

## Introducing the Semlib project: semantic web tools for digital libraries

Christian Morbidoni <sup>a,1</sup>, Marco Grassi <sup>b,1</sup>, Michele Nucci <sup>c,1</sup>, Simone Fonda <sup>d,2</sup>,  
and Giovanni Ledda <sup>e,1</sup>

<sup>1</sup> Semedi Group, Università Politecnica delle Marche, Italy

<sup>a</sup> christian.morbidoni@gmail.com, <sup>b</sup> m.grassi@univpm.it,

<sup>c</sup> mik.nucci@gmail.com, <sup>e</sup> g.ledda@univpm.it

<http://www.semedia.dibet.univpm.it/>

<sup>2</sup> NET7, Italy

<sup>d</sup> fonda@netseven.it

<http://www.netseven.it>

**Abstract.** It is a common opinion that today's digital libraries (DL) can no longer be simple "expositions" of digital objects. Users should no more be passive readers, they need to interact with the library, add their annotations and tags, personalize their experience and collaborate with each other. Web 2.0 technologies, such as social bookmarking and online discussions, are already being applied in DLs to allow users to annotate digital objects. However, the lack of semantic structure of such annotations and a clear social model to share and aggregate community contributions makes it difficult to take full advantage of such collaboratively created knowledge.

The SemLib project aims at developing a modular and configurable annotation system that can be easily plugged into existing digital libraries in order to allow end-users as well as digital libraries content curators to produce meaningful and customizable aggregations of semantically structured annotations produced by communities. In this paper we introduce the SemLib project, discussing the principles and ideas behind the proposed annotation system, and present a prototypal implementation.

**Keywords:** Digital libraries, Semantic Web, Ontology, Data Model

### 1 Introduction

Nowadays, Digital Libraries (DL) are applied in many different contexts ranging from academic institutions to public libraries, archives, museums and industries. Traditionally DLs, as well as Web itself at its beginning, have been based on the expert paradigm according to which experts create content, DL experts provide access to it, and individual users consume it [1]. The advent of Web 2.0 has led to a Copernican revolution in the Web universe that has pushed users more and more toward its center and transformed them from passive content consumers into primary actors in data and metadata creation. As a result, tagging, linking and commenting resources have become common activities for Web users and a

valuable source of metadata that can be exploited to drive resource ranking, classification and retrieval. Annotation creation and sharing in a research context is an established practice since the pre-digital era, therefore its not surprising that in the last years the application of Web 2.0 models has been widely investigated in the context of digital humanities.

One of the ideas at the base of the research and development activities in this field is that user-created annotations, if properly structured and machine-processable, can enrich Web content and enhance search and browsing capabilities. Also allowing users write-access to the collection in DLs can provide users a more engaging experience and “capture diffuse and ephemeral information’ [2]. Supporting social annotations has proved to be an enabling feature for scholars to actually benefit from the digital world in their everyday work. Experiments conducted within the Discovery<sup>3</sup> european project have clearly shown that building structured information by annotating Web documents can be a valuable mean of representing aspects of the study process e.g. in e-learning or classroom activities. In [3], authors make a distinction between “social engagement’, where users annotate contents for their own purposes (e.g., to better organize study resources), and crowdsourcing, where social engagement is used within groups of users (communities) to “achieve a shared goal by working collaboratively together as a group’. If social engagement has been addressed to a certain extent by modern DLs, they rarely provide support to exploit such collected knowledge to improve libraries metadata, enrich contents, searching and linking different contents together. However, the topic is of high interest and not entirely new to the DLs community, as witnessed by interesting ongoing projects like Digitalkoot<sup>4</sup>, which is engaging people through online games, which create different kind of structured contents.

Basing on previous research and developments in Semantic Web oriented collaborative annotations (e.g.: SWickyNotes<sup>5</sup>), the SemLib project<sup>6</sup>, shortly presented in section 2, aims at developing a flexible, collaborative annotation system to address single scholars and unregulated user communities as well as curated “authoritative’ annotations to incrementally enrich digital contents.

In this paper we discuss the data and social model designed during the project’s first phase, presenting a preliminary prototype composed by experimental GUIs to create and exploit annotations and a triple-store based annotation server providing RESTful APIs to create, share and consume them. This paper is organized as follows: chapter 2 shortly presents the SemLib project; chapter 3 provides a brief overview of existing cutting-edge tools for resource annotation; chapters 4 and 5 discuss the annotation system architecture and chapter 6 demonstrates the experimental prototype.

---

<sup>3</sup> ECP 2005 CULT 038206 project, EC eContentplus programme

<sup>4</sup> <http://www.digitalkoot.fi/en/splash>

<sup>5</sup> <http://www.swickynotes.org>

<sup>6</sup> <http://www.semllibproject.eu/>

## 2 The SemLib project: use cases and challenges

The SemLib project, funded by the European commission, aims to improve the current state of the art in DLs, through the application of Semantic Web (SW) technologies for data representation and management. One of the main expected outputs of the SemLib project is the design and implementation of an annotation system able to enrich and interconnect digital objects published on the Web, specifically targeting DLs and multimedia archives owned by participating SMEs. As such objects are different, both from technology and from type of provided content points of view, the annotation system has to be designed to be technologically decoupled from the DL (adopting a RESTful architecture), based on established standards in data and metadata representation (such as RDF and Semantic Web ontologies), domain agnostic and adaptable or configurable for a variety of different use cases.

Resources annotation should be supported at different granularity levels in order to enhance resource fruition and interaction. With respect to this requirement, Web standards such as XPointer<sup>7</sup> and Media Fragment URI<sup>8</sup> are being used respectively to unambiguously identify text excerpts in Web pages and subparts of images and audio-video resources. In addition, as digital content can be remixed and replicated inside a DL (e.g. in summary pages or in composite, derivative digital objects), annotations should address not only entire web pages (has it happens for the majority of existing tools), but also small, atomic unit of content, like pictures, single text paragraphs, etc. Also, as SEMLIB aims at addressing different kind of users, they should be allowed to create different types of annotation, structured according to different levels of complexity and provided with diverse expressive flavor and semantics, from natural language comments to semantic tags coming from a restricted vocabulary to full subject-object-value statements based on domain ontologies. Moreover, SemLib should provide tools and models capable of leveraging the process of collaborative and community driven annotation of DLs items. This is an important requirement both for engaging small unregulated end-user communities and for providing effective tools for scholarly communities and DL maintainers to incrementally and collaboratively enrich the quality of metadata (e.g. basing on a crowdsourcing).

The several high level challenges, which have to be tackled in order to accomplish SemLib's goals, can be summarized as follows:

- supporting DLs in aggregating users in communities by providing properly configured tools and uniform domain vocabularies to create interoperable metadata;
- enabling a social model where end-users, as well as content owners, create, share and aggregate annotations into personal, curated “views” of the collective knowledge base;

<sup>7</sup> “XML Pointer Language (XPointer)” <http://www.w3.org/TR/xptr/>

<sup>8</sup> “Media Fragments URI 1.0” <http://www.w3.org/TR/media-frag/>

- providing DLs with visual tools and APIs to exploit the collective knowledge base, slice it accordingly to custom policies and make it available to end-users for searching, browsing and studying online content;
- developing annotation GUIs capable of efficiently handling the trade-off between the ease of use and the creation/management of meaningful structured data.

### 3 Related Work

In recent years, several annotation systems have been developed. These allow Web resource annotation providing different approaches and functionalities to be applied in different application scenarios. Some applications have been developed as extensions of popular social bookmarking tools, as Delicious<sup>9</sup> or StumbleUpon<sup>10</sup>, that count millions of registered users. Other tools have been more specifically conceived for creating and sharing annotations of digital resources for supporting e-learning, collaborative tasks, such as document reviews or editing, and in general working group cooperation. A complete review of the state of the art tools for Web resources annotation goes beyond the purpose of this work and can be found in [4]. Some of the most interesting applications are now presented and discussed, with regard to SemLib project.

EuropeanaConnect Media Annotation Prototype (ECMAP) [5] is an online media annotation suite based on Annotea [6] that allows users to extend existing bibliographic information about digital items like images, audio and videos. ECMAP allows free-text annotations and semantic tagging, enabling Linked Data resource linkage in the user annotation process, in addition to the possibility to draw user-defined shapes on images, maps and videos. Special support is also provided for high-resolution map images, enabling tile-based rendering for faster delivery, geo-referencing and semantic tag suggestions based on geographic location. ECMAP's annotation system presents several similarities with SemLib, in particular in the overall idea of supporting various types of resources. For this reason, it represents an important reference to identify the basic features that SemLib annotation system should have. LORE (Literature Object Reuse and Exchange) [7] is a tool developed inside the Aus-e-Lit Project "to enable scholars and teachers of literature to author, edit and publish compliant compound information objects that encapsulate related digital resources and bibliographic records". The OAI-ORE Resource Map<sup>11</sup> is used as the main data model and a specific ontology has been defined to describe the relationships among objects, called LORE Relationship Ontology. The annotation tool provides a graphical user interface for creating, labeling and visualizing typed relationships among individual objects, using terms from a bibliographic ontology. While the user interface is powerful, it probably lacks in simplicity and would not be so straight-

<sup>9</sup> <http://http://www.delicious.com/>

<sup>10</sup> [www.stumbleupon.com/](http://www.stumbleupon.com/)

<sup>11</sup> Open Archives Initiative Object Reuse and Exchange  
<http://www.openarchives.org/ore/0.9/primer#ResourceMap>

forward to understand for non-expert users. However, LORE is an interesting source of inspiration, since it presents several conceptual similarities with the SemLib annotation system. One Click Annotator [8] is a WYSIWYG Web editor for enriching content with RDFa annotations, enabling non-experts to create semantic metadata. It allows the annotation of words and sentences, referencing ontology concepts and creating relationships among annotated sentences. The Open Knowledge Foundations Annotator<sup>12</sup> project is developing a Web-based, open-source annotation tool that, from a user interaction perspective, has similarities to SemLib annotation tools. It uses XPath to anchor textual annotations and tags to specific parts of a page, providing also a server-side module for storing annotations represented as JSON data.

The idea of semantic tagging is implemented in Faviki<sup>13</sup>, a social bookmarking tool that allows the use of Wikipedia concepts as tags for Web pages. Tags are suggested using auto completion, allowing disambiguation, where the suggested items are ordered by their use frequency. It also proposes tags automatically extracted from the page using Zemanta<sup>14</sup>. Several Web annotation tools exist, which do not make use of structured semantics and handle simple textual annotations. Among those, Diigo<sup>15</sup> (Digest of Internet Information, Groups and Other stuff) is a social bookmarking application, which allows signed-up users to bookmark and tag Web pages. In addition, Diigo allows users to highlight any part of a Web page, attaching sticky notes to it. Diigo provides a simple but interesting annotations sharing model: annotations can be kept private, shared with a group within Diigo or forwarded to someone else with a custom link.

#### 4 Representing semantically structured annotations

Annotations represent a peculiar type of resources that is specifically conceived to add information to other resources. Annotations acquire therefore full significance in relation with the target resource and other contextual information, such as its author, its creation date and the vocabulary terms used. Properly structuring an annotation is therefore necessary at twofold level. On the one hand, an annotation represents an “information container”, whose structured metadata make contextual information explicit. On the other hand, an annotation includes an informative content that expresses a “knowledge bit” about annotated resources. Such knowledge is strongly domain dependent and, when uniformly structured by means of shared ontologies, can be in turn aggregated and used to increase content accessibility and interoperability.

Several ontologies have been developed in the last few years to provide a generic annotation structure and to improve interoperability among different annotation tools [9] [10]. The Open Annotation Collaboration<sup>16</sup> (OAC) project

<sup>12</sup> <http://okfn.org/projects/annotator/>

<sup>13</sup> <http://www.faviki.com/>

<sup>14</sup> <http://www.zemanta.com/>

<sup>15</sup> <http://www.diigo.com/>

<sup>16</sup> <http://www.openannotation.org/>

recently published the first specifications of the OAC data model [11], which at the moment seems to be the most accepted by the Digital Humanities community. In our first implementation the OAC ontology has been adopted and extended. It provides solid support for contextual metadata and for attaching annotations to involved Web resources. Such resources can be entire media objects or fragments (basing on Media Fragments and XPointer). Other ontologies, like the Annotation Ontology<sup>17</sup>, mostly used in bio-science community, have similar structure and comparable expressivity. In such ontologies annotations have a payload (body) that represents the user-created informative content. In practice, this is usually a Web page (e.g. a blog entry) or a textual comment.

One of the first issues we had to tackle was how to represent annotations that have an RDF graph as body. Even if this specific case is starting to be discussed within the community, it has not yet been regulated by the OAC specification that makes no assumption on the kind of body an annotation can have. It can be, for example, a plain text or a resource with its own URI. In RDF, there are different methodologies to model such a situation, from standard reification, to Content in RDF [12] or some ad-hoc solutions. As our primary goal is to prove how RDF triples produced by users can be aggregated using flexible criteria, we found it convenient to adopt named graphs to represent semantically structured annotation content. In our model, each annotation has an "oac:body" that is associated with a named graph, where the informative content is represented in triples. This allow us to exploit standard support for named graphs in SPARQL and in triplestores, thus querying and accessing only little "slices" of the entire collaborative knowledge graph. As discussed in detail later, this is very important to support personal views and target use cases.

The annotation storage is agnostic with respect to the ontologies used to represent the informative content of annotations. However, communities and DLs would greatly benefit from the uniformity of the data schema and vocabulary used in annotations. Our approach allows DLs to deploy specific configurations of the annotation tools provided, enabling users to transparently adhere to predefined data schemas. A range of pluggable entity spaces (like ontologies or thesauri) can be used in practice to provide users with a shared common vocabulary, enabling effective structured descriptions of any knowledge domain at different levels of expressiveness and with different structures. At the current stage, the annotation tool supports both "open", relatively flat vocabularies like Freebase (leveraging the reconciliation APIs<sup>18</sup>) and restricted controlled vocabularies and taxonomies, e.g. based on the SKOS model [13]. The following example in N3 syntax shows how an annotation and its informative content are represented in RDF.

**Listing 1.1.** An annotation example in N3 notation

```
// contextual metadata
ex:ANNOTATION-ID-1 a oac:Annotation ;
  rdfs:label "My test annotation";
  dcterms:created "2011-01-27 10:30:56";
```

<sup>17</sup> <http://code.google.com/p/annotation-ontology/wiki/Homepage>

<sup>18</sup> [http://wiki.freebase.com/wiki/Freebase\\_API](http://wiki.freebase.com/wiki/Freebase_API)

```

dcterms:creator ex:ChristianMorbidoni ;
oac:hasBody ex:ANN-BODY-ID-1 ;
oac:hasTarget http://example.com/1.htm ;
oac:hasTarget http://example.com/1.jpg .

ex:ANN-BODY-ID-1 a oac:body ;
rdfs:comment "This is an optional comment" ;
semLib:graph ex:graph-ann1 .

// informative content
ex:graph-ann1 {
  http://example.com/1.htm tags:hasTag http://www.freebase.com/view/en/
  pippo_baudo ;
  http://example.com/1.jpg foaf:depicts ex:PippoBaudo ;
  http://example.com/1.jpg ex:is-related-to http://example.com/1.htm.}

```

---

## 5 Addressing digital content and fragments

While the system is designed to work on generic web pages, there are some features that pose some requirements on DLs to better handle annotations. Two main issues have emerged from the analysis of the SemLib use cases and previous experiments.

DLs, like other web 2.0 applications, change over time. Presentation can be restyled and content can be re-organized. In addition, the same content (e.g. a page of an essay) can be accessible via different Web location (e.g. a summary page and the whole essay page). If we want annotations to remain consistent in such cases, in particular when they are shared in communities and not under a centralized control, we need a way of unambiguously identify atomic, annotable contents in DL Web pages. For this reasons the annotation system requires DLs to include RDFa tags to wrap atomic content, the granularity being opportunely tuned to address specific needs. Each marked content should have a resolvable URI associated, to which annotations are attached. This allows also for an annotation to be automatically associated to all pages that include the same content, as it might happen, for example, for derivative works.

As it happens for stand-off markup in general, the annotated content can change itself, e.g. typos gets fixed or corrections are made by editors. In such cases, annotations referring to fine granular fragments (e.g. sentences or words) can become invalid or simply no more addressable in the modified version. While editorial changes in some DLs result in new versioned objects, this is not a rule in practice, and preserving annotations through content modifications and revisions can be useful in publication workflows. In SemLib, this issue has not been fully addressed yet, but the model is “tolerant” to content change. We use XPointers to address DOM documents fragments of the marked content, but we also store the original annotated content, checking for broken annotations and possibly alerting the user when they are shown.

## 6 Sharing annotations

In our system users collect their annotations in notebooks, which are private by default but can be made public and shared with others. Notebooks are identified by dereferenciable URLs that applications can use to retrieve RDF-encoded annotations and relative metadata in different formats (RDF/XML, JSON, etc.). Being able to collect annotations in different notebooks helps users in organizing their work and in grouping annotations by topic or task, furthermore it allows users to make available to others subsets of their annotations.

Sharing a notebook is as easy as sharing its URL on the web, similarly to what happens for popular file sharing platforms. At the moment our system does not provide a social network itself where notebooks can be shared, rather the idea is that of relying on existing communication tools and social media that users are already familiar with. For example, if users want to share a notebook with a single person (e.g. a colleague), they can send the url via mail. In other cases, where users wants make a notebook of public domain, twitter, facebook or other social media can be used as publishing channels. This simple mechanism is general enough to enable different collaborative scenarios, but has limitation in terms of security: once a notebook is made public, each user that receive or find somewhere its URL can access the annotations. In later versions of the system, in order to better address real world use cases, owners of a notebook will be able to explicitly grant read and write permissions to other users of the annotation system. When a users receive an invitation to view a notebook (e.g. receiving the URL by mail) they simply click on it and, if signed in to the annotation system, they are redirected to the notebook web page where they can “activate” it. Each user has a personal preference page where he/she manage the list of active notebooks. When a notebook is active its content is visible to the user while annotated resources are browsed. In other words, by properly configuring the environment, users will be able to aggregate their and others annotations and explore them as custom semantic graphs.

## 7 Creating crowdsourced annotation collections

DL owners interact with the annotation system in two ways. On the one hand they deploy custom configuration of the system to deliver domain specific annotation tools to their users, by including Javascript libraries into their Web pages or suggesting shortcuts as bookmarklets to users. Using such annotation tools, communities of users, around single or federated DLs, can transparently produce metadata adhering to agreed schemas and vocabularies. This in turn makes the collectively produced data interoperable with the DL itself.

On the other hand, DLs owners/maintainers can act as content curators. As such, they might want to make their own annotation but also to select relevant end-user contributions, aggregate them and, perhaps implement a proper contribution submission workflow (as it happens, for example, with reviewed



publications). This would in turn enable a reward based scenario that can stimulate users to contribute. While SemLib does not implement any specific publication workflow, the intent is that of providing a framework that applications can base on to implement their own. In practice, content curators would act as “power users’ of the annotation system. They produce their own annotation as regular user do, and they can copy annotations from users-created notebooks to their own notebooks, preserving authorship and other contextual metadata. Such curated notebooks, along with their informative structured content, can be delivered back to users as trusted/official annotations, or directly imported to enrich the DL. In the first case a properly configured GUI, once embedded in the DL, could show the official annotations distinguishing them from users personal notebooks using some visual effect. In the second case, DLs can use simple REST APIs provided by the system to consume RDF encoded annotations and import them into their own database. Experiments in this directions are being made in SemLib, where some of the involved SME’s products are natively based on RDF.

## 8 Prototypal implementation

At the time of writing, the annotation system implementation has reached a prototype stage and, while collaborative features are still not fully implemented, it supports annotation of generic Web contents. It can be used in any existing Web site without modification to its structure and source-code, it is completely decoupled from the Web sites or DLs to be annotated and can be run by end-users through a dedicated bookmarklet. The system is made of two main macro-components: a client-side and a server-side component. When a user launches the bookmarklet, the client-side component is automatically plugged into the web page the user is currently browsing. The client-side component comprises a set of sub-modules developed in Javascript using the dojo framework<sup>19</sup> to facilitate cross-browser support. The client-side module implements the graphical user interfaces to create and browse annotations as well as modules dedicated to the communication with the server. Among these components the most important are the Fragment Handlers, the Resource Selectors and the annotation composer, called Pundit. Their interactions are depicted in Fig. 1.

During the annotation process, Fragment Handlers and Resource Selectors allow users to import different kind of resources into Pundit, where they can be used to compose structured annotations. Fragment Handlers and Resource Selectors can be configured by the system administrator to use specific vocabularies. Fragment Handlers assist users in selecting parts of content (eg. parts of a web page, parts of images, video frames, etc.) and turn them into actual addressable resources (e.g. using XPointer) to be used into annotations. Fragment Handlers also have the role of resolving resource fragments involved in existing annotations so that they can be highlighted in the page. Resource selectors have a similar role: they allow users to import into Pundit selected terms from

<sup>19</sup> <http://dojotoolkit.org/>

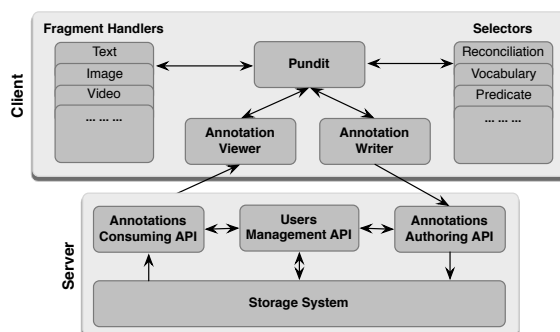


Fig. 1. Simplified architecture of the annotation system

a vocabulary or entity space. Resources are typed, where types are addressable resources as well (as it happens in RDF Schema). The current prototype implements two kind of selectors: one based on the Freebase reconciliation service and one presenting vocabs from a configurable domain taxonomy (e.g. conceptually equivalent to a SKOS vocabulary). Once resources are added to Pundit, users can build structured information in the form of triples (subject, predicate and object), by specifying semantically typed relations that links them, chosen from a predefined, configurable list or RDF properties. Pundit uses domain and ranges of such properties to assist the user and suggest proper relations for different kind of resources. At the current state, the discussed modules can be configured via simple JSON files. However, as the underlying model is an RDF Schema ontology, such a configuration could be easily extracted from a SPARQL endpoint. This might be useful if the DL exposes its data schema and resources via Semantic Web standard mechanisms such as SPARQL and Linked Data. This point will be addressed later in the project. The screenshot in Fig. 2 shows the prototypal user interface to compose semantic annotations. Users can select fragments of the page and import them into Pundit, where they can be dragged to populate statements. Users can also import resources from provided custom taxonomies (like the simple one in the illustration) or from Freebase, and again use them in annotations.

Once triples have been edited, user can save them to the Annotation Server, which is a modular RESTful web-service. It allows annotation storage, user authentication and management in addition to APIs for annotation authoring, consuming and sharing. Such RESTful APIs, partially inspired by previous works as the Annotea Protocol, allow users to create new notebooks and annotations supporting different data formats (e.g. RDF, JSON, etc.), to browse notebooks and related annotations and to personalize users views by activating public notebooks (e.g. shared by others). Such aggregations of activated notebooks can be then exploited by querying them and retrieving semantic data in the form of RDF triples. A typical use of such querying functionalities is that of retrieving all the RDF statements where a particular web resource (or a fragment of it) is involved. Sub-graphs obtained in this way can be immediately explored with existing Semantic Web aware tools. A prototypal annotation navigator, for example, has been implemented using Simile Exhibit.

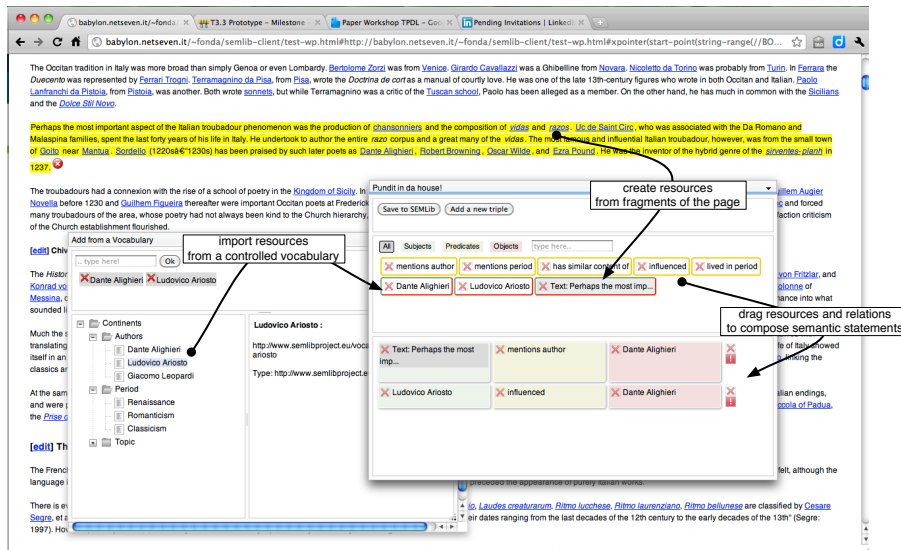


Fig. 2. A screenshot of Pundit in action

The storage module defines a completely generic interface, designed to support different kind of storage systems ranging from traditional relational databases to NoSQL databases (eg. RDF triplestores). In the prototype version, the storage is implemented using the Sesame triplestore<sup>20</sup> as this greatly simplifies handling and exporting RDF data. The storage module, besides keeping users annotations, stores also user profiles and related contextual information (e.g.: user’s metadata, user’s permissions etc.). The Annotation Server supports two single sign-on systems for users authentication, in particular, Open-ID<sup>21</sup> and OAuth<sup>22</sup>. Different authentication systems can be easily implemented developing dedicated plugins. Using single sign-on systems simplifies the integration of the annotation system with existing DL, which may already provide facilities for users authentication.

## 9 Conclusions

In this paper, we introduced the SemLib project, focusing on the proposed data and social model and explaining how those are expected not only to foster annotation sharing between DL communities and user engagement but also to allow the application of crowdsourcing paradigm in the creation of added value for the DLs. As proof of concept of our ideas, we also presented an early prototype implementation of the system discussing the experimental client-side GUIs for annotation creation and the server’s RESTful APIs for annotation storage, sharing and consumption.

<sup>20</sup> <http://www.openrdf.org/>

<sup>21</sup> <http://openid.net/>

<sup>22</sup> <http://oauth.net/>

As SEMLIB is an ongoing project, not all the features here described have been implemented yet, and several challenges are still open in improving annotation creation, visualization and sharing, which will be tackled in future releases of the annotation system. Also, the proposed system will be extensively tested on existing DLs of partner SMEs, which is expected to provide valuable feedbacks and to further boost the development process.

## 10 Acknowledgments

The research leading to these results has received funding from the European Union's Seventh Framework Programme managed by REA-Research Executive Agency<sup>23</sup> ([FP7/2007-2013][FP7/2007-2011]) under grant agreement n. 262301.

## References

1. DELOS, "The DELOS Digital Library Reference Model: Foundations for Digital Libraries, version 0.96". November, 2007.
2. R. A. Arko, K. M. Ginger, K. A. Kastens, and J. Weatherley, "Using annotations to add value to a digital library for education". [Online]. Available: <http://www.dlib.org/dlib/may06/arko/05arko.html>,
3. Rose Holley, "Crowdsourcing: How and Why Should Libraries Do It?", D-Lib Magazine, The Magazine of Digital Library Research. March/April, 2010.
4. M. Grassi, C. Morbidoni, M. Nucci, "Semantic Web Techniques Application for Video Fragment Annotation and Management", Proceedings of the SSPnet-COST 2102 PINK International Conference on "Analysis of Verbal and Nonverbal Communication and Enactment: The Processing Issues" pp.95-103. 2011.
5. B. Haslhofer, E. Momeni, M. Gay, and R. Simon, "Augmenting Europeana Content with Linked Data Resources", in 6th International Conference on Semantic Systems (I-Semantics), September 2010.
6. J. Kahan, M. R. Koivunen, "Annotea: An Open RDF Infrastructure for Shared Web Annotations", Proceedings of the 10th international conference on World Wide Web, Page(s): 623-632, 2001.
7. A. Gerber and J. Hunter, "Authoring, Editing and Visualizing Compound Objects for Literary Scholarship", Journal of Digital Information, vol. 11, 2010.
8. M. L. Ralf Heese, "One Click Annotation" in 6th Workshop on Scripting and Development for the Semantic Web, 2010.
9. Marja-Riitta Koivunen, "Annotea and Semantic Web Supported Collaboration". ESWC 2005, UserSWeb workshop. 2005 Marja-Riitta Koivunen, Ralph Swick, Eric Prud'hommeaux
10. "Annotation Ontology" <http://code.google.com/p/annotation-ontology/>
11. "Open Annotation: Alpha3 Data Model Guide" 15 October 2010 Eds. R. Sanderson and H. Van de Sompel. <http://www.openannotation.org/spec/alpha3/>
12. "Representing Content in RDF 1.0". W3C Working Draft 10 May 2011. <http://www.w3.org/TR/Content-in-RDF10/>
13. "SKOS Simple Knowledge Organization System Reference". W3C Recommendation. 18 August, 2009. <http://www.w3.org/TR/2009/REC-skos-reference-20090818/>

<sup>23</sup> <http://ec.europa.eu/research/rea>