing and modeling by other applications, as well being less prone to errors and misinterpretations by reducing choices and ambiguities. Patterns are thus expected to have two main characteristics:

- **Orthogonality** – each pattern has a specific goal and fits a specific context.
- **Assemblability** – each pattern is to be used only in some locations: that is to say, only within some other patterns (while still allowing for overlapping or mixed-content model items)

As it is, Nine ***abstract*** patterns are defined, and these are used to characterize eleven concrete ***instanceable*** patterns. These patterns allow authors to create unambiguous, manageable and well structured document.

All concrete patterns are organized as part of one of four disjoint abstract classes (*Mixed, Bucket, Flat, Marker*), defined by their ability to contain text or other elements, and which are thus derived by combining the four possible abstract classes for these properties (*Textual, NonTextual, Structured, NonStructured*).

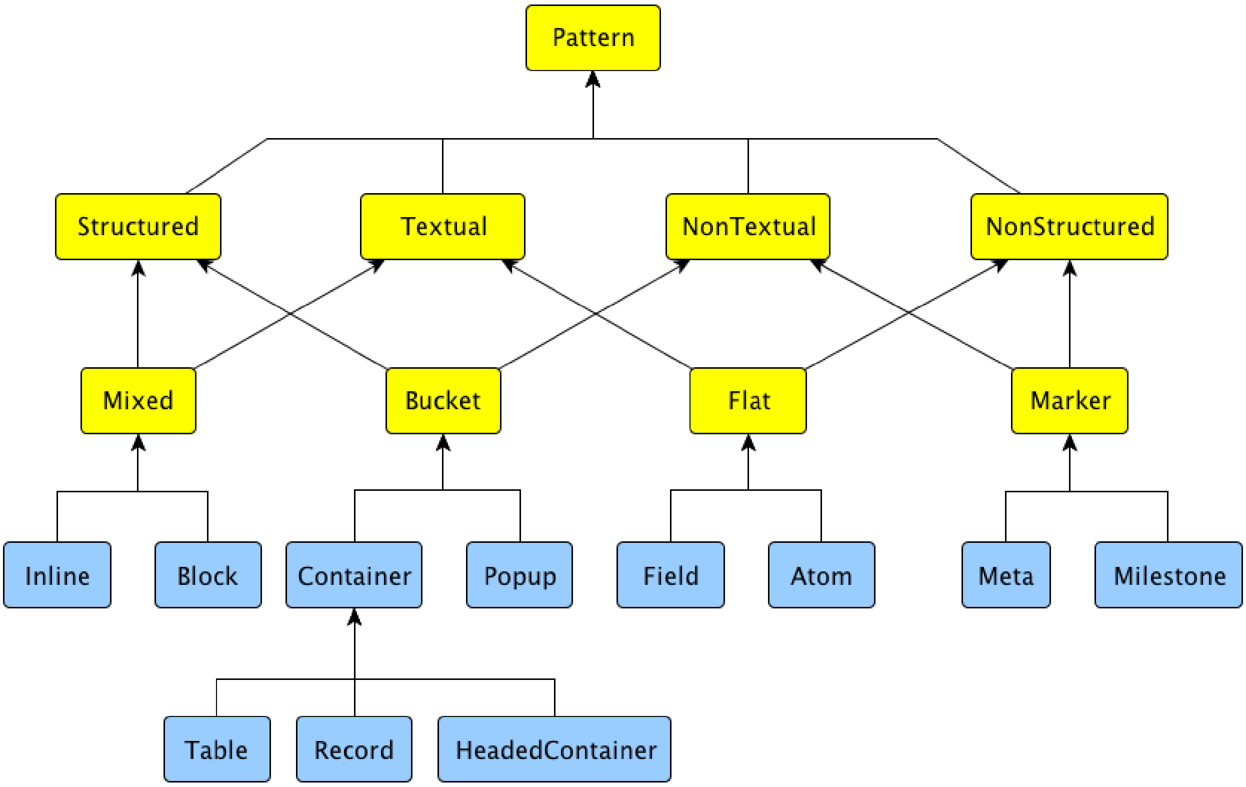


Fig. 15 - The Architecture of the Pattern Ontology

In short, all concrete classes are member of one of these four abstract classes:

- **Mixed**. Individuals of this class can contain other elements and text nodes;
- **Bucket**. Individual of this class can contain other elements but no text nodes;
- **Flat**. Individual of this class can contain text nodes but no elements;
- **Marker**. Individual of this class can contain neither text nodes nor elements.

Formally, these classes are then defined as follows:

```
Mixed ⊑ Structured ⊓ Textual
Bucket ⊑ Structured ⊓ NonTextual
Flat ⊑ Textual ⊓ NonStructured
Marker ⊑ NonTextual ⊓ NonStructured
```

Considering the nature and the purpose of this document, I shall leave further details about ontology design to [DPV11c] and [DPP12]. I will just proceed to give a summary of the different features and meanings of the instanceable patterns. Please note that all subclasses of Container (*Table*, *Record* and *HeadedContainer*) are disjoint, as well as all classes within the same abstract subclass (e.g.: Field is disjoint with Atom).

Table 1 - Summary of instanceable Structural Patterns

Pattern	Description	Examples	Content Model
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. The pattern is meant to represent metadata elements disconnected from the content	script, meta	**Marker** ⊓ ∀ isContainedBy (Container ⊔ Popup)
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. The pattern is meant to represent relevant locations within the text content.	img, br	**Marker** ⊓ ∀ isContainedBy (Inline ⊔ Block). Milestone ⊑ ∃ isContainedBy (Inline ⊔ Block)
Atom	Any simple box of text, without internal substructures (simple content) that is allowed in a mixed content structure but not in a container.	em or span without any internal markup	**Flat** ⊓ ∀ isContainedBy (Inline ⊔ Block). Atom ⊑ ∃ isContainedBy (Inline ⊔ Block)
Field	Any simple box of text, without internal substructures (simple content) that is allowed in a container but not in a mixed content structure.	title	**Flat** ⊓ ∀ isContainedBy (Container ⊔ Popup)
Inline	Any container of text and other substructures, including (even recursively) other inline elements. The pattern is meant to represent inline-level styles such as bold, italic, etc.	a p inside another p, a	**Mixed** ⊓ ∀ isContainedBy (Inline ⊔ Block). Inline ⊑ ∀ contains (Inline ⊔ Atom ⊔ Milestone ⊔ Popup) ⊓ ∃ isContainedBy (Inline ⊔ Block)
Block	Any container of text and other substructures except for (even recursively) other block elements. The pattern is meant to represent block-level elements such as paragraphs.	p, li	**Mixed** ⊓ ∀ isContainedBy (Container ⊔ Popup). Block ⊑ ∀ contains (Inline ⊔ Atom ⊔ Milestone ⊔ Popup)
Popup	Any structure that, while still not allowing text content inside itself, is nonetheless found in a mixed content context. The pattern is meant to represent complex substructures that interrupt but do not break the main flow of the text.	noscript, iframe	**Bucket** ⊓ ∀ isContainedBy (Inline ⊔ Block). Popup ⊑ ∀ contains (Container ⊔ Field ⊔ Meta ⊔ Block) ⊓ ∃ isContainedBy (Inline ⊔ Block)
Container	Any container of a sequence of other substructures and that does not directly contain text. The pattern is meant to represent higher document structures that give shape and organization to a text document, but do not directly include the document content.	a div without text, dl	**Bucket** ⊓ ∀ isContainedBy (Container ⊔ Popup). Container ⊑ ∀ contains (Container ⊔ Field ⊔ Meta ⊔ Block)
Table	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 similar records.	ul, ol	**Container** ⊓ Contains *Homogeneous* Elements: *true* ⊓ Contains Heterogeneous Elements : false
Record	Any container that does not allow substructures to repeat themselves internally. The pattern is meant to represent database records with their non-repeatable fields.	html	**Container** ⊓ Contains Homogeneous Elements: false ⊓ Contains *Heterogeneous* Elements : *true*
Headed Container	Any container starting with a head of one or more block elements. The pattern is meant to represent nested hierarchical elements.	a div containing an h2	**Container** ⊓ ∀ containsAsHeader (Block)

4.6 DOCO – The Document Components Ontology

I have already mentioned DOCO – The Documents Components Ontology [SP11a], both when describing the SPAR (Semantic Publishing And Referencing [Sho10a]) Ontologies and when I introduced and defined the Rhetoric Organization Lens. DOCO is the vocabulary that, according to the Semantic Lens model, should be used to denote the Rhetoric of a scientific paper, by providing information about the organization of the rhetorical components of the document, the rhetorical discourse and the rhetorical structure of the document.

DOCO helps us in this task, by presenting the user with a structured OWL 2 DL vocabulary of document components, both structural (e.g. block, inline, paragraph, section, chapter) and rhetorical (e.g. introduction, discussion, acknowledgements, reference list, figure, appendix), defined by the imported Discourse Elements Ontology (DEO) [SP11b]. Indeed, DOCO is a composite ontology, which imports both the Pattern Ontology I have just introduced, and DEO, on which I will return in a few pages, while here I will just introduce some of the basic concepts behind the core part of DOCO.

This core section of DOCO is mostly used as a way to describe *markup semantics* for document components within the context of the rhetoric hierarchy – that is to say, it allows to denote that, for example, a certain markup item with general identifier <p> item is a paragraph, while this <h2> within a <div> is a section title, and the <div> containing it is a section, and so on.

Most of DOCO is mainly defined as a large set of classes (55), all of which are instanceable, while it uses just the contains and isContainedBy properties of the Pattern Ontology.

Each of DOCO classes usually has several very strict requirements for its usage, both in terms of Pattern and in terms of Discourse Elements, as well of superclasses within the DOCO hierarchy.

On the one side, this reduces ambiguities and the possibility of misinterpretation errors, allowing for more straightforward characterization and less unexplainable choices. On the other hand, however, given the nature of the domain, there is an impressive amount of constraints, and as I had been able to see in my experimental applications, some of DOCO's structures will not be usable if the underlying syntactic markup does not adhere to the ideal modeled by such requirements.

Usually, the DOCO model is based on a certain number of high-level general classes (like "label") and several more constrained specialized sub-classes, (like "chapter label", "figure label", "section label", "table label"), usually disjointed with one another, that require being part of an appropriate Document Component, with an appropriate pattern. Just as an example, I am going to show what is the DOCO organization for a "bibliography" and some related classes, like "bibliographic reference list" (and the "list" class hierarchy) or "section". For the complete documentation of DOCO, see [SP11a]

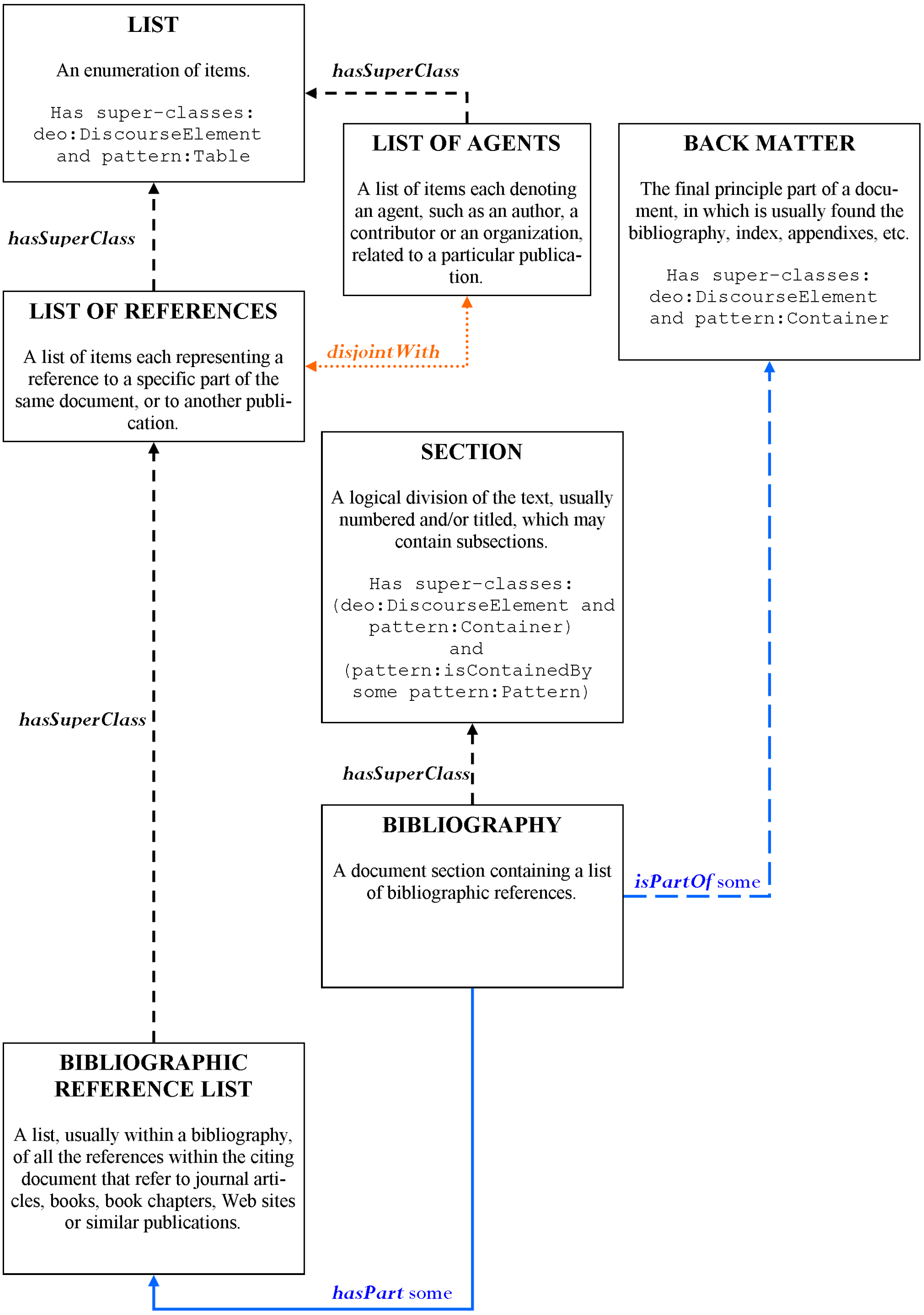

LIST
An enumeration of items.
Has super-classes: deo:DiscourseElement and pattern:Table
hasSuperClass
LIST OF AGENTS
A list of items each denoting an agent, such as an author, a contributor or an organization, related to a particular publication.
BACK MATTER
The final principle part of a document, in which is usually found the bibliography, index, appendixes, etc.
Has super-classes: deo:DiscourseElement and pattern:Container
hasSuperClass
LIST OF REFERENCES
A list of items each representing a reference to a specific part of the same document, or to another publication.
disjointWith
SECTION
A logical division of the text, usually numbered and/or titled, which may contain subsections.
Has super-classes: (deo:DiscourseElement and pattern:Container) and (pattern:isContainedBy some pattern:Pattern)
hasSuperClass
hasSuperClass
BIBLIOGRAPHY
A document section containing a list of bibliographic references.
isPartOf some
BIBLIOGRAPHIC REFERENCE LIST
A list, usually within a bibliography, of all the references within the citing document that refer to journal articles, books, book chapters, Web sites or similar publications.
hasPart some

Fig. 16 - Part of the Architecture of DOCO

4.7 DEO – The Discourse Elements Ontology

The other vocabulary used by the Rhetoric Organization Lens, DEO [SP11b] is a subsidiary ontology imported in DOCO [SP11a], and as such is part of SPAR [Sho10a] as well. The Discourse Elements Ontology is an ontology for describing the most important rhetorical elements of a scientific document, providing a structured vocabulary for the denotation of the rhetorical function of elements (e.g. Introduction, Acknowledgements, Discussion, Appendix, Figures, Results, etc.) and components within a document, thus allowing their role within the overall rhetoric organization of scientific discourse to be described by the means of RDF triples.

DEO defines a single abstract superclass, "Discourse Element" as a subclass of `owl:Thing`, and 30 other classes of individuals, all of them being descendant, directly or indirectly of Discourse Element. Most of the names are self-explaining, and some are imported from SRO, the Salt Rhetorical Ontology. DEO also imports 3 important object properties from Dublin Core, "has relation", and its two sub-properties "has part" and "is part of" which are mostly used by DOCO to define relationships and constraints for document components, as we have seen above.

This is a short list of all DEO Classes. For further documentation, see [SP11b].

- **DiscourseElement**
  - Acknowledgements
  - AuthorContribution
  - Background
  - Biography
  - Caption
    - Legend
  - Conclusion
  - Contribution
  - Data
  - Dedication
  - Discussion
  - Epilogue
  - Evaluation
  - ExternalResourceDescription
    - DatasetDescription
    - SupplementaryInformationDescription
  - FutureWork
  - Introduction

- Materials
- Methods
- Model
- Motivation
- Postscript
- ProblemStatement
- Prologue
- Reference
  - BibliographicReference
- RelatedWork
- Results
- Scenario

4.8 CiTO – The Citation Typing Ontology

The purpose of the Citation Network Lens is to provide a formal encoding of all information related to the citations present within the document, especially regarding their intended purpose and role within the citing document itself. Citation by citation, it is possible to associate with each of them information about the motives behind its choice, and perhaps to link it to its target document.

The Citation Network Lens thus allows the annotation of semantics relevant to the motivation behind every individual citation reference made within a scientific document, giving us the tool to build a semantic citation network where citations within a context could be evaluated more in depth than what would traditionally be possible, thanks to the additional data available to the reader.

To do so, the Semantic Lenses model suggests the use of CiTO - The Citation Typing Ontology (CiTO) [PS12], which is an ontology written in OWL 2 DL designed to enable characterization of the nature or type of citations, both factually and rhetorically. It allows much more than simply asserting in RDF than citations exists, but also encourages to define the factual or rhetorical nature of the citation, and the reasons behind it. In short, CiTO offers the user a way to characterize citations for their factual and rhetorical nature, regardless of them being direct or indirect, implicit or explicit. CiTO properties of a rhetorical nature might also imply a judgment on act of citation, which might be positive, negative, or neutral.

It has evolved from its original formulation (CITO v1.6) and has been recently the target of a remarkable harmonization with the SWAN – Semantic Web Applications in Neuromedicine – Citation Ontology, to obtain version 2.0 [CSP12], before reaching its current status as version 2.1 [SP09]. According to the authors of [CSP12] "*Up until its definition, no public, open, interoperable and complete web-adapted information schema for bibliographic citations and bibliographic references has been made available.*"

CiTO has been initially developed has an ontology for the description of the nature of reference citations in scientific documents, as well as web information resources, and for the publishing of these descriptions within the Semantic Web. It allowed to describe citations in terms of both factual and rhetorical relationships between the citing publication and the cited resources. It also allowed several other characterizations, but many entities not directly related to the citation network that were previously part of CiTO have been moved to other components of the SPAR suite of Ontologies, like FaBiO and C40.

This ontology does not define any Classes for individual entities, but its design revolves around the property "`cites`" and all its possible 33 sub-properties, like "updates, critiques, refutes, extends or obtainsBackgroundFrom" which are to be used to better characterize the semantics of a citation. It also contains its inverse property "`isCitedBy`", from the original Citation Typing Ontology, and all the sub-properties are present in their inverted form as well.

The most recent change in December 2011 has been the addition of two new properties: "*usesConclusionsFrom*" and its inverse "*providesConclusionsFor*", and this led to a version number increment from 2.0 to 2.1 [SP09].

Further changes moved this ontology to its current 2.4 version, with the addition of the "*cito:compiles*" and "*cito:likes*" version, but as these were concurrent to my activity, the version I considered remained CiTO 2.1, in order to remain consistent in my analysis and applications.

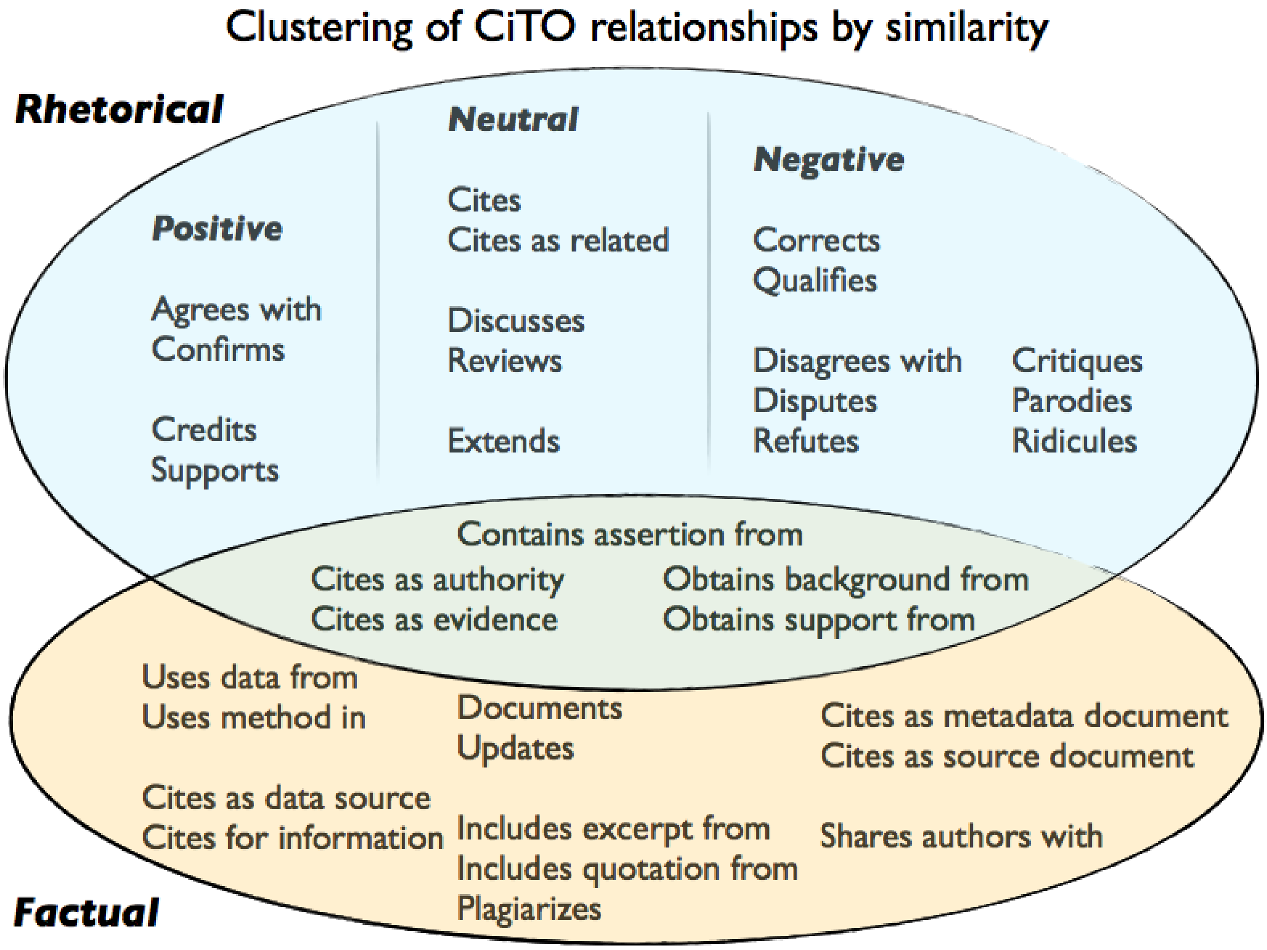


Fig. 17 - CiTO 2.0 properties clustered by their nature

4.9 AMO – The Argument Model Ontology

We know that the Argumentation Semantic Lens aims to describe the argumentation structure within a paper in accordance to Toulmin's Argument Model, which I have already introduced in detail in section 3.5. As I have previously discussed, this model gives us the terms to denote all claims within a document, as well as all those elements supporting, leading to or limiting said claim, and in general allows for a representation of all kinds of scientific argumentations and their components, even when they are nested or overlapping with each other – and to do so is precisely the purpose of the Argumentation Lens.

AMO - The Argument Model Ontology [VP11] is very simply an OWL 2 DL ontology that implements the Toulmin Model of Argument described in [Tou59], and it encodes it through OWL classes and properties, corresponding to those concepts I previously introduced. It does so in order to enable the description of a document's argumentations as a web of inter-linked entities that participate, with a specific role, in one or more arguments. This ontology is also aligned with CITO, and thus is part of the SPAR suite.

The Argument Model Ontology structure postulates two top-tier classes, "Argument" and "Argumentation Entity".

The first is the basic entity corresponding to Toulmin's "practical argument", focusing on the justificatory function of argumentation (first a claim of interest is found, then justification is provided for this claim), and a basic requirement of at least 1 Claim, Evidence and Warrant is established for this entity. A super-property "involves", has Argument as his domain, from which other specific properties (like "has claim") derive.

The second one is a kind of superclass for all specific argumentation model components, like Claim, Warrant or Evidence. Appropriate properties, like "proves" or "leadsTo" are defined to link these entities to one another, to denote argumentations' structure in observance of the model.

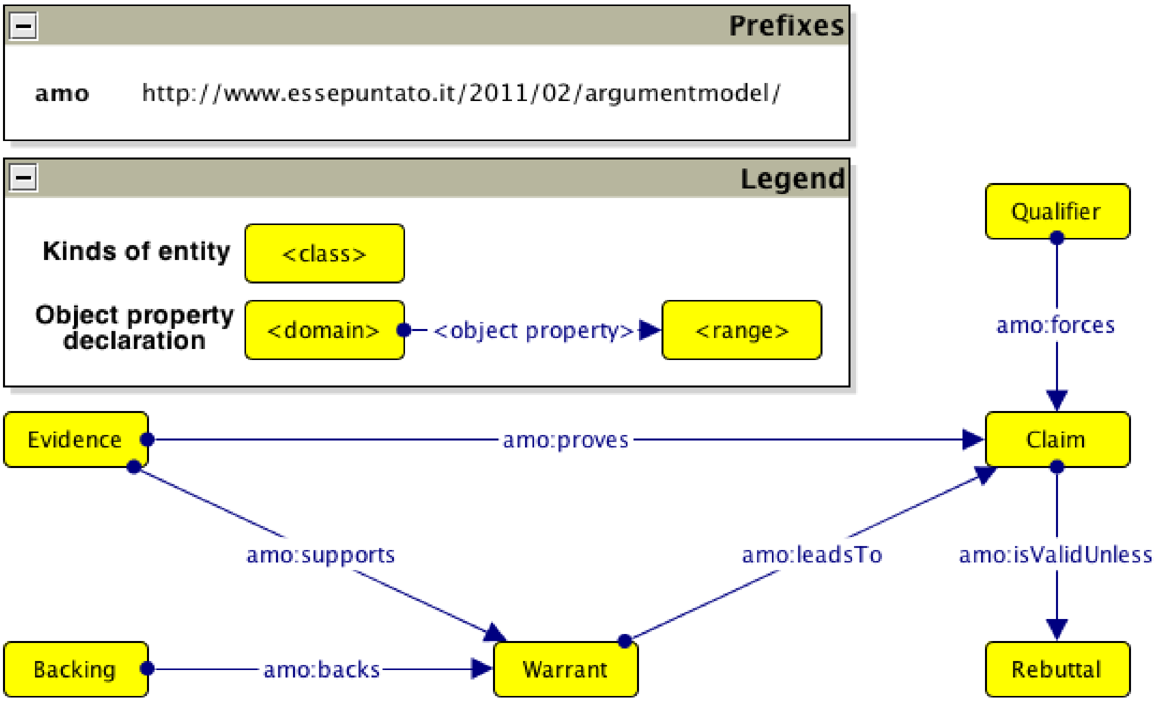


Fig. 18 - The Architecture of AMO

AMO's structure, which gives a lot of flexibility on how to concretely represent an argumentation within its established bounds, is easily summarized in the previous diagram, which closely matches the theory presented in previous part of this dissertation. For further details, see [VP11]

5 SLM – Semantic Lenses Methodology

This dissertation will now proceed with a general discussion on what could be a general purpose methodology for the **application** of Semantic Lenses, on what kind of challenges we are expected to face in doing so, and on how I decided to do a "road test" for the Semantic Lenses model and its related technologies.

I will first briefly discuss which roles the different actors involved in the publishing process, like authors and editors, should take on in the process of annotating Semantic Lenses on a scholarly article. A discussion on the general methodology as well as on the targets of the application, will follow in section 5.2 to 5.5. Finally, in sections from 5.6 to 5.9, I will relate, step by step and lenses by lenses, on the choices, the methodology and the theoretical decisions I had taken to concretely implement the semantic enrichment assertions.

5.1 Applying Semantic Lenses – Authorial or Editorial Activity?

Before examining the methodology for the application of *Structure, Rhetoric, Citation and Argumentative* Lenses that I suggest and before delving deeper in those lenses and their related Ontologies, I think it is important to spend some time and some words in considering an important part of the overall problem of the **Application** of Lenses – namely the simple question: "Who should be involved in this activity?"

Just as a quick reminder, we defined the **application** of Semantic Lenses as the act of annotating and enriching the document with the appropriate metadata specified by Semantic Lenses stack we just illustrated, in contrast with the act of **focusing** of Lenses, which is the act of selecting a specific facet to be viewed and highlighted over the Enriched Document.

While it is clear that the **focusing** a lens is something that will involve any reader of the document, by using appropriately developed tools and interfaces for browsing papers enhanced with lenses (such as the TAL prototype [PVZ12]), there can be more space to debate and discuss the roles involved in

applying lenses over a document, which is the fundamental preparation work that will enable any successive focusing.

In general, the application of any particular lens to an article by adding information about the semantics described by it is an *authorial operation* in the sense that is an act involving individuals acting as agents, responsible for the choice of determined semantic interpretations on a document or its content. As such, tracking lens application is also a problem of *data provenance*, which is about the identification of processes involved in the creation of a resource, and of agents controlling those processes. To do so, the authors of [PSV12a] suggest using the PROV-O Ontology [LSM12], a controlled vocabulary to record the provenance of semantic statements.

However, this does not exactly address what I was aiming to discuss. There is no question about the need to have an author of some sort for the application of Semantic Lenses, and to the benefits of tracking the provenance of semantic assertions on an enriched paper. The main point is to understand the possible relationship between the authorship of Semantic Lenses and the individuals involved in the authorship, editing and publication of the paper being enriched. We have seen that Semantic Publishing involve all levels within the publication chain [Sho09], as both Authors, Reviewers Editors and Publishers might have specific roles that they could fill in producing a semantically enriched scientific document. Moving within the specific domain of Semantic Lenses, it is quite important to identify how all the individuals involved within the process of scientific publication could contribute to the application of Semantic Lenses, and, even more important, there is the need to understand and discuss, for each of the lenses of the stack, if its application is an endeavour for which a determined contributor is best suited, or if, on the other hand, fulfilling the application task is something that could be done at different levels within the publishing chain. In short, the question is: **"Whose duty it is to apply a chosen lens?"**

Sadly, there is no clear-cut answer to this question, but it is surely reasonable to discuss the issue at hand and suggest some guidelines. A very important side of this resides in finding out how much the original authors of the document have to be involved for the generation of semantically accurate Semantic Lenses, and how recommended is their participation in the application of all types of Semantic Lenses.

On the whole, it should be safe to say that the more content specific the Lens is, the more it is important for the Author to be involved in its application. On the other side of the coin, it is possible to say that for the three more context specific lenses, *Research Context, Contribution and Roles* and *Publication Context*, there is no real need to involve the Author of the paper, except perhaps to gather information, and said lenses could be easily created and applied by anyone with the knowledge of the data required to be encoded in them. Publishers (and, in a lesser way, Editors) are all natural candidates for the application of these 3 lenses.

As for the other five, the *Document Structure* lens does not necessarily requires the involvement of the Author, as it is more about assigning the correct structural patterns (as described by the Patterns Ontology [DFP08], see section 4.5) to the element components of the document, and it is something that could a tech-savvy person can infer from the original document content organization, especially if it was originally written in a common format like HTML or DocBook, for which relationships between pattern and markup elements can be deduced by the content model. Promising future developments on automatic textual pattern recognition [DPP12] might also make this an automated task in the future. Considering all these factors, author involvement in the application of this lens is quite limited, as authors (or other individuals) will probably be just asked to review the results and disambiguate some cases.

Table 2 – Summary of Suggested Involvement in the authoring of Lenses

Semantic Lens	Author Involvement	Other Roles Involved
Textual Semantics	**Varies**	Varies
Argumentation	**Highly recommended**	Very Limited involvement
Citation Network	**Recommended**	Other roles might assist
Rhetoric Organization	**Recommended**	Limited involvement
Document Structure	**Limited**	Varies (It might be Automated)
Publication Context	**Not Required**	Mostly Publishers, (Editors)
Contributions and Roles	**Very Limited**	Mostly Publishers, (Editors)
Research Context	**Not Required**	Mostly Publishers, (Editors)

As we get nearer to the actual meaning of the text, to its intentions and to its discourse organization, some kind of involvement by the Authors is obviously recommended, in order to correctly capture and formalize their intended meaning. While in my work of applying lenses to [Mik07] I was obviously forced to "emulate" the intention of the authors, in any semiotically correct application of Semantic Lenses the Authors' opinion about their own words and intention is quite crucial, if we want to avoid misinterpretation when authoring metadata that is meant to convey the specific and precise meaning of the ideas of the Authors themselves.

For the *Rhetoric Organization* Lens, while the Author does not need to be much involved in the part of this lens regarded the Document Components, (as characterized by DOCO [SP11a]) and the markup semantics, which in a sense poses challenges similar to the Structure Lens, his involvement in identifying and formalizing the Rhetoric Discourse of the document (as characterized by DEO [SP11b]) is quite clearly recommended.

The same can be said for the *Citation Network Lens*: although it is possible to theorize what the author intentions for a citation were, thanks to the context of its appearance, it is evidently much better to have said reasons and purpose explicitly defined by the authors themselves, or at least confirmed by other roles with an high domain specific know-how, such as reviewers or editors. As it is, an ideal application of the citation network lens is at least a co-authorial activity.

When we arrive at the *Argumentation Lens*, there is really no excuse for not involving the Authors of the paper. In fact, Authors' involvement and participation in annotating this lens is almost mandatory: arguably, there is no one better than the Authors themselves to explicitly indicate their intended claims and the structure of the argumentations used to sustain them.

As for the final Textual Semantic Lens, while it is expected that some kind of information exchange with the Authors of the paper will be necessary for an ideal application of it, this lens characteristics may actually vary a lot from paper to paper or from domain to domain and there is no definitive answer, like there was none for a favoured ontology. For example, if its application is to be limited in a simple annotation of named entities and their link to domain related ontologies, the Authors' involvement might not be necessary, especially if enough clear and unambiguous data were gathered and encoded in the other lens. On the other hand, if it involves something more, like making a set of precise assertions, with specific properties, the Authors' contribution is probably going to be extremely useful.

5.2 Proposing a general methodology

As a quick reminder we have already defined the **application** of Semantic Lenses as the act of authoring the semantic annotations enriching the document and its component, in opposition with the **focusing** of a Lens, which is the act of using said enhanced data to highlight a specific facet of a scientific document. We also reflected on the responsibilities for the authoring of lenses in regards to different roles within the publication chain (see section 4.4).

I have reason to surmise that in the application activity for semantic Lenses would differ greatly in methodology not just in respect to the role of the person involved and the target document for its application, but also according to a) the tools available to assist in said application and especially b) the time frame of the application – whether the lenses are applied more or less concurrently to the authoring & editing of the target scientific document or instead they are applied subsequently, to the already finished document subsequently, at a much later date.

The first observation is simply the acknowledgement of the well-established fact that our chosen approach when performing a task can be by heavily influenced by the tools we decide to use among the ones at our disposal. "*If you only have a hammer, everything will look like a nail*", or, to put it in a way more related to computer science, the course of action we are more likely to take in order to rename a set of files will be quite different if we are using a file-explorer GUI rather than a shell script, even if we aim for the same end results. The situation for the application of semantic Lenses makes no exception, as no dedicated tools for their application were available, and in order to test them I had to start from scratch and decide the best way to approach the task. The idea of having to manually write and add, one by one, all assertions directly within the RDF/XML or Turtle linearization of a document seemed nightmarish and impractical right from the start, as correcting all kind of errors, modifying the assertions, or simply having an overview (either general or lens by lens) of the added statements would be quite difficult.

However, Semantic Lenses were designed to be able to use all modern Semantic Web technologies. As it is, there is a reasonable amount of tools, languages and libraries for the editing and the manipulation of semantic web documents and ontologies. A possibility could have been to write the instructions for the annotations of the chosen Lenses by using SPARQL queries, e.g. by using the CONSTRUCT keyword, but the end result would still

be quite a low-level approach, with many of the same cons of the aforementioned text-based possibility.
There is also a good number of tools and APIs for Semantic Web available in the Java programming language. Given that among these are both the Apache JENA RDF API for RDF document manipulation and that a Java API for the EARMARK ontological model, whose architecture and advantages I had introduced in section (3.2), is also available, I have chosen to use Java as the privileged way to develop some basic tools to accomplish my intended task of concretely testing the application of some semantic Lens. For instance, let's suppose that we want to assign the Table structural pattern to all `<dd>` elements having a `class` attribute equal to `’table’`. With the methods and the tools that I have been developing, it will be possible to do it as simply as this:

```
applier.buildAnnotation("Table", LensesNames.LA_URI, "expresses",
      LensesNames.DOCO_URI+"Table");
applier.searchWithAttsAndAssert("dd",  LensesNames.EMPTY_URI,  "class",
"table", applier.getLastannotation());
```

This choice allowed me to aim at developing not just a methodology, but some basic tools, which are based both aforementioned APIs, in the form of the SLAM package – Semantic Lenses Application and Manipulation – that might be in future either be re-used or extended. This package will be described in more detail in Section 6.

On to the second observation, I will now explain why I believe that the time frame (and the person tasked to do so) is an important factor in the methodology to be used in lens authoring and lens application on a scientific article.
I want to point out the difference between the creation and application of Semantic Lens metadata concurrently with the authoring of the document as opposed in doing so only afterwards, especially if we consider those Lenses that are more content specific. Indeed, when we examine the timeframe for the application of the first three more context-specific lenses, (*Research Context, Contributions and roles and Publication Context*), their own definition implies that for the information stored within them to be as correct as possible, it should relate to several aspects which can better be collected only after authorship (and possibly, publication) of the document is completed.
However, if we consider the other 5 lenses, the possibility to write them within the same time frame of the document is worthy of contemplation. If the RDF triples for the enrichment of document components with data required by the Semantic Lenses (like those I presented in the examples of section 4.3) could

be assigned during the authoring of the document (or during any closely related activity), the Authors' involvement would be more straightforward, and as a result the information would arguably be far more accurate than a *post-hoc* application. The chances for an ambiguous interpretation of the Authors' intentions would also be slimmer, and such a method could also help the Authors' in examining their motivations for selecting a citation or structuring an argumentation, as they would need to make the reasons behind their choices explicit.

The main problem with this "ideal" approach is, unfortunately, an eminently practical one. First, it supposes that authors already know how to apply semantic lenses, or could become quickly familiar with Semantic Lenses, their definitions and the concepts and meaning encoded by the Ontologies used. And, most important, the fact persists that, at the moment, the application of Semantic Lenses requires a good amount of technical knowledge in computer systems, semantic web technologies and languages, which is an unreasonable expectation from authors not in the field, even when considering the tools I have developed or some of their possible immediate evolutions. Of course, it is possible to envision the future realization of advanced authoring specialized text editors or tools (either as separate software or plugins to existing ones) that could greatly help authors to apply lenses easily, just like it is possible to apply different styles and format to a text in a document editor, as well as reminding them, perhaps with tooltips, of the meaning of the annotation they had just chosen to apply. If these tools were available, or if the objections highlighted above did not apply to the Authors, then it would be possible to add all annotations for all the five content related lenses to a component as it is in the process of being authored (e.g., a paragraph just written is immediately associated with relevant information, for example a "pattern:Block", "doco:Paragraph", "deo:Background"). As it is, this is more of a vision for the future and a final goal to be reached rather an immediate prospect.

While we should not shy away from that ideal final objective, for now, the most practicable approach is to semantically enrich scientific documents **after** they have been completed and finalized, and to pursue the application of lenses with an *ex-post* approach, much like it was done by Shotton *et al* in [SKM09].

Ideally, as discussed in section 5.1, Authors' involvement as sources of assistance for the correct interpretation of their purpose in organizing the discourse, in motivating the choices behind each citation and in identifying what they intended as claims, would be recommended and of great assistance in reaching a good level of accuracy, while the actual technical implementation of semantic lenses would be done by an Editor with enough tech-savvy and domain knowledge to do so. Anyways, we should be mindful that this time consuming collaborative process might not always be completely possible for whatever geographical or practical reasons (such as it was the case for this demonstration), especially for the enrichment of papers published some time ago.

Considering what I had just observed, it is my belief that in this situation the best approach is to analyze the paper on a Lens by Lens basis, with multiple "*visits*" over the target document, each one considering only aspect of the domain, in a sense mirroring the future act of focusing, although from an inverted standpoint. In doing so, the Semantic Lens Editor could concentrate on gathering the most correct information on the specific facet that is being considered, trying both to correctly interpret the Authors intended meanings and not to lose sight of the overall big picture of the article. I think that this kind of approach, especially if from the **bottom-up** in our Semantic Lenses Stack (thus starting from the lest content-related lenses, like the *Document Structure,* then going "up" with the *Rhetoric Organization*, the *Citation Network* and so on…) has the best chance to correctly catch the original intended meaning and to reduce misinterpretation or internal consistency errors – unlike the more "in depth" approach where all lenses are applied in detail to each component within a single passage.

Given these premises, I have chosen to follow this road in my concrete case study on semantic lens application as well. After choosing a target paper [Mik07] in 5.3, selecting some lenses for the tests 5.4 and converting the paper in EARMARK 5.5, I followed an approach based on what I introduced above.

At first I considered each selected lens independently, initially studying a general methodology for its annotation, one not necessarily tied to the target document. For each lens then I would review the document's original source (as markup and text), take down informally its distinguishing features and then start to theorize which statements I would need to add to correctly represent it in conformity to the Semantic Lens model. This would give me an insight on what features should be had by the tools that I would need to develop.

After having gathered this information for each lens, I then proceeded to develop a way to actually annotate them in the document, by the means of SLAM, described in section 6. With that done, I would concretely apply the lenses to [Mik07]. This will be detailed in section 8.

5.3 The target paper – "Ontologies are Us"

For my concrete field test activity I have chosen a well known and widely cited paper on Semantic Web – Peter Mika's "Ontologies are Us" [Mik07], an important study on the topic of *folksonomies* – ontologies emerging from online communities. The author extends the traditional bipartite model of Ontologies in a tripartite model of actors, concepts and instances within a social dimension, studying ontology emergence in del.icio.us and within web pages.

This paper has been cited 84 times within the ACM digital library (of which it is a part), and is also cited 166 times in Scopus, 158 times in Microsoft Academic Search, and 365 times according to Google Scholar – these numbers are a testament on the acknowledged importance of this work, as well as of its quality. It is a well structured paper, adhering to the expected standards for a scholarly article, with a clear and well thought out discourse organization, as well as having enough of a diversity in its contents to make for an interesting test-bed for a wide variety of semantic denotations.

The paper is available for online consultation in HTML format by subscribers of ScienceDirect, and this version was the basis for my enhancement activity.

5.4 The target lenses

Given the magnitude of the task of annotating correctly the test paper with all 8 lenses, an activity that goes far beyond the scope of this work and which would have left me little opportunity to develop SLAM and TAL, I had chosen to focus my field tests in applying lenses on only four of the Semantic Lenses defined in section 4.3.

The Lenses I have chosen were *Document Structure, Rhetoric Organization, Citation Network* and *Argumentation.*

Research Context, Contribution and Roles and *Publication Context* lenses were discarded as they are the three more context-related ones (as already shown in section 4.3) – they also offer information that is on the whole related much more to the document as a whole rather than its context and components, and the metadata payload they could carry would offer little in terms of inter-document interaction, being more focused on intra-document interlinks and applications.

The *Textual Semantics* Lens was discarded for a whole different reason: as we have seen, it is the one that is less rigidly defined, and is extremely domain specific. While its application might certainly have offered interesting possibilities in terms of user interaction, (such as in [SKM09]), the fact remains that it would have been hard to gather some general purpose lessons or methodology from it, as it is the less universal and the more specific of the whole Semantic Lens stack.

Switching back to the four the chosen Lenses, they have several advantages: They are content specific enough to require some annotation and denotation within the document's components, thus allowing for a reasonable variety of application cases and an heterogeneity of challenges to be solved to do annotate them; they address some extremely relevant facets of a scientific publication regardless of its specific domain, such as the rhetorical discourse, the citations or the argumentation model; and they offer relevant opportunities in terms of the focusing activities related to their presence, both at the inter-document and at the intra-document level.

5.5 Porting the target document to EARMARK

We have already seen the major advantages of the EARMARK model for the representation of annotated documents in the appropriate section (3.2), but just as a very fast reminder, EARMARK offers us an excellent way to express semantic assertions about the document and its content, as well as about relationships between its components, allowing a very straightforward integration with Semantic Web Technologies like RDF (any embedded semantic markup could also be converted into RDF triples). It also enables a very straightforward and effective way to address overlapping markup, thanks to the use of Ranges, as well as allowing to express text fragments out of order or reversed.

The EARMARK software package also has a full featured set of Java API based on the Apache Jena RDF API. The EARMARK API offers the user several useful methods for the basic manipulation of EARMARK document models, and a very good starting point to extend with my work.

There is also a very useful EARMARK converter tool, XMLTOEARMARK, which is able to convert in an equivalent EARMARK document model any well-formed XML document, and which I used to port the target document [Mik07] into the EARMARK format.

First of all, I cleansed the paper of most non-essential html markup clutter, especially the one related to the science direct website features, frames, in order to reduce the target paper structural complexity as much as possible while keeping intact all his contents, data and internal reference structure. I then converted it to EARMARK with the use of EARMARK's XMLTOEARMARK tool, which worked perfectly and outputted the representation of the original document within an Earmark ontological model.

5.6 The Application of the Document Structure Lens

It should be clear that the objective here is not to create an univocal 1:1 representation of HTML 5 or XHTML with the Structural Patterns. That is simply NOT possible. (X)HTML content models are much more flexible than Structural Patterns could ever be, due to difference in the original design goals (the HTML schema is a lot less strict than Patterns), and, for many HTML elements, having an a unequivocal, generalized assignment of a single Pattern to all possible instances of that element is impossible.

Take, for example, the simple `<div>` element of HTML. Instinctively, we might be tempted to say that it matches the *Container* Structural Pattern, which is defined as "*Any container of a sequence of other substructures and that does not directly contain text. The pattern is meant to represent higher document structures that give shape and organization to a text document, but do not directly include the document content*". However, two problems immediately arise – the content model for `<div>` implies that it might contain almost any other element in the HTML body, including some that could have been branded with unacceptable patterns, like *Milestones* or *Popups*, and, perhaps even more important, a div might directly contain text. So it's safe to say that a `<div>` can't be always assigned to the *Container* pattern. Can it always be a *Block* then? The answer is again no, as a `<div>` might be nested within another `<div>`, but a *Block* cannot contain another *Block* elements, even recursively. Is it always an Inline? But an *Inline* cannot have any *Container* patterns inside it, while a `<div>` can hold elements which can easily be other containers, and often it is used only to give shape to the document structures….

Having shown that the research of a 1:1 univocal, document-detached representation of the whole HTML with Structural Patterns is an effort in futility, let us detail a more feasible approach. Given the observations just made, it becomes necessary, from a general methodology standpoint, to consider the assignment of Patterns only within the context of the document, and not with an *a priori* approach. Of course, for some kind of markup elements, the identification of their structural pattern is easier than others, and might even be universally acceptable – it is hard to imagine a `<br/>` element as anything different than a *Milestone* pattern.

In [DPP12], an interesting method for automatic pattern recognition (from DocBook, whose schema is less lax and ambiguous than HTML) has been developed concurrently to my activity: The idea is to search for a subset of the schema on which a certain element could be manually given a preferred, predetermined pattern assignment. A three step algorithm for pattern recognition is run on the document. The resulting assignment of a certain pattern depends on the content model of the element and to the pattern of their containers and contained element. Finally, a disambiguation takes place with three separate reduction activities, like a pattern shift reduction if an element is assigned to compatible patterns in different places in the document. E.g. if an element is assigned both to the *Block* and the *Field* pattern, the *Block* pattern is selected, since *Block* does respect all the requirements for *Field* as well. This helps to mitigate one of the problems in pattern recognition: Different authors use the same element in different ways, or the same authors might use the same element in an ambiguous way within the same document. Aside from that, we have already commented on the fact that this path is not as feasible with HTML as we wish it to be, since even in the same documents some general purpose markup elements (such as `<div>` and `<span>`) can be used in widely different ways, precisely because they were designed to be used in such a fashion.

As it is, the approach I had decided to adopt still took into account the results from [DPP12], but adapted it to the circumstances, and, as for the application of all other lenses it was a non-automated authorial activity.

Firstly, I assigned a single general "standard" pattern to as many HTML markup elements as it was reasonably possible, such as the already mentioned *Milestone* pattern to the `<hr>` and `<br>` elements, and listed the more likely candidates pattern to choose from for the other markup elements.

Then, it was the time to go within the context of the target document structure, and to identify how the "ambiguous" elements were used inside it, and what structural patterns the author had used to build up its content. Even with a well-structured, regularly organized documents such as this, I was bound to encounter an heterogeneity of uses for at least some of HTML markup elements, and the results confirmed my suspicion – fortunately, such heterogeneity was more limited than expected, as it was mostly concentrated with `<p>` and `<div>` elements.

What was important, at this point, was also to understand what I needed to be able to search within the EARMARK representation of the document, in order to find and distinguish the EARMARK MarkupItems that were to be annotated by the Structure Lens assertions, by myself or other users in the future. The aim here was to minimize the need to address the items single URI by single URI, and to do so only when absolutely inevitable, since doing so for all target components would greatly increase the time and the complexity of the applications.

To do so, I would surely need to be able to find EARMARK MarkupItems by general Identifier, which was already possible with the EARMARK API. The analysis of the target document under the Document Structure also suggested me that the ability to select a set of item by the content of one of their attribute could greatly help in selecting the right type of items for the assignment of a Structural Pattern. Consider, for example, the "class" attribute in HTML – it is often use to designate a subcategory of usage for certain general purpose elements, as was the case within [Mik07].

These necessities will drive the development of SLAM, which is detailed in section 6, while the end result of the application of the Lens will be discussed in section 8.2.1.

In closing, I also wish to state that, by my analysis, I was able to observe on a strong limitation for the definition of the *Table* pattern, which, in my opinion, strongly reduces its applicability. To quickly refresh our memory, the *Table* pattern content is defined as "`Container ⊓` **`Contains Homogeneous Elements: true`** `⊓ Contains Heterogeneous Elements : false`" – it means that all content within a Table should be a repetition (regardless of its size) of homogeneous elements. This fits perfectly, for example, simple structures like Ordered Lists `<ol>` or Unordered Lists `<ul>` in HTML, which usually contain as first level children only a set of List Items `<li>` .

Paradoxically, problems in assigning this pattern might arise with actual table elements (`<table>`), or with little more structured elements like Definition Lists `<dl>`: For example, the definition list content is usually made by the regular alternation of `<dt>` and `<dd>` elements. There is homogeneity of substructures, but these are made by more than one element – however, this kind of regularity is not acknowledged by the *Table* Pattern. A similar observation could be made for (X)HTML tables and its allowed content model. Even if there is regularity in the repetition of homogenous sub-structures within the content, the way the structural pattern *Table* is currently defined would not allow this kind of content, thus forcing us to opt for the assignment of the more general *Container* pattern. It is my belief that the pattern would carry more significance if this issue could be resolved at the definition level, by relaxing the requirements in order to allow within it repetitions of a specific combination of substructures, as long as it offers some regularity.

5.7 The Application of the Rhetoric Organization Lens

The theoretical bottom-up methodology for the application of the Rhetoric Organization Lens I propose requires, first of all, that some kind of Structural Pattern assignment (even if not final) had already taken place for the document components. This requirement is also a simple logical consequence of the fact that almost all DOCO entity classes which are used by this lens have strict requirements, based on Patterns themselves, on what kind of components could be associated with them.

That said, given the dual nature of this Semantic Lens, exemplified by the dual ontology used (DOCO with DEO), I suggest as a general methodology to perform a double iteration over the target document when gathering information to apply this Semantic Lens: first for DOCO, the more component-related part, and then for DEO, the part which is more relevant to the rhetorical discourse organization.

This lens is even more content-related than the Structure Lens, but there's still space for a couple of general observations, not necessarily tied to the target document for the application activity. First of all, it seems that some elements identified as a *Milestone* pattern within the previous step, such as `<br>`, will probably not be able to be associated to any DOCO class, due to the conflict inherent in the *Milestone* role itself – if they don't acquire any meaning through their attributes, those elements have no special meaning "*per se*", so the most relevant information about them is their position within the document – while DOCO deals more directly with the structural function of elements. It should also be noted that, given the problems already highlighted in the application of the *Table* Pattern, it is unlikely that any actual table of a document could be denoted as a *doco:Table*, since that class requires the assignment of said *Table* Pattern.

Another important observation that could be made even before moving onto the document was that the very strict constraints required by many of DOCO classes' definitions would match only for documents structured in a way that conforms to the ideal model envisioned by DOCO – which is certainly a subset of all valid document models allowed by HTML. For example, a *doco:TextBox* could only be a Container, but we know that a Container cannot directly contain text, so the only "acceptable" model imagined by DOCO to have a text box is to have a *Container* within which is located another element allowing a *Textual* content pattern. This same issue is even more relevant if we consider the strict requirements over very important denotations like *doco:Section*, *doco:FrontMatter* and *doco:BackMatter*. Unfortunately, some of these limitations will emerge in the analysis of the target document, as it has a shallow structural depth, where most the content of the main body is contained within `<p>` elements which are direct children of a single `<div>`, thus not allowing the use of the *doco:Section* attribution due to the lack of appropriate Container pattern components.

It also appears that while DOCO has a good amount of classes allowing us to identify many roles within a scientific paper, from the abstract to the table of contents, it lacks a couple of useful characterizations for common "building blocks" usually included within the front matter, such as keywords or dates/publication information.

The DEO ontology level is instead, at least in my opinion, much more streamlined and of easier application over a document. I had few observations to make before effectively starting the application activity, due to the required connection between this lens metadata and the context of its application. The only thing I could take note of was that documentation and the classes description for DEO was a little too concise, which resulted in entities definitions which in some cases were a little vague and in some others were not too much eloquent in their description. In a sense, this allows for more freedom, but on the other hand, it leaves more space for misinterpretation. Once again, the general methodology I adopted is to proceed over the whole target document, and to annotate the rhetorical organization of the discourse, mostly over the tables, the figures, and the paragraphs composing the main body of the article.

Moving on to the activity related to the target document, I decided to limit the scope of my application of the Rhetoric Lens to the paragraph level, out of a desire to avoid unnecessary cluttering and out of simplicity. The only relevant exceptions were captions and labels for tables and figures, as well citations and internal references, which were all annotated as well. As for the search capabilities that I supposed would be required, they remain pretty much the same exemplified in the previous section.

For all the details on the actual implementation of the application, as well as the extended observations on what DOCO classes could or could not be applied to the target document, and some remarks on the practical application of DEO and on the overall rhetoric discourse of [Mik07], see section 8.2.2

5.8 The Application of the Citation Network Lens

The Citation Network Lens, which is based on CiTO, relies on ontology object properties, and, as such, their intended use is as predicates within RDF statements from one object to the other.

In the first inception of the Semantic Lens model, the characterizations of citations with CiTO were supposed to be gathered at the overall document level. However, I had decided against it, as doing so would be tantamount to the loss of information within the context of the citation act, which is usually within a specific part of the main text (and as such, in a specific point of the rhetoric discourse).

Instead, I opted for a more information rich solution, with the citation network lens assertions associated to each inline citation reference occurrence, so that the metadata enriching each citation could be referred within the context that originated that very citation. Doing so opens up interesting possibilities for the study of the interactions between this lens and the Rhetoric Organization and the Argumentation ones within the same document. For example, it is reasonable to expect that a citation contained within a paragraph denoted as the expression of a *deo:Background* could also be one with the property *cito:ObtainsBackgroundFrom*.

If the need arises, it is easy to "let go and lose" this extra information, and merge or gather all the metadata at an higher level, such as the document one – trying to do the opposite would be quite impractical.

This path also gives the possibility to denote differently the same citation according to the context, extending the versatility of the Citation Network application. It is plausible to theorize that the same source might be cited for different reasons in different parts of the citing document – this approach enables us to catch this additional subtlety of the facet, by simply using the desired citation property within the appropriate context.

It is of course possible to have more than one property characterizing a citation for each of them: not only that, I expect that cases where only one of the CiTO properties could apply will be quite rare.

Considering the abovementioned methodology, and given that it emphasizes the context of the citation as the subject of the statement, I have subsequently decided to use CiTO properties in a direct way (*cito:cites* and sub-properties) rather than in their inverted way (*cito:isCitedBy*)

Once again, I must observe that the descriptions of each CiTO property within the documentation of the Ontology are quite too short and a little too concise in their wording, and I highly recommend their expansion, as of now they offer little to distinguish each other and give little relevant information – thus leaving a lot of leeway to the user.

More details for the concrete implementation on this Semantic Lens, as well as the result of its application and the final discussion are to be found in section 7.3.3

5.9 The Application of the Argumentation Lens

The Argumentation Lens is the more content related one of the four I will be applying on [Mik07]. As it is, its application fundamentally depends on the textual content of the main body of the target paper.

On a general methodology level, the idea is to read accurately through all the document content, first identifying all important claims, then completing the structure of each argumentation by highlighting data, warrants, qualifiers and rebuttals. Not all the text has to be part of a relevant argumentation. I mostly focused on trying to interpret correctly the Author's intention, and aimed to model them accurately. Trying to do so after the Rhetoric Organization Lens has already been applied gave me a significant guidance in maintaining an overall coherence and plausibility for my inference work.

Of course, as already discussed, the Argumentation structure is not necessarily linear. On the contrary, argumentations and argumentation components often overlaps, with some components being shared between more than one argumentation, either in the same or even in different roles. Some components, such the evidence for a claim, can be found outside the main text, e.g. within a table. Finally, according to Toulmin's Model, some whole argumentations can end up being simple components (warrants, backing, and evidence) for a larger one.

In order to implement this, I would certainly need to identify ranges corresponding the text chunks related to each component and create them if they are not already existing. The EARMARK model allows me to operate with overlapping ranges, and it is a great asset in the task of enhancing this aspect of an article. The idea is to create new entities within the EARMARK model, typing them as argumentations, in order to explicitly denote their structure by using the "*hasClaim, hasWarrant, hasEvidence*, etc." AMO properties, and to use appropriate, predictable identifiers.

6 SLAM – Semantic Lenses Application & Manipulation

6.1 Introduction – Tasks, Aims, Necessities and Priorities

In the previous sections I have introduced the Semantic Lenses, with their set of related technologies and onthologies, and I provided several examples of their use. After that I especially focused on a general methodology for their **application**, and stated my aim to field test this application activity on in a real test case (with [Mik07] as the target document), at least for the *Document Structure, Rhetoric Organization, Citation Network and Argumentation* Lenses.

As a very short reminder: I have previously defined the **application** activity as one of the two fundamental task within the Semantic Lenses model – the other being its complementary activity, the **focusing** on an applied lens. The application of Semantic Lenses over a document is thus the act of enriching it by annotating methodically the appropriate metadata (as specified by the Semantic Lens model) which would allow for the explicit semantic denotation within one of the possible aspects of a document. The **focusing** of lenses, in turn, consists in having a chosen set of meaningful information emerge from a lens application, in order to highlight additional data or enabling new interactions over a specific facet.

Of course, in order to apply lenses over a target document, the knowledge of what to write and a methodology to do so is not the only thing required to perform this task successfully. In order for a lens to be applied, the additional data, in the form of RDF statements, has to be actually added to a document. In order to do so, some kinds of tools are necessary. As I have already explained at length in section 5, the methodology is also related to the available tools – and there were no specialized packages available at the beginning of my work. As also detailed in the previous part of this work, I quickly discarded the possibility of manually writing and adding, one by one, all assertions within the source linearization of an EARMARK document, as it would have been exceptionally impractical, error-prone as well as having very poor significance.

My instrument of choice to apply lenses over [Mik07] is thus a newly developed package in the Java programming language, which I christened SLAM – Semantic Lenses Application and Manipulation. In the specific, it starts as an the extension of two already existing Java API, the JENA RDF framework and the EARMARK API, and is a very simple package whose purpose is to allow me to model semantic lenses annotations and applications, to better manipulate them and to have additional finding methods within an EARMARK document, as well as giving me some very useful syntactic shortcuts for their definition. For example, let's suppose that we want to assign the Block pattern, in the Document Structure Lens, to a subset of `<p>` paragraph elements, those having a "`class`" attribute equal to "`svArticle`" are to be modeled as Blocks. SLAM allows me to define this operation simply with just two lines of code, one for creating an Annotation, and the other instructing the Applier to assert it using a set of MarkupItems as subjects:

```
applier.buildAnnotation("Block", LensesNames.LA_URI,
      "expresses", LensesNames.PATTERN_URI+"Block");
applier.searchWithAttsAndAssert( "p", LensesNames.EMPTY_URI, "class",
      "svArticle  section",  applier.getStorage().getAnnotation("Block")
      );
```

The main purpose of SLAM is on the one hand to define a set of classes to model Semantic Lenses as a whole, the RDF annotations that are part of them, and the application process itself, and on the other to add new functionalities to those already made available for handling and searching markup. This is done with the aim of enabling those search and manipulation capabilities over EARMARK which I have found to be necessary in my methodological analysis (see sections 5.6 to 5.9), as well as some additional utilities, such as a way to record lens application statistics or a class to manipulate sets of EARMARK MarkupItems.

Of course, all of this represents just the general purpose part of SLAM, the basic blocks meant to be used to construct the enriched document – but in order to accomplish my objective, I put SLAM immediately to work, and used it to create a working set of instructions for the application of the four chosen Semantic Lenses over "Ontologies are Us", as a way to test and demonstrate the functionality of the package as well as the applicability of my methodology.

The results for the application of Semantic Lenses over [Mik07] will be presented in section 8, while the rest of this section will focus on detailing the SLAM package and its inner workings. But before going on, allow me to give a very short overview of the two APIs on which SLAM relies in order to function.

6.1.1 Jena API

Apache Jena RDF[15] is an open source Java API for RDF, and it defines itself as "*a Java framework for building Semantic Web applications. Jena provides a collection of tools and Java libraries to help you to develop semantic web and linked-data apps, tools and servers.*" [13]

To put it simply, Jena is a framework providing a large set of Java libraries to assist software developers in building Java code capable of handling RDF, OWL, SPARQL and many other Semantic Web technologies in accordance to the official W3C recommendations.

The Jena Framework offers many functionalities, including:

- **An API for reading, processing and writing RDF data in XML, N-triples and Turtle formats**; which is the core part of Jena and the one I will be using the most in SLAM
- An ontology API for handling OWL and RDFS ontologies;
- A rule-based inference engine for reasoning with RDF and OWL data sources;
- Stores to allow large numbers of RDF triples to be efficiently stored on disk;
- A query engine compliant with the latest SPARQL specification
- Servers to allow RDF data to be published to other applications using a variety of protocols, including SPARQL

The development of Jena started in the HP labs in 2000, and in 2010 the project was adopted by the Apache Software Foundation, and became a top-level project in April 2012.

[15] Apache Jena: http://jena.apache.org/index.html

The Jena framework is quite a large scale project and is made of several different packages, which are listed here:

Package Name	*Description*
chh.jena.rdf.model	The Jena core. Creating and manipulating RDF graphs.
chh.jena.datatypes	Provides the core interfaces through which datatypes are described to Jena.
chh.jena.ontology	Abstractions and convenience classes for accessing and manipulating ontologies represented in RDF.
chh.jena.rdf.arp	A parser for RDF/XML.
chh.jena.rdf.listeners	Listening for changes to the statements in a model
chh.jena.reasoner	The reasoner subsystem is supports a range of inference engines which derive additional information from an RDF model
chh.jena.shared	Common utility classes
chh.jena.vocabulary	A package containing constant classes with predefined constant objects for classes and properties defined in well known vocabularies.
chh.jena.xmloutput	Writing RDF/XML and I/O

Within this project, the part of the Jena package that I will use most is its core, in order to manipulate, create, add, remove and fetch RDF triples within RDF document modeled as graphs.

It is possible to access to RDF triples and graphs, and to their various components and representations, through the use Jena's RDF core API. Among the most notable abstractions used it is worth to mention the Resource, used for representing an RDF resource (whether named with a URI or anonymous); the Literal, which is used data values (numbers, strings, dates, etc); the Statement which models an RDF triple and the Model; which represents the whole RDF directed graph. Additional theoretical reference to the meaning of these entities, according to the RDF specification, was presented in section 3.1.

The Apache Jena RDF API offers basic functionalities for the addition and the removal of triples to and from models, and for the search of triples matching certain patterns. Both input and output support most of the commonly-used RDF syntaxes.

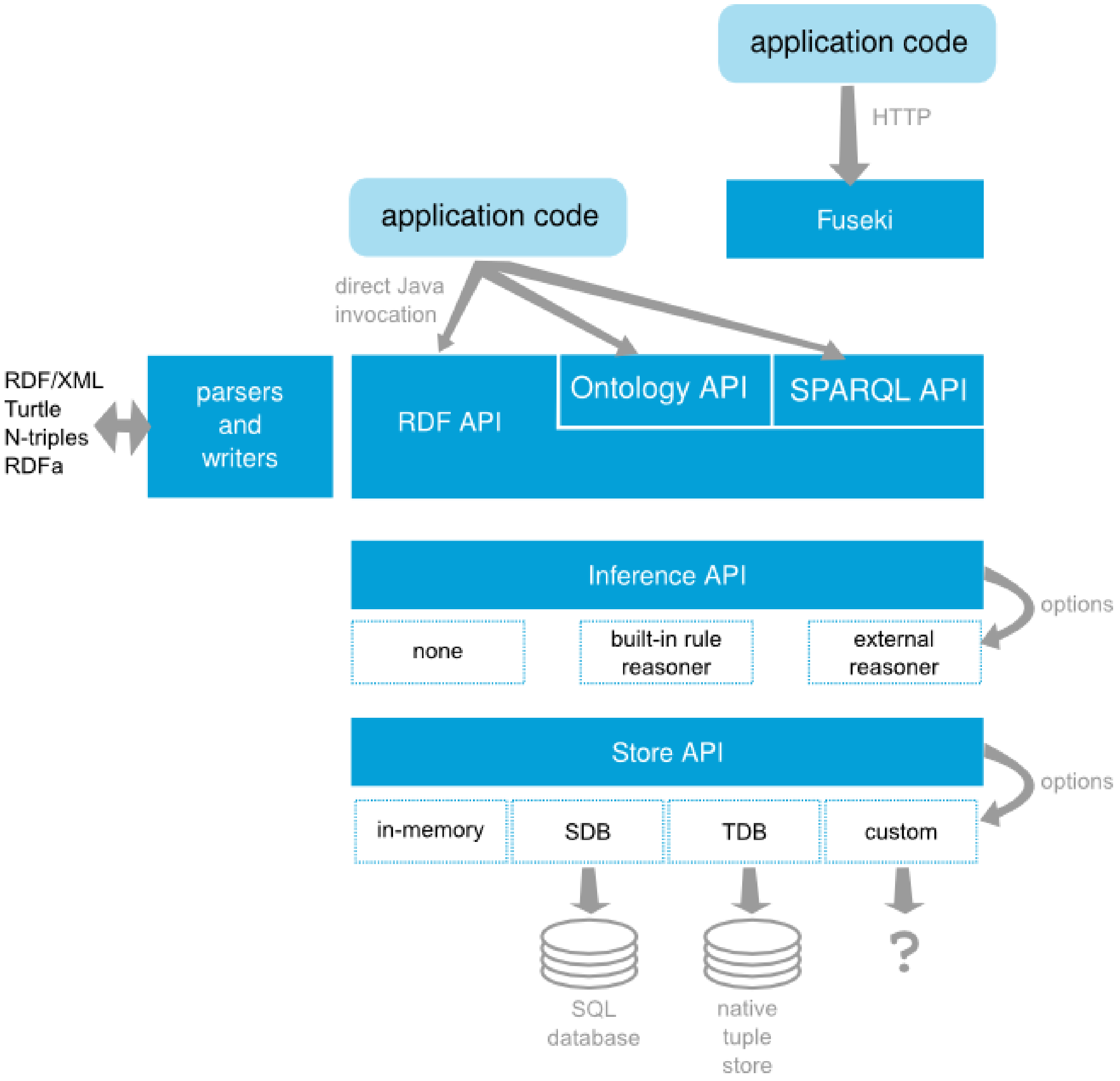


Fig. 19 - Jena's Architecture

There are many other pieces to Jena, whose overall architecture is summarily represented in the picture above, but they are not much relevant within the scope of describing the development and the usage of SLAM.

6.1.2 EARMARK API

I had already extensively presented the EARMARK document format and its overall concepts and architecture in section 3.2. As very quick reminder, allow me state once again that EARMARK (Extreme Annotational RDF MARKup) [PV09] is an ontological model whose purpose is to combine in a single document both the embedded markup used to define the structure of the document (like XML and its derivatives), together with Semantic Web annotations and statements over resources (like RDF), in order to merge all the advantages of both technologies within a single model. [DPV11a]

With the EARMARK ontological approach for meta-markup it is possible to explicitly make structural assertions of markup, describing the structure of a document in a way suitable for the semantic Web, and it's easy to handle overlapping markup through its stand-off notation.

The EARMARK Java API [16], is a framework for the creation, validation and manipulation of EARMARK documents, released under the Apache 2.0 license, and implements completely the current EARMARK model in Java, allowing to use of the EARMARK meta-syntax for non-embedded markup to write stand-off annotations of textual content with fully Semantic Web W3C-compliant technologies. The EARMARK Java framework is a precise model for the format, implementing all the ghost and shell classes we have already introduced, following exactly the data structure defined in the EARMARK ontology. The EARMARK API relies on Jena as well, which uses quite extensively.

The classes *EARMARKDocument*, *Range* and *MarkupItem* (which is extended by *Comments*, *Attributes* and *Elements*) were developed by implementing a specific interface, named *EARMARKNode*, directly derived from the JAVA DOM implementation, in order to maintain the EARMARK data structure as close as possible to a well-known and used model for XML documents.

This extensively documented API allows to create, read, store, manipulate and modify EARMARK documents and all their components directly with Java, and as such was the ideal foundation over which SLAM could be built.

SLAM relies heavily on the use of the EARMARK API, both in terms of data structures and in the methods used to access the EARMARK representation of both the target and the annotated document, and considerably extends some of the basic functionalities offered by EARMARK API in the field of selection and fetching of EARMARK Nodes. Some SLAM methods are simply wrappers and shortcut to call for EARMARK methods together with SLAM data structures and constructs.

[16] S. Peroni, EARMARK API: http://earmark.sourceforge.net/

6.2 SLAM features and architecture

In the previous part of this section I introduced the goals and the necessities driving the development of SLAM, and I stated the intended purpose for this small package. Before exploring SLAM in further detail, I will now give a more detailed overview of its intended capabilities.

I had already said that the goals of SLAM could be quickly summarized within these few points:

- Provide a way to represent Semantic Lens and their components
- Provide a way to apply a Semantic Lens on an EARMARK Documents (and store the results)
- Provide additional search and manipulation capabilities over EARMARK Documents
- Use all these features to assist in the task of writing and applying semantic lenses
- To do so in a re-usable, general way compatible with the methodology I discussed
- To reuse Jena and EARMARK API as much as possible

Now the first issue at hand is how to represent Semantic Lenses, and how to represent the act of their application over a document. They way I chose to handle this, was to focus on the task of applying lenses over a document, and to develop a very lightweight data-architecture around it.

What are the two requirements of this application task we have been discussing so much within this dissertation? Well, first of all, a target document is mandatory. Then the lens to be applied is needed as well, and this means choosing, writing and readying the set of additional information that are to be added on the document. The application of a lens is not just the act of adding an already available markup to a document, but includes the authoring of these semantic annotations as well.

Thus, it might be possible to argue that a way to represent any lens is with a collection of common themed semantic markup (such as RDF statements), ready to be inserted in a document. Obviously, though, these statements, if considered as a whole, are extremely tied to the document and their components.

For example, let us consider the simple assignment of a structure pattern, like we have seen in section 4.3.4.

```
1.   <h-sec-2-uri> a earmark:Element ; # Title of Sec 2
2.        la:expresses pattern:Block ;
```

The second line states that this earmark Element is the expression of a Block pattern within the Document Structure Lens. If we are to consider this a single statement, it is the usual RDF triple made of subject (the h-sec-2), predicate (`la:expresses`) and object (the `pattern:Block`). However, the drawback to this is that in order to write this a pure RDF Statement as part of a lens we would need to precisely specify all the information required by an RDF triple, including the specific URI of the subject element. Now, while considering a lens a collection of RDF triples is certainly correct from a theoretical viewpoint, it is not necessarily the better approach to model one if the purpose is the use of Java to assist in the whole process of applying it, from the creation of the annotations to be added in the target document to their actual merging. Indeed, if we were to do so, we would have very little advantages when comparing this method to the manual annotation of lenses statements directly within a document's linearization. As explained in the methodology discussion, I am looking for a smarter approach, something that could, for example, allow me to instruct that *"all elements with a certain general identifier and a certain attribute are an expression of a selected pattern"*.

Thus, the approach I suggest is quite different. First of all, the basic building block for a lens is not the RDF statement, but just two of its classic components, the predicate (which is a Jena Property) and the object (which could be either a generic Jena RDF Node or a more specific Earmark MarkupItem). This data abstraction will be tied to a specific lens type and to a short name used for fetching purpose, and modeled as the *LensAnnotation* Class. This choice allows for a lot of flexibility. For example, if we consider the previous example, the Annotation would consist just in the "`la:expresses pattern:Block`" part of the statement, without being tied to any specific object. It would then be possible, with the appropriate methods, to perform the application of this pattern on a large set of elements, appropriately selected within the document. Or, just to make another example, it could be possible to re-use the same Annotation, to denote several different elements sharing the same property, just by fetching a different subject, such as it might happen for the rhetoric characterization of a paragraph, or for an argument component part of more than one argumentation.

All the Annotations for a Lens are collected in an HashMap in order to be re-usable, and they are stored using their aforementioned short informal name as a key, which is also used to fetch them back. This HashMap is the heart of the data structure known as a Semantic Lens Annotation Collection (*LensAnnotationCollection* Class), which represents one of the components required for the application of a lens.

Such an Annotation could be written as, we have already seen, simply through this instruction, which orders the Applier, the main actor of the SLAM package, to instance a new Annotation object, and to store it within its internal Collection:

```
applier.buildAnnotation("Block", LensesNames.LA_URI,
            "expresses", LensesNames.PATTERN_URI+"Block");
```

To model the application process, each theoretical Semantic Lens over a target document is to be associated with a Semantic Lens Application (*SemLensApplication* Interface), which is the class interface for grouping all the Java instructions that are to be executed over a document. The main method of this class is "*annotate*", which contains all the commands to be performed in order to create the annotations part of the lens and the instructions on how to link them within the document.

The object receiving, performing and executing these instructions is a Semantic Lens Applier (*SemLensApplier* Class), which is the heart of the SLAM core package. So, if I wanted to assign that block pattern to all <h2> elements, I simply have to write:

```
applier.massAnnotate("h2", LensesNames.EMPTY_URI,
      applier.getStorage().getAnnotation("Block"));
```

An Applier for a Lens associates an Annotation Collection with an EARMARK Document, and offers to the Application class a wide range of methods to perform the subtasks required within the application of a lens. These include the methods for the instancing of a new *LensAnnotation*, a lot of shortcuts and combination methods to find elements (or sets of them) and assert a lens over them with a single call (thus simplifying the coding task and improving its overall readability), methods to fetch annotations from the Collection (or to recover the last used one) and methods to create new elements, properties or ranges. Several preferences on the debugging output and on Statistical recording can be set within the applier as well, through an instance of the *SemLensApplierPreferences* Class.

The Applier is also an extension of the EARMARK Finder (*EarmarkFinder* Class and its related Preferences), which is the class, tied to an EARMARK Document, that contains most of the new research and manipulation functionalities of SLAM. Originally these were included within the Applier, but were then unpacked in order to reach a better separation of purpose and to improve future re-usability of these methods. The Finder offers a wide range of search methods over an EARMARK Document. Some of them are simply syntactic sugar for already existing EARMARK API methods, slightly reworked in order to better accommodate the needs of SLAM, while others offer completely new features.

The new search options enabled by the SLAM *EarmarkFinder* are:

- Find a Set of Earmark Markup Items having a desired General Id and a specific Attribute (e.g.: with the `findItemsWithAtts` method)
- Find a Set of Earmark Markup Items having a desired General Id, and a specific Attribute whose content is equal to some specified values (e.g.: again with the same polymorphic `findItemsWithAtts` method, but called with an additional parameter)
- Find a Set of Earmark Markup Items having a desired General Id, and a specific Attribute whose content matches some kind of content pattern with wildcard support (e.g.: by using the `findItemsWithWildAtts` method)
- Find a Set of Earmark Markup Items sharing a range of similar Ids, from a start to a end. (e.g.: through the use of the `findItemsWithARangeOfIds` method)
- Reverse find of two of the above options: the possibility to select a set of all items with a General Id EXCEPT the ones having a certain Attribute or specific contents for said attribute (e.g.: with the `findItemsExcept` method).

Some additional manipulation options can be made through the Markup Set Reducer (*MarkupItemSetReducer* Class) of the SLAM Utilities, which allows to systematically refine the results of a search, by removing from them all items within another Set of Markup Items.

The rest of the utility sub-package consists in a very simple class to assist in I/O operations (the *EARMARKDocumentLoaderWriter* Class) and in two classes which act as storage for constant definitions (the *LensesNames* and the *LensesTypes*) class.

All these options give the user a lot of flexibility in writing a Lens Application. For example, let's suppose that all Table Boxes in a document are `<div>` elements with an `id` attribute whose contents correspond to a numeric progression, like "table_tbl1", "table_tbl2", and so on. If I wanted to assign the DOCO class "Table Box" to all of them, I could use the abovementioned features to write these two lines of code.

```
applier.buildAnnotation("Table Box", LensesNames.LA_URI, "expresses",
          LensesNames.DOCO_URI+"TableBox");
applier.searchWithWildAttsAndAssert("div", LensesNames.EMPTY_URI, "id",
          "table_tbl*", applier.getLastannotation());
```

First, a new Annotation is created by the Applier and inserted into its internal storage (a Lens Annotation Collection), then the applier invokes a shortcut method which in turn calls the method *findItemsWithWildAtts*, that we have already introduced as a method which will return a set of Earmark Markup Items with a shared General ID, and a specific Attribute matching a specific content pattern. The outputted set is then used as the subject for the assertion of the Annotation just created, recovered through the *applier.getLestannotation()* call.

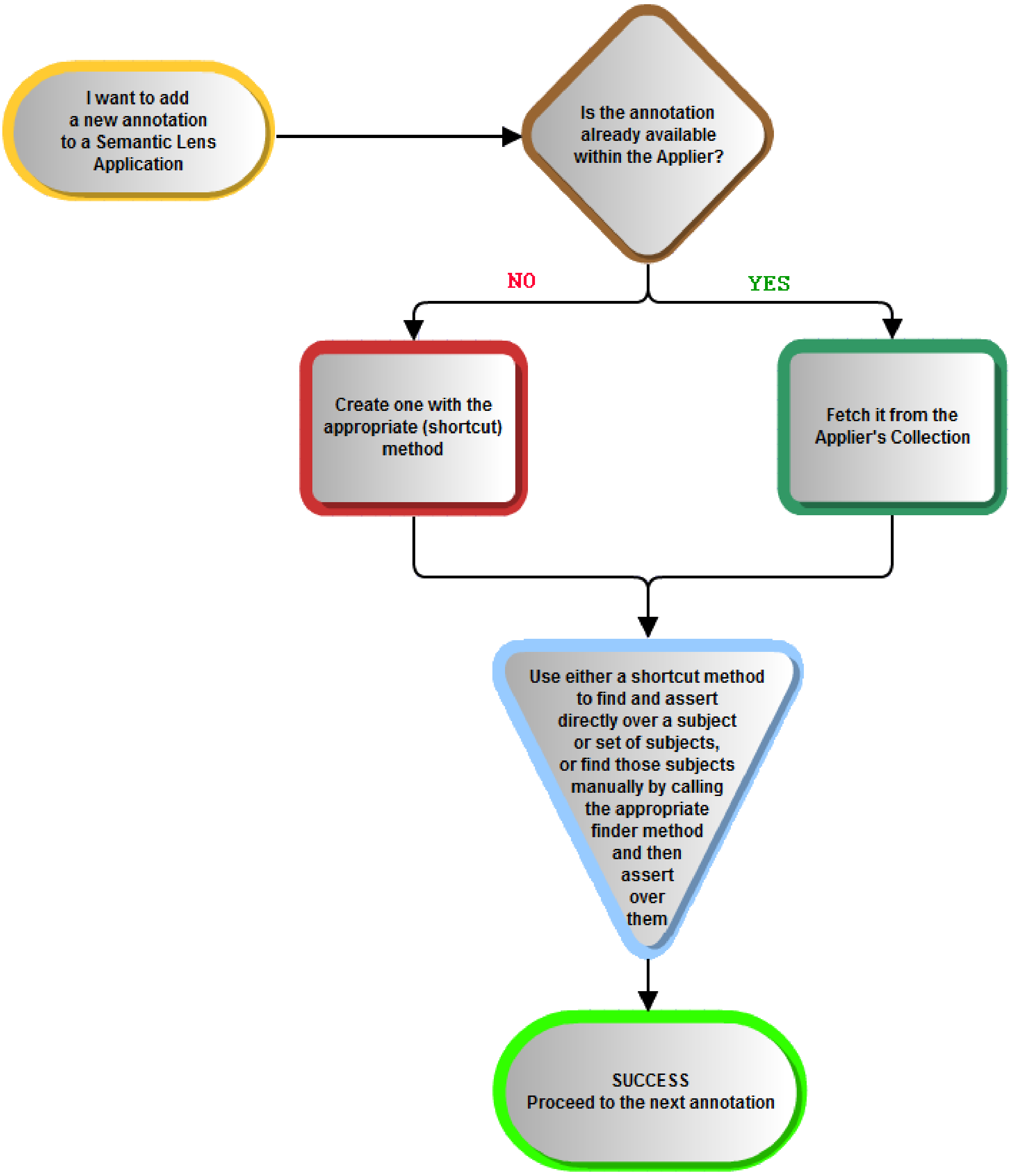

I want to add a new annotation to a Semantic Lens Application
Is the annotation already available within the Applier?
NO
YES
Create one with the appropriate (shortcut) method
Fetch it from the Applier's Collection
Use either a shortcut method to find and assert directly over a subject or set of subjects, or find those subjects manually by calling the appropriate finder method and then assert over them
SUCCESS
Proceed to the next annotation

Fig. 20 - Informal Representation of SLAM's workflow

The overall design architecture of SLAM has allowed me to streamline the application task as much as possible, as straightforward ease of use within the limits of a Java framework was a desirable outcome for the development of SLAM. In fact, the main workflow for creating a Lens Application over a document could be summarized as a loop of adding or re-using new Lenses Annotation to an Applier, and choosing the subjects of these assertions, as exemplified by the flowchart in the previous page.

6.3 Details on the code and on SLAM Classes

In the previous sub-sections I gave a fast overview of SLAM from a general perspective and shortly summarized which features are made available by this package. I also explicated the basic workflow of the application activity within the SLAM framework. In the following part, I will discuss SLAM classes, sub package per sub package, a little more in detail, class by class.

6.3.1 Core Package

The SLAM core Package is made up by 10 classes, each with a specific role:

- **LensAnnotation:** This is a basic class to group together the objects used to create Assertions on an EARMARK document. This class represents a Semantic Lens Annotation (which is, in practice, a generalized RDF statement WITHOUT the subject), that is to be used within a Semantic Lens Application. Every annotation represents a couple made of a predicate, or "property", together with an "object", which could be either a generic Jena RDFNode or an Earmark MarkupItem (not both!). This couple is then to be used in building an assertion on a document or on any of its components. These Annotations are identified by a String name key and stored in a repository (within the LensAnnotationCollection class) inside the Applier of each Application. For Example "`la:expresses doco:TextChunk`" is a Lens Application, and could just be named "Doco TextChunk". To each LensAnnotation is also assigned an appropriate LensType. This class offers several constructor methods, although it is usually built from within a Semantic Lens Applier.
- **LensAnnotationCollection:** This class is a just a data structure to store a collection that groups all the Lens Annotations related to a single Applier (within a single Lens Application). This collection is implemented by a Java HashMap, where the names of the *LensAnnotation*(s) are the keys to the map, and the annotation itself is the value stored. To each Collection is associated a LensType, and a warning will be had if an annotation of the wrong type is put in the Map.

- **SemLensApplication:** This is an Interface defined for the application of any kind of Semantic Lens, by using an Applier (*SemLensApplier*) to annotate the document with several Lens Annotations, built within the Applier and stored within its Lens Annotation Collection. The concept of applying a Semantic Lens is represented by this Interface, which is then followed in the hierarchy through the abstract class "*BasicLensApp*". Said abstract class is then to be extended by any specific Lens Application that the author may want to create on a specific document. Its main method is the "annotate" method, the public method that contains all the instructions to annotate a Lens, using an Applier contained by the concrete class implementing this interface.
- **BasicLensApp:** This abstract class implements the interface *SemLensApplication*, but in order to be concretely used by any project it has to be extended by a concrete class, one for any single semantic lens we want to apply to a given document. It is simply a skeleton class for the implementation of the main Interface *SemLensApplication*.
- **EarmarkFinder:** This class provides an extension of the EARMARK APIs in term of methods to find and select nodes in an EARMARKDocument, especially geared towards finding Markup Items and sets of them. The results of these searches are then reused to annotate Semantic Lenses on specific elements or sub sets of items. This class is extended by the *SemLensApplier* class. It was originally part of the Applier, but it was then decoupled to offer more flexibility for other applications. The class itself is a large collection of public methods, ready to be used to select elements on a EARMARK document or Jena Model. In addition to providing semantic sugar for some of the already available methods of the EARMARK APIs, this class adds several features, which were listed in the previous pages. Prominently, amongst these are the ability to select sets of elements by the presence of attributes, by the contents of said attributes (with wildcard support) and the possibility to select all the elements NOT having a certain match of attributes and contents.
- **SemLensApplier:** As we already said, this is the main class of the SLAM package, and it is an extension of the *EarmarkFinder* class. The Semantic Lens Applier as a class serves a dual purpose: on the one hand, it acts as an operable data structure abstraction to couple an EARMARK Document with a LensAnnotationCollection, and on the other hand, it contains the implementation (directly or as a result of extending the Finder) of all the methods used by the concrete Semantic Lens Application class. As such, the Applier is the actor performing the directions dictated by the Application *annotate* method. Using the

Applier methods, the Semantic Lens Application is thus built to annotate specific Semantic Lens Annotations on a Document. A Semantic Lens Applier is included inside each Semantic Lens, to allow for a closer relationship and an improved customization of its own operativity. Each Applier includes two objects, a SemLensApplierPreferences, used to customize the output and the behaviour of a specific Applier, and a AppMetaInfo, which includes a stats record for the Applier's activities and other meta-information on it. An applier is specific to one and only one type of Lens, as defined in LensesTypes, and as such it's a class ready to be extended if the needs arises. The applier itself is a large collection of public methods and shortcut-methods, ready to be used to select elements on a EARMARK document or Jena Model, to create or re-use Annotations, and to apply specific annotations to single elements or to set of them. Consequently this class aims to cover all the basic needs of the authorial process of creating a Semantic Lens for a specific document, and is the main actor in the workflow discussed in the previous page

- **SemLensApplierPreferences:** Class with all the options and preferences on the behaviour of a *SemLensApplier* instance. It is mainly used to set the options of the output log on System.out while performing an annotation activity. This class extends the *EarmarkFinderPreferences* Class, just like the *SemLensApplier* extends the *EarmarkFinder*
- **EarmarkFinderPreferences:** Same as above, this is a class with all the options and preferences on the behaviour of an *EarmarkFinder*. It is mainly used to set the options of the output log on System.out while performing an annotation activity.
- **AppMetaInfo:** This class is a generic Black Box for meta-information on a Semantic Lens Applier (which is represented by the *SemLensApplier* class). Instances of this class are located inside said Appliers, and they contain an *AuthorMetaInfo* object to store information on the Application author, as well as offering several statistical recording methods for the Applier during an application activity.
- **AuthorMetaInfo:** This class is a generic and simple optional Black Box to hold and store information about the author of a Semantic Lens Application. Most of the meaningful data should either be held or within the annotated document itself or in a structured form elsewhere (FOAF, for example), this object just offers pointers to them.

All classes and methods are fully documented, and a complete javadoc documentation is available.

6.3.2 Utilities Package

The utilities package is composed of just four classes, as well as one executable class with a main method in the sub-package "*exec*":

- **MarkupItemSetReducer:** This is a very simple utility class that contains tools used to systematically reduce a MarkupItem Set in a single or in multiple steps, by removing from it several other smaller Sets. It contains a *baseset* field with a Set of Markup Items, which is the original set from which the items should be removed, and methods to either remove all items except the ones specified in a smaller subsets or to remove from the *baseset* just those items part of the subset.
- **EARMARKDocumentLoaderWriter:** This is an utility class that assists the user in loading an EARMARKDocument from the file-system, or in writing it in several different formats. It can be used either to convert an EARMARKDocument to another representation (i.e.: RDF-XML to TURTLE), or it can be used to save an EARMARKDocument after it has been modified, for example, after several lenses are applied to it. It also allows to set namespaces. It is suggested to use the namespace maps defined in the *LensesNames* class. Out of simplicity, both output and input files should be in the same directory location.
- **LensesNames:** This public final class simply contains an enumeration of all the namespaces used by the technologies introduced so far, like semantic lenses and related ontologies, earmark, linguistic acts, and so on, as well as their abbreviation prefixes. It also contains some namespace/prefix maps ready to be used by the LoaderWriter.
- **LensesTypes:** This public final class just contains an enumeration of constants for encoding each allowed Lens Type within the Semantic Lens model. These types are associated to core classes like Annotations and Appliers.
- **LaunchEarmarkContentReaderExec:** This is a very simple executable class whose purpose is to recover chosen snippets from an earmark document and to analyze them on screen. It was used for tests during the application task over [Mik07]. It includes an EarmarkFinder and offers several methods for analyzing Markup Items and Ranges within an Earmark Document and to display them on screen through the System.out. The aim of this simple class was to assist in the authoring process of a Lens, in order to help in deciphering ranges references within an EARMARK document and translating them into a snippet of text.

All classes and methods are fully documented, and a complete javadoc documentation is available.

6.4 From Theory to Practice – *Mikalens* sub-package and its structure

I had already stated that SLAM was developed also in order to create a set of tools that could allow me to apply four chosen Semantic Lenses (*Document Structure, Rhetoric Organization, Citation Network* and *Argumentation*) over [Mik07], in order to put the theoretical model and my suggested methodology into practice.

Having readied a workable framework structured as discussed above, the board was set and I could now use SLAM to accomplish my intended task, and apply the four abovementioned lenses on the intended target document. While the results and the details of this activity, together with code samples, will be shown and detailed in section 8, both from an overall perspective as well as on a lens by lens basis, I deem it appropriate to illustrate here how I structured the SLAM sub-package that I employed to reach my goal, and which I named "`mikalens`".

It is a good way to show how SLAM could be used to tackle an annotation activity, as well as being a concretized proof of concept on both the workflow and the functionalities of the package itself.

As a consequence of the software architecture that I have designed and discussed before, the core of this code is composed by four concrete classes (*StructureAppOnMika, RhetoricApp…, CitationApp…, ArgumentApp…*), one for each of the Lenses that is to be applied on the document. These four classes all extend the *BasicLensApp* abstract class, which in turns implements the *SemLensApplication* Interface we have defined some pages ago.

Each of this classes represents the Lens Application of a specific lens, and is associated with its own Semantic Lens Applier, tied to the target document (which is the EARMARK model of the HTML version of [Mik07]). Most important of all, each of these classes has its own implementation of the *annotate* method, containing all the instructions for its application. These instructions use the Applier both to create new Annotations (or to fetch them from its storage Collection) and to find the targets (either Sets or single Items or Nodes) within the document which will be the subjects of these annotations.

Of course, all these four classes need to be instanced and the process has to be initialized somewhere, and that's why there is a fifth class within the package, the executable *LaunchLensesExec* Class. This contains the main method, and its structure is quite simple. First it uses the *EARMARKDocumentLoaderWriter* Class to ready the target document and to specify the output document, then it simply creates, in a sequence appropriate to the Semantic Lenses Stack, all four of the application classes, together with their appliers, then it specifies the appropriate preferences in terms of debugging output and statistical recording, and calls on their annotate methods. Finally, the end result is outputted both in TURTLE and RDF/XML linearizations.

7 TAL – Through a Lens

7.1 User Interaction with Lenses – The Focusing Activity

So far the main topic of this dissertation has been the **application** of Semantic Lenses as a way to enhance scientific document. The ultimate purpose of Semantic Publishing activities is improving the user experience and comprehension as they read semantically enriched articles. In accordance to this, the **application** of Semantic Lenses is the fundamental activity which provides us the with the right kind of metadata and their desired organization within a scientific document.

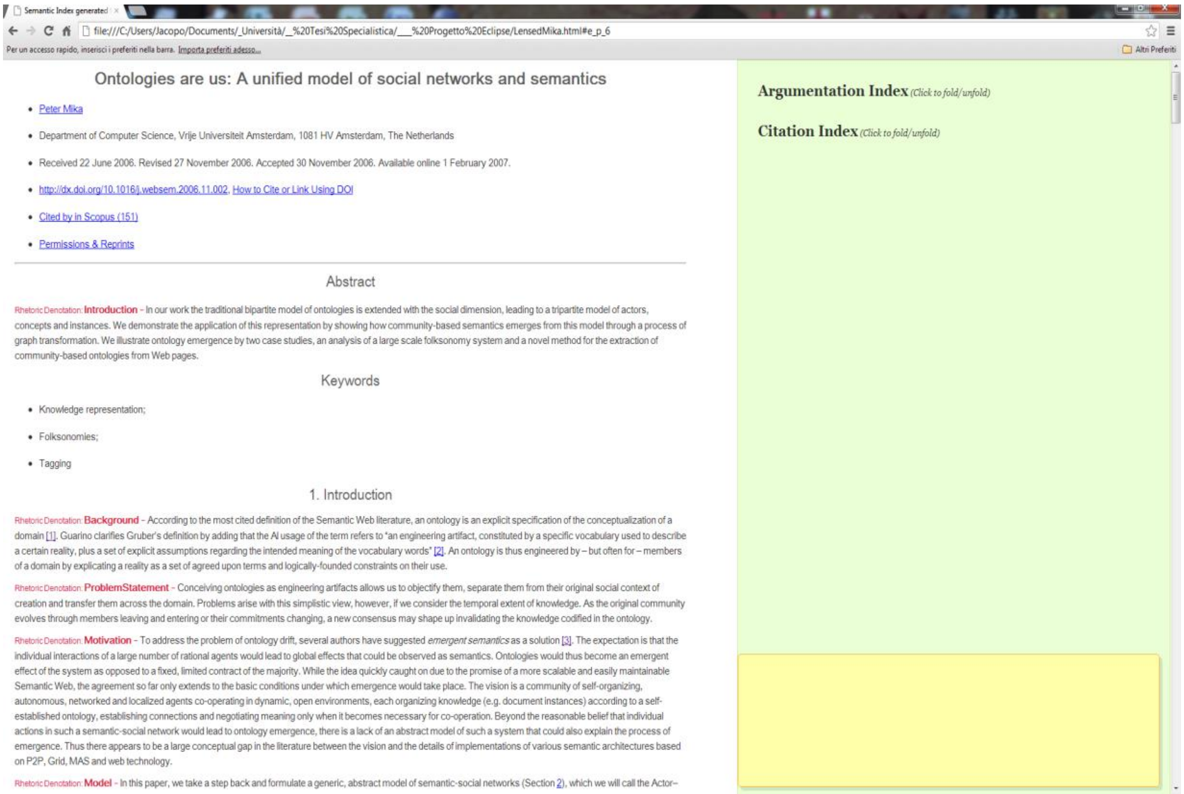
This being done, it is now time to consider the other part of the Semantic Lens model, the **focusing** of a lens, which is the complementary activity to the **application**, and has already been introduced in Sections 2 and 4.1. For the sake of clarity, allow me to recall its purpose.

The **focusing** of a lens is the activity through which the user will be able to choose which facet of the document will be enhanced, allowing the emergence of its specific semantics, the highlighting of additional related information. It will also enable a new set of interactions on it over the document itself. To put it in another way, the **focusing** is the act through which the reader is able to put to a good use the metadata methodically embedded in a Semantic Lens enhanced document, by allowing him to **focus** over a single aspect of the document, in order to have its reading and comprehension experience enhanced by the related additional information which will be presented to his attention. As a logical consequence for this definition, a successful **application** is required before any kind of **focusing** might take place.

In short, the **focusing** is the set of activities and tools that allow the users to tap into the organized metadata repository resulting from the **application** task, and to obtain some advantage in terms of comprehension, readability, interactivity or any other semantic enrichment.

The main focus of the research and development activities that led to this thesis has been the **application** of semantic lenses, but after considering some of the possibilities, I also opted to develop a small prototype for the creation of an enhanced and interactive document from the results of my semantic lenses road tests.

As it is, I have developed TAL – Through a Lens – which is another extension of the SLAM package (see section 6). It allowed me to use Java to generate automatically an enriched HTML version of the original "Ontologies are Us" article by Peter Mika [Mik07]. This enhanced version was originated by using the semantic information methodically stored in the annotated EARMARK version of [Mik07] which was the output of my application activity for four Semantic Lenses (see also section 4, 5, 8).


Semantic Index generated
file:///C:/Users/Jacopo/Documents/_Università/_%20Tesi%20Specialistica/___%20Progetto%20Eclipse/LensedMika.html#e_p_6
Per un accesso rapido, inserisci i preferiti nella barra. Importa preferiti adesso...
Altri Preferiti

Ontologies are us: A unified model of social networks and semantics

- Peter Mika
- Department of Computer Science, Vrije Universiteit Amsterdam, 1081 HV Amsterdam, The Netherlands
- Received 22 June 2006. Revised 27 November 2006. Accepted 30 November 2006. Available online 1 February 2007.
- http://dx.doi.org/10.1016/j.websem.2006.11.002, How to Cite or Link Using DOI
- Cited by in Scopus (151)
- Permissions & Reprints

Abstract

Rhetoric Denotation: **Introduction** - In our work the traditional bipartite model of ontologies is extended with the social dimension, leading to a tripartite model of actors, concepts and instances. We demonstrate the application of this representation by showing how community-based semantics emerges from this model through a process of graph transformation. We illustrate ontology emergence by two case studies, an analysis of a large scale folksonomy system and a novel method for the extraction of community-based ontologies from Web pages.

Keywords

- Knowledge representation;
- Folksonomies;
- Tagging

1. Introduction

Rhetoric Denotation: **Background** - According to the most cited definition of the Semantic Web literature, an ontology is an explicit specification of the conceptualization of a domain [1]. Guarino clarifies Gruber's definition by adding that the AI usage of the term refers to "an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words" [2]. An ontology is thus engineered by – but often for – members of a domain by explicating a reality as a set of agreed upon terms and logically-founded constraints on their use.

Rhetoric Denotation: **ProblemStatement** - Conceiving ontologies as engineering artifacts allows us to objectify them, separate them from their original social context of creation and transfer them across the domain. Problems arise with this simplistic view, however, if we consider the temporal extent of knowledge. As the original community evolves through members leaving and entering or their commitments changing, a new consensus may shape up invalidating the knowledge codified in the ontology.

Rhetoric Denotation: **Motivation** - To address the problem of ontology drift, several authors have suggested *emergent semantics* as a solution [3]. The expectation is that the individual interactions of a large number of rational agents would lead to global effects that could be observed as semantics. Ontologies would thus become an emergent effect of the system as opposed to a fixed, limited contract of the majority. While the idea quickly caught on due to the promise of a more scalable and easily maintainable Semantic Web, the agreement so far only extends to the basic conditions under which emergence would take place. The vision is a community of self-organizing, autonomous, networked and localized agents co-operating in dynamic, open environments, each organizing knowledge (e.g. document instances) according to a self-established ontology, establishing connections and negotiating meaning only when it becomes necessary for co-operation. Beyond the reasonable belief that individual actions in such a semantic-social network would lead to ontology emergence, there is a lack of an abstract model of such a system that could also explain the process of emergence. Thus there appears to be a large conceptual gap in the literature between the vision and the details of implementations of various semantic architectures based on P2P, Grid, MAS and web technology.

Rhetoric Denotation: **Model** - In this paper, we take a step back and formulate a generic, abstract model of semantic-social networks (Section 2), which we will call the Actor–

Argumentation Index *(Click to fold/unfold)*

Citation Index *(Click to fold/unfold)*

Fig. 21 - The TAL prototype

I can thus define TAL as a prototypical application that enables improved navigation and comprehension of a scientific document enriched by semantic lenses, allowing the user some basic tools to perform some focusing activities with its features, designed to assist the user in performing tasks that would benefit from an improved understanding of a the subject document, at the same time hiding the intrinsic complexity of a document enriched with RDF statements.

This section focuses more on the design and implementation aspects of TAL and on its generation, and on its features. The final application of TAL over [Mik07], as well as the results of a short user testing session, will be detailed in section 8. The rest of this section is structured as follows: In section 7.2 I will explore the ideas and the design of TAL in more detail, in section 7.3 I will focus on its features, in section 7.4 I will describe the overall structure of the Java code for its generation, while in section 7.5 I will give a brief overview on how some of the interactions were realized with CSS and JQuery[17].

[17] JQuery: http://jquery.com/

7.2 Through a Lens - TAL

With the development of Through a Lens, I have chosen to find some concrete applications for the many heterogeneous improvements on user experience we have already theorized by the application of Semantic Lenses. Of course, given that I had only a single annotated document on which to work, and that it was annotated only on some specific lenses, I had to make some choices on which features to implement through TAL.

First of all, the presence of only a single source document resulted in the obvious consequence of choosing to privilege intra-document applications in this prototyping, rather trying to create mock-ups of inter-document interactions which would not be founded within the any concrete data. I then selected some possible ways to enhance the presentation of the target paper by using the information within the applied Lenses at my disposal: *Document Structure, Rhetoric Organization, Citation Network* and *Argumentation*. The natural idea that came into my mind was to consider HTML as a way to produce a prototypical interactive front-end interface: this choice allowed me to combine the simplicity of se and the ease of development typical of HTML within a familiar environment for most user, regardless of their familiarity with Semantic Web Technologies, as well giving me a quick meter of comparison between the original HTML article and the enhancements that could be done on it.

I then had to choose which kind of focusing activity I could enable or promote with TAL. Considering that the aim here was to provide with meaningful enrichment all kind of users, regardless of their familiarity with Semantic Web technologies, I decided to discard anything related to the *Structure* lens, since the data stored on Structural Patterns assignment for each of the document components is more of significance in other circumstances, and I opted to fix my efforts on the most content related lenses.

From the description of the Semantic Lenses stack, the right place to start was the Argumentation Lens. The main idea here was to provide a quick way to summarize the argumentation structure of the paper, but in a way that would not be either too confusing for the reader and one which would be easy to interact with along with the original non-augmented text. Considering the aims of the scientific discourse, as seen in [DeW10], I decided to create a interactive argumentation index with all the claims made by the paper – its purpose would be to give a quick summary of the Author's intentions to the reader, allowing him to skim quickly through this enhanced index and see if there is any specific claims catching his attention more than enough to justify a deeper studying of the document. Of course, this index would also need to be expand to reveal the full structure of each argumentation behind each claim.

Moving on to the citation network, I acted with two different objectives: First, the idea of an organized, interactive and meaningful index for a specific facet that I had adopted for the Argumentation Lens still appealed to me, and I chose to develop a similar function for the Citation Network as well. The aim for this Citation Lens index would be to present all the citations, grouped by frequency, together with their CiTO denotations and pointers to the occurrences within the text. Second, the same relevant information could be made to appear as an on-hover contextual tooltip when moving the mouse over a citation within the main body of text, so that its purpose might be immediately be made explicit.

The Rhetoric Organization Lens has been involved with its DEO characterizations at the paragraph level, which would help to contextualize the flow of the rhetoric discourse of the article. These might be made explicitly available at the beginning of each paragraph, and could also provide some contextual inter-lens information when operating over other facets, for example by the means of rhetoric organization tooltips when considering the Argumentation lens.

These features are discussed in further details in the following section.

After I selected what should be done, the next obvious step was to choose how it could be made so. Since I had already developed a Java package, SLAM, which gave me the means to apply and to extensively manipulate semantic lenses annotations within an EARMARK document, the logical choice was to start from there. As a consequence of this, I decided to develop another small Java package and to task it with the generation of an enhanced static HTML page from the annotated EARMARK document obtained by the application of lenses on [Mik07]. This static HTML page would include all the original document contents, as well as all the relevant information gathered from the Lenses which would then be re-used within the interface. The idea was to have this information stored in additional HTML snippets, which would then either be part of the two indexes, or embedded in the page, either as an explicit or an hidden content to be displayed on occasion (like the tooltips).

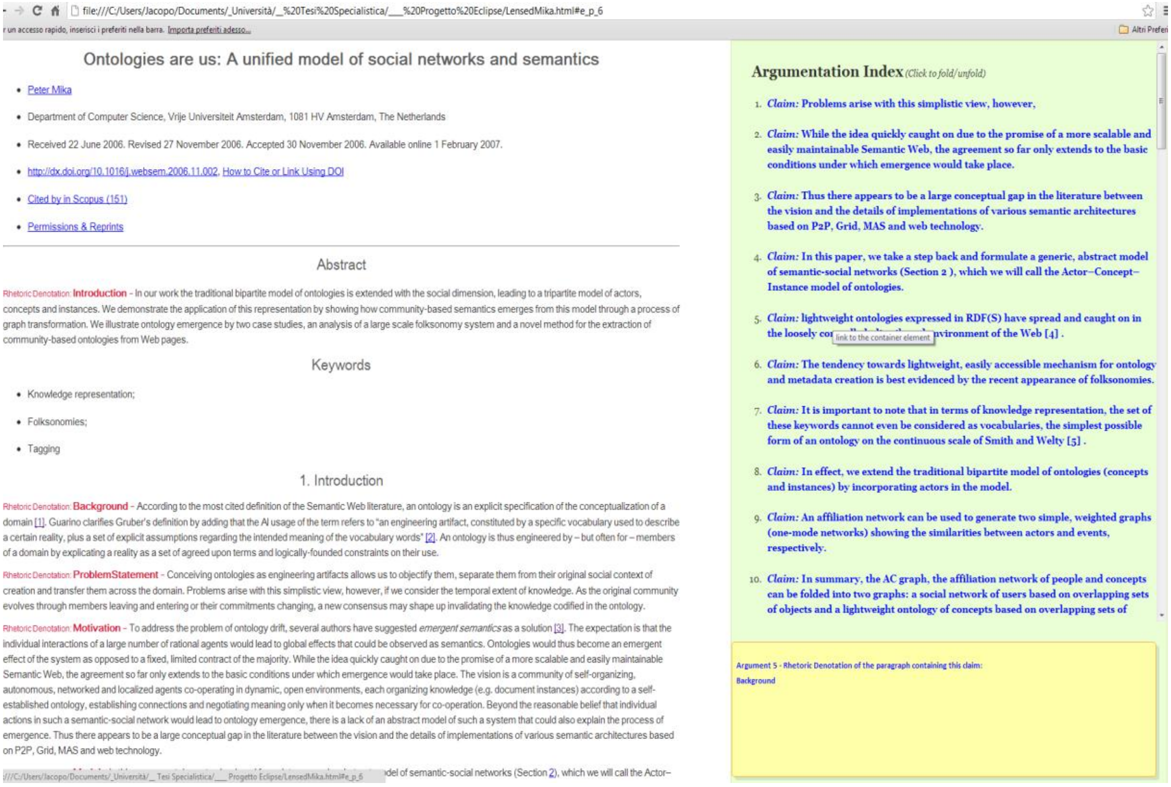
In order to deliver a reasonably effective interactivity while keeping the overall architecture of the prototype simple, the presentational part of TAL was done purely with the use CSS stylesheets in tandem with JQuery, which were mostly used to show and hide the additional content in response to the user interaction. All additional content is visibly marked as such, by the use of different backgrounds, different fonts and different colors, and most of it is located in a separate area of the screen, either in the top right section (the indexes), or in the bottom right (the tooltips)

7.3 TAL Features

In the previous part of this section I gave a quick overview of TAL from a general perspective (both from a software design and architectural standpoint), and mentioned which features were part of my design goal. In this section, I will go over each of them a little bit more in detail.

7.3.1 Explorable Argumentation Index

The Argumentation Index is one of the core features of TAL – its purpose is to provide both a summary of the document's claims and a fast access to the argumentation organization of the paper, as well as offering the reader the means to interact with it and to make the argumentation model explicit, allowing to see for each argument which part of the text correspond to the defined roles.


file:///C:/Users/Jacopo/Documents/_Università/_%20Tesi%20Specialistica/___%20Progetto%20Eclipse/LensedMika.html#e_p_6
r un accesso rapido, inserisci i preferiti nella barra. Importa preferiti adesso...
Altri Prefer
link to the container element
///C:/Users/Jacopo/Documents/_Università/_ Tesi Specialistica/___Progetto Eclipse/LensedMika.html#e_p_6

Ontologies are us: A unified model of social networks and semantics

- Peter Mika
- Department of Computer Science, Vrije Universiteit Amsterdam, 1081 HV Amsterdam, The Netherlands
- Received 22 June 2006. Revised 27 November 2006. Accepted 30 November 2006. Available online 1 February 2007.
- http://dx.doi.org/10.1016/j.websem.2006.11.002, How to Cite or Link Using DOI
- Cited by in Scopus (151)
- Permissions & Reprints

Abstract

Rhetoric Denotation: **Introduction** - In our work the traditional bipartite model of ontologies is extended with the social dimension, leading to a tripartite model of actors, concepts and instances. We demonstrate the application of this representation by showing how community-based semantics emerges from this model through a process of graph transformation. We illustrate ontology emergence by two case studies, an analysis of a large scale folksonomy system and a novel method for the extraction of community-based ontologies from Web pages.

Keywords

- Knowledge representation;
- Folksonomies;
- Tagging

1. Introduction

Rhetoric Denotation: **Background** - According to the most cited definition of the Semantic Web literature, an ontology is an explicit specification of the conceptualization of a domain [1]. Guarino clarifies Gruber's definition by adding that the AI usage of the term refers to "an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words" [2]. An ontology is thus engineered by – but often for – members of a domain by explicating a reality as a set of agreed upon terms and logically-founded constraints on their use.

Rhetoric Denotation: **ProblemStatement** - Conceiving ontologies as engineering artifacts allows us to objectify them, separate them from their original social context of creation and transfer them across the domain. Problems arise with this simplistic view, however, if we consider the temporal extent of knowledge. As the original community evolves through members leaving and entering or their commitments changing, a new consensus may shape up invalidating the knowledge codified in the ontology.

Rhetoric Denotation: **Motivation** - To address the problem of ontology drift, several authors have suggested *emergent semantics* as a solution [3]. The expectation is that the individual interactions of a large number of rational agents would lead to global effects that could be observed as semantics. Ontologies would thus become an emergent effect of the system as opposed to a fixed, limited contract of the majority. While the idea quickly caught on due to the promise of a more scalable and easily maintainable Semantic Web, the agreement so far only extends to the basic conditions under which emergence would take place. The vision is a community of self-organizing, autonomous, networked and localized agents co-operating in dynamic, open environments, each organizing knowledge (e.g. document instances) according to a self-established ontology, establishing connections and negotiating meaning only when it becomes necessary for co-operation. Beyond the reasonable belief that individual actions in such a semantic-social network would lead to ontology emergence, there is a lack of an abstract model of such a system that could also explain the process of emergence. Thus there appears to be a large conceptual gap in the literature between the vision and the details of implementations of various semantic architectures based on P2P, Grid, MAS and web technology.

del of semantic-social networks (Section 2), which we will call the Actor–

Argumentation Index *(Click to fold/unfold)*

1. *Claim:* Problems arise with this simplistic view, however,
2. *Claim:* While the idea quickly caught on due to the promise of a more scalable and easily maintainable Semantic Web, the agreement so far only extends to the basic conditions under which emergence would take place.
3. *Claim:* Thus there appears to be a large conceptual gap in the literature between the vision and the details of implementations of various semantic architectures based on P2P, Grid, MAS and web technology.
4. *Claim:* In this paper, we take a step back and formulate a generic, abstract model of semantic-social networks (Section 2), which we will call the Actor–Concept–Instance model of ontologies.
5. *Claim:* lightweight ontologies expressed in RDF(S) have spread and caught on in the loosely co nvironment of the Web [4] .
6. *Claim:* The tendency towards lightweight, easily accessible mechanism for ontology and metadata creation is best evidenced by the recent appearance of folksonomies.
7. *Claim:* It is important to note that in terms of knowledge representation, the set of these keywords cannot even be considered as vocabularies, the simplest possible form of an ontology on the continuous scale of Smith and Welty [5] .
8. *Claim:* In effect, we extend the traditional bipartite model of ontologies (concepts and instances) by incorporating actors in the model.
9. *Claim:* An affiliation network can be used to generate two simple, weighted graphs (one-mode networks) showing the similarities between actors and events, respectively.
10. *Claim:* In summary, the AC graph, the affiliation network of people and concepts can be folded into two graphs: a social network of users based on overlapping sets of objects and a lightweight ontology of concepts based on overlapping sets of

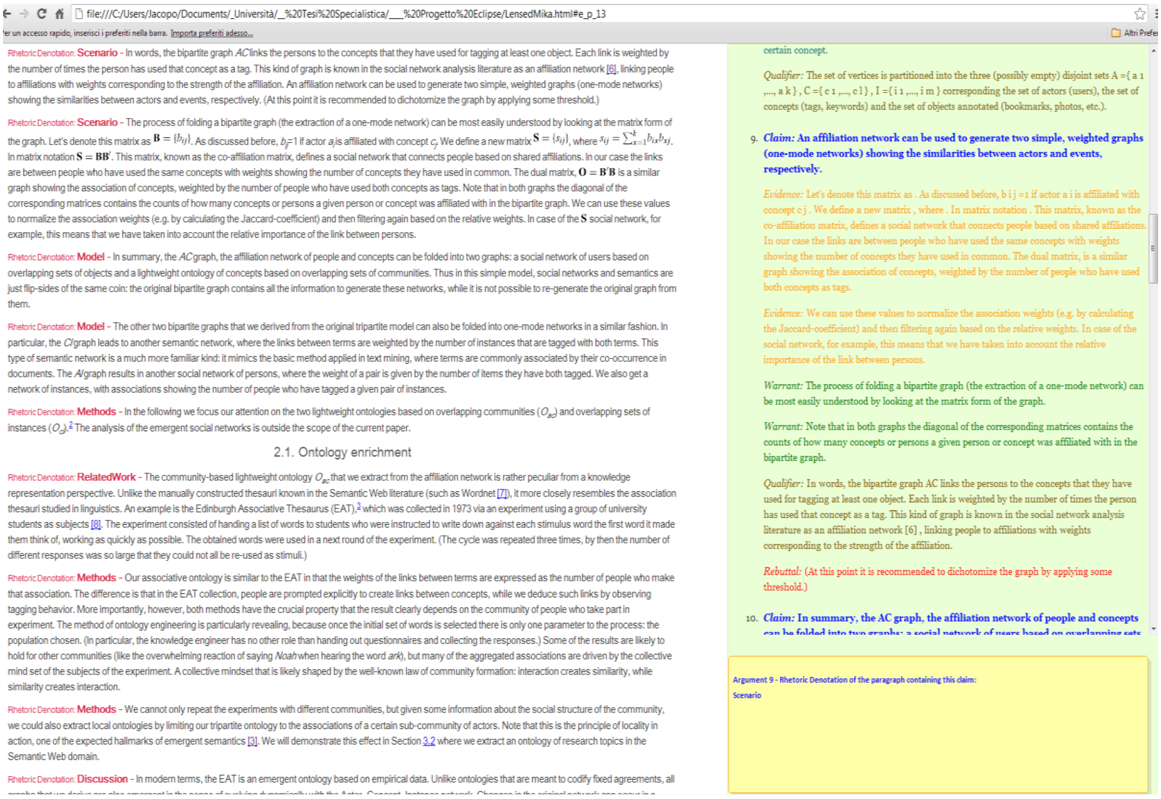
Argument 5 - Rhetoric Denotation of the paragraph containing this claim:
Background

Fig. 22 - The Argumentation Index of TAL

The Argumentation Index contents, extracted by processing the Argumentation Lens annotations over the target document, are stored in a

<div> separated from the rest of the paper contents, and is located in the top right part of the screen. The separation from the original content is marked by a different background and a different font.

By default the Index is folded, with only the title visible, inviting to the interaction with an explicit "*click to fold/unfold message*". If unfolded, the Index lists all the arguments within the paper by their claims (in bold, colored in deep blue), and they are ordered by the way in which they appear in the document.



file:///C:/Users/Jacopo/Documents/_Università/_%20Tesi%20Specialistica/___%20Progetto%20Eclipse/LensedMika.html#e_p_13

'er un accesso rapido, inserisci i preferiti nella barra. Importa preferiti adesso... Altri Prefer

Rhetoric Denotation: **Scenario** - In words, the bipartite graph *AC* links the persons to the concepts that they have used for tagging at least one object. Each link is weighted by the number of times the person has used that concept as a tag. This kind of graph is known in the social network analysis literature as an affiliation network [6], linking people to affiliations with weights corresponding to the strength of the affiliation. An affiliation network can be used to generate two simple, weighted graphs (one-mode networks) showing the similarities between actors and events, respectively. (At this point it is recommended to dichotomize the graph by applying some threshold.)

Rhetoric Denotation: **Scenario** - The process of folding a bipartite graph (the extraction of a one-mode network) can be most easily understood by looking at the matrix form of the graph. Let's denote this matrix as $\mathbf{B} = \{b_{ij}\}$. As discussed before, b_{ij}=1 if actor a_i is affiliated with concept c_j. We define a new matrix $\mathbf{S} = \{s_{ij}\}$, where $s_{ij} = \sum_{x=1}^{k} b_{ix} b_{xj}$. In matrix notation $\mathbf{S} = \mathbf{BB}'$. This matrix, known as the co-affiliation matrix, defines a social network that connects people based on shared affiliations. In our case the links are between people who have used the same concepts with weights showing the number of concepts they have used in common. The dual matrix, $\mathbf{O} = \mathbf{B}'\mathbf{B}$ is a similar graph showing the association of concepts, weighted by the number of people who have used both concepts as tags. Note that in both graphs the diagonal of the corresponding matrices contains the counts of how many concepts or persons a given person or concept was affiliated with in the bipartite graph. We can use these values to normalize the association weights (e.g. by calculating the Jaccard-coefficient) and then filtering again based on the relative weights. In case of the $\mathbf{S}$ social network, for example, this means that we have taken into account the relative importance of the link between persons.

Rhetoric Denotation: **Model** - In summary, the *AC* graph, the affiliation network of people and concepts can be folded into two graphs: a social network of users based on overlapping sets of objects and a lightweight ontology of concepts based on overlapping sets of communities. Thus in this simple model, social networks and semantics are just flip-sides of the same coin: the original bipartite graph contains all the information to generate these networks, while it is not possible to re-generate the original graph from them.

Rhetoric Denotation: **Model** - The other two bipartite graphs that we derived from the original tripartite model can also be folded into one-mode networks in a similar fashion. In particular, the *CI* graph leads to another semantic network, where the links between terms are weighted by the number of instances that are tagged with both terms. This type of semantic network is a much more familiar kind: it mimics the basic method applied in text mining, where terms are commonly associated by their co-occurrence in documents. The *AI* graph results in another social network of persons, where the weight of a pair is given by the number of items they have both tagged. We also get a network of instances, with associations showing the number of people who have tagged a given pair of instances.

Rhetoric Denotation: **Methods** - In the following we focus our attention on the two lightweight ontologies based on overlapping communities (O_{ac}) and overlapping sets of instances (O_{ci}).[2] The analysis of the emergent social networks is outside the scope of the current paper.

2.1. Ontology enrichment

Rhetoric Denotation: **RelatedWork** - The community-based lightweight ontology O_{ac} that we extract from the affiliation network is rather peculiar from a knowledge representation perspective. Unlike the manually constructed thesauri known in the Semantic Web literature (such as Wordnet [7]), it more closely resembles the association thesauri studied in linguistics. An example is the Edinburgh Associative Thesaurus (EAT),[3] which was collected in 1973 via an experiment using a group of university students as subjects [8]. The experiment consisted of handing a list of words to students who were instructed to write down against each stimulus word the first word it made them think of, working as quickly as possible. The obtained words were used in a next round of the experiment. (The cycle was repeated three times, by then the number of different responses was so large that they could not all be re-used as stimuli.)

Rhetoric Denotation: **Methods** - Our associative ontology is similar to the EAT in that the weights of the links between terms are expressed as the number of people who make that association. The difference is that in the EAT collection, people are prompted explicitly to create links between concepts, while we deduce such links by observing tagging behavior. More importantly, however, both methods have the crucial property that the result clearly depends on the community of people who take part in experiment. The method of ontology engineering is particularly revealing, because once the initial set of words is selected there is only one parameter to the process: the population chosen. (In particular, the knowledge engineer has no other role than handing out questionnaires and collecting the responses.) Some of the results are likely to hold for other communities (like the overwhelming reaction of saying *Noah* when hearing the word *ark*), but many of the aggregated associations are driven by the collective mind set of the subjects of the experiment. A collective mindset that is likely shaped by the well-known law of community formation: interaction creates similarity, while similarity creates interaction.

Rhetoric Denotation: **Methods** - We cannot only repeat the experiments with different communities, but given some information about the social structure of the community, we could also extract local ontologies by limiting our tripartite ontology to the associations of a certain sub-community of actors. Note that this is the principle of locality in action, one of the expected hallmarks of emergent semantics [3]. We will demonstrate this effect in Section 3.2 where we extract an ontology of research topics in the Semantic Web domain.

Rhetoric Denotation: **Discussion** - In modern terms, the EAT is an emergent ontology based on empirical data. Unlike ontologies that are meant to codify fixed agreements, all

certain concept.

Qualifier: The set of vertices is partitioned into the three (possibly empty) disjoint sets A ={ a 1 ,..., a k } , C ={ c 1 ,..., c l } , I ={ i 1 ,..., i m } corresponding the set of actors (users), the set of concepts (tags, keywords) and the set of objects annotated (bookmarks, photos, etc.).

9. ***Claim:* An affiliation network can be used to generate two simple, weighted graphs (one-mode networks) showing the similarities between actors and events, respectively.**

   Evidence: Let's denote this matrix as . As discussed before, b i j =1 if actor a i is affiliated with concept c j . We define a new matrix , where . In matrix notation . This matrix, known as the co-affiliation matrix, defines a social network that connects people based on shared affiliations. In our case the links are between people who have used the same concepts with weights showing the number of concepts they have used in common. The dual matrix, is a similar graph showing the association of concepts, weighted by the number of people who have used both concepts as tags.

   Evidence: We can use these values to normalize the association weights (e.g. by calculating the Jaccard-coefficient) and then filtering again based on the relative weights. In case of the social network, for example, this means that we have taken into account the relative importance of the link between persons.

   Warrant: The process of folding a bipartite graph (the extraction of a one-mode network) can be most easily understood by looking at the matrix form of the graph.

   Warrant: Note that in both graphs the diagonal of the corresponding matrices contains the counts of how many concepts or persons a given person or concept was affiliated with in the bipartite graph.

   Qualifier: In words, the bipartite graph AC links the persons to the concepts that they have used for tagging at least one object. Each link is weighted by the number of times the person has used that concept as a tag. This kind of graph is known in the social network analysis literature as an affiliation network [6] , linking people to affiliations with weights corresponding to the strength of the affiliation.

   Rebuttal: (At this point it is recommended to dichotomize the graph by applying some threshold.)

10. ***Claim:* In summary, the AC graph, the affiliation network of people and concepts can be folded into two graphs: a social network of users based on overlapping sets**

Argument 9 - Rhetoric Denotation of the paragraph containing this claim:

Scenario

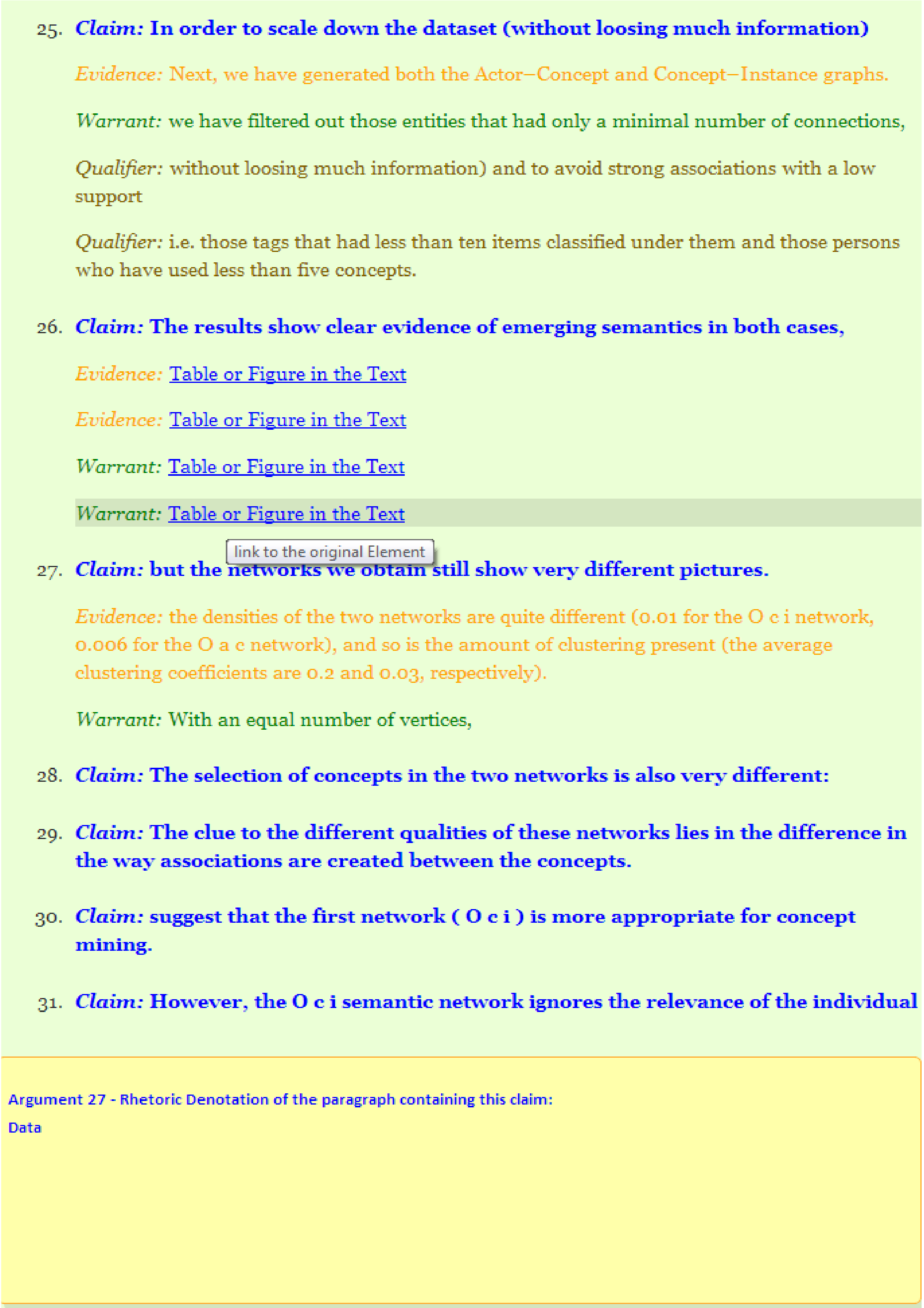
Fig. 23 - The Argumentation Index, Expanded

Clicking on each item in the ordered list of claims provokes two responses:

First, each claim is an anchor, so the main text on the left hand side scrolls until the beginning of the paragraph containing the claim.

Second, clicking on an argumentation claim causes the index item to expand, unveiling the structure for that argumentation. Each argumentation is represented as a list of its components, ordered by type and by position in the text. Each type of component (e.g. *Warrant, Evidence, Qualifier*, etc.) is explicitly labeled, and colored in a way to be immediately distinguishable from other types. If the component is a snippet of text, it is reported in its entirety within the index list. If it is a larger structure, such as another argumentation, a table or an image, a link pointing to that resource within the main text body is provided.

When the mouse hovers on a claim, a tooltip is also displayed in the tooltip area, which is located on the bottom right of the screen. The tooltip provides a quick reminder of the rhetorical connotation (as a DEO class) of the paragraph containing the claim (e.g. “Background”), as well as another anchor link to it. Clicking on the tooltip also expands the corresponding argument item in the index. All opened items can be unfolded by clicking again on them.



25. *Claim:* **In order to scale down the dataset (without loosing much information)**

    Evidence: Next, we have generated both the Actor–Concept and Concept–Instance graphs.

    Warrant: we have filtered out those entities that had only a minimal number of connections,

    Qualifier: without loosing much information) and to avoid strong associations with a low support

    Qualifier: i.e. those tags that had less than ten items classified under them and those persons who have used less than five concepts.

26. *Claim:* **The results show clear evidence of emerging semantics in both cases,**

    Evidence: Table or Figure in the Text

    Evidence: Table or Figure in the Text

    Warrant: Table or Figure in the Text

    Warrant: Table or Figure in the Text

27. *Claim:* **but the networks we obtain still show very different pictures.**

    Evidence: the densities of the two networks are quite different (0.01 for the O c i network, 0.006 for the O a c network), and so is the amount of clustering present (the average clustering coefficients are 0.2 and 0.03, respectively).

    Warrant: With an equal number of vertices,

28. *Claim:* **The selection of concepts in the two networks is also very different:**

29. *Claim:* **The clue to the different qualities of these networks lies in the difference in the way associations are created between the concepts.**

30. *Claim:* **suggest that the first network (O c i) is more appropriate for concept mining.**

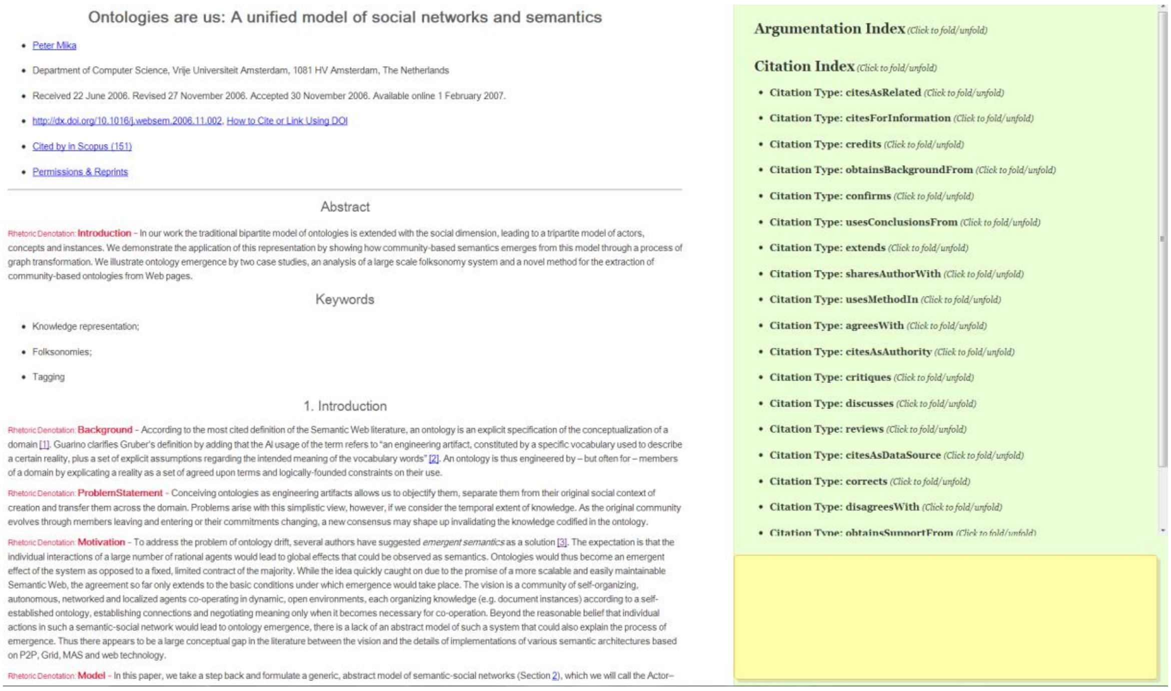
31. *Claim:* **However, the O c i semantic network ignores the relevance of the individual**

Argument 27 - Rhetoric Denotation of the paragraph containing this claim:
Data

Fig. 24 - TAL's Argumentation Index, Details

7.3.2 Explorable Citation Index

The Citation Index is the natural counterpart for the Argumentation Index, but realized over the Citation Network Lens. The purpose of the Citation Index is to give an organized and interactive Index for the whole set of citations made by the document, and to offer a level of readability and interactivity similar to the one seen in the Argumentation Index, by explicitly showing all the citations within the text, grouped by their related CiTO properties and ordered by frequency in the document, together with pointers to their occurrences within the text.



Ontologies are us: A unified model of social networks and semantics

- Peter Mika
- Department of Computer Science, Vrije Universiteit Amsterdam, 1081 HV Amsterdam, The Netherlands
- Received 22 June 2006. Revised 27 November 2006. Accepted 30 November 2006. Available online 1 February 2007.
- http://dx.doi.org/10.1016/j.websem.2006.11.002, How to Cite or Link Using DOI
- Cited by in Scopus (151)
- Permissions & Reprints

Abstract

Rhetoric Denotation: **Introduction** – In our work the traditional bipartite model of ontologies is extended with the social dimension, leading to a tripartite model of actors, concepts and instances. We demonstrate the application of this representation by showing how community-based semantics emerges from this model through a process of graph transformation. We illustrate ontology emergence by two case studies, an analysis of a large scale folksonomy system and a novel method for the extraction of community-based ontologies from Web pages.

Keywords

- Knowledge representation;
- Folksonomies;
- Tagging

1. Introduction

Rhetoric Denotation: **Background** – According to the most cited definition of the Semantic Web literature, an ontology is an explicit specification of the conceptualization of a domain [1]. Guarino clarifies Gruber's definition by adding that the AI usage of the term refers to “an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words” [2]. An ontology is thus engineered by – but often for – members of a domain by explicating a reality as a set of agreed upon terms and logically-founded constraints on their use.

Rhetoric Denotation: **ProblemStatement** – Conceiving ontologies as engineering artifacts allows us to objectify them, separate them from their original social context of creation and transfer them across the domain. Problems arise with this simplistic view, however, if we consider the temporal extent of knowledge. As the original community evolves through members leaving and entering or their commitments changing, a new consensus may shape up invalidating the knowledge codified in the ontology.

Rhetoric Denotation: **Motivation** – To address the problem of ontology drift, several authors have suggested *emergent semantics* as a solution [3]. The expectation is that the individual interactions of a large number of rational agents would lead to global effects that could be observed as semantics. Ontologies would thus become an emergent effect of the system as opposed to a fixed, limited contract of the majority. While the idea quickly caught on due to the promise of a more scalable and easily maintainable Semantic Web, the agreement so far only extends to the basic conditions under which emergence would take place. The vision is a community of self-organizing, autonomous, networked and localized agents co-operating in dynamic, open environments, each organizing knowledge (e.g. document instances) according to a self-established ontology, establishing connections and negotiating meaning only when it becomes necessary for co-operation. Beyond the reasonable belief that individual actions in such a semantic-social network would lead to ontology emergence, there is a lack of an abstract model of such a system that could also explain the process of emergence. Thus there appears to be a large conceptual gap in the literature between the vision and the details of implementations of various semantic architectures based on P2P, Grid, MAS and web technology.

Rhetoric Denotation: **Model** – In this paper, we take a step back and formulate a generic, abstract model of semantic-social networks (Section 2), which we will call the Actor–

Argumentation Index *(Click to fold/unfold)*

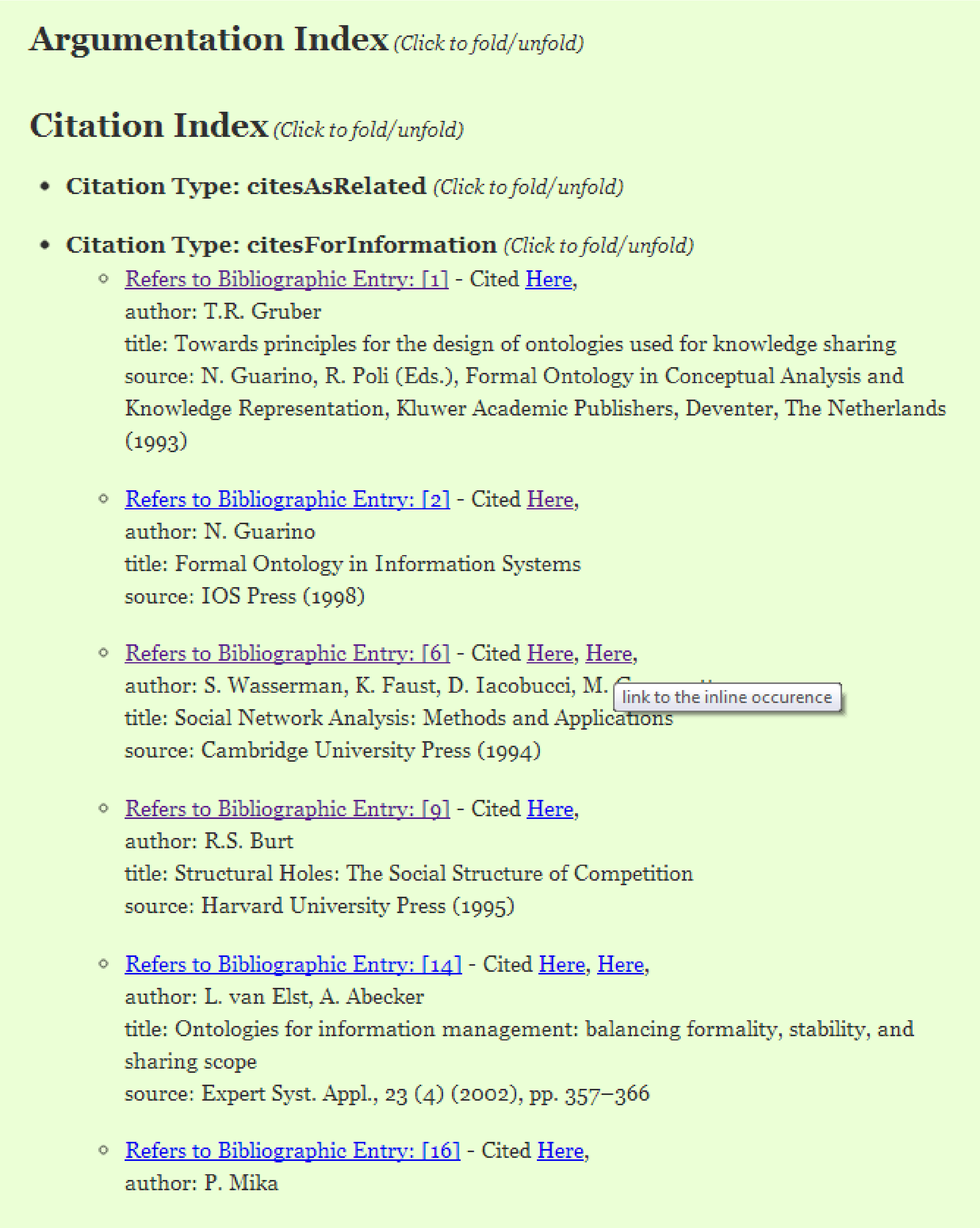
Citation Index *(Click to fold/unfold)*

- **Citation Type: citesAsRelated** *(Click to fold/unfold)*
- **Citation Type: citesForInformation** *(Click to fold/unfold)*
- **Citation Type: credits** *(Click to fold/unfold)*
- **Citation Type: obtainsBackgroundFrom** *(Click to fold/unfold)*
- **Citation Type: confirms** *(Click to fold/unfold)*
- **Citation Type: usesConclusionsFrom** *(Click to fold/unfold)*
- **Citation Type: extends** *(Click to fold/unfold)*
- **Citation Type: sharesAuthorWith** *(Click to fold/unfold)*
- **Citation Type: usesMethodIn** *(Click to fold/unfold)*
- **Citation Type: agreesWith** *(Click to fold/unfold)*
- **Citation Type: citesAsAuthority** *(Click to fold/unfold)*
- **Citation Type: critiques** *(Click to fold/unfold)*
- **Citation Type: discusses** *(Click to fold/unfold)*
- **Citation Type: reviews** *(Click to fold/unfold)*
- **Citation Type: citesAsDataSource** *(Click to fold/unfold)*
- **Citation Type: corrects** *(Click to fold/unfold)*
- **Citation Type: disagreesWith** *(Click to fold/unfold)*
- **Citation Type: obtainsSupportFrom** *(Click to fold/unfold)*

Fig. 25 – The Citation Index of TAL

The position and the way to open the Citation Index is the same of the Argumentation one. Once it expands, the index reveals a first list of CiTO citation properties, such as “*citesAsRelated*” or “*sharesAuthorWith*”. This list is ordered by frequency of use within the enhanced document – the most used properties appear first.

Clicking on any voice within the list of properties used in the article unfolds a nested sub-list with the references to all citation items exhibiting that property. To each item is associated a summary of the bibliographic reference information originally contained within the text, together with pointers to both the complete bibliographic reference (as made in the original article), as well as anchor links to each occurrence of the citation within the main text of the document. All opened items can be unfolded by clicking again on them.



Argumentation Index *(Click to fold/unfold)*

Citation Index *(Click to fold/unfold)*

- **Citation Type: citesAsRelated** *(Click to fold/unfold)*
- **Citation Type: citesForInformation** *(Click to fold/unfold)*
  - Refers to Bibliographic Entry: [1] - Cited Here,
    author: T.R. Gruber
    title: Towards principles for the design of ontologies used for knowledge sharing
    source: N. Guarino, R. Poli (Eds.), Formal Ontology in Conceptual Analysis and Knowledge Representation, Kluwer Academic Publishers, Deventer, The Netherlands (1993)
  - Refers to Bibliographic Entry: [2] - Cited Here,
    author: N. Guarino
    title: Formal Ontology in Information Systems
    source: IOS Press (1998)
  - Refers to Bibliographic Entry: [6] - Cited Here, Here,
    author: S. Wasserman, K. Faust, D. Iacobucci, M. G
    title: Social Network Analysis: Methods and Applications
    source: Cambridge University Press (1994)
  - Refers to Bibliographic Entry: [9] - Cited Here,
    author: R.S. Burt
    title: Structural Holes: The Social Structure of Competition
    source: Harvard University Press (1995)
  - Refers to Bibliographic Entry: [14] - Cited Here, Here,
    author: L. van Elst, A. Abecker
    title: Ontologies for information management: balancing formality, stability, and sharing scope
    source: Expert Syst. Appl., 23 (4) (2002), pp. 357–366
  - Refers to Bibliographic Entry: [16] - Cited Here,
    author: P. Mika

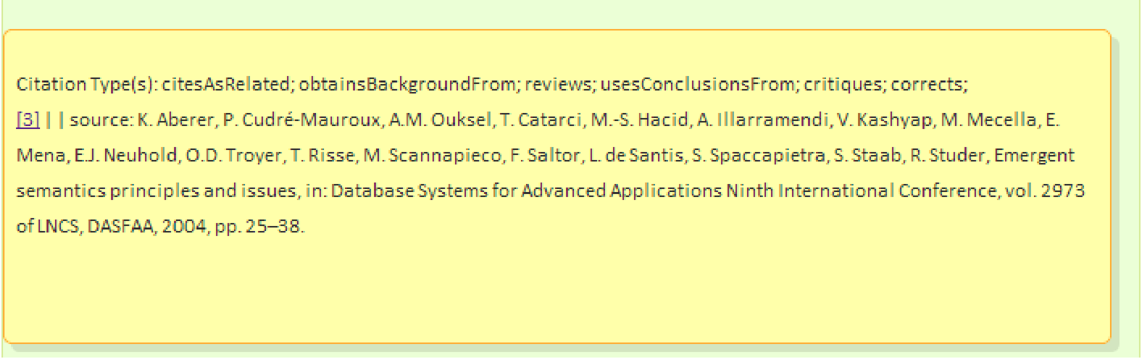
Fig. 26 - TAL's Citation Index, Details

7.3.3 Rhetoric and Citation Tooltips

The bottom right of the screen "*real-estate*" has been designed as an area reserved for displaying contextual tooltips which might present additional information on mouse-hover above some relevant elements.

We have already seen the Rhetoric denotation tooltips within the Argumentation Index: hovering over a Claim within the index reveals information on the DEO class associated with the paragraph containing the claim.

Another type of tooltip enabled for this demonstration is a Citation Network one. Hovering over a single citation reference (usually marked as "[#]") reveals its citation network information within the context. A tooltip appears, containing all CiTO properties associated to that citation occurrence, as well as a link to its entry within the original bibliographic reference list of the document.

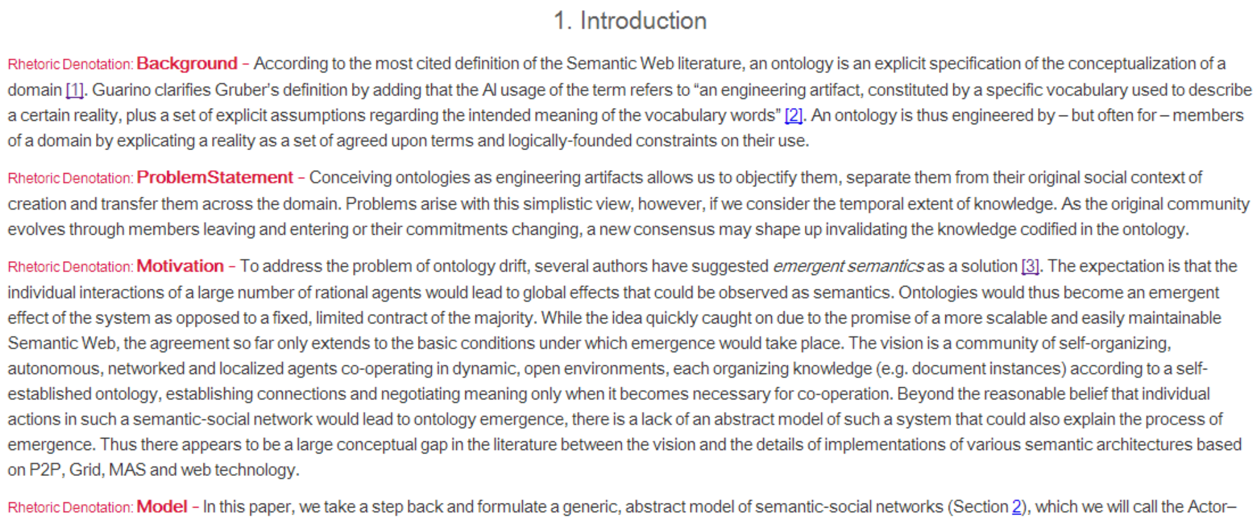


Citation Type(s): citesAsRelated; obtainsBackgroundFrom; reviews; usesConclusionsFrom; critiques; corrects;
[3] | | source: K. Aberer, P. Cudré-Mauroux, A.M. Ouksel, T. Catarci, M.-S. Hacid, A. Illarramendi, V. Kashyap, M. Mecella, E. Mena, E.J. Neuhold, O.D. Troyer, T. Risse, M. Scannapieco, F. Saltor, L. de Santis, S. Spaccapietra, S. Staab, R. Studer, Emergent semantics principles and issues, in: Database Systems for Advanced Applications Ninth International Conference, vol. 2973 of LNCS, DASFAA, 2004, pp. 25–38.

Fig. 27 - The Tooltip Area with a Citation Tooltip in TAL

7.3.4 Contextual Rhetoric Denotations

TAL also uses and highlights data from the Rhetoric Organization Lens. As well as enabling the Rhetoric tooltips discussed above, another type of Rhetoric denotation for the paragraph has been made available. The DEO characterizations at the paragraph level, are made explicit in a bright red text with a different font at the beginning of each paragraph, to help the reader to contextualize the flow of the rhetoric discourse within the document main content.



1. Introduction

Rhetoric Denotation: **Background** – According to the most cited definition of the Semantic Web literature, an ontology is an explicit specification of the conceptualization of a domain [1]. Guarino clarifies Gruber's definition by adding that the AI usage of the term refers to "an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words" [2]. An ontology is thus engineered by – but often for – members of a domain by explicating a reality as a set of agreed upon terms and logically-founded constraints on their use.

Rhetoric Denotation: **ProblemStatement** – Conceiving ontologies as engineering artifacts allows us to objectify them, separate them from their original social context of creation and transfer them across the domain. Problems arise with this simplistic view, however, if we consider the temporal extent of knowledge. As the original community evolves through members leaving and entering or their commitments changing, a new consensus may shape up invalidating the knowledge codified in the ontology.

Rhetoric Denotation: **Motivation** – To address the problem of ontology drift, several authors have suggested *emergent semantics* as a solution [3]. The expectation is that the individual interactions of a large number of rational agents would lead to global effects that could be observed as semantics. Ontologies would thus become an emergent effect of the system as opposed to a fixed, limited contract of the majority. While the idea quickly caught on due to the promise of a more scalable and easily maintainable Semantic Web, the agreement so far only extends to the basic conditions under which emergence would take place. The vision is a community of self-organizing, autonomous, networked and localized agents co-operating in dynamic, open environments, each organizing knowledge (e.g. document instances) according to a self-established ontology, establishing connections and negotiating meaning only when it becomes necessary for co-operation. Beyond the reasonable belief that individual actions in such a semantic-social network would lead to ontology emergence, there is a lack of an abstract model of such a system that could also explain the process of emergence. Thus there appears to be a large conceptual gap in the literature between the vision and the details of implementations of various semantic architectures based on P2P, Grid, MAS and web technology.

Rhetoric Denotation: **Model** – In this paper, we take a step back and formulate a generic, abstract model of semantic-social networks (Section 2), which we will call the Actor–

Fig. 28 - Contextual Rhetoric Denotations in the TAL prototype

7.4 Prototype generation from the test-bed through Java

After choosing which features should be made available in the TAL prototype, as well as its overall architecture, the next step was to actually implement the code for its generation. I have already explained that the basic concept was to create an enhanced HTML document from the original paper and its version with annotated lenses, and to store in this HTML output the additional information, gathered from the applied lenses, ready to be displayed and manipulated within the browser.

In order to manufacture the TAL prototype HTML, I have decided to rely on Java, to re-use as many as the methods I had already defined in SLAM, as well as to be able to continue working with both the EARMARK and Jena API. As already stated, my objective was to automate as much as possible the prototype generation – the only manual additions were the full-sized images for the article figures.

The main idea was first to use all three aforementioned libraries to extract all the relevant information required to populate the Indexes, the Tooltips and the Denotations; hence to reprocess said information and store it in appropriate data structures; the next step was to pass all of them to an HTML formatter which would produce all the appropriate HTML snippets and then rearrange them appropriately within the original (which in turn would need to have additional identifiers to anchor intra-document links), to finally output the enhanced TAL hypertext.

This approach was chosen in order to separate the presentation task as much as possible from the information gathering task. As the small TAL package is structured now, it would be easy to re-use the same semantic emergence methods to extract all the very same relevant data structures, and then to have them outputted in a completely different way, without the need to rewrite anything with the exception of the Formatter and the executable, which are located in a different class.

I tried to implement TAL classes and methods to be as document-independent as possible, but, of course, it was not possible completely do so, mainly for two reasons:

The first motivation is tied the Semantic Lenses themselves. There is no unique or univocal way to annotate a set of Lenses on a document – even disregarding the obvious different choices different authors could make on the content meanings, there are often several alternative paths that could be taken to apply a lens and still result in annotations compliant with Semantic Lenses stack. For

instance, the Citation Network Lens could have been made by using the inverted set of properties (isCitedBy and its sub-properties instead of cites and its sub-properties), switching the subjects with the objects, and it would still have been a valid choice. As a consequence of this, the prototype searched for the semantic lenses annotation in a way coherent with the methodology I have used to annotate them. In short, the way I fetched these annotations was dependant on the methodological choices I made (and already detailed previously within this dissertation), as well as to the limitations of both APIs and to the implementation compromises I already discussed.
The second reason is mainly due to the scope of this work. There are only some lenses annotated on the enriched version of the target document, and there is only one enriched document to work with. As it is, some information that could be ideally encoded in a different way (such as bibliographic reference details for the citations used within [Mik07]), was instead gathered and extracted right from the text. Also, in order to avoid making huge changes in the original document layout, its structure was not fundamentally changed. Thus, the way the TAL prototype is built closely and intentionally resembles the original, with separate additions made evident to the user. However, the methodology at the core of TAL is completely re-usable, and the document-tied aspects are limited to what discussed above.

The TAL package itself is made by a core group of 8 classes, with an additional class defining the executable, positioned within a sub-package. A brief list of the classes will follow:

TAL Core Package

- **SemanticIndex:** This is the main "***focusing***" class of the TAL package. It finds all the relevant Lens information and stores them in appropriate data structures. This class is tasked with the creation of the data structures used by the TAL prototype over an annotated EARMARKDocument. It uses a SLAM EarmarkFinder, which is associated with the EARMARKDocument version of the target article in order to extract information from the applied Lenses. It has several public methods to build the data structures that are to be processed by the HTMLTALFormatter, as well as several private utility methods that are used in order to accomplish this task. *(Note on the name: TAL original purpose was solely to create a "Semantic Index of Argumentations" - it was later extended and renamed to avoid ambiguities.)*
- **ArgumentComponent:** This class is a just a data structure to contain all significant information about a single Component of an Argumentation, according to Argumentation Lens and AMO. It stores the type of the

argument component, its properties, its identifier within the document, its component, its container (if any) and its textual content (if any, if not - a link to its content), as well as the DEO annotation of its container (if it's a Claim). This class implements the comparable interface in order to be sorted within the Argumentation Index (by AMO property).

- **SemIndexArgument:** This class represents a single Argumentation (as specified by AMO and the Argumentation Semantic Lens) on a Document annotated with an Argumentation Lens. An argumentation, for the purpose of the Argumentation index automatic generation, is made of a list of claims (1+) and several other components, both of these are represented as ArrayList(s) of ArgumentComponent(s), which a specialized class acting as data container. This class is a data structure of its own, and is used by the SemanticIndex to make the definitive collection of Argumentations on a Document. A collection of this SemIndexArgument(s) is then to be given to a Formatter for output generation.
- **CitoComponentA:** This is the data structure class for the CiTO components organized as a Citation Index. It contains all the meaningful data that are to be displayed within a Citation Index sub-item. A Collection of these component is created by the SemanticIndex class and the processed by the HTMLTALFormatter into the final TAL Citation Index.
- **SortedEntryOfCitoA:** This class is used to implement the Comparable Interface for a SortedMap of <Integer,CitoComponentA>, and is used to sort entries within the Citation Index. The SortedMaps represent a single item entry in the TAL Citation Index These entries, which represents CiTO properties are first sorted by the number of references sharing the same property in descending order. Then, the entries with the same number of occurrences are sub-sorted in alphabetical order.
- **CitoComponentB:** This class is the data container for the CiTO information to be reworked into inline tooltips. It contains all the relevant information that are to be included in the tooltip, extracted both from the Citation Network Lens and from the original text for each citation made by the original document. These components are created by the SemanticIndex class and then processed into HTML fragments to be embedded in the TAL prototype by the HTMLTALFormatter.
- **RhetFlagsComponent:** This class is simply a data structure to contain all significant information gathered from the Rhetoric Organization Lens denotations over a paragraph. It stores the identifier of the paragraph and its DEO denotation, to be rebuilt into the Rhetoric Flags

for TAL. It is created by the SemanticIndex class and a collection of these is sent to the HTMLTALFormatter to build the final output.

- **HTMLTALFormatter:** This is an abstract class which is just used to collect all the methods finalizing the construction of the HTML prototype of TAL. There is a method to re-adjust the original markup of the HTML version of Mika's Ontologies are us, and a method for each of the features of TAL to be built from the appropriate data structure into HTML snippets. A method to merge all of these snippets in the final TAL HTML, and a method to write the HTML to a file are also available

TAL.EXEC sub-package

- **LaunchIndexCreatorExec:** The executable for the creation of TAL from Mika's "Ontologies are us" and its enriched version with lenses applied on it. It requires both the original HTML file and the annotated version of the document. This executable produces the TAL HTML prototype over the target document.

I will now illustrate the basic data processing flow within this package in order for the final TAL prototype HTML to be outputted.

The heart of this package lies, as already observed, within the SemanticIndex class. This class is instanced by associating with it an EARMARKDocument, with Semantic Lenses applied on it, and a SLAM EarmarkFinder to explore it. It then offers several public methods to create the data structures for representing the meaningful information that is to be reprocessed and converted to a presentation format by the HTMLTALFormatter and its methods.

Within the SemanticIndex class, getters and setters aside, there are five main public methods, two for the generation of the Argumentation Index, and one each for all other features. Within the TAL prototype creation process, this class is expected to do all those activities related to the extraction of information both from the applied lenses and from the document itself, in order to construct appropriately planned data containers ready to be used to build out the enriched HTML interface.

The Argumentation Index is represented as an ArrayList of SemanticArgument, each of them holding two collections (one sorted, the other not sorted) of ArgumentComponent to represent the internal structure of the Argumentation as denoted by the Argumentation Lens application over the target document.

The SemanticIndex class first finds all the argumentations within the annotated document by calling the *findAllArguments* method, whose results are then processed into the final output format by the *readyArgumentCollection* method.

This is a method to ready said ArrayList of SemIndexArgument for the HTMLFormatter, in order to output the Argumentation Index of TAL. This method processes a Set of MarkupItems of the `rdf:type "amo:Argument"` (found by *findAllArguments*) and extracts the meaningful information to create a list of SemIndexArguments. Each SemIndexArgument is populated by ArrayLists of ArgumentComponents, which are the data structure for the data to be processed by the HTMLTALFormatter, whether claims or other components. In order to do so, two additional private methods are called: First, it uses *prepareSingleArg* to create and fill each SemIndexArgument by processing a MarkupItem (representing said argumentation within an EARMARK document) into the argumentation components, which will be unpacked and reprocessed as collections of ArgumentComponent storage classes. It also uses the *findDeoInfoOnComponentContainer* method to gather the information for the Rhetoric Tooltips to be shown on mouse hover on claims within the Argumentation Index.

The SemanticIndex class is also responsible to ready all other data structures. The one used to obtain the Rhetoric Denotation for each paragraph is perhaps the simplest one, as it is purely a collection of RhetFlagsComp, outputted by the *readyRhetFlagsPara* method, which searches for all DEO classes expressed over paragraphs within the target document.

Citation Tooltips are represented with an ArrayList of CitoComponentB, which is in turn built by the SemanticIndex class, as usual. In order to gather the appropriate information, it uses the *intradocClaimsB* method to create said ArrayList. First, it recovers all earmark elements and gets for each of the statements on which said element is the subject. Then, for each set of statements, it examines the predicate, and if it is a CiTO property part of the Citation Network lens, it is flagged as relevant. These assertions are saved, and, if some CiTO related assertions are found, the element reference is saved as well. For each element containing some interesting CiTO assertions, the method recovers several other data - for example, it uses the object of the CiTO statement to extract bibliographic information about it contained within the target document, as well as its internal reference label. These data are obtained by calling some other private methods, like *getBiblioRefLabel*, *parseCitoBiblioData* (these two methods are strictly document-related) and *getTrueTextContent*. This last one is perhaps the most interesting of the utility methods, as it allows to recover the actual textual content of an EARMARK Element, without having it mixed with its attributes text content.

The Citation Index data structure is the most complex one. While it is based on CitoComponentA, which is a container class much similar to CitoComponentB, and it is populated in a similar way to their cousins, the way these are organized to represent the final Citation Index is significantly different. The Citation Index is represented as a Map of String together with a nested SortedMap made in turn of Integers and CitoComponentA. This complex data structure is used in order to have each CiTO property (the first level Map, based on Strings) mapped to a structure (the second level SortedMap) of which contains the data on each occurrence using that property to denote a citation made within the original text. All this is obtained as the result of the *intradocClaimsA* method of the SemanticLens class.

After all required data structures are correctly created from the applied lenses, the core of TAL generation is tasked on the HTMLTALFormatter abstract class and its static methods.

The HTMLTALFormatter has one method for each of the feature, (*buildArgumentIndex* for the Argumentation Index together with its own tooltips, *buildFragmentA* for the Citation Index, *buildFragmentB* for the Citation Contextual Tooltips, and *buildRhetFlags* for the Rhetoric Denotations at the paragraph level). Each one of these methods requires in input the data formats presented above, and they all output a DOM Document with the appropriate HTML fragments ready to be merged within the TAL prototype. However, before launching them, the executable first calls the *improveMika* method of the HTMLTALFormatter, which is a document-dependent method that reads as a DOM Document (via the *parseXML* method) the original "cleaned" HTML of version of [Mik07], the same that was used to create the EARMARK document on which the lenses were applied. This *improveMika* method simply prepares the base of the TAL prototype, by adding a number of appropriate identifier attribute, to act as the receiving anchors for intra-document links in tooltips and indexes.

Finally, everything is merged in a single file and put into place by the *generateMergedDoc* method of the HTMLFormatter class, which results in the TAL prototype ready within a single DOM Document object, which is then written to the filesystem by the *writeHTMLToFile* method.

Of course, all the package comes fully documented with the appropriate javadoc.

7.5 Other Implementation details

Having manufactured the improved and enriched HTML source containing all the additional data required for TAL's features to function, the final stage of this development task was to create a suitable presentational interface for it, enabling the user interaction within the focusing activity.

To do so, I have decided to avoid using any kind of server-side technology, and I chose to rely only on Cascading Style Sheets and JavaScript (especially on the JQuery library). The presentation style is very minimal, with an emphasis on distinguishing the original content from the enriched information, by the means of different colors, fonts and backgrounds.

I made use of three CSS stylesheets, applied in cascade, each one absolving a different role.

- The first to be applied is a basic stylesheet for the textual content of the main article, which is mainly an extremely reduced and simplified version of the presentation style for the original target document.
- The second one is a stylesheet for the overall liquid layout of TAL, including the positioning of the area reserved for the Lenses Index. It also includes instructions regulating the fonts and the coloring of the elements.
- he third one regulates the style of tooltips and the overall presentation of the tooltip area.

Initially, all transitions from visible to invisible element display were planned to be executed purely by CSS, but in order to have a smoother user-experience, allowing for better control of the interaction I switched to JQuery to render these effects.

All JQuery code related to TAL is within a single file, and it is not exceptionally complex. First of all, a "change" function is defined, which is used to toggle the display of index lists and sub-list items. Then that function handler is bound to the on-click even over the appropriate html components to enable interactivity over the index.

In order for the tooltips to display properly (they should appear on mouse-enter over the desired interactive element, but they should not disappear on mouse-exit, or else no interaction would be possible with their content), I used the mouse-enter event over the appropriate element classes, and I check a global variable referring to the last tooltip displayed to remember which element is to be hidden when the tooltip is changed, and to avoid inappropriate repetitions when hovering in and out of the same element.

All JQuery code uses the console to output debug information.

8 Evaluation of SLAM and TAL

8.1 Enacting the methodology and using the framework to field test lenses

In section 4 I have presented the Semantic Lens model for the enrichment of scientific document and its two fundamental activities (the **application** of Semantic Lenses, defined as the act of authoring the semantic annotations enriching the document or its components and embedding them within it, and the complementary **focusing** of a Lens, which is the act of using said enhanced data to highlight a specific aspect of the enhanced document), introduced its related ontologies, discussed on its purpose. In section 5, I proposed a general purpose methodology for their concrete usage, especially within the application of four specific lenses (*Document Structure, Rhetoric Organization, Citation Network* and *Argumentation*).

In section 6 I expounded the development and the features of SLAM – a Java framework for Semantic Lenses Application and Manipulation, which I designed and developed in order to have a tool to facilitate and streamline the application of Lenses over EARMARK documents.

In section 7, I presented the TAL – Through A Lens – prototype HTML interface for the focusing of certain lenses over an annotated version, explained the underlying technology and listed its useable features.

In short, we are now familiar with a model and its related technologies, and we also have a methodology, together with the tools that enable us to act upon this methodology, and to produce concrete results. The board is thus set and the pieces are ready to discuss the results of the proof of concept test for Semantic Lenses that I performed, applying the four abovementioned lenses over the EARMARK conversion of Peter Mika's "Ontologies are Us" article [Mik07], and obtaining both its enhanced version with lenses applied on it and a TAL prototype generated from it.

This was a goal I had declared since the introduction of this dissertation, and I had already written at length (see section 4.10) about the motivations leading to this choice. This is the first large scale testing activity that has been done over multiple facets Semantic Lens model, and one of the aims of this specific part of my project was to discover which kind of challenges and obstacles would surface during the practical realization of what theorized so far. This activity would also allow to exemplify some of the enhancements and of the advantages of these Semantic Publishing techniques, in a way similar to what has been done in [SKM09].

Some of these issues or difficulties, many of them related to a specific level of Semantic Lenses, could be theorized and summarily analyzed even before going in directly over a document-level application – has I had already anticipated in section 5.

In the following pages I will focus more on the details of the application process (including several aspects involving the authoring, the vocabularies practical applicability and completes, as well as document analysis), explaining my authoring and implementation choices and compromises, together with code samples, and also detailing any interesting finding and inter-lens relationship I could have observed. Obviously, I had chosen to proceed following the **bottom-up** methodology already introduced as much as possible, and these results will be presented in section 8.2 (and its subsections) on a lens by lens basis, rising in the Semantic Lens stack from the lowest level of interest through the more content related ones.

After that, I will analyze some numerical statistics, as recorded by SLAM during the application activity, and discuss any interesting figures, which might perhaps give some interesting insight on the relationships between different levels of the Lenses Model, or on the most used denotations.

In section 8.4 I will proceed by discussing the overall results of my application test, in order to gain from the lessons learned during this lengthy experience, and I will also suggest and justify some changes in both the methodology and in the ontologies that I personally believe could improve future activities.

In section 8.5 the TAL prototype will be put to the test, and I shall detail both the nature of the user testing, as detailed also in [PVZ12], as well as its results.

Finally, in section 8.6 the overall findings of this road-test experience will be summarized.

8.2 Applying the annotations, lens by lens

I shall now proceed in discussing how the application of the four Semantic Lenses was authored and implemented over the target document. For each of the lenses I will make some reminders to the methodology (see sections from 5.6 to 5.9), provide some example lines of code from SLAM, and discuss the decisions I made in order to complete the application task, as well as any shortcomings of the methodology (or the model), or any implementation compromise I might have adopted, and consider the final results. It should be noted that it was not feasible to have the optimal conditions theorized in section 5.1 in terms of Author involvement, and most of the choices I made are strictly tied by my own personal interpretation, even if I tried to "emulate" the intentions of the Author in order to perform them as correctly as possible.
As I said earlier, I am going to follow my **bottom-up** method in covering the Semantic Lenses Stack, so I shall start with the *Document Structure* Lens.

8.2.1 The Application of the Document Structure Lens

First of all, the application of this lens does not aim to create a general purpose univocal of (X)HTML with Structural Patterns – this would be a serious misconception, as explained in detail in section 5.6. However, what should be done in order to avoid dealing with each markup element at the time, which is obviously extremely impractical considering the hundreds of element composing even a short document such as [Mik07], has been already pointed out: the basic idea is to find a subset of all the possible (X)HTML Element -> Pattern assignments that is valid for the document object of this application.
In order to do so, I have started by those elements whose patternization is universally acceptable, like, as already exemplified, the `<br/>` element which corresponds to a Milestone pattern. However, a good deal of markup elements of the original were still left out by this process, including the most significant ones, like `<div>`, `<p>` and `<span>`. In addition to that, I have already mentioned that the document structure of the original is somewhat peculiar – all the markup is held within the `"centerPane"` `<div>`, and the main text body, as well as its subsections, are not grouped by containing div elements, but are simply a succession of paragraph. In short, the overall document structure, even if orderly, has a very shallow depth.

Thus I went on to analyze the content models used by each element within the target document, as grouped by general identifier, and reached the following conclusions as summarized within Table 3.

Table 3 – Pattern Assignment over the target document

(X)HTML TAG	*Pattern Assignment*	*Notes*
HTML	*Record*	The <html> root tag is, regardless of the document, a perfect match for the definition of a Record, which is a Container whose contents are heterogeneous and non repeatable.
BODY	*Container*	Corresponds to the quintessential Container
HR, BR	*Milestone*	We have already seen that elements like these are the quintessential Milestone pattern
IMG	*Milestone*	The way the <img> element is used, it responds perfectly to the definition of a content-less Element (with the exception of attributes) whose position in the document is meaningful
UL	*Table*	We have defined the Table Pattern as a Container that mandates a repetition of homogeneous sub-structures within its content. Elements like <ul> or <ol> fit this pattern perfectly, as they are made by an indefinite repetition of <li> elements.
OL	*Table*	Same as above
LI	*Block*	As a subclass of the Bucket ghost pattern, a Block is a container of text and other substructures which does NOT allow other block elements in its content. This corresponds perfectly with the usual content model of list items for ordered and unordered lists and table components. <li> elements within this document match the Block pattern perfectly.
TABLE	*Container*	As discussed in section 5.6, it was impossible to characterize a <table> element as a Table pattern, due to the limitations of the Table patterns, which requires the repetition of just a single element. As a consequence of this, I was forced to scale back the pattern assignment from the more appropriate Table sub-class of Patterns to the Container Pattern
TBODY	*Table*	The same reasoning made for the lists applies very well to the table sub-elements (but not, sadly, for the <table> element as a whole, as shown above)
THEAD	*Table*	Same as above
TR	*Table*	Same as above

TH	*Block*	As a subclass of the Bucket ghost pattern, a Block is a container of text and other substructures which does NOT allow other block elements in its content. This fully corresponds with the usual content model of list items for ordered and unordered lists and table components. `<th>` elements within this document match the Block pattern perfectly.
TD	*Block*	Same as above
DL	*Container*	As discussed in section 5.6, it was impossible to characterize a `<dl>` element as a Table pattern, due to the limitations of the Table patterns, which requires the repetition of just a single element – while a `<dl>` contains a regular repetition of two: `<dt>` and `<dd>`. Consequently, I was forced to scale back the pattern assignment from the more appropriate Table Pattern to the Container Pattern
DT	*Block*	As a subclass of the Bucket ghost pattern, a Block is a container of text and other substructures which does NOT allow other block elements in its content. This fully corresponds with the usual content model of list items for ordered and unordered lists and table components. `<dt>` elements within this document match the Block pattern perfectly.
DD	*Block*	Same as above
DIV	*Container*	In general the content allowed within a `<div>` (X)HTML element makes it one of the most versatile ones, but in order to assign a Pattern we have of course to consider its specific use. As it is, in [Mik07] a `<div>` is almost never used to contain directly any kind of text. However, many `<span>` and `<a>` elements are directly contained within them, and we have no choice but to classify these elements as Inline if we don't want to overly focus on an element by element approach which would imply an increased complexity of this activity. As a result of this, I will assign the Container pattern to all div elements that have no direct Inline children, and the Block pattern to the others. **This first group is the large majority.**
DIV	*Block*	As explained above, the Block pattern is assigned to the `<div>` elements that have within their content model Inline children elements, like `<span>` and `<a>`. **There are only 3 elements within this group.**

P	*Block*	**If class = ‘svArticle’** Fortunately, the document is very structured, and ALMOST all <p> elements are located inside the main container of the document, and most of the time they don't contain any other element that might conflict with the block pattern, such as lists or other divs. This said, I can safely assign <p> the Block pattern to all paragraph with the class "svArticle section", while I will be assigning the "Inline" one to the others, which are some kind of labels for tables, images or footnotes contained in dd/dt elements
P	*Inline*	**If class != ‘svArticle’** As stated above, these other paragraph elements are used as labels or captions for tables, images or lists. Considering that lists are Block elements, these <p> elements can only be assigned an Inline pattern
SPAN	*Inline*	The way the <span> element is used within this document does not allow for it to be limited just as an Atom pattern, as it often contains some kind of internal markup, like <a> or <em> elements. As it stands, it was consequently assigned the Inline Structural Pattern
EM	*Atom*	This is the only element that is consistently used as a proper Atom Structural Pattern throughout the whole document – it contains only text
SUB	*Inline*	<sub> and <sup> do sometimes contain the <em> element, and not just text. In order to model them correctly when considering their use within the document, they are assigned the Inline Pattern
SUP	*Inline*	Same as above.
A	*Inline*	The same reasoning applied for <span> holds for the <a> element. While most of <a> elements contain only text, and could thus be classified as Atoms, there are some containing Milestones or other having subscript or superscript elements. Without any obvious way to distinguish them, as some share the same class attributes, I chose to assign the most comprehensive pattern
H1, H2, H3	*Block*	The headlines elements, like <h2>, are good examples of Block elements within the target document, for their mixed content models and their predictable positioning, as well as the absence of other Block level elements nested within them
NOSCRIPT	*Popup*	Within the document, is often used as a structural only element container (within Blocks/Inline) with no textual content, but which can contain other elements, like Milestones such as <img> elements.

SCRIPT	*Meta*	A very good example for a content-less Element whose meaning depends on its presence within the document (and the value of its attributes) and not on its position.

We can see that no relevant Marker, Field or HeadedContainer Patterns were assigned to any of the elements. It is also possible to observe that the Structure Pattern assignments are dominated by the Container (and Table), Block, and Inline Patterns, with only few elements (both in General Id and in quantity) receiving any different patternization – this might be because the afore-mentioned Structural Patterns are the most "versatile" ones, and thus are the most useful to represent a very lax content model structure like the one that could be found within (X)HTML. Detailed statistics over this Lens will be presented in section 7.3.

In order to show a quick example on how SLAM was used to apply the authoring choices made for the Document Structure lens, let us consider some snippets about the patternization of some MarkupItems, like the <p> paragraph elements.

First of all, the *annotate* method is called, the applier is readied and miscellaneous options are set:

```
public void annotate() throws […] {
      /* Applier is recovered */
      applier = getApplier();
      /* Options for the debug/logging are set */
      applier.getOptions().setLog_assertOnNode(false);
      applier.getOptions().setLog_findSingleItem(false);
```

This was just an introductory snippet of code. Moving on to the most meaningful parts, let us create a *LensAnnotation* to represent the Block Structural Pattern

```
applier.buildAnnotation("Block", LensesNames.LA_URI,
      "expresses", LensesNames.PATTERN_URI+"Block");
```

With this line I am instructing the *SemLensApplier* associated with this Lens Application to create a new Lens Annotation for the predicate "*expresses*" of the Linguistic Acts Ontology, and having as object the "Block" Structural Pattern. In Turtle, it would be equivalent to this:

```
[INSERT SUBJECT] la:expresses pattern:Block
```

As we have discussed above, I have chosen to assign the Block Pattern to all the headline elements like <h1>, <h2>, <h3> and so on. In SLAM, it could be done by using a very simple combination short-cut method from the applier, with a single line of code.

```
applier.massAnnotate("h2",LensesNames.EMPTY_URI,
     applier.getLastannotation());
```

This line of code uses the "*massAnnotate*" to search the target EARMARKDocument for all the MarkupItems with "h2" as a General Identifier method of the Applier and no special namespace, and to use all of them as subjects for the last annotation used by the applier (in this case, our "Block" Pattern). This is combination method acts as a shortcut for a common operation, the one of asserting an annotation using as subjects all items with a specific General Id (and namespace) – It uses the EARMARK API to recover the set of Items, then it simply iterates on this Set and asserts the specified *LensAnnotation* on the document for each item, using it as subject.

To obtain the same results, I could also have written :

```
applier.massAnnotate("h2", LensesNames.EMPTY_URI,
     applier.getStorage().getAnnotation("Block") );
```

In this case, I am explicitly instructing the Applier to get the annotation I christened as Block from its internal Storage (a *LensApplicationCollection*) and to pass it to the *massAnnotate* shortcut method.

Of course, the usage of the shortcut methods is completely optional. Please observe:

```
applier.assertOnSet (applier.findItemsByGID("h2"),
     applier.getStorage().getAnnotation("Block") );
```

This snippet of code obtains the same results as the other two presented above, but it is even more explicit. With this I instruct the Applier, through the *assertOnSet* method, to assert a specific Lens (the second parameter, which fetches our Block *LensAnnotation*) on all the MarkupItems part of a specified set (the first input parameter, which is obtained by calling one of the finder methods, *findItemsByGID*, that simply returns all elements with the same General Identifier).

What we have seen so far was the simplest part, where all Elements sharing a General Identifier have the same Structural Pattern. However, I have just argued that this is not always the case.

Let us consider the case of the <p> paragraph elements. I have written that only the ones having a "`class`" attribute equal to "`svArticle`" are to be modeled as Blocks in the Document Structure Lens. How can I obtain this result with SLAM? Thanks to the Finder, it is as simple as this:

```
applier.searchWithAttsAndAssert("p", LensesNames.EMPTY_URI,
      "class", "svArticle section", applier.getLastannotation()
);
```

In order to achieve our goal, we use a shortcut method from the Applier that relies on the ability of the Finder to retrieve a set with all the *MarkupItems* sharing the same General ID and having a specified Attribute with a certain value, in this case, the "class" attribute with the "svArticle section" value. The shortcut method "*searchWithAttsAndAssert"* does exactly this, and asserts all the result of the search on the *LensAnnotation* inputted as a final parameter (once again, our "Block" pattern Annotation). As before, I could have avoided the shortcut method and I could explicitly called on the Finder within a call to "*assertOnSet*"

8.2.2 The Application of the Rhetoric Organization Lens

Onwards to the Rhetoric Organization Lens: In accordance to the methodology already detailed in section 5.7, I chose which denotations to assign for specific elements over two subsequent iterations over the document markup, one for the DOCO part of this lens, the other for DEO. However, in order to improve the readability of this Semantic Lens Application implementation and to the betterment of the source code overall readability, I therefore re-organized the SLAM instructions to be sequential to the document structure.

As a consequence of this implementative refactoring, the final "*annotate*" method for the Rhetoric Organization Lens Application sequentially builds and asserts Annotations, to either individual or sets of elements, by their order of appearance within the organization of the target document itself.

That said, within section 5.7 I had already discussed some of the expected problems of applying the DOCO part of this Lens unto a document, many of which were confirmed during the actual application activity. Most of these are a consequence of the very strict requirements, by the means of mandatory Pattern assignments over containers and descendants that are requested by some DOCO classes. Considering that the target document, as already mentioned, has a very shallow structural depth (it can be imagined as a very wide and very short tree), many of the mandated requirements for structured container boxes could not be met.

This is a small list of the consequences related to these issues:

- No Sections or TextBox could be identified and assigned. As a consequence, no section titles or subtitles could be assigned either, even when the role of said components (such as the case for the <h2> and <h3> elements was extremely explicit.
- No valid global Front Matter could be defined, although I was able to assign a Back Matter correctly. In the end, I opted to assign 4 different Front Matters to different components located at the beginning of the text.
- Among the many instances of mathematical formulas used within the document, only in 1 case could the Formula and Formula Box denotations be used
- The same problem goes for inline embedded images – It is impossible to characterize them as such, and for some labels as well.

- Footnotes require to be within a Popup Pattern. As the Footnotes within this document were defined at the end of the Back Matter, through the use of definition list elements, no Footnotes could be validly defined in the target document. The same could be said for the Footnotes inline references within the text (which are made through direct text within subscript or superscript elements)

Another relevant choice that I had to make during the application process involved deciding on how to characterize inter-document links, whether they consisted in inline bibliographic citation references such as "[##]" (which are extremely important and meaningful, especially in order to build the Citation Network Lens) or just references from one section to another (e.g. [Table 2] or "see Section 3.1"), or reference to Footnotes located at the back of the document (see above). Structurally, they were all very similar within the target document, consisting in `<a>` link elements embedded within a `<span>`. It was possible to distinguish one type from the other with some analysis on their id and class attribute contents, analysis made possible by the methods featured in the SLAM Finder, but the issue didn't lie in the impossibility to differentiate them.

The main problem here was choosing which of DOCO denotations should be assigned to these three inter-document links. I have just discussed why it was impossible to assign the Footnote denotation to the Footnotes, due to Structural Pattern Constraints.

Unfortunately, DOCO also lacks a specific class to express the fact that an element absolves the function of an inline citation reference. As it is, these kind of components carry a little more importance than other simple inter-document references, especially looking forward to inter-document application potential. This absence is to be added to the already observed lack of characterizations for a couple of common "building blocks" usually included within the front matter, such as keywords or dates/publication information.

As it is, I renounced to use any DOCO class and I was thus forced to assign the generic DEO "Reference" denotation to both types of inter document references – which is a pity, as having a specialized class could have reduced the amount of document specific implementations in TAL, would have rendered the model more expressive and the emergence of meaning more explicit.

Considering all that I have reported, my personal impression after the application task was completed is that the current DOCO requirements make for a system that is a little awkward in its concrete application over documents. While intended and designed in order to reduce ambiguities in their application, I believe that the end result is an overshoot and that they end up in

creating a web of requirements that is a little too strict for many of them to be actively used on a document, thus ending in a loss of potentially useful denotations.
Still, I had been able to annotate most of the document components with meaningful information.

As for the DEO level, it confirms my impression of being more streamlined and less problematic in its application, and there was little to report beside what already discussed in section 4.12. The main concern here originates from the fact that some (not all) of the entities are described within the DEO documentation a little vaguely, and it occasionally caused some uncertainty on which was characterization would best catch the Author's intention in organizing the scientific discourse, when considering the paragraph level.
During the application activity, I have strongly felt the absence of a characterization for discourse elements expressing a digression, an aside or a demarcation from what has been the main subject of the discourse so far. These pieces, usually the size of paragraph, are usually either a way to point out some specific aspect of something (a deo:Clarification ?) which might be relevant but not completely related or a way to set out a disjuncture on a certain aspect distinguishing itself from the main flow of the rhetoric (a deo:Differentiation ?). Anyways, It is my belief that the encoding of some of these properties in future versions of DEO could be a relevant addition. These (and others) suggestions will be again analyzed in section 7.4.

In order to show some sample code for this part as well, let us consider some relevant passages, like the assignment of the "doco:Paragraph" entity to all the <p> elements acting as paragraphs.

```
applier.buildAnnotation("Paragraph", LensesNames.LA_URI,
      "expresses", LensesNames.DOCO_URI+"Paragraph");
applier.searchWithAttsAndAssert("p", LensesNames.EMPTY_URI, "class",
      "svArticle section",
      applier.getStorage().getAnnotation("Paragraph"));
```

As we can see, this is very similar to the procedures we have exemplified in the previous section, which is to be expected, as I have already expounded on the fact the workflow to create a Lens Application with SLAM is quite regular (see section 6.2). Here we see that first the appropriate annotation is built, and then the applier is ordered to use a set of all the appropriate <p> elements as a subject for asserting that annotation. From this snippet we can see how easily the annotations can be re-used and how regular the structure of the code is.

In order to give the proper rhetorical denotation for the DEO level on each paragraph, I had of course to operate single element by single element. In order to do this, the code used is, for instance:

```
      applier.buildAnnotation("DEO Model", LensesNames.LA_URI,
            "expresses", LensesNames.DEO_URI+"Model");
      applier.searchAndAssert(
            "http://www.essepuntato.it/2010/04/SWWEx#e_p_5",
            applier.getLastannotation() );
```

In this case the Applier builds a "`la:expresses deo:Model`" Annotation, which is then immediately recalled by the applier as the last input parameter for the "*searchAndAssert*" combination method, which acts as a shortcut method to fetch a single MarkupItem and use it as the subject for asserting the Annotation.
Of course, once again it could have been written in a more explicit way:

```
applier.assertOnNode(
      applier.findSingleMarkupItem(
            "http://www.essepuntato.it/2010/04/SWWEx#e_p_5"),
      applier.getStorage().getAnnotation("DEO Model")
);
```

The above snippet of code obtains the same result as the previous one, but without using the a shortcut method. It calls on the *"assertOnNode"* method of the applier, which instructs the Applier to assert the annotation passed as the second parameter on the target document, by using the item inputted in the first parameter as a subject. "*findSingleMarkupItem*" is just a method of the finder that acts as simple syntactic sugar for the EARMARK API.

Finally, let's take a look at something a little more complex and refined, in order to fully show the potentialities of SLAM and of its Finder and utilities.
In the first part, we want to assert as DOCO Figure Boxes all the appropriate <div> elements within the document. They don't have a recognizable class, but I know that they all have Id attributes whose content starts as "figure_fig<something>". So I use the methods that allow me to search Elements by General ID and Attribute content wildcards.:

```
applier.buildAnnotation("Figure Box", LensesNames.LA_URI, "expresses",
            LensesNames.DOCO_URI+"FigureBox");
applier.searchWithWildAttsAndAssert("div", LensesNames.EMPTY_URI, "id",
            "figure_fig*", applier.getLastannotation());
```

In this example I used the shortcut method "*searchWithWildAttsAndAssert*" which works in a similar way to the other combination methods already shown within this section. Of course, there is also an explicit finder method "*findWithWildAttsAndAssert*" if the user prefers to avoid shortcuts.

We can also perform tasks that are even more refined. Let us consider the case of the "doco:Figure" annotation. There were 2 `<img>` XHTML elements with the same class "figure large" for each actual image within the target document, one inside a `<noscript>`, the other outside. Since there was only a grand total of 6 figure in the document, I could have annotated them manually id by id, but I instead opted to show the adaptability of SLAM as an extension of the already excellent EARMARK API. I noticed that the alt attribute had a different content for each of those images, and called on the utility MarkupItemSetReducer Class to select just one subset of the 6 images.

```
applier.buildAnnotation("Figure", LensesNames.LA_URI, "expresses",
            LensesNames.DOCO_URI+"Figure");
MarkupItemSetReducer reducer_figlarge = new MarkupItemSetReducer(
      applier.findItemsWithAtts("img", LensesNames.EMPTY_URI, "class",
            "figure large"));
applier.assertOnSet(
      reducer_figlarge.keep(applier.findItemsWithWildAtts("img",
      LensesNames.EMPTY_URI, "alt", "Full-size image (*K)")),
      applier.getLastannotation()
);
```

First of all, I created the appropriate Lens Annotation for the Figure. Then I instance a new object of the *MarkupItemSetReducer* Class, and its base set (the one to be reduced) is defined as the result of the "*findItemsWithAtts*" call, which returns a Set of Markup Items with a shared General Id (*img* in this case), and having an Attribute (*class*) with a specified content (*"figure large"*).

Finally, within the assertOnSet call, the Reducer is instructed, through the "*keep*" method, to discard all items within its base Set with the exceptions of those being returned by the *findItemsWithWildAtts* call.

8.2.3 The Application of the Citation Network Lens

Keeping up with the bottom-up approach, I now progress to the next level of the stack, in order to discuss my implementation for the application of the Citation Network Lens over the target paper.

The concrete application of this lens proceeded quite straightforwardly as I aimed to adhere to the methodology illustrated in section 5.8, and the idea of choosing an information-rich solution by placing the Citation Network markup within the context of the inline citation occurrences themselves allowed for quite a straightforward development process.

However, two quite relevant compromises, differing from both the theoretical methodology and the model, were made in order to reach the final result. These involve both the objects and the chosen subjects of the CiTO ontology object properties used to characterize the citations.

The first and more obvious one is about the objects of the properties. In theory, the Citation Network Lens model would have required me to refer to the URI (or IRI) of the scientific document being cited. For example, I could have used their DOI, if available.

However, while referring to other documents URI makes sense within the idea of integrating a system within the LOD, I also had to consider the scope and the purpose of this activity, and the fact that it also relates to prototype generation over a single document. As a consequence of this, and in order to enable a simpler way to create inter-document interactions, I decided to use as objects for my citation network annotations the URI identifiers for the Bibliographic Reference elements in the back matter. This would allow me to operate more simply within the annotated document, and to follow thorough the citation network links to associate citation properties to bibliographic reference information in a fairly simple way.

The second one is less of a theoretical compromise but more of an implementation-related choice. In order to reduce complexity, code clutter and for the sake of simplicity and clarity, I opted for the use of the very inline reference elements as subjects for the citation network CiTO assertions. Of course, this is not meant to intend that it is just that element doing the citation act – which is still semantically done by the document. It is more simply just a mean to the end of avoiding the need to use a double link structure (whole document & occurrences) which could have been constructed by combining the whole SPAR ontologies.

Obviously, the choice of adopting these implementation compromises does not negate the validity of either the model or of the methodology so far described – the same result could be obtained by combining what have been done so fare with BiBO, C40, and further integration with LOD, but it would have been far outside the scope of this demonstration.

That clarified, the only other issue with the application of the Citation Network lens is the unfortunate state of the ontology documentation for CiTO properties, which is extremely concise. As it's too often the case, these property descriptions are very short in their wording (e.g., for the "*discusses*" property, the documentation states only that "*The citing entity documents information about the cited entity*" which is not that much of an useful clarification), and they consequently offer the user a poor guidance in distinguishing each other and in understanding which property would be better suited in a specific instance. The current situation leaves a lot of leeway to personal interpretation, and thus results in a lot of potential for ambiguities.

I would also like to observe that the, as far as my opinion goes, CiTO seems to lack a specific property to deal with documents which are cited not in a negative way or in order to disprove them, but as related for their complementary or different approach on a subject. The addition of a property able to address this meaning, something like "`cito:unlike`" or "`cito:differentlyFrom`". In a sense, this is somewhat of a continuation of the issue addressed with my suggested extension of DEO (see the previous section).

Moving on to some relevant examples. Let's see, for instance, the code relative to the denotation of citation 13 within the target document.

```
            // Citation [13]
      applier.buildEMAnnotation("Shares Author With [13]",
            LensesNames.CITO_URI, "sharesAuthorWith",
            "http://www.essepuntato.it/2010/04/SWWEx#e_li_64");
      applier.assertOnSetEM(applier.findItemsWithAtts("a",
            LensesNames.EMPTY_URI, "href", "#bib13"),
            applier.getLastannotation());

      applier.buildEMAnnotation("Extends [13]", LensesNames.CITO_URI,
             "extends",
      "http://www.essepuntato.it/2010/04/SWWEx#e_li_64");
      applier.assertOnSetEM(applier.findItemsWithAtts("a",
            LensesNames.EMPTY_URI, "href", "#bib13"),
            applier.getLastannotation());

      applier.buildEMAnnotation("Uses Data From [13]",
LensesNames.CITO_URI,
            "usesDataFrom",
      "http://www.essepuntato.it/2010/04/SWWEx#e_li_64");
      applier.assertOnSetEM(applier.findItemsWithAtts("a",
            LensesNames.EMPTY_URI, "href", "#bib13"),
            applier.getLastannotation());

        applier.buildEMAnnotation("Cites As Data Source [13]",
            LensesNames.CITO_URI, "citesAsDataSource",
            "http://www.essepuntato.it/2010/04/SWWEx#e_li_64");
        applier.assertOnSetEM(applier.findItemsWithAtts("a",
            LensesNames.EMPTY_URI, "href", "#bib13"),
            applier.getLastannotation());
```

The *buildEMAnnotation* and *assertOnSetEM* methods the straightforward equivalent of the methods we have already examined in the previous pages, but they are the specialized counterpart for Annotations having as object an Earmark Item instead of a generic RDF Node. In this snippets, first all the appropriate Annotations are created, and then they are applied on all the instances where citation #13 appears (which occurs just once, so I could have also annotated it manually on the single <a> item through a *searchAndAssert*, but when the instructions are written this way they are both easier to read and to write).

Some interesting findings on the correlation between Rhetoric Organization, Citation Network and Argumentation Lens will be expounded in section 8.4

8.2.4 The Application of the Argumentation Lens

Finally, we reach the last facet of the target document for which I am applying a Semantic Lens – the Argumentation one. The methodology selected to apply the Argumentation Lens has already been introduced in section 8.9, but just as a brief summary, the basic idea is to structure each single argumentation within the main text, starting first by identifying the claims, and then by denoting each component within the argumentation, all the while observing Toulmin's Argument Model and using the AMO ontology which represents it. Of course, components and whole claims can be shared and re-used between different argumentations.

I have already explained that most components will be identified within the main textual content of the article, although some other elements (tables, images, lists) might participate in the argumentation structure of the paper, and that, on the other hand, there might be some part of the main text body which are not relevant to any of the major argumentations modeled.

In order to concretely implement this Argumentation Lens Application, I proceeded in accordance with the aforementioned methodology, and I modeled each argumentation starting by its claim, and specifying within each argumentation model its relationship with other components of the text, be them text parts, other argumentations or actual elements. In order to store these models I decided to use an ad-hoc Earmark Element Node created for each one of them and assign to them the type "*amo:Argument*" – these could be thought as document-level markup elements grouping together meta-information, like those corresponding to the Meta pattern, or those html elements within the `<head>` part of an HTML documents. Anyways, these newly created Earmark Elements do not alter the structural markup of the document.

In theory, this passage could have been avoided by using simple RDF Nodes within the Jena Model of the Document, instead of using full Earmark Elements. However, it was an implementation related choice, as there was apparently some minor incompatibility between the Jena Model API and the EARMARK Document API that caused some of the changes to be lost when switching from a Model to a Document and vice-versa. This is probably going to be corrected, but in order to reach a workable implementation, I had decided to adapt and accept this minor compromise for implementation reasons.

Arguably, the longest, hardest and more-error prone part of authoring this Lens Application was the need to manually associate each relevant text part to an appropriate new range, if there were none already tied to it. The large majority of the argumentation components within the text did not correspond perfectly to any textual content of any existing markup element, and as such the aforementioned range was non-existent and had to be created. In order to identify the correct indexes, I had to rely on index functions of an external text editor after importing the Docuverse into it, and still I had to search manually for the start and the beginning of each of them within the character stream, note down the beginning and the position, and iterate the process for each fragment identified within the text. This was probably the biggest usability issue I have faced.

While the EARMARK range system worked perfectly and with remarkable flexibility, and was very intuitive and easy to deal with overlapping markup, it is also very much evident, in my opinion, that if some actual large scale effort to enrich documents with the Argumentation Lens is to be done, some tools are to be developed in assisting in accomplishing this task within the application activity.

Dealing with other components or already defined ranges, on the other hand, was relatively very easy, as I just referred to their identifiers within the EARMARK document.

To see SLAM in action for the annotation of the Argumentation Lens, let us consider the same example used in section 3.5, a snippet from [Mik07], precisely, the first argument within the 4th section of the target paper – the argumentation number 42. The pieces of the text are colored and identified according to their roles.

[Qualifier] In absence of a golden standard, evaluating the results of ontology learning or ontology mapping is a difficult task:[/Qualifier] [Claim] inevitably, it requires consulting the community or communities whose conceptualizations are being learned or mapped.[/Claim] [Evidence] In order to evaluate our results, we have thus approached in email 61 researchers active in the Semantic Web domain, [/Evidence] [Qualifier] most of whom are members of the ISWC community and many of them are in the graph-theoretical core of the community.[7] [/Qualifier] [Evidence] The single question we asked was *In terms of the associations between the concepts, which ontology of Semantic Web related concepts do you consider more accurate?* [/Evidence] [Rebuttal] Lacking a yardstick, there is no principled correct answer to this question that we expected to receive. [/Rebuttal] [Warrant] Instead, we were interested to find out if there is a majority opinion emerging as an answer and if yes,

which of the two ontologies (produced by the two different methods) would that majority accept as more accurate. [/Warrant]

In order to model this argumentation, which is quite straightforward, in the sense that it does not re-use components from other argumentations or non textual-components, there are several steps to be taken. First of all, I had to instance the new element representing the argumentation, and to create all the new ranges that identify the various textual sub-components of the argumentation (a range for each one of them).

```
applier.newEmElement("my_arg42", "", LensesNames.SWEX_URI2,
            Collection.Type.List);

applier.newPointerRange(LensesNames.SWEX_URI2 + "r_arg42_q01-p_54",
            35321, 35438,
      "http://www.essepuntato.it/2010/04/SWWEx#d_text");
applier.newPointerRange(LensesNames.SWEX_URI2 + "r_arg42_c01-p_54",
            35439, 35556,
      "http://www.essepuntato.it/2010/04/SWWEx#d_text");
applier.newPointerRange(LensesNames.SWEX_URI2 + "r_arg42_d01-p_54",
            35557, 35673,
      "http://www.essepuntato.it/2010/04/SWWEx#d_text");
applier.newPointerRange(LensesNames.SWEX_URI2 + "r_arg42_q02-p_54",
            35674, 35791,
      "http://www.essepuntato.it/2010/04/SWWEx#d_text");
applier.newPointerRange(LensesNames.SWEX_URI2 + "r_arg42_d02-p_54",
            35792, 35954,
      "http://www.essepuntato.it/2010/04/SWWEx#d_text");
applier.newPointerRange(LensesNames.SWEX_URI2 + "r_arg42_r01-p_54",
            35955, 36059,
      "http://www.essepuntato.it/2010/04/SWWEx#d_text");
applier.newPointerRange(LensesNames.SWEX_URI2 + "r_arg42_w01-p_54",
            36060, 36277,
      "http://www.essepuntato.it/2010/04/SWWEx#d_text");
applier.newPointerRange(LensesNames.SWEX_URI2 + "r_arg42_d03-p_55",
            36278, 36873,
      "http://www.essepuntato.it/2010/04/SWWEx#d_text");
```

Both the “*newEMElement*” and “*newPointerRange*” methods of the applier are just convenient wrappers for calls directly over the EARMARK API, and they allow me to instance the new objects. Now that I have all the pieces of the argumentation at my disposal, it is time to model its basic structure and to assign to each piece its intended role. In order to do so, I shall proceed by creating an Annotation that defines the role of a component (e.g. “has Qualifier” for the qualifier #1), and assert it using the newly created core Argumentation element as Subject.

```
applier.buildAnnotation("Arg42 AMO:Qualifier 01",
      LensesNames.AMO_URI, "hasQualifier",
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_q01-p_54");
applier.searchAndAssert("http://www.essepuntato.it/2010/04/SWWEx#my_arg
42",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 AMO:Claim 01", LensesNames.AMO_URI,
      "hasClaim", "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_c01-
p_54");
applier.searchAndAssert("http://www.essepuntato.it/2010/04/SWWEx#my_arg
42",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 AMO:Evidence 01",
      LensesNames.AMO_URI, "hasEvidence",
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d01-p_54");
applier.searchAndAssert("http://www.essepuntato.it/2010/04/SWWEx#my_arg
42",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 AMO:Qualifier 02",
      LensesNames.AMO_URI, "hasQualifier",
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_q02-p_54");
applier.searchAndAssert("http://www.essepuntato.it/2010/04/SWWEx#my_arg
42",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 AMO:Evidence 02",
      LensesNames.AMO_URI, "hasEvidence",
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d02-p_54");
applier.searchAndAssert("http://www.essepuntato.it/2010/04/SWWEx#my_arg
42",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 AMO:Rebuttal 01",
      LensesNames.AMO_URI, "hasRebuttal",
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_r01-p_54");
applier.searchAndAssert("http://www.essepuntato.it/2010/04/SWWEx#my_arg
42",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 AMO:Warrant 01",
      LensesNames.AMO_URI, "hasWarrant",
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_w01-p_54");
applier.searchAndAssert("http://www.essepuntato.it/2010/04/SWWEx#my_arg
42",
      applier.getLastannotation());


applier.buildAnnotation("Arg42 AMO:Evidence 03",
      LensesNames.AMO_URI, "hasEvidence",
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d03-p_55");
```

```
applier.searchAndAssert("http://www.essepuntato.it/2010/04/SWWEx#my_arg
42",
      applier.getLastannotation());
```

From this example it is possible to see that the workflow should be quite familiar to us, and is akin to the one characterizing the other applications. First of all, an Annotation is built through the Applier, by having the appropriate property (e.g.: "hasClaim" for the claim) associated to the right object (e.g.: the PointerRange identifying the text of the claim itself). Then it is asserted in the document with the Argument element wrapper as a subject. In doing this for all the argumentation components, we obtain exactly the model we desired in the first place.

Finally, If we wanted to, we could also obtain an even more explicit model of the Argumentation, by stating not just the roles within the Argumentation, but assigning the properties that denote the interactions between all the components, in accordance to the model presented in sections 3.5 and 4.9. For instance, we might want to state that the Claim "is valid unless" the Rebuttal.

```
applier.buildAnnotation("Arg42 Q Forces C", LensesNames.AMO_URI,
      "forces", "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_c01-
p_54");
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_q01-p_54",
      applier.getLastannotation());
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_q02-p_54",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 W Leads To C", LensesNames.AMO_URI,
      "leadsTo", "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_c01-
p_54");
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_w01-p_54",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 E Proves C", LensesNames.AMO_URI,
      "proves", "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_c01-
p_54");
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d01-p_54",
      applier.getLastannotation());
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d02-p_54",
      applier.getLastannotation());


applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d03-p_55",
      applier.getLastannotation());
applier.buildAnnotation("Arg42 E Supports W", LensesNames.AMO_URI,
      "supports", "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_w01-
p_54");
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d01-p_54",
      applier.getLastannotation());
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d02-p_54",
      applier.getLastannotation());
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_d03-p_55",
      applier.getLastannotation());

applier.buildAnnotation("Arg42 C Valid Unless R", LensesNames.AMO_URI,
      "isValidUnless",
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_r01-p_54");
applier.searchAndAssert(
      "http://www.essepuntato.it/2010/04/SWWEx#r_arg42_c01-p_54",
      applier.getLastannotation());
```

While doing this extra round of annotations for each argument is well possible, it might not be necessary, as in practice it does not always provide additional details, as these extra properties might also be inferred by applying an ontology reasoner over the enhanced document.

To conclude, in order to complete the Application, all Argumentation elements (whose instantiation has been moved to the start of the annotate method, to avoid errors) are denoted as being of the appropriate `amo:Argument` type, as discussed above. In order to do so:

```
applier.buildAnnotation("RDF Type: Argument",
      "http://www.w3.org/1999/02/22-rdf-syntax-ns#", "type",
      "http://purl.org/spar/amo/Argument");
applier.assertOnSet(applier.findItemsWithARangeOfIds(
      "http://www.essepuntato.it/2010/04/SWWEx#my_arg", "00", 01, 61,
""),
      applier.getLastannotation());
```

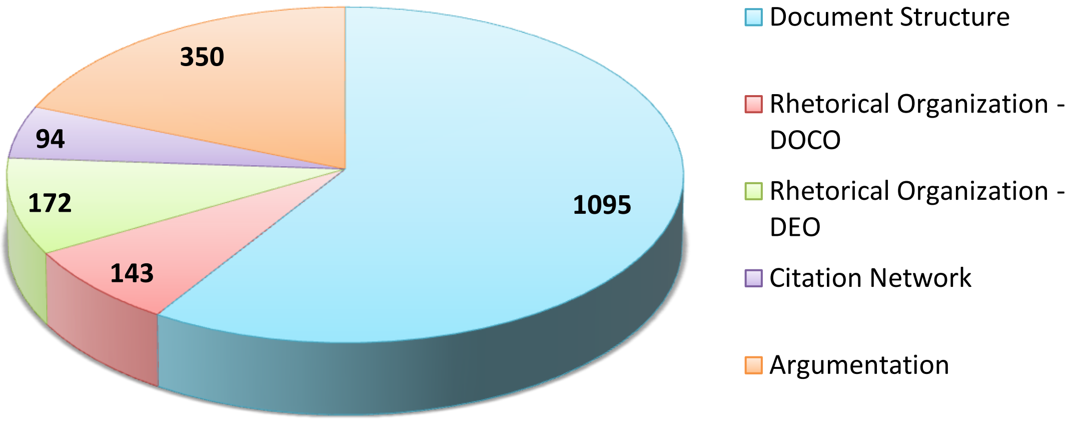
I simply create the appropriate annotation for the "`rdf:type amo:Argument`" and the apply it on all arguments – in this example I wanted to show another one of the features of SLAM's Finder, *findItemsWithARangeOfIds* – its ability to search for items by predictable ranges and patterns of Ids.

8.3 Overall statistics

In this subsection I'll quickly present some numerical statistics that SLAM recorded on the results of my activity for the application of Semantic Lenses over [Mik07]. In total, **1856** Annotations created and asserted over the target document in order to complete the application of all four Semantic Lens (*Document Structure, Rhetoric Organization, Citation Network, and Argumentation*).
The annotations were distributed this way:

Semantic Lens	Annotations
Document Structure	1095
Rhetorical Organization - DOCO	143
Rhetorical Organization - DEO	172
Citation Network	94
Argumentation	350

Graph 1 - Overall Distribution of Lenses Annotations



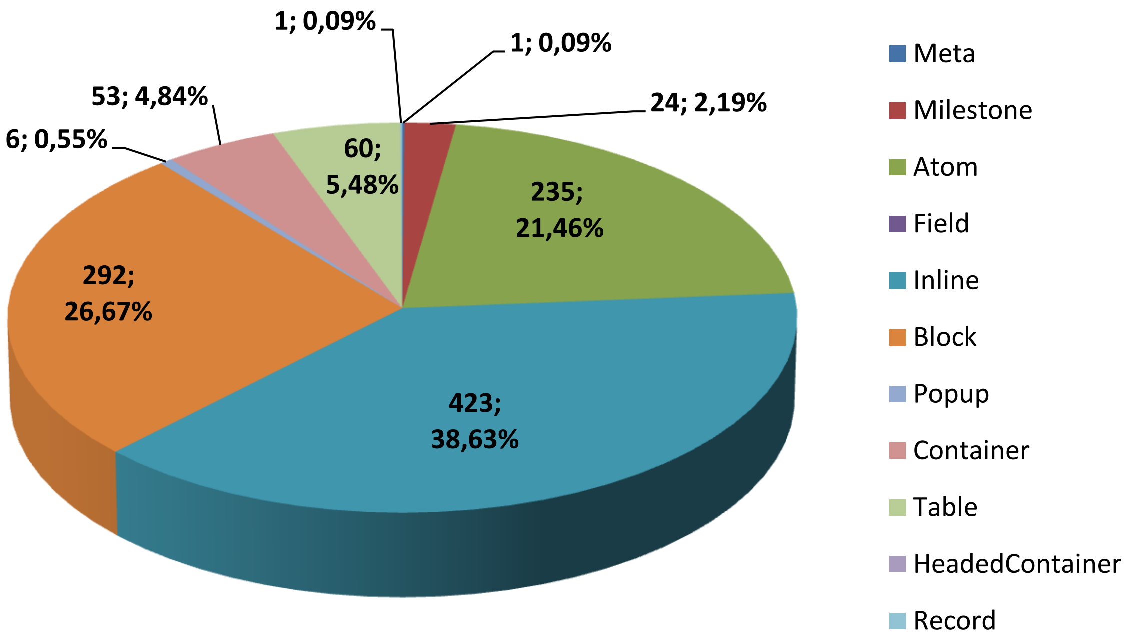
As we can see from Graph 1, the great numerical majority of the annotations were within the Document Structure Lens. This was to be expected, as almost all elements within the document markup were associated with a Pattern and consequently received at least one annotation, which was certainly not the case for the other lenses. For example, all `<em>` and `<sub>` elements received a pattern association in the form of a document structure lens assertion, but none was characterized in any other way. Most of the <span> elements did not receive any Rhetorical denotation as well.

Let us examine the distribution of the annotations on a lens by lens basis. Coherently to the **bottom-up** approach which I held within this whole dissertation, I will start with the **Document Structure Lens.**

First, let me present the overall **numerical incidence of each Structural Pattern** within the final annotated document, here shown in Graph 2.

Meta	1
Milestone	24
Atom	235
Field	0
Inline	423
Block	292
Popup	6
Container	53
Table	60
HeadedContainer	0
Record	1

Graph 2 - Distribution of Structural Patterns

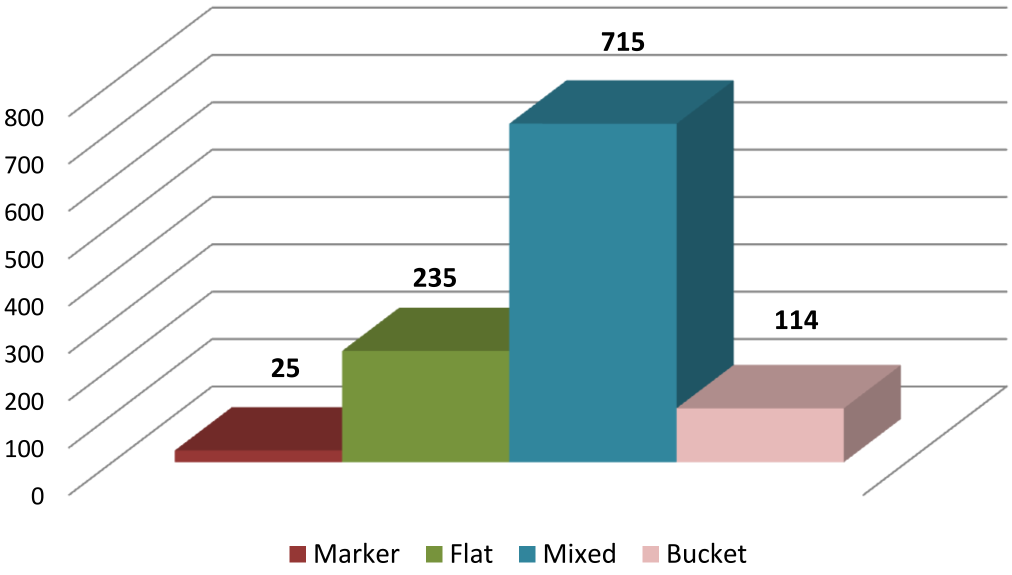


As already observed, the Inline, Block and Atom Patterns dominate the assignments, with the Container pattern and its sub-pattern Table the only other true relevant denotations. This result is pretty consistent with the HTML markup language (which favors and encourages mixed content models) and with the shallow-depth, large-width structure of the target document. As already observed, the HTML version of [Mik07] is well structured, but has few containers (and no section containers whatsoever), and mostly consists in a large sequence of paragraph elements, separated by few titles (which are blocks). See section 8.2.1 for the chosen assignments of Pattern, as grouped by element General Identifier.

What we have just observed becomes even more evident if we group the patterns by their Ghost class of reference:

Marker	25
Flat	235
Mixed	715
Bucket	114

Graph 3 - Distribution of Structural Patterns By Ghost Classes



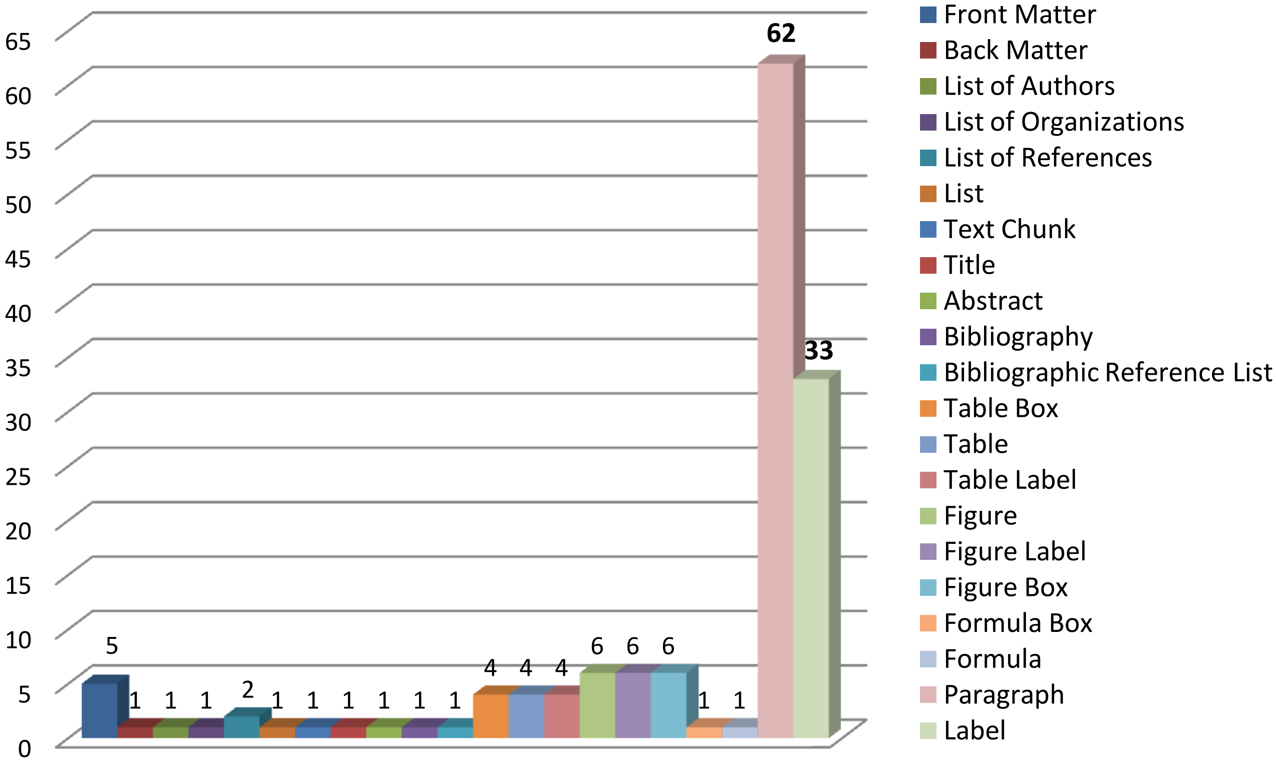
As we can see from Graph 3, the privileged content model is well highlighted if we group patterns by their containing Ghost Class, with the Mixed Pattern (*Inline* & *Block*) accounting for over than 65% of the total number of Pattern associations. Once again, this is consistent and somewhat expected with both the HTML in general and this document in particular, and it reflects its internal composition in terms of elements. For example: There are far more `<p>` elements than `<div>` ones.

Moving on to the **Rhetoric Organization Lens**, we should distinguish from the Annotations related to DOCO and to the Document Components role and the Annotations related to the Discourse Elements, which are done through DEO. **I will first discuss the grand totals for the DOCO part of this lens:**

Front Matter	5	**Table Box**	4
Back Matter	1	**Table**	4
List of Authors	1	**Table Label**	4
List of Organizations	1	**Figure**	6
List of References	2	**Figure Label**	6
List	1	**Figure Box**	6
Text Chunk	1	**Formula Box**	1
Title	1	**Formula**	1
Abstract	1		
Bibliography	1	***Paragraph***	62
Bibliographic Reference List	1	***Label***	33

As we can see from the table, the most widely used characterizations are “Paragraph”, to denote the paragraphs within the main text body, and “Label”, which is used within definition lists and within elements of the bibliography. It is interesting to note how consistent the most structured assertions are when grouped by “role”, such as by Table or by Figure. This is the result of the highly structured requirements imposed by DOCO in order to make such denotations.

Graph 4 - Distribution of Rhetorical Assertions - DOCO

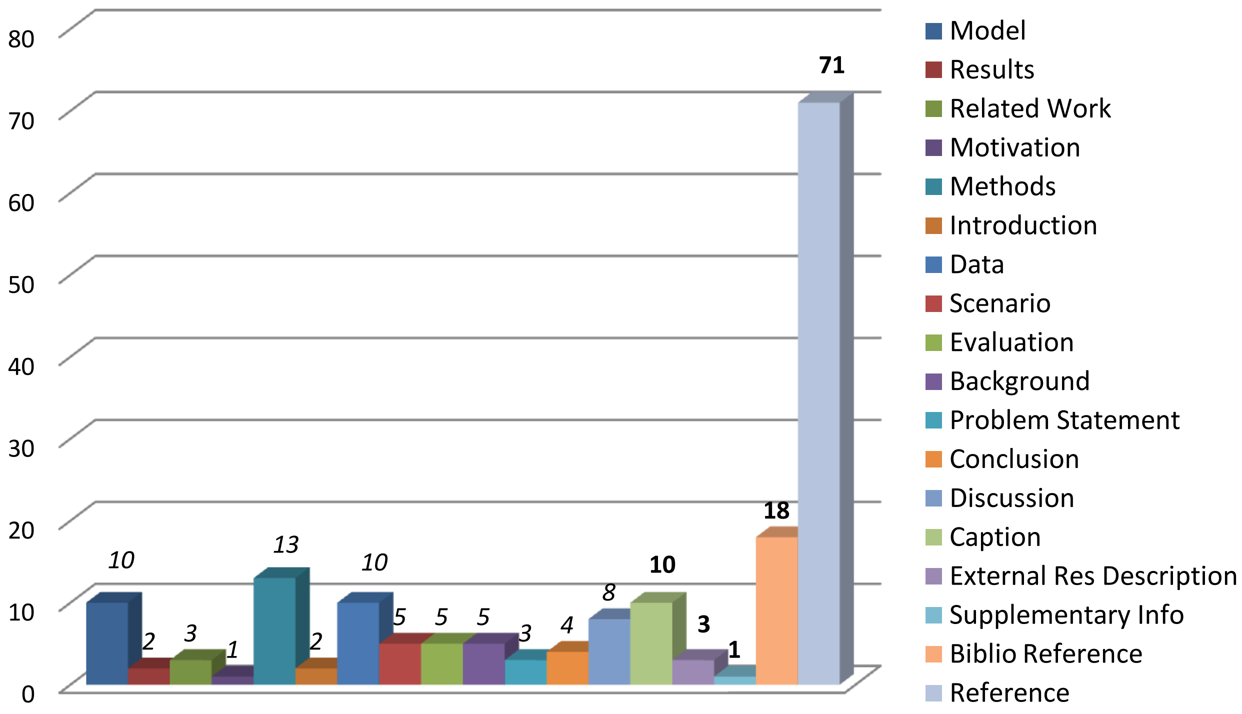


Graph 4 should be able to impress a little more explicitly the overwhelming numerical majority of the two more widespread DOCO annotation. It is also possible to observe that the List characterization is also quite used – if we sum up the occurrences for all its classes and subclasses, we obtain a total of 6 occurrences, comparable with Tables and Figures in the article. I have already discussed the difficulties in applying the Formula denotation correctly, due to the restrictions of DOCO itself, so there is only 1 Formula explicitly asserted. The disproportion between the Front and Back Matters is also due to the structure of the target document.

Let us move onwards in order to consider the DEO level. On the left hand side of the table are the rhetorical denotations for all the paragraphs of the main text, on the right hand side all the others:

Introduction	2	**Caption**	10
Model	10	**External Res Description**	3
Results	2	**Supplementary Info**	1
Related Work	3	**Biblio Reference**	18
Motivation	1	**Reference**	71
Methods	13		
Data	10		
Scenario	5		
Evaluation	5		
Background	5		
Problem Statement	3		
Conclusion	4		
Discussion	8		

Graph 5 - Rhetorical Organization Lens - Overall Distribution of DEO Assertions

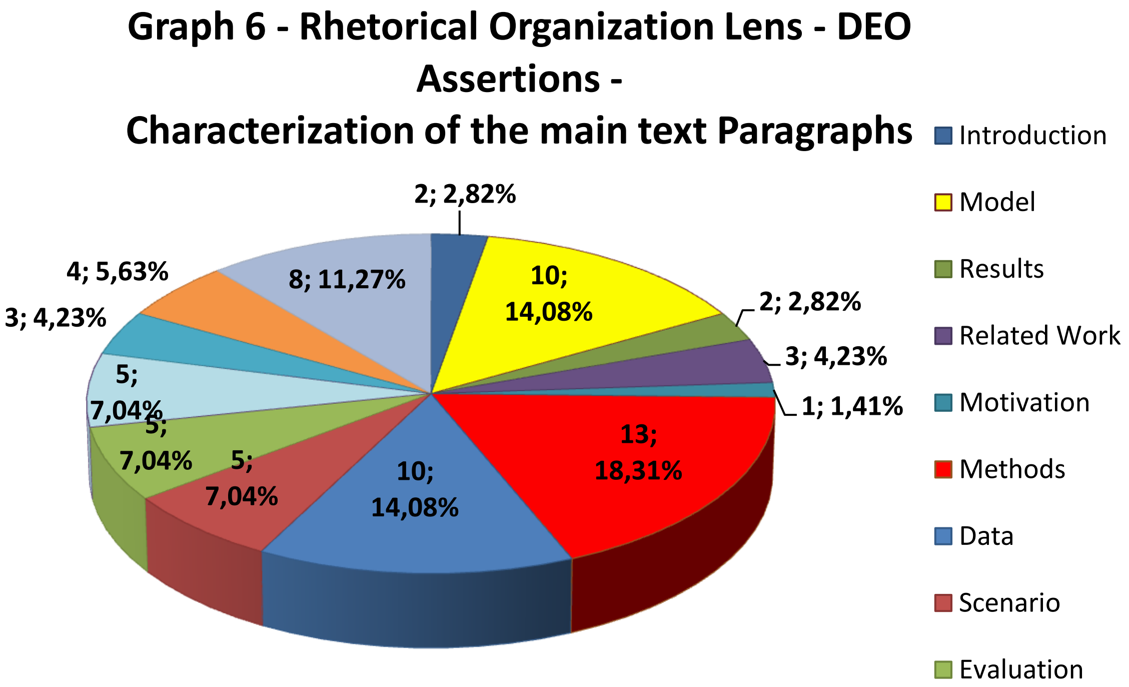


In Graph 5 I have translated the overall numerical distribution of all DEO related assertions over the target document, (by considering the whole table above as a data source). We can see that the most used assertion is

"Reference", but this is not unexpected, and should not be extremely significant, as this fact is the consequence of my decision to use it to annotate almost all relevant <a> links, including inline citations, footnotes, and inter-document references. It can also be observed that we have exactly 10 captions, which are exactly the 6+4 Figure and Table labels of the DOCO part of this lens. Only 18 out of 32 possible DEO classes were used within my application.

The next graph, Graph 6, is surely more interesting, as it's the distribution of the **DEO characterizations** used for to express the rhetorical role **of the main text paragraphs.** It highlights some pretty interesting results.

We can see that the most used denotations are paragraphs whose subject is classified as Methods, Model, Data and Discussion, closely followed by an equal share of Background, Evaluation and Scenario descriptions. The first four correspond quite well to the usual and most expected rhetorical building blocks of the scientific discourse – a model is offered, a methodology is explained, data is gathered, and results are discussed. When we add the other 3 most used denotations, as well as the Conclusions, the Problem Statements, and the Related Works we have now all most widely used pieces that are expected within a scientific article, especially one published within a journal – whose editors and publisher usually require to adhere to a specific organization of discourse.


Graph 6 - Rhetorical Organization Lens - DEO Assertions - Characterization of the main text Paragraphs

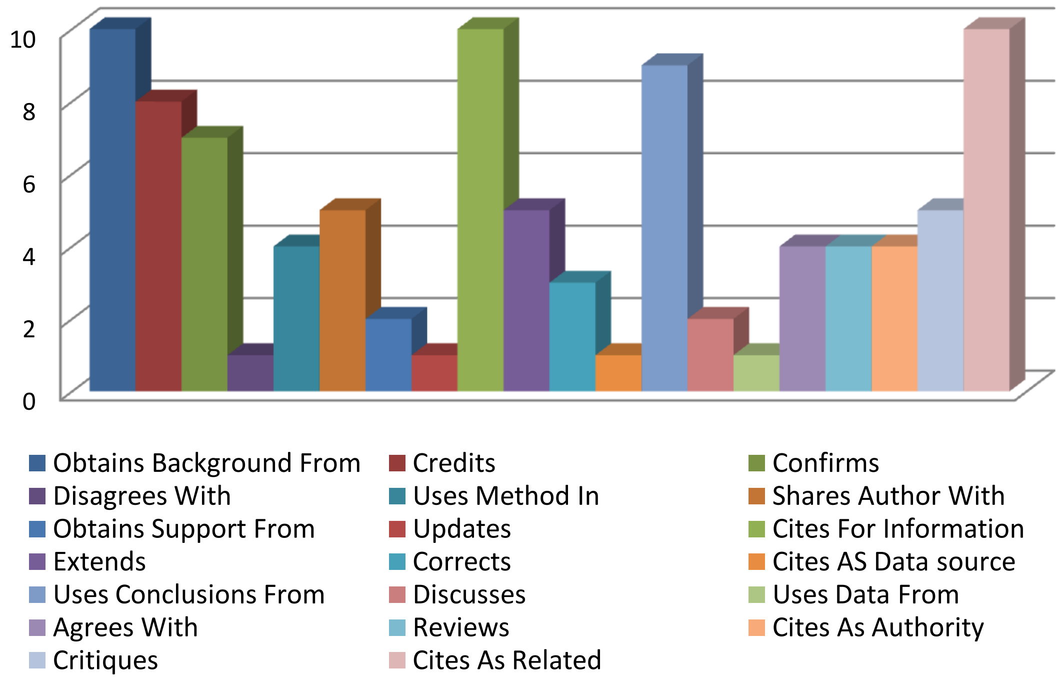
It is also possible to observe that the proportions between these elements might also be worthy of note. They could be roughly classified within 3 tiers: The four most used denotations all with more than 10%, six others between 4 and 10% of the occurrences, and all others with less than 3% of the occurrences. Of course, a single document is too small a sample to make any concrete observation, but further investigation could be in order, especially on a set of articles taken from the same journal, and might open up interesting research paths.

Then, If we move on to examine the **Citation Network Lens and the CiTO** properties used, I can observe that I used just 20 out of the 33 possible properties defined in the Citation Ontology:

Obtains Background From	10	**Corrects**	3
Credits	8	**Cites AS Data source**	1
Confirms	7	**Uses Conclusions From**	9
Disagrees With	1	**Discusses**	2
Uses Method In	4	**Uses Data From**	1
Shares Author With	5	**Agrees With**	4
Obtains Support From	2	**Reviews**	4
Updates	1	**Cites As Authority**	4
Cites For Information	10	**Critiques**	5
Extends	5	**Cites As Related**	10

As we can see from Graph 7, it's once again possible to note that some properties end up being used more often than others, and we could once again try to group up these properties within 3 larger groups, based on their occurrences, with Obtains Background From, Confirms, Uses Conclusions From, Cites for Information, Credits and Cites as Related belonging to the group of those most used. However, these results should be treated a little more cautiously than DEO ones – first of all, some citations occurred more often than others (but that is a deliberate consequence of the Author's intentions), and, more importantly, the CiTO documentation is a little more sketchy, thus giving me more space for personal interpretations.

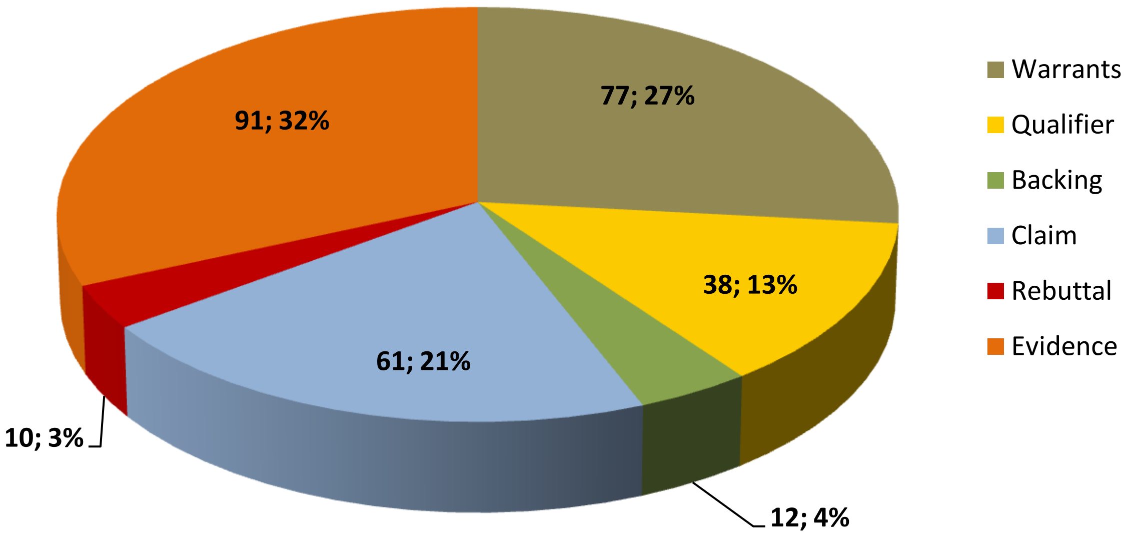
Graph 7 - Citation Network Properties Distribution



Closing this subsection is the **Argumentation Lens**: Here are the related data about its composition, excluding the 61 Annotations "`rdf:type amo:Argument`" which I used to identify the Argument wrapper elements and which were not included in Graph 8:

Warrants	77
Qualifier	38
Backing	12
Claim	61
Rebuttal	10
Evidence	91

Graph 8 - Argumentation Model Components Distribution within the Argumentation Lens



We can see that the three mandatory components of Toulmin's Argument Model, *Claim, Evidence* and *Warrant*, are the three most used properties with which Argumentation component are identified: Together, they represent exactly the 80% of all AMO related annotations used. Evidence, or Data, is the most frequent identification for an argumentation component role. It is also possible to observe that, among the three other denotations, Qualifiers are used far more often, and they represent more than half of the occurrences within this subset.

All these result appear pretty consistent with the expectations of modeling a successful and convincing scientific discourse. The presence of enough relevant Evidence is mandatory in order to convince of the validity of Claims, and it seems quite unsurprising that in a scientific article the majority of the argumentation model components carry out this role – especially if we consider the fractal nature of the Argumentation model, with some components or whole argumentation being re-used within a different argumentation, possibly with a different role. The dominance of Qualifiers is also not unexpected, as within scientific discourse many of the claims are valid only under specific circumstances or hypothesis.

8.4 Lessons Learned and possible, recommended improvements

After all the selected lenses applications over the target document have been applied and individually analyzed, it is possible to shortly summarize some common observations over the result of this task.

First of all, after completing the application activity, I am satisfied to report that the very simple workflow for which SLAM was designed (which in turn was derived from the general methodology, see section 5) accomplished its intended role, allowing me to combine readability, simplicity and flexibility in writing the application instruction. The possibility to re-use lenses annotations and to assert over Set of items (or nodes) appropriately selected was invaluable in cutting down the length of such a vast work down to a manageable size. **For example, the 1095 Document Structure lens assertions are the result of less than 100 lines of code**. This readability, compactness and usability advantage in the approach I adopted is even true when we consider the necessity of editing, re-using or correcting the existing code. In the previous sections, I have also provided ample and heterogeneous examples of the flexibility of SLAM Earmark Finder in finding and selecting sets of relevant elements to be then consequently used as subjects for the annotations defined within a lens.

This, of course, is not just the consequence of using a set of tools (SLAM), but also a direct result of applying the general methodology I proposed. The **bottom-up** approach within the Semantic Lenses stack that I chose to follow proved to be much useful in the practical activity, as it became possible to use the already annotated lenses as guidance for the decision making process in the authoring of the following ones. For the case of the Rhetoric Organization Lens, for example, planning its application without considering the Pattern assignment from the underlying Document Structure layer would have been very difficult, considering the strong requirements of DOCO.

So, with the conclusion of this application activity, I can safely say that I had been able to validate the effectiveness of this methodology. Indeed, most of the problems or the difficulties I had encountered in accomplishing the aforementioned goals, were caused either by the nature of the structural markup of the target document (such as the impossibility to properly define Sections), or by incompatibilities (being solved) between EARMARK and Jena, or just minor inconveniences, like those related with the search of the appropriate terms within the Semantic Lenses vocabulary, which could perhaps be improved, as I will discuss in the next few pages. The other implementation compromises I had to take were those related to the limited scope of this demonstration.

In closing, it might also be interesting to take note of the possible relationships between the three most content-related lens used in my activity, which is something I already hinted at in their dedicated sub-sections. I have summarized in the following tables the citation properties within the context of their use in the target paper, together with DEO and AMO denotations. The part written in bold text represent some of the instances where the new terms I suggest in section 8.4.1 could have been used.

Table 4 – Summary of observed cross-lens relationships

Citation	**CiTO Properties of the Citation**	**DEO class of the paragraph**	**Argumentation Role of the statement.**
1	Cites As Authority, Confirms, Obtains Background From, Cites For Information	Background	NONE
2	Obtains Background From, Cites For Information, Disagrees With	Background	Arg01_Rebuttal01
3	Obtains, Background From, Critiques, Cites As Related, Reviews, Corrects, Uses Conclusions From	Motivation	Arg02_Qualifier01
4	Obtains Background From, Confirms, Shares Author With, Agrees With, Extends	Background	Arg05_Claim01
5	Cites As Authority, Uses Conclusions From	Background/ **Clarification?**	Arg06_Qualifier01; Arg07_Claim01
3	Obtains, Background From, Critiques, Cites As Related, Reviews, Corrects, Uses Conclusions From	Background/ **Clarification?**	Arg07_Evidence02
6	Cites For Information, Obtains Background From, Uses Conclusions From Uses Method In, Credits	Scenario	Arg09_Qualifier01
7	Critiques, Cites As Related, **Differently From?**	Related Work	Arg12_Qualifier01
8	Updates, Cites As Related	Related Work	Arg12_Qualifier02
3	Obtains, Background From, Critiques, Cites As Related, Reviews, Corrects, Uses Conclusions From	Methods	Arg14_Evidence01
6	Cites For Information, Obtains Background From, Uses Conclusions From Uses Method In, Credits	Discussion	Arg17_Claim01

9	Uses Method In, Credits, Cites For Information	Discussion	Arg17_Claim01
10	Credits, Cites As Related	Discussion	Arg17_Evidence01
11	Credits, Cites As Related	Discussion	Arg17_Evidence01
12	Uses Method In, Uses Conclusions From, Credits, Cites As Related	Methods	NONE
13	Shares Author With, Extends, Uses Data From, Cites As Data Source.	Methods	Arg37_Qualifier01
5	Cites As Authority, Uses Conclusions From	Discussion	Arg47_Evidence01
14	Cites For Information, Credits Obtains Support From, Confirms	Discussion	Arg47_Evidence01
4	Obtains Background From, Confirms, Shares Author With, Agrees With, Extends	Discussion	Arg47_Backing01
15	Agrees With, Reviews, Cites As Related	Discussion	Arg53_Qualifier01
16	Shares Author With, Confirms, Extends, Cites For Information	Conclusion	Arg58_Claim02
7	Critiques, Cites As Related, **Differently From?**	Related Work	Arg59_Warrant01
17	Discusses, **Differently From?**	Related Work	Arg59_Rebuttal01
18	Discusses, **Differently From?**	Related Work	Arg59_Rebuttal01

We can see that the act of citing other documents is quite evenly distributed when we consider the argumentation components, but these do tend reflect the nature of the citation. For example, Citations which are used to draw support, re-use conclusion from or are cited as data source are either within Evidences or Qualifiers, which is something to be expected while, instead, citations whose background is obtained are more evenly distributed within the argumentation model. In retrospect with the DEO rhetoric organization, the act of citation seem to be mostly done either in those part that introduce the problem (Background, Motivation) and in those that discuss and expound on the results of the Author's work (Discussion, Conclusion), with the obvious addition of those paragraphs dedicated to presenting other Related Works.

8.4.1 Suggested changes in the SPAR Ontologies

Finally, in order to conclude this sub-section, I present a short summary table to group all the suggested changes and additions that I believe could profit further application activities, based on the experience in performing the tasks described so far.

Table 5 – Summary of the proposed changes and additions in the Ontologies used

Lens and Ontology	Suggested Additions	Suggested Changes
Document Structure – Pattern Ontology	Nothing specific.	Allowing for the Table pattern to accept homogenous substructures made by the repetition of more than a single element (e.g. <dl> and the repetition of <dt> and <dd> inside them)
Rhetoric Organization – DOCO	New ontology classes to identify specific intra-document links, and especially one to denote the role of inline citation references. The addition of classes to identify a block of keyword or a timeline might be useful as well	A global relaxation and review of some of the structural requirements, both as Patterns or as other components
Rhetoric Organization – DEO	New ontology classes to identify an aside, a digression or the refinement of a concept, either with a single class or with multiple new ones. For example, it could be possible to define a "**`deo:Clarification`**" and a "**`deo:Differentiation`**" class	An improvement in the wording of the documentation, in order to reduce possible ambiguities
Citation Network – CiTO	New ontology properties to deal with citations that are not negative but emphasize a differentiation between the cited and the citing document. For example, those on document cited for their complementary approach over a subject. In a sense, this is somewhat of a continuation of the issue previously addressed in DEO. For example, it could be possible to define a new property, something on the line of "**`cito:unlike`**" or, perhaps, or "**`cito:differentlyFrom`**".	A strong improvement and overhaul of the documentation, with lengthier and more meaningful descriptions for the citation properties.
Argumentation – AMO	Nothing specific.	Nothing in particular.

8.5 User Testing the TAL prototype

The Through A Lens – TAL – Prototype Interface I obtained by the enhanced version of the target document was the subject of an additional test activity, one aimed at its user testing.

In [PVZ12] an user testing session was undertaken with the purpose of gathering some preliminary data about the usability and the effectiveness of TAL, which is not yet a complete application, but is still at the prototype stage. The TAL page was generated from the annotated document (representing the enhanced version of [Mik07]) by using the Java Framework I developed, and which is already described in section 6, and it was put online. After that, three different unsupervised tasks involving navigation and the focusing of lenses over it were planned for execution by the test subjects.

The test subjects which graciously volunteered to assist us are nine people with heterogeneous backgrounds (from PhD students to some publishing houses employees), which were asked to perform these three pre-planned tasks, without any supervision, and without any previous familiarity with the TAL application, its interface, or even without any previous knowledge of the Semantic Lens model on the whole. No "administrators" observing the subjects or providing guidance for their actions were present while they undertook these tasks.

These are the tasks given to the test subjects:

Table 6 – Tasks in the User Testing of the TAL prototype

Task	Object of the Task	Time	Successes
Warm-up Task	Use TAL to find the paragraph containing the 2nd claim and write down all the citations within that paragraph, noting and reporting the motivations behind those citations. **This is a combined task involving several features of TAL**	5 minutes	Not defined
1	Write down all the motivations behind the citation of the reference #[8] in the target document **This is a task which mainly relies on the correct focusing of the Citation Network lens.**	5 minutes	**5/9 (55%)**

Task	Object of the Task	Time	Successes
2	Write down the textual evidences of a specific claim within the target document – find and note down the evidence on the claim whose original text is: *"It is important to note that in terms of knowledge representation, the set of these keywords cannot even be considered as vocabularies, the simplest possible form of an ontology on the continuous scale of Smith and Welty [5]"* **This was a task aimed at the testing of main feature of TAL, the Argumentation Index, and thus on the focusing of the Argumentation Lens.**	5 minutes	**9/9 (100%)**
3	Identify, by writing down their first words, all the paragraphs containing Problem Statements discussed in the paper. **This was a task aimed at working with the Rhetoric Lens, and specifically, with its Denotations located at the beginning of each paragraph.**	5 minutes	**6/9 (66%)**

After all the tasks were performed, the test session was concluded by asking the subjects to fill in two short questionnaires, one with multiple choice answers and the other textual, in order to collect their thoughts on their experience of using TAL to complete these tasks (max. 10 minutes). All the questionnaires and all the outcomes of the experiments are available online[18].

Out of the 27 total main tasks (3 tasks given to each of 9 subjects), 20 were completed successfully (e.g., the right answers were given), while 7 had incorrect or incomplete answers, giving an overall success rate for task completion of 74%.

These 20 successes were distributed as follows: 5 in Task 1, 9 in Task 2 and 6 in Task 3.

The usability score for TAL was computed using the System Usability Scale (SUS) [Bro96], a well-known questionnaire used for the perception of the usability of a system. It has the advantage of being technology independent (it has been tested on hardware, software, Web sites, etc.) and it is reliable even with a very small sample size [Sau11].

[18] Results of user testing activity on TAL, from [PVZ12]: http://www.essepuntato.it/sac2013/questionaries

In addition to the main SUS scale, we also were interested in examining the sub-scales of pure Usability and pure Learnability of the system, as proposed recently by Lewis and Sauro [LS09]. As shown in the following table, the mean SUS score for TAL was 70 (in a 0 to 100 range), surpassing the target score of 68 to demonstrate a good level of usability [Sau11]. The mean values for the SUS sub-scales Usability and Learnability were 69.44 and 72.22 respectively.

Table 7 – SUS Scores resulting from the user testing

Measure	**Mean**	**Max Value**	**Min Value**	**Standard deviation**
SUS Value	70	95	50	13.58
Usability	69.44	93.5	53.13	12.18
Learnability	72.44	100	37.5	24.83

Even if the TAL interface is still at the early prototype stage, and it is not yet a complete application, the outcomes reported from the user testing session can on the whole considered positive, and these results are an encouragement for further development of TAL, as well as giving valuable indication on which aspects of the interface are to be improved in order to enhance its usability and effectiveness during the focusing of applied lenses.

9 Conclusions

Within my thesis dissertation I have shown three relevant research results in the field of Semantic Publishing. After introducing the scientific and technological context of this work, and after having presented the Semantic Lens model [PSV12a] for the enhancement of scientific papers, I have discussed the development of a set of methodologies for the application of that model, the development of tools (**SLAM** and **TAL**) for performing the two main tasks of Lens **application** and **focusing**, together with a proof of concept prototype obtained from a concrete application of these methods and tools in a case study for four Semantic Lenses (*Document Structure, Rhetoric Organization, Citation Network and Argumentation*), with the HTML version of Peter Mika's "*Ontologies are us*" [Mik07] as the object of my tests.

In this document I have illustrated many of the possible advantages of the Semantic Lens model for document enrichment and demonstrated the feasibility of the suggested methodology for the application of metadata, expressed as RDF statements over selected vocabularies, aimed to enrich the meaning of a document and of its components. I accompanied it with examples on how to concretely implement the methodology, as well as a discussion on the difficulties encountered and on the way to overcome them in order to reach our intended goal. Treasuring from this experience, I recommended some improvements that might help in future activities, and also pointed out some interesting future developments that might be made possible by the continued research on Semantic Lenses.

In section 5, I discussed the advantages and the necessities of involving Authors within the Semantic Lens application activity, then I went on by proposing a general methodology for Lens application, both on a general level and on a lens by lens case. I emphasized the advantages of using the EARMARK document model [PV09] for representing the target document, given its advantages in handling overlapping markup, its integration of traditional and semantic web notations and technologies, and its powerful, versatile and well-document Java API. I strongly advocated a **bottom-up**, information-rich approach within the Semantic Lenses stack, with a workflow as dethatched as possible from the specificities of the target document. Such peculiarities will have nevertheless to be addressed, especially considering how strongly tied to the document are the most content-related Lens, but the less document-specific solutions are required, the more re-usable is the general methodology.

As a consequence of this, in section 6 I expounded on the development of **SLAM** – "***S****emantic* ***L****enses* ***A****pplication and* ***M****anipulation*" – a Java package that acts as a tool for the **application** of semantic lenses, an activity consisting in the authoring of the semantic assertions enriching the document and its components and their appropriate embedding within it. The SLAM package is a framework incarnating the methodology I proposed, and it extends both the Jena and the EARMARK API. SLAM models as closely as possible the approach I suggested, and makes it easy to annotate documents and their components in a straightforward and modular way. It relies on Appliers to enact the instructions written within Applications, in order to add Annotations within a document. Its workflow was designed to be both simple and versatile, and it has proven effective in reaching the intended results and encourages re-use of Lenses Annotations within it, allowing for improved readability and error-correction of the code when compared to more primitive approaches. It also offers many new possibilities for the selection and extraction of Items within an EARMARK node, through its Finder, as well as several additional utilities, like the recording of statistical information over a Lens Application.

The TAL – **T**hrough **A L**ens – is the prototype interface that I developed for the **focusing** of lenses applied over a document. I have defined focalization as the set of activities using the metadata embedded within the lenses assertions in order to highlight specific facets of a document, aiming to offer enhanced user interactions and explicitly emphasize aspect-related meaning emergence. I explained what kind of features can such a prototype offer, such as an explorable Argumentation Index, listing claims and other argumentation components, a Citation Network Indexes with all citation properties motivating the purpose of each citation, informative tooltips over Rhetoric and Citations and contextual Rhetoric Organization denotations for the document's paragraphs. I also detailed how this prototype has been obtained, by the development of the TAL Java package which is capable to extract the relevant information from an annotated document and to reprocess it into an output format. This allows for a stark separation between content extraction and its formatting, thus allowing for possible future extension and exports in other formats. I also quickly described the methods and the technologies that are used by the Formatter to create the presentational interface, including JQuery.

All these research and development activities where not just limited to the pure theoretical planning of a methodology or to the implementation of untested tools. On the contrary, they were instrumental in order to extensively field test the Semantic Lens model for the first time, and to verify if it was possible to produce significant concrete results from it. In section 8 I depicted this field-test activity over [Mik07], first focusing on the application process lens by lens, examining the applicability, through the available tools, of both of the methods and the model (including the vocabulary) presented so far, together with several examples taken from my work. I was able to conclude that activity successfully, keeping close to my intended approach, and validating the feasibility of the suggested workflow, which produced a completely enriched document on all four levels examined, in observance to the intended goals. I was also able to examine in details all the difficulties encountered through this process, and to expound on how they were solved. I went on to examine the numerical statistics resulting from this application, and have drawn several conclusions on the most used denotations, and how these fit within the bigger picture of scientific discourse, and observed on how these results might encourage further research and analysis on bigger samples of enhanced documents. In order to profit from the experience and the know-how gathered from my work, I also collected several suggestions for the improvement of the technologies and of the vocabularies used by the Semantic Lens model, which I justified and subsequently summarized. From the enriched document I obtained I was also able to generate a concrete example of the TAL prototype interface, which underwent user-testing, as related in [PVZ12], and showed a reasonable amount of success in terms of Usability and Learnability, especially if we take into account its limitations inherent to its alpha-prototype stage.

Thus, what I presented in section 8 was a practical field test for both the methodology and the two Java frameworks developed, SLAM and TAL. It was also the first concrete exercise ever attempted to enact the Semantic Lenses model for the enhancement of scientific papers, and it resulted in the successful production of both an enriched document and a prototype interface for its browsing.

This operation was undertaken to the fullest extent allowed by the sub-set of lenses considered for this exercise, and I believe it has served admirably in helping to gather useful information for the perfecting of the methodology and the tools, as well being successful in demonstrating their appropriateness, as discussed in the previous pages.

However, even if it has been a very methodical, fully featured work on application and focusing, it was also not the only goal of this thesis demonstration, (but more a means to an end), and as such it was limited in scope as well as in the domain universe of its application, which was a single paper.

Consequently, this activity is only a building block in the effort of developing a full set of tools and interfaces, scalable, portable and usable, in order to achieve a concrete realization of the current Semantic Lens model architecture, aiming to provide authors, editors and publishers of scientific document with an approach to semantic enrichment of publisher that could capture all the relevant aspects of a document and incarnate all advantages of the Semantic Publishing idea.

By all means, there is still much that could be done within the development of Semantic Lenses in the context of Semantic Publishing, as well as many more application and research paths that are just now opening up for future investigation. I shall now give a brief overview of these future opportunities, most of which would obviously require a larger sample of enriched document, which could perhaps be a set of papers from the same journal or a set of conference proceedings, in order to gather data from a large enough number of sources to make relevant and well grounded observations that might be more easily generalized.

I have already hinted at one of these possibilities in section 8.4: The research of correlations between different levels of the typical scientific paper, as well as the study of the possible correlations between assertions related to different lenses within the same part of a document, is one of the most interesting prospective paths. Discovering which correlations exist, or might be reasonably expected, between different facets of the Semantic Lens model, and then studying the strengths of these correlations and the conditions under which they are present would probably prove a very great asset.

The study on these correlations would also be one of the possible cornerstones for another important future development of the concrete application of Semantic Lens, which would be researching and perfecting automated recognition not just of Structural Patterns, but of the characterizations related to other lenses as well (such as the Rhetoric Organization). The aim is to become able to automatically identify the denotations of components within a lens, or at least a range of the most likely assignments, thus assisting the users and speeding up the application task over existing documents. It order to do this, an important step would be the development of reliable heuristics, able to formulate reasonably accurate hypothesis on what role could have a document component within a lens when its role in the lower level lenses had already been identified. This is still another possible use for the bottom-up methodology advocated within this dissertation.

With a large enough sample, further investigation of the most widely used assertions within each lens (like I did in section 8.3), could also prove interesting. Such results could help us identify likely patterns in each specific level of significance within scientific documents, including, perhaps, enough data to reinforce our understanding of how we organize scientific discourse, to improve the abovementioned heuristics, to discover relevant relationships between concepts located at different levels of understanding, or to put to the test new user interaction options.

In the rest of this section, I will illustrate some other possible applications of Semantic Lenses, divided by intra-document ones and inter-document functions.

Given that the Through A Lens – TAL – framework and resulting prototype interface has already been extensively discussed, I will start by detailing what possible extensions of the TAL prototype and what new features could be developed in the near future. The result of some user testing over the prototype has just been presented in the previous section, and it's logical to start from there.

First of all, its Learnability factor could probably be improved by adding an in-document documentation on the meaning of the ontology terms and properties used, either through links or tooltips. This documentation can be obtained from the ontologies themselves, and, for example, might be extracted on demand from their online version. The Citation Index could also be improved, by adding further navigation options, like the possibility of displaying only all the properties present within a single bibliographic reference, or shared by set of them. Search options for specific properties might also be included.

Another quality-of-use improvement could be implementing the highlighting of the relevant text snippets or document components within the main document area when the user is hovering over their counterpart within the Argumentation Index. In general, all the interface can be improved, and a more complete set of tools for the focusing task could be designed, perhaps accompanied by appropriate graphics. Also, all lenses and filter could be made to appear or disappear on command: for example, the Contextual Rhetoric denotations located at the beginning of each paragraph might be hidden or shown on command. Other indexes like the Citation and the Argumentation Index could be built for other levels, like the Rhetoric or the Semantic ones, and all relevant meta-information about the document, like those captured within the first three more-context related semantic lenses (*Research Context, Contribution and Roles, Publication Context*) could be gathered, if present, and shown on user request. Other filters for could be developed – for example, it might be possible to color the main text within the paper with argument model related keys, or highlight with different colors or border elements within the text according to their Structural Pattern during a focusing activity on the Document Structure lens.

There are also some features involving a certain amount of inter-documental interaction that might be easily added to TAL. For instance, the possible presence of the *Textual Semantic* Lens would encourage the development of tooltips with information on the entities being used or identified by the semantic vocabularies, perhaps extracting them straight from the reference ontologies, and Citation Tooltips could ideally point to the resources being cited, or even to their meta-document information. In short, there are many possibilities in user interaction that might be explored when considering the focusing task as the selection of a lens in order to highlight a specific semantic aspect of a scientific document, and with TAL we have just started to scratch the surface of this rich and valuable vein of applications.

While the enabling of worthy and meaningful interactions through the focusing task is the ultimate goal of the Semantic Lenses model, it is not the only advantage of its adoption, even within a single document perspective. Having an enhanced document ready at hand might open up at least two other innovative possibilities, which might not be directly tied to the focusing of lenses, but are the consequence of gathering and using the information embedded within them.

First of all, the correct application of the Document Structure lens, together with the DOCO related part of the Rhetoric Organization one, can be of assistance in document conversions between different formats. Software, applications and framework tasked with converting document formats might use the information encoded within the lenses, as expressed by component's assertions, to better perform the conversion, especially if some general pattern assignment reference can be made for a format (e.g. [DPP12]).

For instance, an HTML `<div>` might be converted in DocBook as an `<abstract>` if it is flagged as a "`pattern:Container doco:Abstract`", rather than being converted as a `<blockquote>` if it has "`pattern:Block doco:BlockQuotation`" as its assigned lenses assertions.

Another promising inter-document application that might be possible thanks to the Rhetoric Organization Lens is the automatic validation of the rhetorical level of a paper, especially the structure of its discourse. It would be possible to imagine and define meaningful requirements such as "*a well structured paper has to have a Problem Statement within the 1st section, must express at least 1 Background in the next part, and must not present Conclusions before data are Discussed somewhere in between*". Then these requirements could be verified, for example at the document ontology level. Ways to formalize such schemas and to apply these validation checks to submitted documents might prove very useful within the publishing process, as many journal publishers usually require that works submitted to them adhere to certain standards.

Of course, we can surmise that other levels might be subject of validation as well, perhaps even by mixing more facets to form up complex requirements, such as one stating that a document *"must have at least X citations with property Y or Z in its sections which are identified as* `'deo:Background'` *"*.

As already said, many additional operations for Semantic Lenses would become far more relevant if a sizeable set of documents were enhanced according to its model, or in a way that could be related to it.

I have just reasoned on some possible improvements for TAL (or other focusing interfaces) and explained how some these additional features for interfaces might benefit from the ability of fetching data at runtime to other resources, like getting digital abstracts from cited documents, or definitions for named entities. After all, that is what the Linked Open Data and the Semantic Web is all about. Obviously, the systematic use of Semantic Web technologies by Semantic Lenses offers documents enriched with them integration and access to the LOD and the Semantic Web, and allows for them to be fully accessible within it as well. This opens up possibilities related to the inter-documental use of statements specific to the Lens model.

The information encoded in the first three Lenses of the stack, those that are more-context related and centered on wrapping data about the whole document (*Research Context, Contribution and Roles, Publication Context*) could be gathered or indexed, and presented on request by any other documents, interface or application, acting as an introductory informative entry point for the publication. Or, to extend this example, a short digital abstract could be built by combining a summary of these information together with a list of relevant claims and the contents of the components marked as "`doco:Abstract`".

Another extremely important theme is the possibility to revolutionize the metrics of scientific citation measuring. We know that the measurement of citations between peer-reviewed papers is an important way to evaluate the impact of a scientific article, and that the productivity of scientists and research projects is estimated on a similar basis. However, currently available methods might only take into account the simple fact that a paper cites another, and evaluate this act with estimates on the importance of the paper performing the citation. But there is a very significant difference between a scientific document cited as an important source, or a seminal work within its field, and one cited only in order to be disproved or dismissed as ridiculous. By making the reasons behind a citation explicit and providing such information in a way that is readily and unambiguously available, the citation network lens might offer the foundation to develop improved indexes better suited to correctly estimate the importance and the impact of a publication, as it would be reasonable to weight differently citations according to the motivation behind the act of citation itself. As it is, this lens has a great potential in terms of inter-documental semantics.

Another possible inter-documental application tied to the citation network might be discovering if there are intersections between the citations present in two documents, and if they are differently denoted. For instance, we might easily find out that article X and the conference proceeding Y both cite the document C, but it the motivations behind that citation might differ.
Finally, with large sets of document it might also be possible to venture deeper in a semantic statistical analysis of the metadata used, as already hinted.

In closing, I think that this lengthy thesis demonstration makes a convincing case on how promising, versatile and worthy of attention the Semantic Lenses model is, as well as paving the way for its further development, by providing some basic building blocks (in terms of methods and tools) for its use. Most issues encountered (and solved) and most implementation compromises were related either to the nature and the structure of the target document, or to limitations due to the early stage of maturity and relative novelty of the technologies available or developed (as in tools and vocabularies), most of them being untested or in the prototype stage. These are all open to further improvement, but did not show any fundamental defect or insurmountable limitations in their design.
To conclude, it is my belief that with this work I have been able to obtain, show and detail how the Semantic Lens model is a worthy addition to the effort of encouraging the Semantic Publishing revolution, and that it offers both several concretely appreciable and measurable results in its actual application, several all-round advantages over other approaches, as well as many promising opportunities for further development and growth.
We have seen the impressive rate at which scientific information is being produced every day, and thus it is easy to understand how important is to be able to quickly retrieve and sift through this impressive amount of data and reasoning already at our disposal, especially in order to find out what how it can relate to our intended scientific hypothesis and organization of discourse. After all, *"Human reason can neither predict nor deliberately shape its own future. Its advances consist in finding out where it has been wrong"* [19]. From this acceptance that in science there are no theories that cannot be disproven, and that refutability is part of the scientific method, "*in so far as a scientific statement speaks about reality, it must be falsifiable*" [20], also comes the desire for being able to access, examine, comprehend and, if the need arises, eventually discuss and disprove what other scientists have proposed. By enabling us to improve our correct understanding of these existing findings, by making easier to correlate separate results with related ones in order to put together a more complete picture of the problem

[19] F. A. Von Hayek (1960); *The Constitution of Liberty*
[20] K. Popper (2002); *The Logic of Scientific Discovery*

domain, and by improving the interactivity of document contents as well as encouraging the emergence of semantics and of knowledge, Semantic Publishing technologies, including Semantic Lenses, might truly herald a revolution in scientific productivity and accessibility.

Conclusioni

Nell'ambito di questa dissertazione di tesi ho mostrato risultati di ricerca di rilievo nell'ambito del settore del Semantic Publishing. Dopo aver introdotto il contesto scientifico e tecnologico in cui si colloca questo lavoro, e dopo aver presentato il modello delle Lenti Semantiche [PSV12a] per l'arricchimento semantico di documenti scientifici, ho discusso la ricerca e la definizione di un insieme di metodologie per l'applicazione di questo modello (SLM). Ho illustrato lo sviluppo di due package Java (SLAM e TAL) il cui scopo è fornire strumenti utili al compimento delle due principali attività di **applicazione** e **focalizzazione** delle Lenti. Ho infine sottoposto questi concetti e questi strumenti ad una prova pratica, ottenuta testando la reale applicazione di questi metodi, tramite i suddetti strumenti, ed ottenendo infine dei prototipi concreti, la cui casistica si fonda sull'applicazione di quattro Lenti Semantiche (*Strutturale, Retorica, Citazionale e Argomentativa*) sulla versione HTML di "Ontologies are us" di Peter Mika [Mik07].

In questo elaborato ho esposto in dettaglio molti dei possibili vantaggi delle Lenti Semantiche come modello per l'arricchimento documentale, ed ho mostrato la fattibilità dell'applicazione della metodologia da me proposta per l'applicazione di metadati, espressi come statement RDF nell'ambito di vocabolari selezionati, al fine di arricchire ed esplicitare il significato di un documento scientifico e dei suoi componenti.

Ho accompagnato questa esposizione con esempi su come implementare concretamente la metodologia, sfruttando gli strumenti a disposizione, ed anche con una discussione sulle difficoltà incontrate e su come queste sono state superate al fine di raggiungere l'obiettivo prepostomi. Facendo tesoro di questa esperienza, ho anche raccomandato alcuni miglioramenti e cambiamenti che potrebbero aiutare nello svolgimento di attività future, così come ho suggerito alcuni sviluppi possibili di una continuazione della ricerca sulle Lenti Semantiche.

Nella sezione 5 ho discusso i vantaggi del coinvolgimento degli autori nelle attività di applicazione di Lenti Semantiche, ed ho successivamente proposto una metodologia – **SLM** o "***S**emantic **L**enses **M**ethodology*" – per l'attività di applicazione, sia a livello generale che lente per lente. Ho enfatizzato i vantaggi dell'uso del modello documentale di EARMARK [PV09] per rappresentare il documento oggetto dell'attività di arricchimento, viste le sue qualità nella gestione dell'overlapping markup, la sua integrazione dei pregi delle tecnologie web tradizionali e del web semantico, e le sue potenti, versatili e ben documentate API Java. Ho fortemente suggerito l'adozione di un approccio **bottom-up** carico di informazione nell'ambito dello stack delle Lenti Semantiche, il cui flusso di lavoro è il più distaccato possibile dalle specificità del documento bersaglio. Tuttavia queste particolarità dovranno comunque essere affrontate, specialmente considerando quanto fortemente correlate al documento sono le quattro lenti prese principalmente in esame in quanto più legate al contenuto. Tanto meno sono richieste soluzioni specifiche rispetto al documento bersaglio, tanto più riutilizzabile risulta la metodologia generale.

Come conseguenza di questo, nella sezione 6 ho esposto in dettaglio lo sviluppo di – **SLAM** or "***S**emantic **L**enses **A**pplication and **M**anipulation*" – un package Java volto a fornire uno strumento completo per l'**applicazione** di lenti semantiche, definita come una attività consistente nella redazione delle asserzioni semantiche che arricchiranno il documento ed i suoi componenti e nella loro appropriata aggiunta al suo interno. Il package SLAM è un frame work che dà corpo alla metodologia che ho proposto, e che estende le api di Jena e di EARMARK. SLAM modella l'approccio da me suggerito il più fedelmente possibile, e facilita l'annotazione di documenti e dei suoi componenti in un modo diretto e modulare.

Si basa su Applicatori per mettere in atto le istruzioni contenute all'interno di Applicazioni, al fine di aggiungere Annotazioni all'interno di un Documento. Il suo flusso di lavoro è stato progettato per essere sia semplice che versatile, e si è mostrato efficace nel raggiungere i risultati sperati, incoraggiando il riutilizzo di Annotazioni di Lenti al suo interno, consentendo una migliore leggibilità ed una più facile correzione degli errori di codice rispetto ad approcci manuali più primitivi. Offre inoltre nuove possibilità per la selezione, l'estrazione e la manipolazione di oggetti da un nodo EARMARK, il tutto attraverso il suo Cercatore. Fornisce anche altre utilità addizionali, come la registrazione di informazioni statistiche sul risultato di una Applicazione di Lente.

TAL o "***T**hrough **A** **L**ens*" è invece il prototipo di interfaccia che ho sviluppato per la **focalizzazione** di lenti già applicate su un documento. La focalizzazione è quell'insieme di attività che sfruttano i metadati immagazzinati tramite le asserzioni espresse dalle lenti con lo scopo di evidenziare specifiche sfaccettature di un documento, in modo da incrementare le possibilità di interazione utente ed enfatizzare esplicitamente l'emergere di significato relativo allo specifico aspetto preso in considerazione. Ho spiegato quali tipi di funzionalità può offrire questo prototipo, come un Indice Argomentativo navigabile, in grado di elencare tesi e componenti di ogni argomentazione; un Indice Citazionale elencante tutte le proprietà che motivano la selezione di una citazione; tolti informativi sulla lente Retorica e su quella Citazionale, e denotazioni contestuali dell'organizzazione Retorica dei paragrafi del documento. Ho anche mostrato come questo prototipo è stato ottenuto, ossia tramite lo sviluppo del package Java TAL, che è in grado di estrarre le informazioni rilevanti da un documento annotato con Lenti Semantiche, per poi processarle producendo un formato di output, in modo da avere una separazione netta fra l'estrazione dei contenuti e la loro presentazione, facilitando così future estensioni o l'esportazione in altri formati. Ho anche rapidamente descritto i metodi e le tecnologie che sono usate dal Formattater per creare l'interfaccia di presentazione, fra cui JQuery.

Tutte queste attività di ricerca e sviluppo non si sono limitate solo alla pura pianificazione teorica di una metodologia, o all'implementazione di strumenti non applicati concretamente. Al contrario, queste sono state la premessa fondamentale al fine di poter mettere per la prima volta alla prova il modello delle Lenti Semantiche, e verificare la possibilità di ottenere risultati significativi dalla sua adozione. Nella sezione 8 ho esposto i dettagli di questa attività di test pratico avente come oggetto [Mik07]. Innanzitutto mi sono concentrato sul processo di applicazione, lente per lente, tramite l'uso degli strumenti sviluppati, ed aderendo alla metodologia da me proposta ed alle raccomandazioni del modello (vocabolari compresi). Ho presentato vari esempi presi dal risultato del mio lavoro.

Sono stato in grado di concludere questa parte dell'attività con successo, mantenendomi vicino all'approccio prefissomi, e validando così la plausibilità del flusso di lavoro suggerito, ottenendo come prodotto finale un documento completamente arricchito su tutti e quattro i livelli esaminati, in accordo con gli obiettivi che mi ero prefigurato di raggiungere. Sono stato inoltre in grado di esaminare in dettaglio tutte le difficoltà incontrate nel corso di questo processo, e di spiegare come sono state superate.

Ho proseguito presentando le statistiche ed i dati numerici raccolti durante questa attività di applicazione, ed ho potuto trarre diverse conclusioni sulle connotazioni più usate, e su come queste possano rientrare in un quadro più ampio della modellazione del discorso scientifico, osservando come questi risultati potrebbero incoraggiare ulteriori ricerche ed analisi su campioni più grandi di documenti arricchiti.

Al fine di trarre profitto dall'esperienza e dal know-how accumulato nel corso del mio lavoro, ho anche raccolto diversi suggerimenti per il miglioramento delle tecnologie e delle ontologie utilizzati nel modello delle Lenti Semantiche, che ho motivato e riassunto in una tabella apposita.

Dal documento arricchito da me ottenuto sono anche stato in grado di generare un esempio concreto del prototipo dell'interfaccia TAL, che a sua volta è stato sottoposto ad una sessione di user testing, relazionata in [PVZ12], e che ha mostrato un ragionevole ammontare di successo in termini di Usability e Learnability, specialmente se teniamo in considerazione le sue limitazioni inerenti al suo essere ancora ad un primitivo stadio di prototipizzazione.

Infine, ritengo che questa lunga dimostrazione di tesi costituisca una convincente argomentazione a sostegno di quanto promettenti, versatili e degne di attenzioni siano le Lenti Semantiche, e penso che possa altresì iniziare ad indicare una strada per il futuro sviluppo di questo modello, fornendo le prime fondazioni (in termini di metodologia e di strumenti) per il loro uso concreto. Inoltre, la maggior parte delle problematiche riscontrate (e risolte) e la maggioranza dei compromessi implementativi adottati sono stati dovuti o alla natura ed alla struttura del documento bersaglio, oppure causati da limitazioni dovute alla relativa giovinezza delle tecnologie a disposizione o sviluppate (vuoi come strumenti che come vocabolari), alcune delle quali mai testate o in stato di puro prototipo. Queste risultano essere tutte aperte a ulteriori miglioramenti, ma nessuna di esse ha mostrato fondamentali difetti o limiti insormontabili nella loro concezione di base.

Per concludere, è mia ferma convinzione l'essere riuscito, con questo lavoro, a dimostrare approfonditamente come il modello delle Lenti Semantiche sia una aggiunta degna di nota allo sforzo d'insieme mirato ad incoraggiare la rivoluzione del Semantic Publishing. Infatti, tramite la sua adozione e la sua messa in pratica, risulta essere in grado di offrire diversi risultati tangibili apprezzabili e misurabili, vari vantaggi rispetto ad altri approcci, e parecchie opportunità promettenti in termini di sviluppo e crescita futura.

Abbiamo potuto già osservare a quale impressionante velocità l'informazione scientifica venga prodotta ogni giorno, ed è quindi facile capire quanto importante sia essere in grado di recuperare e selezionare questo enorme ammontare di dati e di ragionamenti già a nostra disposizione, specialmente al fine di scoprire come questi si possono relazionare con le nostre intenzioni in termini di ipotesi scientifiche od organizzazione del discorso scientifico nell'esposizione dei risultati di una ricerca. Dopo tutto, *"La ragione umana non può né prevedere né deliberatamente plasmare il proprio futuro. I propri passi in avanti consistono nello scoprire dove si era sbagliata fino a quel momento"* [21]. Quindi, dall'accettazione che nella scienza non esistono teorie che non possano essere confutate, e che l'inficiabilità è parte del metodo scientifico, in quanto *"finché una asserzione scientifica si occupa del reale, deve essere refutabile"* [22], deriva altresì il nostro desiderio di essere in grado di trovare, esaminare, comprendere, e, se necessario, discutere e smentire quanto altri scienziati hanno prodotto. Tutto questo ci consente di migliorare la correttezza della nostra comprensione dei

[21] F. A. Von Hayek (1960); *The Constitution of Liberty*

[22] K. Popper (2002); *The Logic of Scientific Discovery*

risultati finora disponibili, rendendo più facile correlare risultati separati con altri potenzialmente collegati al fine di mettere insieme un quadro più accurato di un problema, e migliorando l'interattività del contenuto dei documenti, allo stesso tempo incoraggiando l'emergere del significato e della conoscenza in esso codificata, le tecnologie del Semantic Publishing, incluse le Lenti Semantiche, potrebbero essere le avanguardie di una vera rivoluzione nel campo della produttività e dell'accessibilità scientifica.

Bibliography – Ontologies References

[DFP08] – A. Di Iorio, F. Vitali, S. Peroni (2008); *Pattern Ontology*.
Version: 1.4, 21 May 2012. http://www.essepuntato.it/2008/12/pattern[23]

[Gan07] – A. Gangemi (2007); *Linguistic Acts Ontology*.
Version 1.1 http://ontologydesignpatterns.org/cp/owl/semiotics.owl

[Per08] – S. Peroni (2008); *EARMARK Ontology.*
Version 1.8.1, 24 February 2011. http:/www.essepuntato.it/2008/12/earmark

[Per11] – S. Peroni (2011); *EARMARK Overlapping Ontology.*
Version 1.0, 02 May 2011. http:/www.essepuntato.it/2011/05/overlapping

[Sho10a] – D. Shotton (2010); *Introducing the Semantic Publishing And Referencing Ontologies: SPAR*. First introduced with this blog post (2010):
http://opencitations.wordpress.com/2010/10/14/introducing-the-semantic-publishing-and-referencing-spar-ontologies/
Available at:
http://sempublishing.svn.sourceforge.net/viewvc/sempublishing/SPAR/index.html

[SP09] – D. Shotton, S. Peroni (2009); *CiTO the Citation Typing Ontology.*
Version: 2.4.1, 28 September 2012 http://purl.org/spar/cito

[SP11a] – D. Shotton, S. Peroni, (2011); *DoCO, the Document Components Ontology.*
Version 1.0, 5 May 2011. http://purl.org/spar/doco

[SP11b] – D. Shotton, S. Peroni (201); *DEO, the Discourse Elements Ontology.*
Version 1.0, 5 May 2011. http://purl.org/spar/doco

[VP11] – F. Vitali, S. Peroni (2011); *AMO, the Argument Model Ontology.*
Version 1.0, 4 May 2011. http://www.essepuntato.it/2011/02/argumentmodel

[23] Unless differently noted, all links were last visited on 30/09/2012

Bibliography

[AH04] – G. Antoniou, F. van Harmelen (2004); *A Semantic Web Primer.*
MIT press, April 2004, ISBN: 978-0-262-01210-2

[BBP12] – D. Beckett, T. Berners-Lee, E. Prud'hommeaux, G. Carothers (2012); *Turtle, Terse RDF Triple Language*. W3C Working Draft 10 July 2012
Available at: http://www.w3.org/TR/turtle/ [24]

[BCL94] – T. Berners-Lee, R. Cailliau, A. Luotonen, H.F. Nielsen, A. Secret (1994); *The World Wide Web.* Communications of the ACM, Volume 37 Issue 8, Aug. 1994. Pages 76 - 82
DOI: http://dx.doi.org/10.1145/179606.179671
Available at: http://www.lsi.upc.edu/~gabarro/Wap/p76-berners-lee.pdf

[Bec04] – D. Beckett (2004); *RDF/XML Syntax Specification (Revised)*.
W3C Recommendation 10 February 2004
Available at: http://www.w3.org/TR/REC-rdf-syntax/

[Ber01] – T. Berners-Lee (2001); *The Semantic Web.* Scientific American, Volume 284, Number 5, Feature Article, May 2001

[Ber06] – T. Berners-Lee (2006); *Linked Data, Designs Issue.* Personal view published on the W3C website. Available at: http://www.w3.org/DesignIssues/LinkedData.html

[Ber07] – T. Berners-Lee (2007); *Giant Global Graph*. Blog post from 2007
Available at: http://dig.csail.mit.edu/breadcrumbs/node/215

[BK08] – T. Berners-Lee, L. Kagal (2008); *The Fractal Nature of The Semantic Web*.
Ai Magazine - AIM , vol. 29, no. 3, pp. 29-34, 2008
Available at: http://www.aaai.org/ojs/index.php/aimagazine/article/view/2161/2017

[BLK09] – C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, et al (2009); *DBpedia – A Crystallization Point for the Web of Data.* In Journal of Web Semantics: Science, Services and Agents on the World Wide Web, 7 (3): 154.165.
DOI: http://dx.doi.org/10.1016/j.websem.2009.07.002

[BRL09] – B.C. Björk, A. Roos and M. Lauri (2009); *Scientific journal publishing: yearly volume and open access availability*. *Information Research***, volume 14, number 1,** paper 391.
Available at: http://informationr.net/ir/14-1/paper391.html

[24] Unless differently noted, all links were last visited on 30/09/2012

[Bro96] – J. Brooke (1996); *SUS: a "quick and dirty" usability scale*. In Usability Evaluation in Industry: 189-194. London, UK: Taylor and Francis. ISBN: 978-0748404600

[CSP12] – P. Ciccarese, D. Shotton, S. Peroni, T. Clark (2012); *CiTO + SWAN: The Web Semantics of Bibliographic Records, Citations, Evidence and Discourse Relationships* – Accepted for publication, to appear in Semantic Web Journal, 2012
Available at: http://www.semantic-web-journal.net/sites/default/files/swj175_0.pdf

[DeW10] – A. De Waard (2010); *From Proteins to Fairytales: Directions in Semantic Publishing*.
IEEE Intelligent Systems, pp. 83-88, March/April, 2010.
DOI: http://doi.ieeecomputersociety.org/10.1109/MIS.2010.49 -
available at http://lpis.csd.auth.gr/mtpx/sw/material/IEEE-IS/IS-25-2.pdf

[DBK06] – A. De Waard, L. Breure, J.G. Kircz, H. Van Oostendorp (2006); *Modeling rhetoric in scientific publications*. Proceedings of the International Conference on Multidisciplinary Information Sciences and Technologies, INSCIT2006; 25-28, October 2006
Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.105.2156

[DBC09] – A. De Waard, S. Buckingham Shum, A. Carusi, J. Park, M. Samwald (2009); *Hypotheses,*
evidence and relationships: The HypER approach for representing scientific knowledge claims.
In Proceedings of 8th International Semantic Web Conference, Workshop on Semantic Web Applications in Scientific Discourse. Lecture Notes in Computer Science, Springer Verlag: Berlin, 26 Oct 2009, Washington DC.
Available at: http://oro.open.ac.uk/18563/

[DG09] – B. D'Arcus, F. Giasson (2009); *Bibliographic Ontology [BIBO] Specification*.
Available at: http://bibliontology.com/specification

[Dum04] – E. Dumbill (2004); *DOAP – Description of a Project – Project Design Documents*
Available at: https://github.com/edumbill/doap/wiki/Project-design

[DPP12] – A. Di Iorio, S. Peroni, F. Poggi, F. Vitali (2012); *A first approach to the automatic recognition of structural patterns in XML documents.* In Proceedings of the 2012 ACM symposium on Document engineering, Pages 85-94 (DocEng 2012). New York, New York, USA.
Available at: http://palindrom.es/phd/wp-content/uploads/2010/07/dcng18-diiorio.pdf
DOI: http://dx.doi.org/10.1145/2361354.2361374

[DPV11a] – A. Di Iorio, S. Peroni, F. Vitali (2011); *A Semantic Approach to Everyday Overlapping Markup*. Journal of the American Society for Information Science and Technology, Volume 62, Issue 9, 2011
DOI: http://dx.doi.org/10.1002/asi.21591
Available at: http://palindrom.es/phd/wp-content/uploads/2010/07/jasist_earmark.pdf

[DPV11b] – A. Di Iorio, S. Peroni, F. Vitali, (2011); *A Semantic Web Approach To Everyday Overlapping Markup*. Journal of the American Society for Information Science and Technology, 62 (9): 1696–1716.
DOI: http://dx.doi.org/10.1002/asi.21591
Available at: http://palindrom.es/phd/wp-content/uploads/2010/07/jasist_earmark.pdf

[DPV11c] – A. Di Iorio, S. Peroni, F. Vitali (2011); *Using semantic Web technologies for analysis and validation of of structural markup.* In International Journal of Web Engineering and Technology, Volume 6 Issue 4, Pages 375-398, October 2011.
DOI: http://dx.doi.org/10.1504/IJWET.2011.043439

[GHK07] – T. Groza, S. Handschuh, H. L. Kim (2007); *SALT: Semantically Annotated LATEX for scientific publications*, Lecture Notes in Computer Science, 2007, Volume 4519/2007, 518-532.
Available at: http://www.springerlink.com/content/t220214924577133/
DOI: http://dx.doi.org/10.1007/978-3-540-72667-8_37

[Gru07] – T. Gruber (2007); [*Definition of] Ontology***.** Encyclopedia of Database Systems, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag, 2009.
Available at: http://tomgruber.org/writing/ontology-definition-2007.htm

[HB11] – T. Heath and C. Bizer (2011); *Linked Data: Evolving the Web into a Global Data Space (1st edition).* Synthesis Lectures on the Semantic Web: Theory and Technology, 1:1, 1-136. Morgan & Claypool.
DOI: http://dx.doi.org/10.2200/S00334ED1V01Y201102WBE001
Available at: http://linkeddatabook.com/editions/1.0/

[HKP09] – P. Hitzler, M. Krötzsch, B. Parsia, P. F. Patel-Schneider, S. Rudolph (2009); *OWL 2 Web Ontology Language Primer.* W3C Recommendation 27 October 2009
Available at: http://www.w3.org/TR/owl2-primer/

[IFL98] – IFLA Study Group on the Functional Requirements for Bibliographic Records (1998); *Functional Requirements for Bibliographic Records [FRBR].* Final Report.
Available at: http://archive.ifla.org/VII/s13/frbr/frbr_current_toc.htm

[KC04] – G. Klyne, J. J. Carrol (2004); *Resource Description Framework (RDF): Concepts and Abstract Syntax.* W3C Recommendation 10 February 2004
Available at: http://www.w3.org/TR/rdf-concepts/

[LS09] – J. Lewis, J. Sauro (2009); *The Factor Structure of the System Usability Scale*. Proceedings of the 1st International Conference on Human Centered Design, HCD09.
DOI: http://dx.doi.org/10.1007/978-3-648-02806-9_12

[LSM12] – T. Lebo, S. Sahoo, D. McGuinness, (2012). *PROV-O: The PROV Ontology. W3C Working Draft 03 May 2012. World Wide Web Consortium.*
Available at: http://www.w3.org/TR/prov-o

[Mik07] – P. Mika (2007); *Ontologies are us: A unified model of social networks and semantics*. Journal of Web Semantics, Volume 5, Number 2, June 2007: 5-15.
DOI: http://dx.doi.org/10.1016/j.websem.2006.11.002

[MM04] – F. Manola, E. Miller (2004); *RDF Primer.* W3C Recommendation 10 February 2004
Available at: http://www.w3.org/TR/rdf-primer/

[PGG08] – D. Picca, A. Gliozzo, A. Gangemi, (2008); *LMM: an OWL-DL MetaModel to Represent Heterogeneous Lexical Knowledge.* In Proceedings of the Sixth international conference on Language Resources and Evaluation (LREC 2008). Marrakech, Morocco.
Available at: http://www.lrec-conf.org/proceedings/lrec2008/summaries/608.html

[PMM11] – S. Pettifer, P. McDermott, J. Marsh, D. Thorne, A. Villeger and T.K. Attwood (2011); *Ceci n'est pas un hamburger: modelling* [sic.] *and representing the scholarly article.*
Learned Publishing, 24 (3): 207-220.
Available at:
http://www.ingentaconnect.com/content/alpsp/lp/2011/00000024/00000003/art00009
DOI: http://dx.doi.org/10.1087/20110309.

[PGV11] – S. Peroni, A. Gangemi, F. Vitali (2011); *Dealing with Markup Semantics.* In Proceedings of the 7th International Conference on Semantic Systems (I-Semantics 2011). September 2011, Graz, Austria.
DOI: http://dx.doi.org/10.1145/2063518.2063533
Available at:
http://palindrom.es/phd/wp-content/uploads/2010/07/earmark_isemantics2011_cr.pdf

[POJ09] – E. Pafilis, S.I. O'Donoghue, L.J. Jensen, H. Horn, M. Khun *et al* (2009); *Reflect: augmented browsing for the life scientist.* Nature Biotechnology, 27(6): 508-510.
Available at: http://www.nature.com/nbt/journal/v27/n6/full/nbt0609-508.html
DOI: http://dx.doi.org/10.1038/nbt0609-508

[PS12] – S. Peroni, D. Shotton (2012); *FaBiO and CiTO: ontologies for describing bibliographic resources and citations*. In Journal of Web Semantics: Science, Services and Agents on the World Wide Web.
DOI: http://dx.doi.org/10.1016/j.websem.2012.08.001

[PSV12a] – S. Peroni, D. Shotton, F. Vitali (2012); *Faceted documents: describing document characteristics using semantic lenses.* Proceedings of the 2012 ACM symposium on Document engineering, Pages 191-194. (DocEng 2012). New York, New York, USA.
DOI: http://dx.doi.org/10.1145/2361354.2361396
Available at:
http://palindrom.es/phd/wp-content/uploads/2010/07/docsp095-peroni.pdf

[PSV12b] – S. Peroni, D. Shotton, F. Vitali (2012); *Scholarly publishing and the Linked Data: describing roles, statuses, temporal and contextual extents.* In Proceedings of the 8th International Conference on Semantic Systems (i-Semantics 2012): 9-16. New York, New York, USA: ACM.
DOI: http://dx.doi.org/10.1145/2362499.2362502

[PV09] – S. Peroni, F. Vitali (2009); *Annotations with EARMARK for arbitrary, overlapping and out-of order markup.* Proceedings of the 9th ACM symposium on Document engineering, Pages 171-180. (DocEng 2009) Munich, Germany, 2009.
DOI: http://dx.doi.org/10.1145/1600193.1600232
Available at: http://palindrom.es/phd/wp-content/uploads/2009/07/eng030-peroni.pdf

[PVZ12] – S. Peroni, F. Vitali, J. Zingoni (2012); *Semantic lenses to bring digital and semantic publishing together*. Submitted for evaluation to the Symposium on Applied Computing 2013 (SAC 2013).

[RRF08] – R.B. Reis, G.S. Ribeiro, R.D.M. Felzemburgh, F.S. Santana, S. Mohr, et al. (2008); *Impact of Environment and Social Gradient on Leptospira Infection in Urban Slums*. PLoS Negl Trop Dis 2 - (4): e228.
DOI: http://dx.doi.org/10.1371/journal.pntd.0000228

[Tou59] – S. E. Toulmin (1959); *The Uses of Argument*. Cambridge University Press, 1959. Second edition, 2003. ISBN: 978-0521534833

[Sau11] – J. Sauro (2011). *A Practical Guide to the System Usability Scale: Background, Benchmarks & Best Practices*. ISBN: 978-1461062707

[Sho09] – D. Shotton (2009); *Semantic Publishing: The coming revolution in scientific journal publishing*. Learned Publishing 22: 85-94, 2009.
DOI: http://dx.doi.org/10.1087/2009202 -
Preprint available at:
http://purl.org/net/semanticpublication/Shotton_Semantic_publishing_evaluation.pdf

[Sho10b] – D. Shotton - *CiTO, the Citation Typing Ontology.* Journal of Biomedical Semantics 2010, 1 (Suppl. 1): S6.
Available at: http://www.jbiomedsem.com/content/1/S1/S6 [Cito v1.6]
DOI: http://dx.doi.org/10.1186/2041-1480-1-S1-S6.

[SKM09] – D. Shotton, K. Portwin, G. Klyne, and A. Miles (2009); *Adventures in semantic publishing: exemplar semantic enhancement of a research article.*
PLoS Computational Biology 5 (4), 2009.
DOI: http://dx.doi.org/10.1371/journal.pcbi.1000361 –
Available at http://www.ploscompbiol.org/doi/pcbi.1000361

[W3C09] – W3C OWL Working Group (2009); *OWL 2 Web Ontology Language Document Overview.* W3C Recommendation 27 October 2009
Available at: http://www.w3.org/TR/owl-overview/

Acknowledgements – Ringraziamenti

Al sempre disponibilissimo Fabio Vitali, per avermi sempre dato fiducia, per avermi proposto sfide affascinanti, e per avermi spronato a trovare il modo di superarle. Per la sua chiarezza e cordialità, ed anche per aver sempre trovato il tempo di seguire il mio lavoro.

A Silvio Peroni, per la cortesia, l’affabilità e la prontezza con cui mi ha accompagnato durante tutte le mie ricerche, e per l’enorme mole di know-how che ha sempre messo a mia disposizione. Ad Angelo di Iorio, a Francesco Poggi ed agli altri ricercatori del dipartimento, per la cordialità con cui hanno condiviso con me informazioni e risultati preziosi.

Ad Alex, per una intesa salda che dura ormai da una vita. A Daniele, per ogni gentilezza e per ogni stimolante conversazione. A Fabio, paragone di serenità, di virtù e di dedizione. A Giuliano, per tutta la carica di energia che sai sprigionare. A voi, per tutte le risate condivise e per l’amicizia che mi dimostrate: siete quattro, ma valete sette volte sette.

A Gloria per la fiducia e per ogni confidenza, per la tua esemplare intraprendenza e per la tua determinazione. Per ogni volta che le tue parole, il tuo sguardo e la tua vicinanza mi hanno aiutato a leggere dentro il mio animo.

A Enrico ed a Piergiorgio, per aver contribuito ad accrescere ed arricchire la nostra amicizia, nata proprio in questa facoltà, anche dopo che ognuno ha preso strade diverse. E per ogni chiacchierata rilassatamente nerd.

A *Ciri*, Giacomo, Paolo, Eugenio, Alessio, *Italo* e Silvia, Martina, Anna e tutti gli altri che con la loro simpatia e vicinanza fanno splendere le estati di Cervia.

A Leonardo, Ines, Irene, Piero, Mara e tutti i parenti fiorentini, per tutto l’affetto e la stima, contraccambiate, e per la vostra capacità di farmi sentire come a casa ogni volta che ci rivediamo.

Alla Combriccola, perché anche se così sparpagliati, vi sento incredibilmente vicini. Grazie di tante serate di divertimento e di tanti buoni consigli.

Ad Ilaria, per la gioia di un’affinità ritrovata. A Cecilia ed Anna, registe straordinarie ed instancabili, ed a tutti i talentuosi *Frà teatranti*, per l’estro e l’ispirazione. Ad Elois per le belle conversazioni e per le *DonCosciottate*.

Ed a tutti coloro che magari non leggeranno mai queste righe, ma la cui amicizia e vicinanza ha aggiunto gusto e significato al mio percorso di vita.