Pundit: augmenting web contents with semantics

Marco Grassi, Christian Morbidoni and Michele Nucci
Semedia, Università Politecnica delle Marche, Italy

Simone Fonda
Net7 SRL, Italy

Francesco Piazza
Semedia, Università Politecnica delle Marche, Italy

Abstract

Scholars are using the Web every day to search, read, collaborate, and ultimately do their research. While some of the basic activities that the scholars do, such as reading and writing papers, are already well supported in the digital world, some essential scholarly primitives, such as annotation, augmentation, contextualization, and externalization, do not yet have clear support in terms of software tools. What scholars ultimately do during their research activity is to iteratively and collaboratively create new knowledge. With the advent of the Digital Humanities, we now have the opportunity—and technology—to capture at least a part of this knowledge and make it available as machine-processable data so to be better explorable and discoverable. In this paper, we present and discuss Pundit: a novel semantic annotation tool that enables scholars to collect, annotate, and contextualize Web resources. Deep-linking is used in conjunction with an RDF-based data model to allow granular selection of content (e.g. text excerpts, image fragments). Pundit aims at enabling scholars to produce meaningful machine-readable data that captures the semantics of their annotations. By providing a customizable annotation environment, where domain specific vocabularies can be loaded, and easy ways of integrating with existing Web archives or libraries, Pundit enables users to publish their annotations and collaboratively build a semantic graph. Such a graph can be consumed via HTTP APIs and standard SPARQL, thus allowing existing Linked Data applications to easily work with the data and Web clients in general to build specific visualizations.

Correspondence:
Dr. Marco Grassi
DII - Dipartimento Ingegneria dell'Informazione, UNIVPM Università Politecnica delle Marche, Via Brecce Bianche, 60131 Ancona, Italy
E-mail:
m.grassi@univpm.it

1 Introduction

The growing importance and acceptance of the Web, as a communication medium, and of digital tools, as valuable and even essential instruments to perform research, have lead to a revolution in the Humanities. Digital Humanities scholars use the Web every day to search, read, collaborate, and ultimately do their scientific research. While some of the basic scholarly activities, such as reading and writing papers, are already well supported in the digital world, some essential primitives, identified and discussed in literature (Unsworth, 2000; Palmer *et al.*, 2009), such as annotation, augmentation, and externalization, are not yet fully supported by existing software.

Literary and Linguistic Computing, Vol. 28, No. 4, 2013.
doi:10.1093/llc/fqt060 Advance Access published on 18 September 2013

A notable step forward in this context is the Open Annotation specification (Sanderson *et al.*, 2013), which proposes a technology independent, RDF-based language to encode different types of annotations on different types of media. Furthermore, Semantic Web technologies and Linked Data are widely recognized by the Digital Humanities community as solid foundations for representing, publishing and sharing data on the Web (Gradmann, 2010), as witnessed by recent efforts within the Europeana[1] network that have lead to the definition of the Europeana Data Model (EDM).[2] However, while a certain consensus is being reached at the data representation level, there is still a lack of tools that enable scholars to produce and use semantic data and annotations. Especially if we look for tools that work at a Web scale, rather that being tightly integrated into specific systems, contents, or knowledge silos.

What scholars ultimately do during their research is to create new knowledge. This could result from annotating, arguing, cataloguing, grouping primary and secondary sources, or from establishing novel connections among texts, works of art and cultural objects in general.

Such knowledge, in turn, serves as input to other scholars to elaborate new ideas and, again, to produce additional knowledge.

In our research we investigated what tools were needed to capture and make use of such knowledge on the Web. The main goal of our work is to allow scholars to collaboratively create machine-readable knowledge and to enable its exchange and reuse leveraging the Web of Data infrastructure. Such structured knowledge on top of the digital contents, can be thought of as a semantic overlay on top of the Web. In order to enrich it, scholars should be able to define meaningful relations among resources, as well as among fragments of them, by deep-linking to text and media. Here is where RDF and other Semantic Web technologies come in, allowing distributed information to be easily connected and merged.

This enables scenarios where scholars collaboratively create an interlinked semantic graph while annotating Web resources. In this way they create links from the Web of Documents (made of natural language text and digital media) to the existing Web of Data (made of structured resources datasets, like DBpedia[3] and Freebase[4]), thus contributing to what was called the Global Data Space (Heath *et al.*, 2011).

Data created in this way can be queried and consumed by a variety of applications, which already rely on such standards, to explore, visualize, and analyse the graph in many different ways, fostering the creation of specialized externalizations of such works. The annotation primitive plays a fundamental role in this scenario, as it explicitly involves the creation of new knowledge. In general, however, the digital tools that scholars have at their disposal very often do not support rich semantic annotations, being limited to comments and tags, and often do not leverage the potential of the Web of Data.

In this article, we present and discuss Pundit, a novel semantic augmentation tool developed in the Semlib Project[5] (Morbidoni *et al.*, 2011) and currently being enhanced and applied in the DM2E project,[6] which allows scholars to create semantically structured data out of their annotations on Web contents. In Pundit, we are experimenting with different user interfaces to establish typed relations among texts and media and to link them to the Web of Data as well as to controlled domain vocabularies (Grassi *et al.*, 2012).

Pundit supports scholars in the:

- *Augmentation (annotation) of online content*, ranging from simple comments to semantic tags to custom typed relations between different kinds of 'items', including text excerpts, images, and fragments of images;
- *Contextualization*, by creating links to the Web of Data or to ad-hoc domain vocabularies or taxonomies of entities, which can be followed by machines to reach additional data;
- *Simple aggregation*, by collecting items of interest from content available on the Web. These includes text excerpts, image fragments, and entire Web pages;
- *Collaboration*, since annotations created with Pundit can be shared and public collections of annotations are searchable. These functions have been recently aggregated into a separate Web application.[7]

- *Reuse,* since annotations are consumable via REST API or via SPARQL in the form of RDF data.

Annotations can be collected in notebooks and kept private or shared on the Web. Public annotations, along with their semantic data, are stored and served as RDF data with a REST API and SPARQL. This allows Pundit to be decoupled, whose role is to create semantic annotations, from other applications that might consume such annotations to produce specialized visualization or exploration tools, thus providing externalization of the knowledge produced by scholars. Presenting the results of a specific research activity (externalization), is something that cannot be generalized and addressed by a single system as needs and paradigms change consistently from case to case. That is why it is necessary to build on top of common APIs and a standard data model in order to create interactive, domain-dependent applications.

This article is organized as follows: Section 2 explains more in details the concept of semantic annotations that Pundit enables. Section 3 discusses related works, Section 4 discuss the Pundit data model and design principles, Section 5 showcases its main functionalities, and Section 6 provides demonstrative examples of specific applications built on top of the annotations created with Pundit. Finally, Section 7 overviews the results of qualitative and user-based evaluations of the system.

2 Semantic Annotations

While the term 'annotation' is often associated with the act of commenting or tagging, in Pundit annotations should be thought of as structured 'bits' of knowledge and they are represented as sets of RDF triples. A triple is a simple statement that has a subject, a predicate and an object. For example, a triple can state that a text (subject) has been written by (predicate) Immanuel Kant (object), or that a portion of an image (e.g. a stamp) depicts Kant, or again, that a region of an image (e.g. a manuscript) is transcribed in a given Web page. Annotations can have different meanings and express very different information about the digital objects, but they are represented with a uniform data model.

When different users independently create different annotations, these can still be seen as independent sets of triples (with a creating date and a creator/annotator) but can also be merged into a collaboratively built knowledge graph (as shown in Fig. 1). Each annotation always maintains its context, namely its authorship and the connection to the Web location where such annotation has been created. Each relation (or link) established between two Web resources is potentially a gateway for a machine to collect more data, thus 'expanding' the graph. This is shown in Fig. 1 where the resource Immanuel Kant is linked to Freebase that provides additional RDF data right away (e.g. birth and death places and dates).

3 Related Works

Since the advent of the so-called 'Web 2.0', Web content annotation has become a common practice. In particular, textual comments and plain tags are supported nowadays in several mainstream applications like Facebook and Flickr. Recently, a growing number of tools appeared that allow users to collect and share media from generic Web pages. Such tools, sometimes referred to as 'Web clipping tools' (as they mainly focus on content selection and aggregation more than annotation), share with Pundit the idea of mixing distributed contents. However, they do not attempt to create semantic connections among objects. Some examples are Clipboard,[8] Pinterest,[9] and Bundler.[10] SpringPad[11] can be classified as a clipping tool, but in addition it tries to extract and make some intelligent use of metadata (e.g. displaying geographic data on a map). Bookmarking and annotation tools explicitly targeted to scholars are Zotero[12] (Ritterbush 2007) and Mendeley.[13] While they are very different in their architecture and user interface, both focus on organizing and sharing bibliography and on providing a sort of clipping functionality to collect papers and, in some cases, to automatically extract basic metadata.

In the following, we revise the tools that are more related to our work, that is, those that implement structured semantic annotation, supporting users in

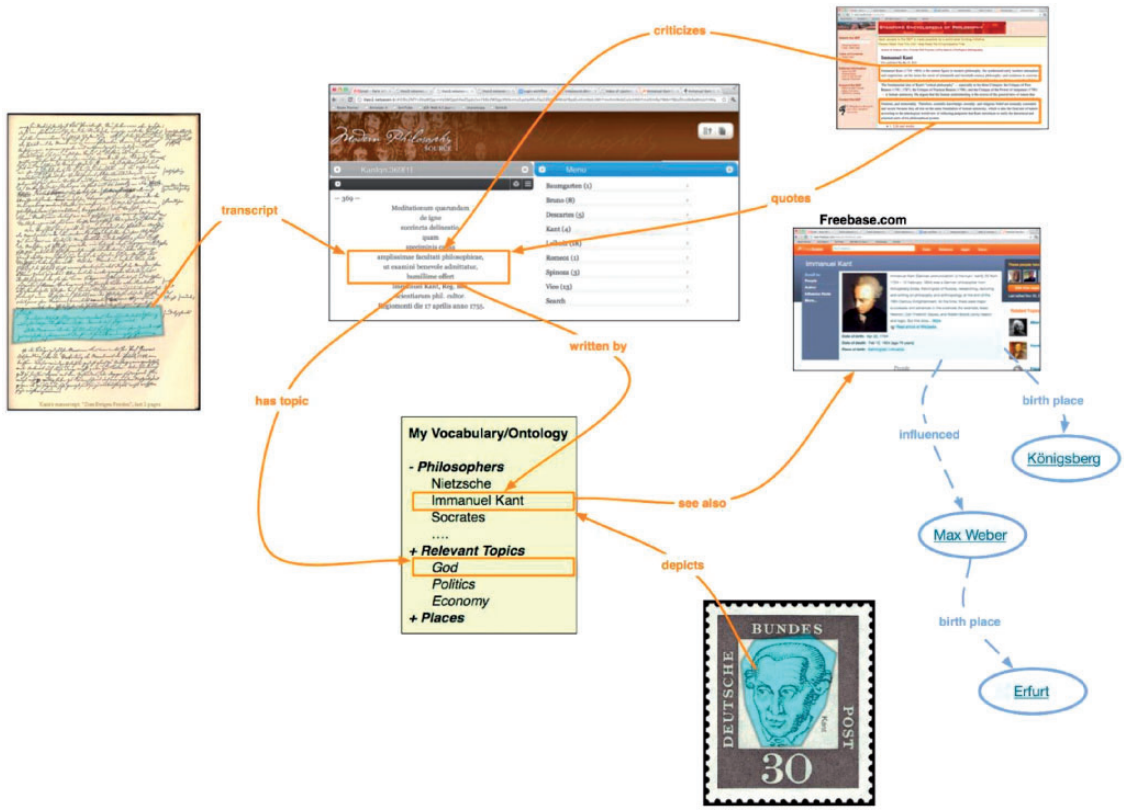

Fig. 1 Augmenting original content with semantically structured annotations

the creation of relations among objects, and, more in general, go beyond clipping and tagging. Such applications are usually based on Semantic Web technologies to represent data. However, an exhaustive state of the art in semantic annotation goes beyond the purpose of this paper and can be found in the literature (Uren, 2006; Andrews *et al.*, 2011).

The semantic tagging paradigm, which exploits publicly available Linked Data sources to retrieve unambiguous tags, has been implemented in Faviki.[14] Zemanta[15] uses natural language processing techniques to automatically extract semantic tags from pages. Europeana Connect Media Annotation Prototype (ECMAP) (Haslhofer *et al.*, 2010), an online media annotation suite based on Annotea (Kahan *et al.*, 2001), allows users to augment textual comment linking DBpedia resources.

Other tools also allow the use of restricted vocabularies or ontologies in the annotations. One Click Annotation (Luczak-Rŝsch *et al.*, 2010) and CWRC-Writer (Canadian Writing Research Collaboratory Writer) (Rockwell *et al.*, 2012) allow annotating entities in text excerpts by choosing among predefined categories (like person, location, etc.) or creating new ones. LORE (Literature Object Reuse and Exchange) (Gerber *et al.*, 2010), a Mozilla plugin developed inside the Aus-e-Lit Project, allows users to annotate Web page fragments adding textual comments and specifying tags selected from the AustLit thesaurus or entered as free text.

Some annotations tools enable also the editing of more expressive annotations other than textual comments or tags. LORE allows users to create the so-called 'compound objects', by bookmarking Internet resources and describing them using standard terms coming from a bibliographic ontology. A graphical user interface is provided to create and visualize typed relationships among individual

objects based on LORE Relationship Ontologies. CWRC-Writer provides an experimental interface for the creation of subject-predicate-object statements. If most of the tools focus on the annotation of text, some of those support the annotation of other types of digital items. ECMAP in particular permits also the annotation of maps, video fragments and images.

Although not based on Semantic technologies and not supporting semantic annotations, Open Knowledge Foundation (OKFN) Annotator[16] is worth mentioning in particular for its proposed architecture and vision that presents several similarities to Pundit. Like Pundit, it has been conceived as a JavaScript library that can be added to any Web page, both adding it into HTML and injecting it using a bookmarklet, to make it annotable.

4 System Overview

4.1 The annotations data model

An annotation can be split in two components: the annotation *metadata* contains contextual information that refers to the act of creating it, while the annotation *content* contains the triples that the user created within his or her annotation. The Open Annotation data model,[17] which recently evolved from two previous initiatives, namely the OAC (Open Annotation Collaboration[18]) model and the Annotation Ontology,[19] reached a mature state and is used in Pundit as backbone to represent the annotations.

As shown in Fig. 2, the `oa:hasTarget` property is used to specify the digital resource the augmentation involves. In Pundit it can be a text excerpt, an image or a specific region of an image. The extensible `oa:Selector` construct allows the application to actually identify and properly visualize the target in its context. Different types of selector are used. In particular, an XPointer is used to resolve text-fragments and images within a Web page, while a specific polygonal selector is used in the case of images region that provides the relative coordinates of the points composing the polygon within the image.

The `oa:hasSource` property is used to identify what resource the selector is relative to (e.g. an image in the case of polygonal selector, a Web page in the case of XPointer selector). The content of the augmentation is represented as triples and enclosed in a named graph, connected to the augmentation itself with the `oa:hasBody` property.

Using named graphs to enclose triples allows clearly identifying the triples that belong to a given annotation as well as to aggregate them into 'composite' graphs when needed, thus being able to query them using standard SPARQL.[20] For example, one could query for all the annotations whose target is a specific image and whose author is one (or more) specific user, and then extract all the resources that 'are depicted in' the image according to the selected annotation.

4.2 The Pundit server

The core component of the system is the Pundit annotation server that stores and manages users, annotations and notebooks. The server is an RDF-based store and supplies a REST API layer that provides a convenient protocol to consume and create annotations. Such API allows to 'slice' the overall annotation repository, e.g. querying single notebooks or by accessing all the annotations that involve a specific set of resources. Beside the REST API, each notebook, as well as the graph resulting from the merging of all the public notebooks, can be queried via SPARQL. While the REST API is targeted mainly to the Pundit client module, SPARQL endpoints enable a more flexible use of the annotations.

4.3 The Pundit annotation environment

The Pundit client is a JavaScript application that builds on top of the server API to provide an on purpose environment for users to create annotations on Web pages. Rather than implementing an ad-hoc annotation tool within a specific digital library or archive, Pundit is designed as a configurable tool to be plugged into existing Web sites in order to annotate heterogeneous contents from different Web sources. While this choice poses several challenges from a technical viewpoint, it has a strong rationale. Scholarly researchers, which are at the centre of our scenario, should not be bound to a specific archive or corpus, but rather they should be provided with a coherent environment

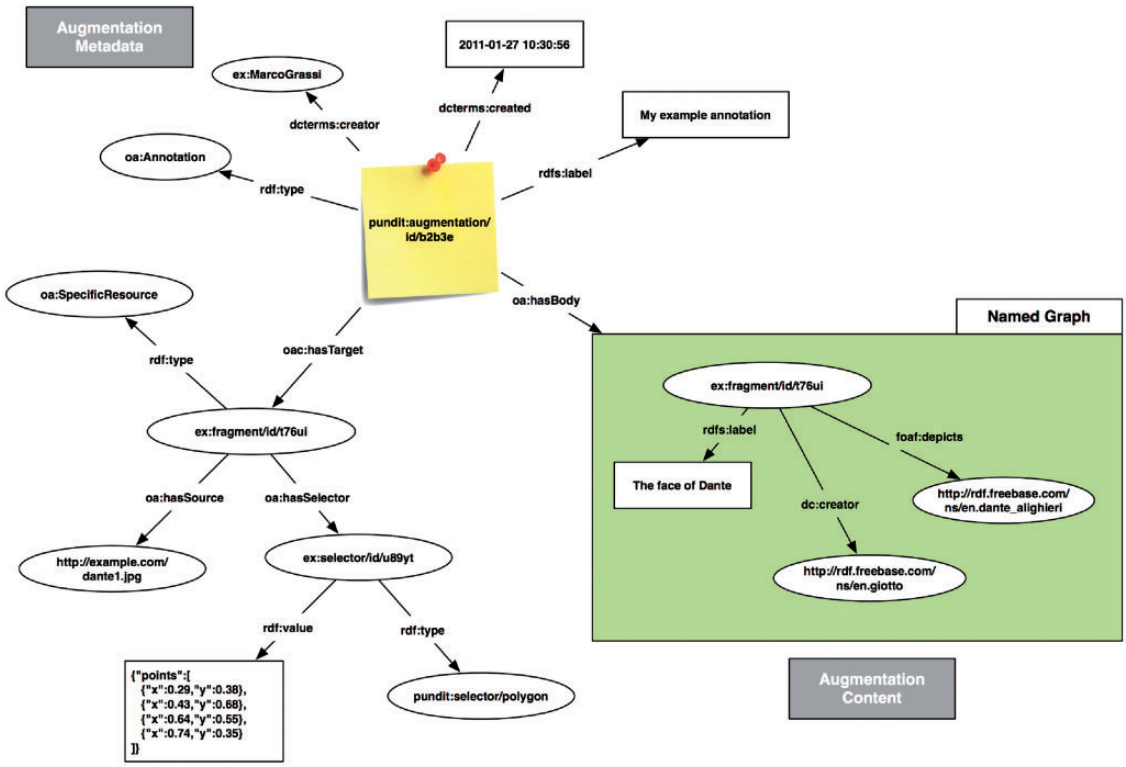

Fig. 2 Annotation data model

to build an annotation overlay on top of (possibly) the whole Web.

There are several ways in which Pundit can interact with Web contents:

- *As a JavaScript library*: Opportunely configured and added to Web pages, Pundit provides a GUI where visitors can edit annotations simply by accessing the pages. A single annotations server can be shared by several Web sites.
- *As a REST API*: In some cases, existing Web pages are not very easy to be annotated right away. They might, for example, include JavaScript code that prevents Pundit from working properly or applications based on closed technologies, such as Adobe flash. In these cases, a REST API is provided to feed content into Pundit to be annotated. In this way Pundit can be integrated simply by including hyperlinks to API calls (as happens in digital libraries like wittgensteinsource.org).
- *As a bookmarklet*: Scholars can also use Pundit independently from content providers. The Pundit bookmarklet, in an experimental state at the time of writing, is a bookmark link that users can quickly install in their browsers and start annotating the pages they visit on the Web.
- *As a browser extension*: Pundit browser extensions (for Google Chrome and Mozilla Firefox) enable users to automatically activate Pundit on every Web page they load. In addition, using the option panel, users can configure what version and configuration of Pundit they want to use and can restrict the activation of Pundit only on selected pages.

4.4 Addressing different communities

While common requirements exist for all the communities where Pundit has been evaluated so far, it appears also clear that different communities have different needs when it comes to creating and

exploring knowledge. This means that a good annotation environment needs to be finely tunable, with respect to at least the following three main aspects.

4.4.1 *Annotation vocabularies and relations*

Specific communities often use common thesauri or vocabularies to model their domains. In these cases, the scholars want to share and use the same terminology and entities in their annotations. A similar consideration is valid for the types of relations that the scholars expect to use to interconnect annotated items. Pundit addresses this issue in two ways. On the one hand, it includes built-in search for common Linked Data providers (among the others: Freebase, DBpedia, and Wordnet[21]) on the other hand it allows including custom taxonomies in the form of a simple JSON file. The same format is used to configure the predicates available to users. While, for implementation reasons, we are not using a standard RDF serialization, the conceptual model of the vocabularies is compliant to RDF. Automatically extracting a vocabulary from an existing RDF Schema[22] or OWL[23] ontology or from a SKOS[24] schema is relatively straightforward and has been done in some case studies, such as the Wittgenstein-Source.org digital library.

In Fig. 3a we show a simple custom taxonomy and, in Fig. 3b, the results of a search in Freebase. In both cases the items can be used in annotations and interconnected with text or images within a Web page.

4.4.2 *Annotation functionalities*

Considering that scholars have very diverse interests and matters of study, it is clear that the kinds of annotations they are interested in are different, ranging from simply commenting text passages, to mark entity occurrences in a text, to connect two similar images, and so on. Given the flexibility of the data model, all these use cases can be addressed with appropriate triples: the challenge is rather to provide a user interface that scholars can find easy to use. In Pundit, we address this by offering a set of annotation modules that can be activated or deactivated on the fly by reading a configuration file. However, we are far from claiming that all needed modules are already there. We rather focused on building an extensible system where new plug-ins can be added later.

4.4.3 *Knowledge visualization and exploration*

Externalization of the semantic data is perhaps the most difficult aspect to address in a generic way. This is way we think that relying on standards, like RDF, is important to foster the development of vertical, domain-specific applications. Pundit includes a basic annotation-browsing tool, called 'Ask', which solves the common issue of searching and looking into public notebooks. This also acts as an access point to specific visualizations that can be plugged into the system. In Section 7, a demonstrative vertical application is presented in the context of philosophy; other applications are being created at the time of writing and will be listed in the project Web site.

A vertical application bases on the definition of a data schema, which identifies the RDF properties and classes that are expected to be found in the annotations in order to be compliant. Once this is

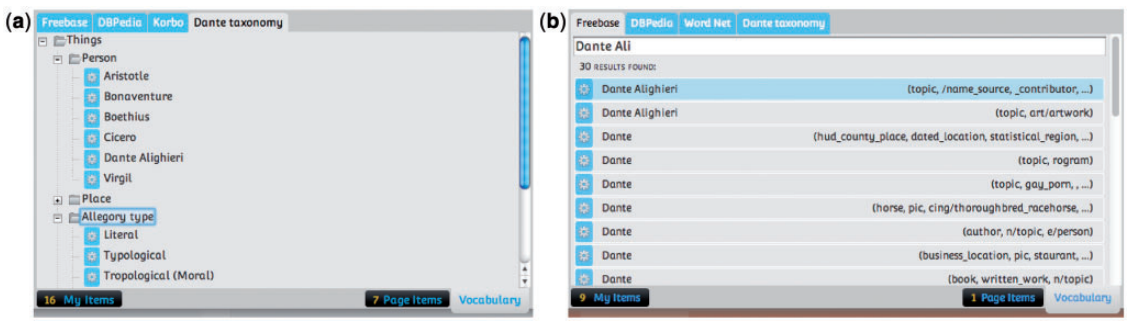

Fig. 3 Custom vocabularies and external entities stores

defined, Pundit can be configured or deployed with vocabularies that conforms to such a data schema and delivered to users by integrating into Web sites or providing a custom configured bookmarklet (as previously discussed). This simple workflow proved to be able to quickly deliver the tool to scholars in different domains.

4.5 Dealing with dynamic Web content

A common issue in dealing with dynamic Web content is that page presentation changes over time: pages can be restyled, with respect to layout and mark-up, and content can be re-organized and moved to different pages. Also, the content within a Web page has commonly a certain degree of granularity and the very same content can often be repeated in several pages. For example, an index page often displays a short document summary that contains only relevant paragraphs while another page could contain all of them. Similarly, the same image can appear in different pages with different resolution and dimension. Finally, along with the actual content (e.g. a document, a digitized manuscript, a picture), Web pages contain accessory content like navigation menus, advertising banners, and page headers.

This inherent instability of the Web makes it at least complicated to guarantee the stability of annotations in general. Pundit assumes that the URL of an annotated resource does not change over time and at the moment does not address issues such as retrieving past version of an annotated resource. Recently, these kinds of challenges have been addressed by Memento (Sanderson *et al.*, 2010), a framework based on the idea of extending the HTTP protocol with additional headers to support servers to provide different versions of the same resource.

4.5.1 *Named contents*

By marking-up fractions of Web contents, providers can explicitly expose them as an annotable named content and make Pundit 'work better' with the content. The following is an example of such a simple markup that can be included in Web pages, where the 'about' attribute, compliant with the RDFa[25] standard, contains a stable URL that identifies the named content.

```
<div class="pundit-content" about="http://
example.org/annotable-contents/xyz">
    {HTML CONTENT}
</div>
```

This simple markup is enough to enable interesting capabilities. Annotations made with Pundit will be attached to the named content and will be stable over changes in the rest of the HTML page. Additionally, the same named content can be embedded in multiple pages, possibly across distinct digital libraries. Annotations will be automatically available for all the instances of the same named content.

In Fig. 4, we illustrate a more complex use case where a single Web page contains multiple named contents. Content providers can re-mix images, text, and composite contents (e.g. a text with nested images) to produce dynamic representations without breaking the anchor points of the annotations.

5 User Interface

Pundit allows users to annotate several types of multimedia contents at different levels of granularity by providing specific modules to assist them in selecting the content and creating simple or complex annotations. The selection of text fragments, images, and polygonal image regions is illustrated in Fig. 3, while temporal and spatial video fragments annotation has been addressed in the Semtube prototype[26] (Grassi *et al.*, 2012), a Pundit extension to annotate YouTube videos.

Figure 5 shows a screenshot of Pundit. As one can see, annotations are marked in the page with yellow icons and can be viewed on the side bar. Colours and zooming effects are used to make it easier to associate an annotation to the text or the image it corresponds to.

In this section, we present the main built-in annotation modules that are currently available in Pundit and resulted from addressing the user requirements of the early adopter communities.

5.1 Comments and semantic tags

Commenting is, perhaps, the most common primitive scholars rely on to take notes when reading and studying. Pundit offers an easy way to add simple

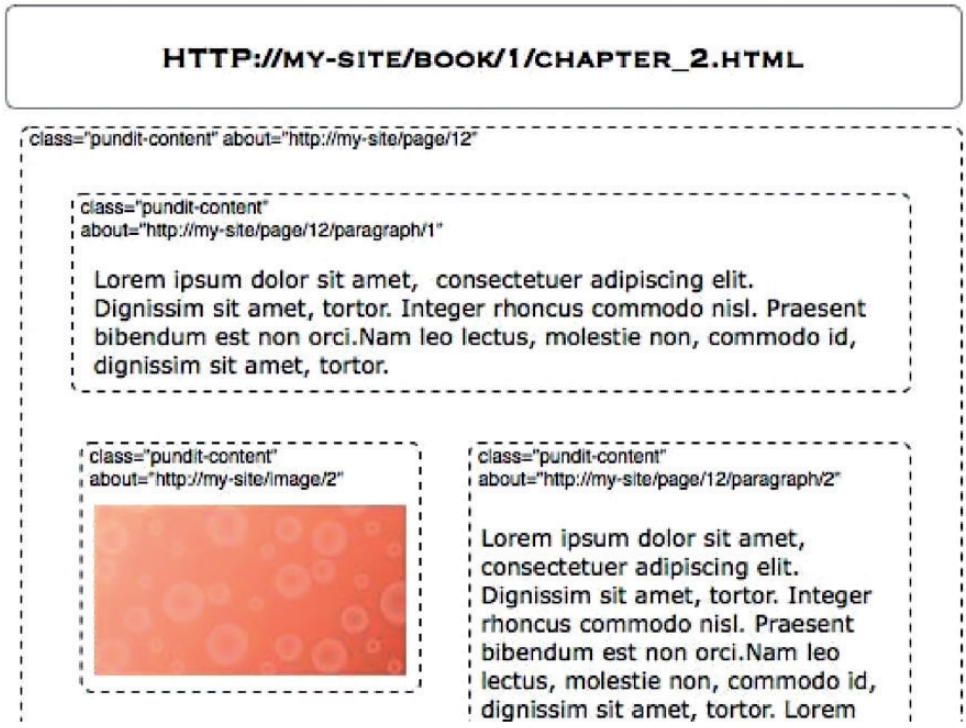

Fig. 4 Named contents mark-up

textual comments. However, the idea is to try to engage users in creating more explicit semantics. The common tagging paradigm is used to allow searching for related Linked Data resource and adding them to the annotations (Fig. 6). Another way of adding semantics is to use the 'recognize entity'. This provides a quick way to automatically get suggestions on entities that might be mentioned in a given text or in a comment entered by the user. Pundit currently uses external terminology extraction services like DBPedia Spotlight[27] and DataTxt.[28]

5.2 Connecting two texts

Connecting two text fragments, for example, to express a similarity or a citation between two sentences from the same or from different pages belonging to the same Digital Library, is a common case of annotation. Pundit provides a friendly Graphic User Interface (GUI) to create such kind of annotations (see Fig. 7). The predicate used to connect the two fragments can be pre-configured (to support repetitive annotations of the same kind, e.g. a quotation), or chosen by a list of possible relations.

5.3 The triple composer

The most generic and powerful way to create annotations is provided by the *Triple Composer*, shown in Fig. 8. This is a generic user interface to compose statements (with the subject-predicate-object form) that express relations among different kind of items. Such items can be text fragments, images of a Web page or entities selected from custom controlled vocabularies, from Linked Data entity stores like Freebase, DBpedia, and Wordnet.

The triple composer includes three boxes corresponding to the subject, the predicate and the object of a statement (triple). Items can be dragged from the vocabulary tabs into the subject and object boxes or chosen from a drop down suggestion panel that appears by clicking on one of the boxes. The triple composer also ensures the annotations to be semantically correct, by checking the rdf:type of subject

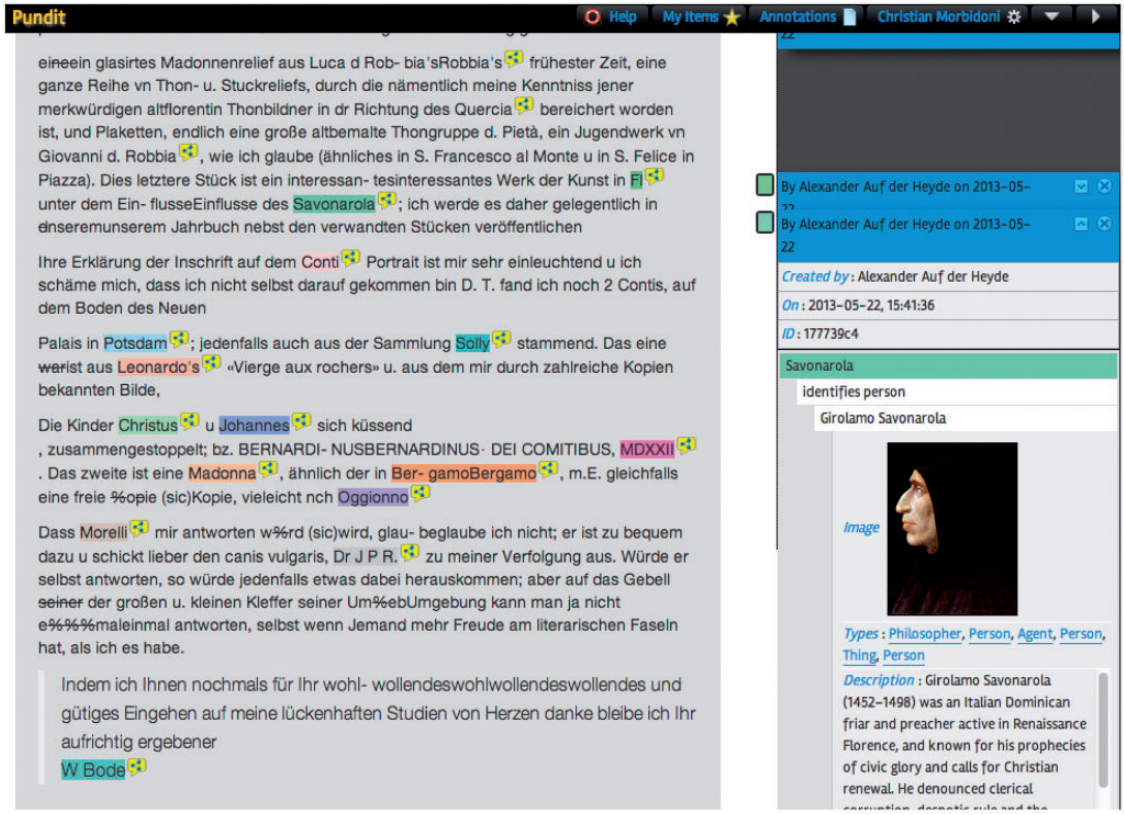

Fig. 5 A screenshot of Pundit: the annotations in the page are marked with yellow buttons and their content is shown in the side bar.

and object and the rdf:range and rdf:domain of the property. For example, it will accept as the object of a 'has birth place' relation only a location and not a person. Combined with the possibility of running Pundit on virtually every Web page (enabled by the bookmarklet version and the browser extensions) and with the ability of bookmarking items (provided by the *MyItems*), the *Triple Composer* allows users to create relations among Web resources of any type and to create relations among items that span multiple pages and multiple Web sites.

5.4 Notebooks and collaboration

The authors have recently conducted a survey on multimodal video annotation (Grassi *et al.*, 2011) that shows how collaborative annotation creation is not commonly supported in the majority of existing annotation tools. In Pundit a keep-it-simple approach has been followed that allows for open communities as well as for single scholars or small teams to work collaboratively. Annotations are collected in notebooks that can be private or public. Notebooks can be shared using an approach that resembles popular file sharing Web systems. Each notebook has a unique URL that can be passed to collaborators via e-mail or other channels. Such URL redirects users to a notebook activation page that will allow the corresponding annotations to be visible. This is illustrated in Fig. 9.

While at the moment this simple model keeps the permission management easier, clear directions in improving this aspect are indicated by mainstream Web collaboration platforms, such as Google Drive, where each item can be shared with multiple users.

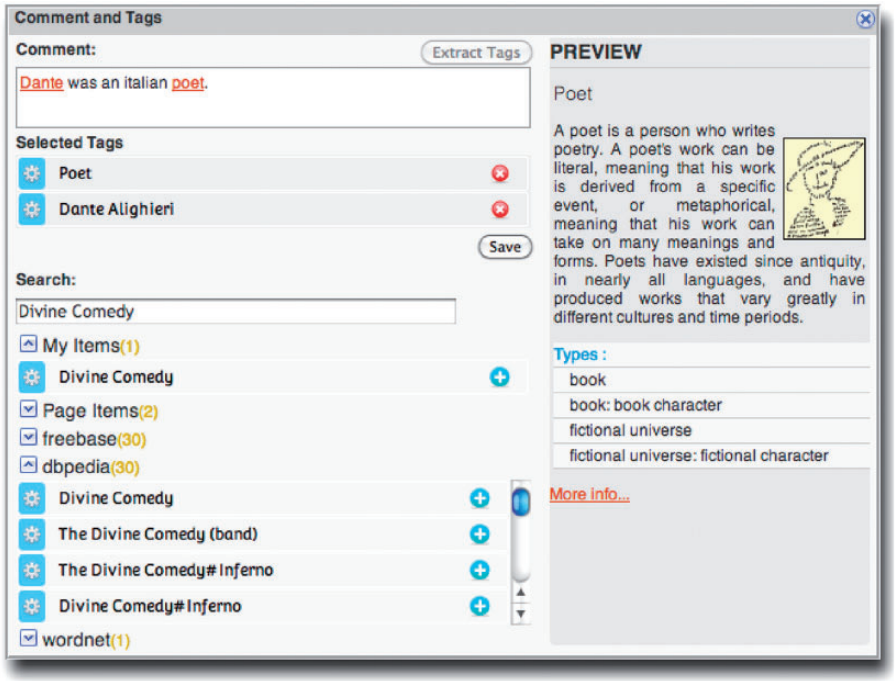

Fig. 6 Comment/tag panel

Fig. 7 Creating a connection between two text fragments with Pundit

6 Externalizing Semantic Annotations

As already remarked above, knowledge created by semantic annotations, as shown in Fig. 10, is stored as RDF named graphs and made available by means of SPARQL endpoints and RESTful APIs. In this way, knowledge can not only be easily accessed by external applications but also mixed and meshed with other data coming from external sources to be reused in different contexts from the one they originated from, as it was done in the SEMLIB project (Fossati *et al.*, 2012). This is something referred to as 'serendipity'. In this section we provide some demonstrative examples to prove how it is possible to build, on top of the same annotation tool and data model, vertical applications in diverse domains.

6.1 Edgemaps visualization: a demonstrative use case

Recently, the Digital Humanities community's attention has been captured by interactive graph visualizations such as Edgemaps (Dörk *et al.*, 2011). In a popular demo,[29] the influences among philosophers are shown in a map that helps in visualizing and exploring paths in the history of philosophy. The demo shows influence relations coming from Freebase, a well know general-purpose Linked Data repository. While for a 'generic' user such a visualization is enough, we can't probably say the

same for scholars that consider such relations as a matter of study and might probably ask: 'Why exactly do you say that Marx influences Gramsci?', 'What is the evidence of that in the actual primary sources?', 'Who said that?'.

Based on this idea, we customized Pundit by including a vocabulary of relations extracted from the CiTO ontology.[30] This includes predicates like 'cites' and 'quotes', as well as other more specific ones like 'discusses', 'cites as sources', 'agrees with', etc. A heterogeneous group of scholars then used the Pundit bookmarklet to annotate primary sources on Wikisource.org (which collects in an open data portal, non copyrighted materials from a variety of authors). Finally, we extended the code of the Edgemaps demo to load influences relations from users' augmentations made with Pundit instead of reading them from Freebase.

Each time the user creates a relation using some of the properties from the CiTO ontology, connecting two texts from different philosophers, a corresponding edge is created in the edgemap. Each time two philosophers are connected by an 'influenced by' relation, the corresponding annotations are shown so that the scholar can immediately get evidence of 'why the relation is there', as shown in Fig. 11. It is also possible to load multiple notebooks from different scholars, thus in fact enabling a collaborative scenario, where annotation authorship is always tracked back and each user can decide what notebook to see or trust.

The collaborative graph produced by scholars during a focus group is available at the project Web site.[31]

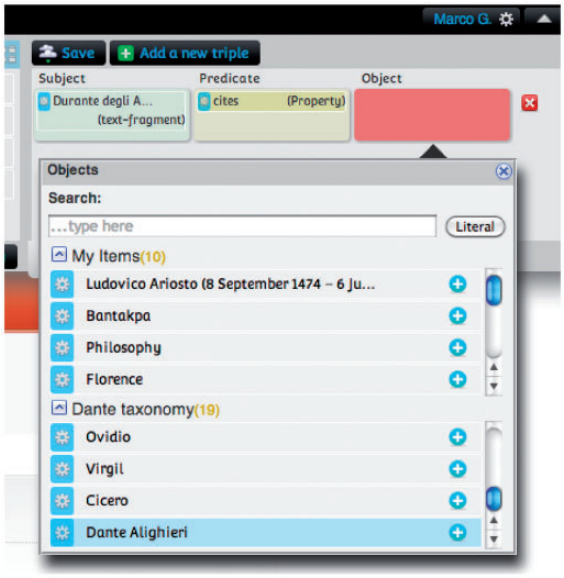

Fig. 8 The triple composer

6.2 Other examples

The same 'pattern' can be applied to several contexts and to address very diverse use cases. Let us consider for example an economics or politics journalist. By using pundit to annotate online journals he can

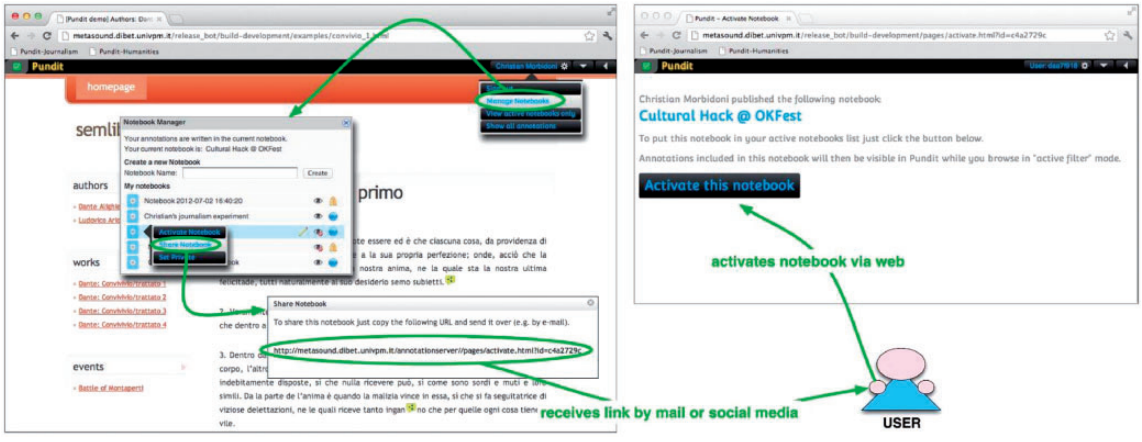

Fig. 9 Annotation sharing with Pundit

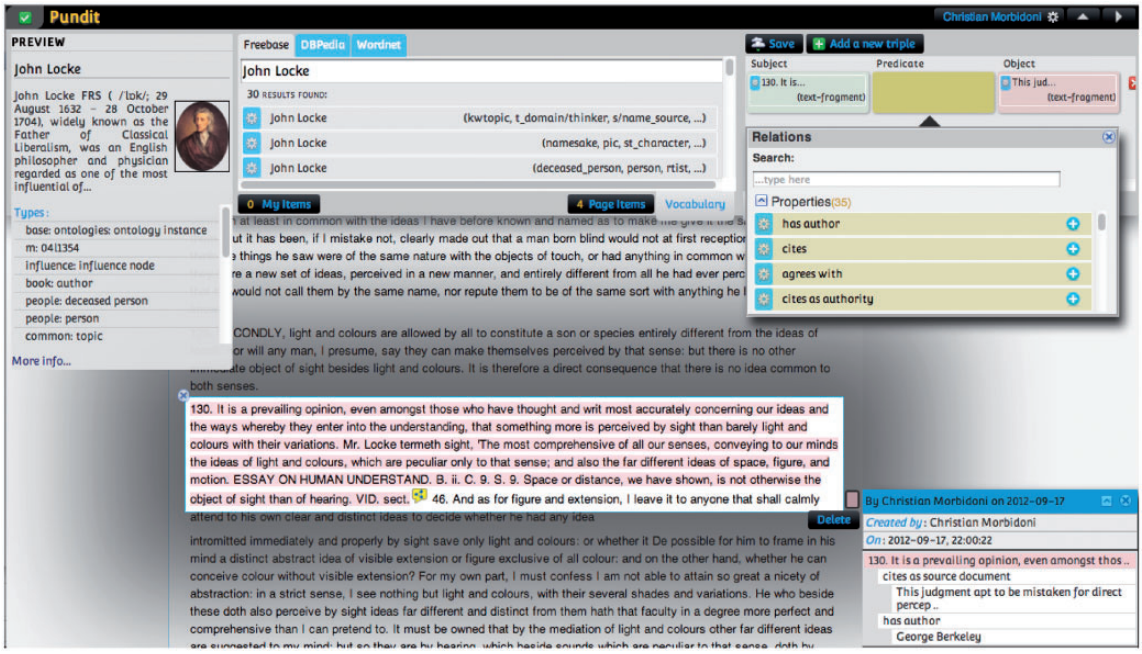

Fig. 10 Creating a semantic annotation by composing a triple (or statement) with Pundit.

quickly build data about public declarations from politicians or economists. We individuated a simple set of fields (or properties) to tag, describe and associate time to such declarations. Then we created a demonstrative Web application where annotated declarations are shown in a timeline and associated with the trend of a financial indicator in order to reveal existing connections among them (Fig. 12).

Another simple but effective application that can be used in several contexts (e.g. annotation of ancient maps, history, etc.) is that of graphing annotated objects from a given notebook in a timeline. For this example we used an open-source tool (TimeLine JS32) to let users create 'visual stories' by annotating Web images (Fig. 13). These demonstrative applications are available at http://thepund.it.

7 Evaluation

7.1 Qualitative comparison with existing tools

This section provides a qualitative comparison between Pundit and some of the most recent and relevant semantic annotation tools currently available. In Table 1, the comparison is performed following the same features used by Andrews *et al.* in their survey on semantic annotation tools (Andrews *et al.*, 2011). This makes possible to further extend the comparison also to the tools reviewed in their works. Some additional and more specific features have also been taken into account in Table 2.

If most of the existing tools have been created for content editing, Pundit has been specifically conceived for the annotation of Web content and to attach the annotation to the same page where it has been created, similar to ECMAP. In addition, Pundit provides the maximum level of expressivity in creating annotations (see Table 1). As CWRC-Writer, it supports all the existing approaches (tags, attributes, relations, and ontology) for creating structured annotations providing dedicated GUI. This makes user interaction more complex to handle (e.g. one needs to build statements using the Triple Composer and not simply typing a tag) but, as confirmed by user evaluation, this complexity seems acceptable in scenarios where researchers are used to complex annotations and need

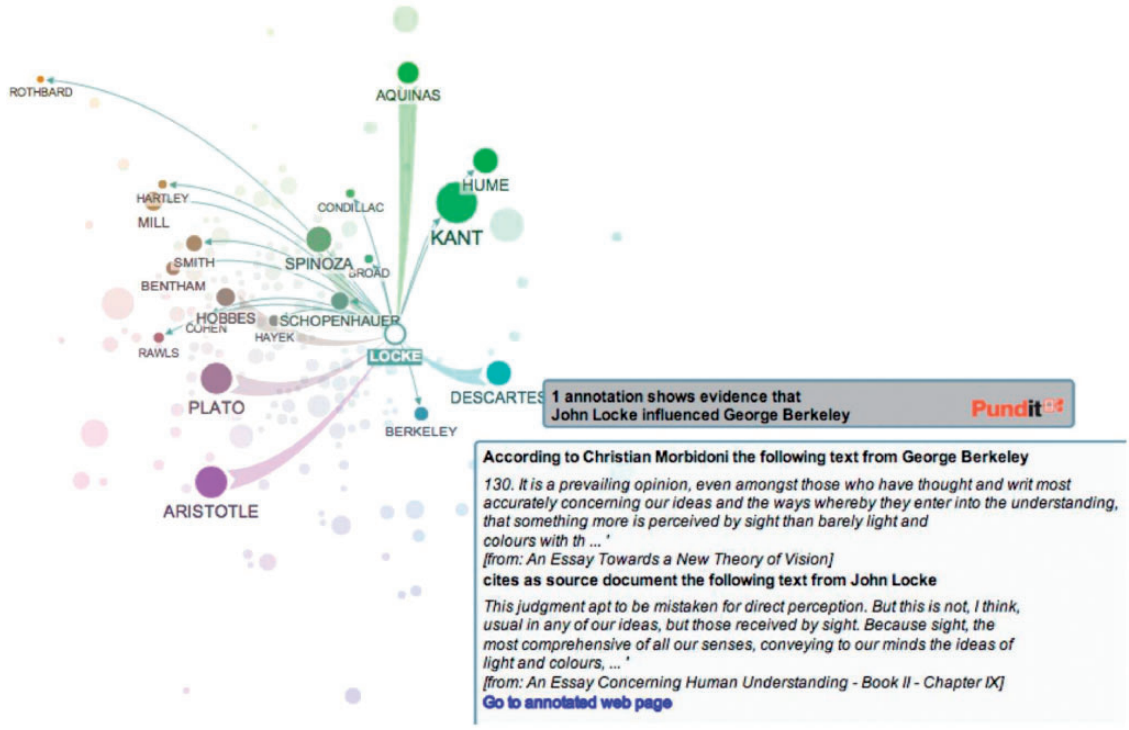

Fig. 11 Showing evidences of philosopher influence with a Edgemap Visualization

to encode non-trivial data. For less experienced users, a more intuitive GUI has been provided.

As remarked in Section 6, the main focus of Pundit is to enhance the creation of annotations and to foster their reuse by external applications. Accordingly, Pundit server provides a powerful and well-documented RESTful API to consume created data and metadata. It also exposes a SPARQL endpoint for each notebook. It is noteworthy that none of previously discussed tools provide such fine-grained access to created annotations. In Pundit, content and annotation search and visualization is mostly referred to external applications. For this reason, in contrary to other tools, like LORE, which also provides more advanced and graphical annotation visualization, Pundit integrates only contextual annotation visualization. However, using the server API it is possible to build general-purpose applications for notebook and annotation visualization. Nevertheless, as previously mentioned, a non-contextual generic annotation and notebook browser is provided with an external tool called 'Ask'. In addition, leveraging the graph nature of the data retrievable from the server, it is possible to combine the annotation information with data coming from other sources to create data visualization GUIs tailored for more specific application scenarios, as discussed in Section6.

Pundit is also the only tool that has been conceived to provide specific support for annotation sharing. In particular, Pundit relies on the mechanism of notebooks to allow the aggregation relevant information. In addition, the notebooks can be made available both to other users and third-party applications by means of a dereference-able URI. Such information can be easily consumed leveraging the uniform model provided by RDF and the clearly defined semantic provided by the use of pluggable ontologies. This constitutes the first step towards a more complete support for collaborative annotation that is going to be implemented in future releases of Pundit.

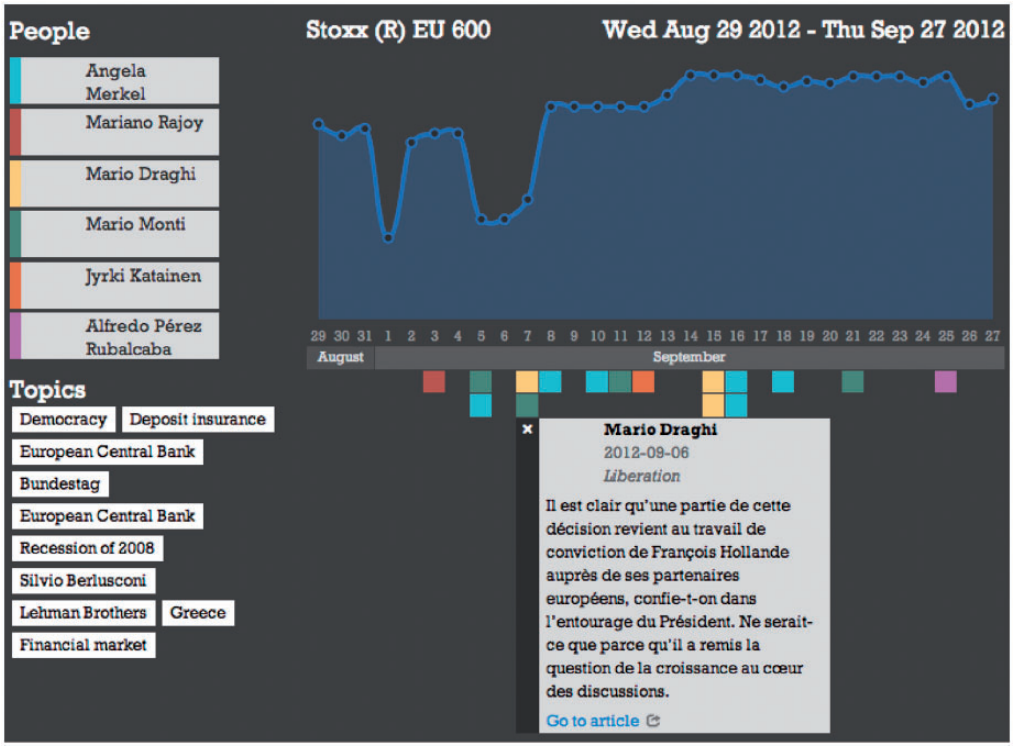

Fig. 12 A demonstrative visualization of annotated newspaper articles

Fig. 13 A demonstrative application: an annotation timeline depicting the history of the city of Ancona, Italy

Table 1 Tool comparison according to structural complexity, vocabulary type, and collaboration type

		Pundit	ECMAP	One click annotation	CWRC Writer	LORE	Annotator
Structural complexity	(Semantic) Tags	X	X	X	X	X	
	Attribute	X			X		
	Relations	X		X	X	X	
	Ontology	X		X	X	X	
Vocabulary type	No KOS		X	X	X	X	
	Authority file						
	Taxonomy	X					
Collaboration type	Single user, private use	X			X		X
	Single user, public use	X	X	X		X	X
	Collaborative	X					

7.2 User evaluation and further developments

Pundit is currently used as an experimental platform within the Agorà[33] and the DM2E[34] projects. Pundit is also being used within the Agorà project by a community of scholars to augment contents of two digital libraries[35] and two focus groups has been organized to let scholars try the software and provide feedback on functionalities and user interface. From these focus groups, it emerged that the augmentation paradigm based on triples is well understood by scholars, but that training is needed to start using Pundit effectively. In some cases, scholars would like to be able to use simpler interfaces to perform specific recurring types of augmentations, such as relating two texts from different essays, as it is too time consuming to create such annotations by using the *MyItems* and the *Triple Composer*. To this end the 'text-to-text' feature (described in previous sections) has been recently added to Pundit and is being successfully used in Agorà.

Another important requirement is to enable users to use precise bibliographic references and existing bibliographic databases as items in augmentations. To address this requirement, integration with BibServer[36] is under development and should allow scholars to use as vocabularies their own bibliographic base. All the participants to the focus groups agreed that being able to include custom vocabularies is an essential feature and it is being used successfully in Agorà to allow using existing taxonomies of concepts developed in the past by scholars. The need to edit vocabularies on the fly within the Pundit interface also has emerged.

Other requirements pointed out relate to the privacy setting and the access control. The simple model implemented by Pundit has to be improved to allow multiple scholars to write the same notebook and to allow sharing a notebook with a specific user. Other issues connected to the collaboration model emerged during open discussions. For example, in the DM2E project, scholars proposed a two-stage publication workflow whereby notebooks can be public or published. When published they should be undeletable in order to be stably cited and quoted by other scholars. However, these kinds of requirements can vary from user to user, which suggests that the model should be as flexible as possible. In general, the great majority of the scholars agree that semantic annotation as implemented by Pundit is useful for their research and that the ability to build digital externalizations is the crucial point.

Online user surveys were done during different events, ranging from public presentations at conferences and workshops, to hands-on sessions and focus groups with scholars. The users were requested to assign a score (from 0 to 5) to express their agreement with a set of statements, as well as to provide richer feedback by answering open questions. Ninety-eight users provided written feedback. The survey is currently ongoing and we expect to collect more questionnaires in the future. In the meantime we briefly discuss preliminary results.

Table 2 Tool comparison according to additional features

	Pundit	ECMAP	One click annotation	CWRC Writer	LORE	Annotator
Annotation purpose	Generic Web content annotation (special attention to DL)	Generic Web content annotation	Content editing	Content editing	Generic Web content annotation	Generic Web content
Representation of annotation metadata (context)	RDF/OWL OA data model	RDF	RDF	RDF	RDF	OA data model Different data stores No RDF
Representation of knowledge expressed by the annotation	RDF/OWL Ontologies (Named Graph)	Plain Text, Semantic Tags, geo tagging	RDF	RDF	RDF	Different data stores No RDF
Annotable resource types	Text-fragments, Images, Image fragments. Prototype for video and video fragment	Text-fragments, Images, Image fragments, video (temporal fragment), maps	Plain Text (copy/paste from different sources)	Plain text, TEI xml	Text fragment, Images	Text-fragments, Images
Vocabulary/ schema con- figurability	JSON configuration file	No	No	Ontology import and creation	No	No
Annotation storage	Dedicated annotation server based on Sesame Triplestore (inference).	Dedicated annotation server based on Mongo DB.	Dedicated annotation storage based on Triplestore	Local files	Dedicated annotation server	Dedicated annotation servers (different technologies). No triplestore
Annotation grouping and sharing	User per- sonal notebooks, flexible selection via API/SPARQL	No	No	No	No	No
Annotation consumption	Linked Data and W3C Standards (RDF, SPARQL), RESTful API	RESTful APIs. RDF exports.	Linked data endpoint	No	RDF	No
Annotation visualization	Contextual annotation visualization on every Web page. External application to explore Notebooks and Annotations (Ask)	Contextual visualization	Contextual and Faceted Visualization	Contextual visualization	Contextual Graphic visualization of compound object	Contextual annotation visualization
Annotation search	External application to explore Notebooks and Annotations (Ask the Pund)	Textual search	Faceted and textual search	No	Not clear	No
Annotation sharing	Collections of annotations can be made public or shared by a deferenciable URL	No	No	No	No	No
Installation	No. Works in every Web browser	No. Works in every Web browser	No. Works in every Web browser	No. Works in every Web browser	Mozilla Firefox via plugin	No. Works in every Web browser
Integration in Web pages	Add the Javascript to the page or lauch the Bookmarklet. Configurable using a JSON file.	No	No.	No	No	Add the Javascript to the page or lauch the Bookmarklet.

More than 70% of the users that saw Pundit for the first time, being trained only by a live demonstration before filling up the questionnaire, assigned a score greater than 3 to the statement 'Overall, I am satisfied with Pundit' and 'I think I would like to use Pundit frequently', so we deduce the software meets the expectations and is perceived as useful. The percentage increases (about 90%) where involved users were scholars participating in a focus group, who spent some time to experiment and perform assignments with Pundit.

About 20% of the overall users rated 4 the following statement: 'I found Pundit unnecessarily complex', and the statement 'I thought Pundit is easy to use' was rated less than 3 by 25% of the users. We are aware that Pundit is not a 'one click' tool as it involves complexity. The direction we think appropriate is to create specialized plugins for addressing limited and well-specified annotation types (e.g. annotating a period in time).

Survey results show that the text-to-text annotation and the entity extraction are the two more understood and easy to use functionalities, while the triple composer is the most difficult to use: the majority of users agrees that it is a powerful way of relating things but find that making annotations requires a considerable amount of time. However, this is mitigated by the availability of the suggestion panel that helps in speeding up the process.

About 40% of the users think that they would benefit (or have already done so) from having written user documentation for the tool. On the other hand we have clues that the annotation environment is not perceived as 'unnecessarily complex': only 14% of the users rated 3 the corresponding statement, while the rest rated it less than 3. More than 60% of the users rated 4 or 5 the statement 'I learned to use Pundit quickly', while the 13% rated it 1.

8 Conclusions

In this paper we presented Pundit, a novel Web-based collaborative semantic annotation tool targeted to scholars. In doing so we highlighted and proposed solutions for what we think are interesting aspects and challenges to be taken into account when dealing with knowledge creation and reuse in the scholarly domain. We then provided preliminary examples of how such collaborative knowledge can drive interactive visualization to bring added value to scholars.

The experimentation with Pundit in Digital Humanities is currently ongoing in two community projects (DM2E and Agorà) and the positive feedback collected so far encouraged us in further developing it and in developing demonstration cases of semantic augmentation to externalize knowledge. Furthermore, we think the concepts and methodologies implemented in Pundit can be applied in other contexts to support professionals, such as journalists and lawyers, in their research activity on the Web.

Acknowledgements

The research leading to these results has received funding from the European Union's Seventh .Framework Programme managed by REA-Research Executive Agency [SEMLIB - 262301 - FP7/2007-2013 - FP7/2007-2011 - SME-2010-1]. The research is also supported by the DM2E project, funded by the European Commission's 'ICT Policy Support Programme' (ICT PSP), agreement No. 297274.

References

Andrews, P., Zaihrayeu, I., and Pane, J. (2011). A classification of semantic annotation systems. *Semantic Web Journal*. http://www.semantic-Web-journal.net/content/classification-semantic-annotation-systems (accessed 23 August 2013).

Dörk, M., Carpendale, S., and Williamson, C. (2011). *EdgeMaps: Visualizing Explicit and Implicit Relations. Proceedings of VDA 2011: Conference on Visualization and Data Analysis, IS&T/SPIE,* 1–12, Jan 2011.

Fossati, M., Giuliano, C., and Tummarello, G. (2012). *Semantic Network-driven News Recommender Systems: a Celebrity Gossip Use Case. International Workshop on Semantic Technologies meet Recommender Systems & Big Data, Workshop at ISWC 2012.*

Gerber, A. and Hunter, J. (2010). Authoring, Editing and Visualizing Compound Objects for Literary Scholarship. *Journal of Digital Information*, **11**: 1–13.

Gradmann, S. (2010). Knowledge = Information in Context: on the Importance of Semantic Contextualisation in Europeana. http://pro.europeana.eu/c/document_library/get_file?uuid=cb417911-1ee0-473b-8840-bd7c6e9c93ae&groupId=10602 (accessed 23 August 2013).

Grassi, M., Morbidoni, C., and Nucci, M. (2012). A Collaborative Video Annotation System Based on Semantic Web Technologie. *Cognitive Computation*, **4**: 497–514.

Grassi, M., Morbidoni, C., Nucci, M., Fonda, S., and Ledda, G. (2012). Pundit: Semantically Structured Annotations for Web Contents and Digital Libraries. In Mitschick, A., Loizides, F., Predoiu, L., Nürnberger, A., and Ross, S. (eds), *Semantic Digital Archives 2012. Proceedings of the Second International Workshop on Semantic Digital Archives (SDA 2012)*. Paphos, Cyprus. September 27, 2012, CEUR-WS.org/Vol-**912**, urn:nbn:de:0074-912-6.

Grassi, M., Morbidoni, C., and Piazza, F. (2011). Towards Semantic Multimodal Video Annotation. Towards Autonomous, Adaptative, and Context-Aware Multimodal Interfaces. Theoretical and Practical Issues. *Lecture Notes in Computer Science*, **6456**: 305–16.

Haslhofer, B., Momeni, E., Gay, M., and Simon, R. (2010). Augmenting Europeana Content with Linked Data Resources, in 6th International Conference on Semantic Systems (I-Semantics) September 2010.

Heath, T. and Bizer, C. (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.

Kahan, J. and Koivunen, M. R. (2001). *Annotea: An Open RDF Infrastructure for Shared Web Annotations, Proceedings of the 10th international conference on World Wide Web*, pp. 623–32.

Luczak-Rŝsch, M., Heese, R., and Paschke, A. (2010). Future Content Authoring. *Nodilities – The Magazine of the Semantic Web*, **(11)**, 17–8.

Morbidoni, C., Grassi, M., and Nucci, M. (2011). *Introducing SemLib Project: Semantic Web Tools for Digital Libraries, Proceedings of the International Workshop on Semantic Digital Archives - sustainable long-term curation perspectives of Cultural Heritage held as part of the 15th International Conference on Theory and Practice of Digital Libraries (TPDL)*. Berlin. 29 September 2011.

Palmer, C., Teffeau, L., and Pirmann, C. (2009). Scholarly Information Practices in the Online Environment: Themes from the Literature and Implications for Library Service Development Report. Development. http://www.oclc.org/resources/research/publications/library/2009/2009-02.pdf (accessed 23 August 2013).

Rockwell, G., Brown, S., Chartrand, J., and Hesemeier, S. (2012). CWRC-Writer: An In-Browser XML Editor. *Digital Humanities 2012 Conference Abstracts*. Germany: University of Hamburg. 16–22 July 2012.

Sanderson, R., Ciccarese, P., and Van de Sompel, H. (2013). *Open Annotation Data Model.* Community Draft. February 2013. http://www.openannotation.org/spec/core/ (accessed 23 August 2013).

Sanderson, R. and Van de Sompel, H. (2010). Making Web Annotations Persistent Over Time. *JCDL 2010 - Digital Libraries - 10 Years Past, 10 Years Forward, a 2020 Vision*. Brisbane, Australia: The University of Queensland. 21–25 June 2010.

Ritterbush, J. (2007). Supporting Library Research with LibX and Zotero. *Journal of Web Librarianship*, **1**(3): 111–22.

Unsworth, J. (2000). Scholarly Primitives: What Methods Do Humanities Researchers Have in Common, and How Might Our Tools Reflect This? Symposium on Humanities Computing Formal Methods Experimental Practice. http://people.lis.illinois.edu/~unsworth/Kings.5-00/primitives.html (accessed 23 August 2013).

Uren, V., Cimiano, P., Iria, J., Handschuh, S., Vargas-Vera, M., Motta, E., and Ciravegna, F. (2006). Semantic Annotation for Knowledge Management: Requirements and a Survey of the State of the Art. *Journal of Web Semantics*, **4**(1): 14–28.

Notes

1. http://europeana.eu/.
2. http://pro.europeana.eu/edm-documentation.
3. http://dbpedia.org, http://it.dbpedia.org.
4. http://freebase.com.
5. Semlib Project: http://www.semlibproject.eu/.
6. DM2E project: http://dm2e.eu/.
7. Ask – http://ask.thepund.it.
8. https://clipboard.com/.
9. http://pinterest.com/.
10. http://bundlr.com/.
11. http://springpad.com/.

12. http://www.zotero.org/.
13. http://www.mendeley.com/.
14. Faviki: http://www.faviki.com.
15. Zemanta: http://www.zemanta.com/.
16. Annotator: http://okfnlabs.org/annotator/.
17. Open Annotation core specification: http://www.openannotation.org/spec/core/.
18. Open Annotation Collaboration: http://www.openannotation.org/.
19. Annotation Ontology: http://code.google.com/p/annotation-ontology/.
20. The SPARQL Query Language, http://www.w3.org/TR/sparql11-query/.
21. http://wordnet.princeton.edu/.
22. RDF Schema: http://www.w3.org/TR/rdf-schema/.
23. OWL 2.0: http://www.w3.org/TR/owl2-overview/.
24. SKOS: http://www.w3.org/2004/02/skos/.
25. RDFa 1.1 Primer: http://www.w3.org/TR/xhtml-rdfa-primer/.
26. Semtube: http://www.semedia.dii.univpm.it/semtube/.
27. DBPedia Spotlight: http://spotlight.dbpedia.org/.
28. DataTxt: https://spaziodati.3scale.net.
29. Marian Dörk EdgeMaps: http://mariandoerk.de/edgemaps/.
30. CiTO Ontology: http://purl.org/spar/cito/.
31. http://www.thepund.it/visualization-demos/philosophers-demo-howto/.
32. Timeline JS: https://github.com/VeriteCo/TimelineJS.
33. Agorà Project: http://project-agora.eu/.
34. DM2E Project: http://dm2e.edu/.
35. Daphnet Modern/Ancient: http://modernsource.daphnet.org/, http://ancientsource.daphnet.org/.
36. BibServer: http://bibserver.org/.