Pundit: Semantically Structured Annotations for Web Contents and Digital Libraries

Marco Grassi[a,1], Christian Morbidoni [b,1], Michele Nucci [c,1], Simone Fonda [d,2], and Giovanni Ledda [e,1]

[1] Semedia Group, Università Politecnica delle Marche, Italy
[a] m.grassi@univpm.it, [b] christian.morbidoni@gmail.com, [c] m.nucci@univpm.it,
[e] g.ledda@univpm.it
http://www.semedia.dibet.univpm.it/
[2] NET7, Italy
[d] fonda@netseven.it
http://www.netseven.it

Abstract. This paper introduces Pundit[3]: a novel semantic annotation tool that allows users to create structured data while annotating Web pages relying on stand-off mark-up techniques. Pundit provides support for different types of annotations, ranging from simple comments to semantic links to Web of data entities and fine granular cross-references and citations. In addition, it can be configured to include custom controlled vocabularies and has been designed to enable groups of users to share their annotations and collaboratively create structured knowledge. Pundit allows creating semantically typed relations among heterogeneous resources, both having different multimedia formats and belonging to different pages and domains. In this way, annotations can reinforce existing data connections or create new ones and augment original information generating new semantically structured aggregations of knowledge. These can later be exploited both by other users to better navigate DL and Web content, and by applications to improve data management.

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

1 Introduction

Since the advent of the digital era, cultural heritage preservation has been increasingly dealing with the conservation and the management of digital contents in Digital Libraries (DLs). These contents can be the digital reproduction of non-digital artefacts and manuscripts or more and more often born-digital multimedia contents. As this amount of data multiplies everyday faster and faster, its proper classification and management is becoming an increasingly complex task but nevertheless more and more crucial to make such information effectively consumable.

[3] www.thepund.it

With such purpose, in recent years, Semantic Web technologies and guidelines have been finding growing application in DL libraries scenario. RDF data model is currently employed by Europeana[4] initiative to aggregate independently provided digital contents. Several DLs have also made their data publicly available over the Web following the Linked Data recipes to join the giant and interconnected knowledge base of the Linked Open Data cloud [1]. Several efforts have also been done to introduce common accepted ontologies and schema for metadata encoding of DL contents, as BIBO[5], OAI-ORE[6] and Europeana Data Model[7].

Since the advent of the Web 2.0, the capability to annotate Web content, even with simple approaches based on plain-text comments or tags, has been growingly recognized as an highly beneficial feature not only for the user, making the navigation a more engaging and profitable experience, but also for the content providers that can leverage on user created metadata to better classify and search their published resources. Nevertheless, in several research scenarios, the annotation of DL contents and more in general of Web resources represents a fundamental activity daily performed by scholars. Also, in most of these cases an higher level of accuracy and granularity is typically required in the annotations to encode information about multimedia resource fragments, such as text excerpts or image regions, according to specific controlled vocabularies.

Most of the existing systems rely on simple textual comments and tags. Such approach is relatively easy to implement and very intuitive for users but it suffers from several issues related with the ambiguity of natural language and limits the accuracy and the efficiency of resource classification and retrieval. The founding idea of this research is that, if properly structured and provided with clearly-defined and machine-processable semantics, annotations can constitute themselves a primary information which can enrich the original contents and provide added value for other users as well as for third party applications. On this line, Semantic Web technologies are employed to foster the flexibility and interoperability of user created annotations, to promote their linkage with the Web of Data and to permit their reuse by other people or applications beyond the context they originated from.

This paper introduces Pundit[8], a novel semantic annotation tool, developed in the context of the Semlib project[9] [1]. Pundit has been conceived not only to permit the annotation of generic Web pages and multimedia resources but to be also specifically tailored to and integrated in existing DLs. Pundit provides support for different types of annotations, ranging from simple comments to semantic links to Web of data entities, to fine granular cross-references and citations. Pundit can be configured to include custom controlled vocabularies

[4] http://www.europeanaconnect.eu/

[5] http://bibliontology.com/specification

[6] http://www.openarchives.org/ore/1.0/primer

[7] http://pro.europeana.eu/edm-documentation

[8] Pundit: http://www.thepund.it

[9] http://www.semlibproject.eu/

and has been designed to enable groups of users to share their annotations and collaboratively create structured knowledge. This paper is organized as follows: Sec. 2 shortly provides a brief overview of related works; Sec. 3 explains the proposed data model for the annotations; Sec. 4 discusses Pundit prototype and its main functionalities.

2 Related Work

Nowadays, Web content annotation has become a common practice users are familiar with. In particular, textual comments and plain tags are supported in several mainstream Web applications like Facebook and Flickr.

In recent years, a growing number of tools have also been specifically created to allow user to annotate digital resources. Some of those found on Semantic Web technologies to improve the efficiency and the productivity of user created annotations. An exhaustive state of the art in Semantic Annotation goes beyond the purpose of this paper and can be found in literature [3], [2]. This section briefly discusses some of the most interesting annotation approaches implemented in the recently developed semantic annotation tools.

Semantic tagging paradigm, which exploits publicly available Linked Data knowledge bases to retrieve unambiguous concept to use in resource tagging, has been implemented in several application. Faviki[10] is a social bookmarking tool that uses DBpedia concepts as tags for Web pages. Zemanta[11] uses natural language processing techniques to automatically extract semantic tags from pages. Europeana Connect Media Annotation Prototype (ECMAP) [7], an online media annotation suite based on Annotea [8], allows to augment textual comment linking Dbpedia resources.

Other tools also allow the use of entities belonging to restricted vocabularies or ontologies in the annotations. One click annotation [9] and CWRC-Writer [10] allow to annotate entities in text excerpts by choosing between predefined categories (as person, location, etc, ...) or creating new ones. LORE(Literature Object Reuse and Exchange)[11], a Mozilla plugin developed inside the Aus-e-Lit Project, allows to annotate Web pages fragment adding textual comments and specifying tags selected from the AustLit thesaurus or entered as free text.

Some annotations tools enable also the creation of more expressive annotations other than textual comments or tags. LORE allows to create the so called "compound objects', by bookmarking Internet resources and describing them using standard terms coming from a bibliographic ontologies. 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-object-predicate statements.

If most of these 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.

[10] http://www.faviki.com/

[11] http://www.zemanta.com/

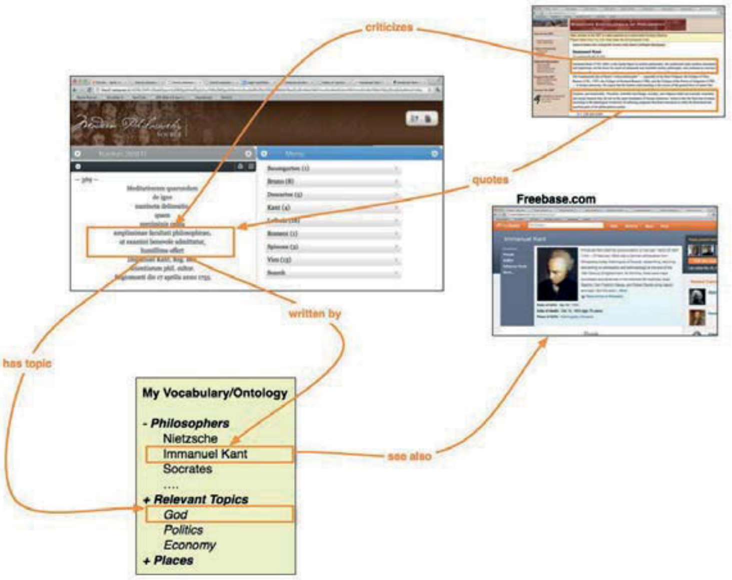


Fig. 1: Creating semantically structured aggregations of knowledge by means of annotations

3 Semantically structured annotations

The main idea in Pundit is that of enabling users not only to comment, bookmark or tag Web pages, but also to create semantically structured data while annotating, thus enriching the so called Web of Data. The ability to express semantically typed relations among resources, relying on ontologies and specific vocabularies, not only enables users to express unambiguous and precise semantics, but also, more interestingly, fosters the reuse of such collaboratively created knowledge within other Web applications. In Pundit annotations contain a set of RDF triples that connects annotated object (e.g. text excerpts) among each other and with entities in the Linked Data Web. Thanks to the nature of RDF data model (where triples can be flexibly combined to form arbitrary graphs) and to the use of URIs as identifiers for both entities and annotated objects, different annotations independently authored by different users, can be combined to form a semantic network that applications can retrieve via SPARQL endpoint and dedicated REST API. The resulting RDF graph is exemplified in Fig. 1.

Annotations acquire full significance in relation with the target resource and other contextual information, such as their author, their creation date and the vocabulary terms used. Such metadata are encoded in RDF relying on the OAC ontology, which provides a framework to represent annotations context in a standard way. The OAC data model uses the `aoc:hasTarget` property to define the target to which the annotation is attached. The target of an annotation can be entire Web pages or media objects, or their fragments (basing on Media Fragments and XPointer standards). Annotations also contain a payload, defined by the `oac:hasBody` property, which represents the user-created informative con-

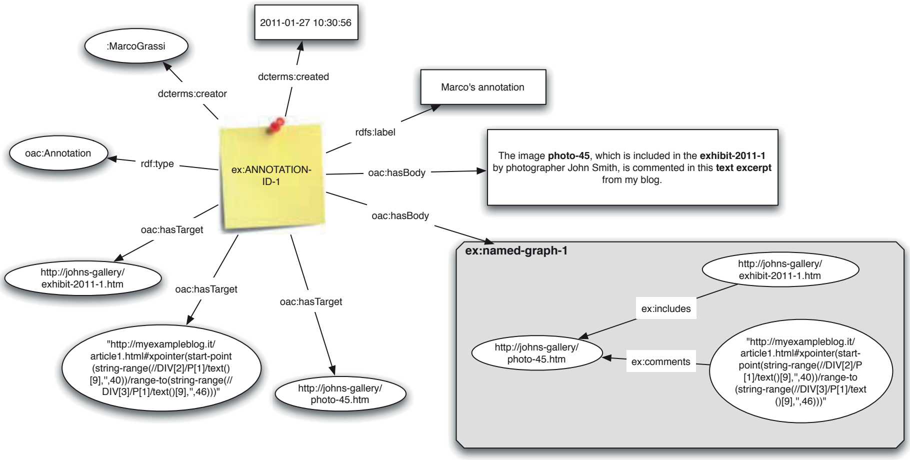

2011-01-27 10:30:56
:MarcoGrassi
dcterms:created
dcterms:creator
Marco's annotation
rdfs:label
oac:Annotation
rdf:type
ex:ANNOTATION-ID-1
oac:hasBody
The image **photo-45**, which is included in the **exhibit-2011-1** by photographer John Smith, is commented in this **text excerpt** from my blog.
oac:hasBody
oac:hasTarget
http://johns-gallery/exhibit-2011-1.htm
oac:hasTarget
oac:hasTarget
"http://myexampleblog.it/article1.html#xpointer(start-point(string-range(//DIV[2]/P[1]/text()[9],'',40))/range-to(string-range(//DIV[3]/P[1]/text()[9],'',46)))"
http://johns-gallery/photo-45.htm
ex:named-graph-1
http://johns-gallery/exhibit-2011-1.htm
ex:includes
http://johns-gallery/photo-45.htm
ex:comments
"http://myexampleblog.it/article1.html#xpointer(start-point(string-range(//DIV[2]/P[1]/text()[9],'',40))/range-to(string-range(//DIV[3]/P[1]/text()[9],'',46)))"

Fig. 2: The representation of an annotation using a named graph

tent. At the time of writing, the OAC (Open Annotation Collaboration) model[12] has been merged with the Annotation Ontology[13] to form the OA (Open Annotation) specification[14] that only recently reached its first stable state and is considered the de facto standard for representing annotations on the Semantic Web. It's worth to remark that, among other news, OA explicitly validates the use of named graphs that has been already put in place in Pundit. At the time of writing, full compliancy with such a specification is currently under development. In Pundit, named graphs are used as "bodies' (using the OA jargon) of annotations, which in our system is composed by RDF triples itself. This allows to keep separated statements belonging to different annotations, while still being able to aggregate them into "composite' graphs and query them using standard SPARQL language. 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 "comments' the image according to the selected annotations. Fig. 2 illustrates how annotations are represented in our system.

While the OAC ontology is used to represent contextual information, the semantic content cannot be represented based on a fixed ontology. Different users communities operating in specific domains need specific shared vocabularies (ontologies) of terms and relations that they can use in annotations. At RDF data storage level, the system is therefore agnostic with respect to the domain ontologies used in structuring annotation informative semantic content, and specific configuration at application level can be used to build an ad-hoc vocabulary for each community addressed. Pundit supports both "open", relatively flat vo-

[12] http://www.openannotation.org/

[13] http://code.google.com/p/annotation-ontology/

[14] http://www.openannotation.org/spec/core/

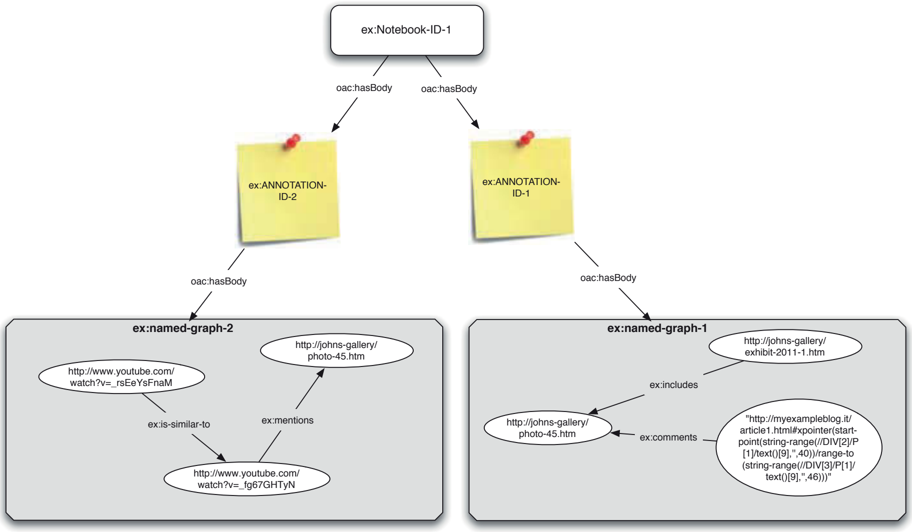

ex:Notebook-ID-1
oac:hasBody
oac:hasBody
ex:ANNOTATION-ID-2
ex:ANNOTATION-ID-1
oac:hasBody
oac:hasBody
ex:named-graph-2
http://johns-gallery/photo-45.htm
http://www.youtube.com/watch?v=_rsEeYsFnaM
ex:is-similar-to
ex:mentions
http://www.youtube.com/watch?v=_fg67GHTyN
ex:named-graph-1
http://johns-gallery/exhibit-2011-1.htm
ex:includes
http://johns-gallery/photo-45.htm
ex:comments
"http://myexampleblog.it/article1.html#xpointer(start-point(string-range(//DIV[2]/P[1]/text()[9],'',40))/range-to(string-range(//DIV[3]/P[1]/text()[9],'',46)))"

Fig. 3: The RDF representation of a notebook including two distinct annotations

cabularies like Freebase (leveraging the reconciliation APIs[15]) and restricted controlled vocabularies and taxonomies, e.g. based on the SKOS model.

In Pundit, "notebooks" are resources that aggregate a set of annotations so that they can be retrieved and queried. By default, each user has a proprietary notebook where all her annotations are collected. Notebooks have a central role in collaborative annotation. These can in fact have read/write privileges and can be used for giving users control over her annotations, allowing to set them as private or public and to select what notebooks are relevant. More precisely, Pundit supports the concept of "active notebook": when a notebook is active for a given user the annotations in it will be shown by default. As a big number of public notebooks might be available, this mechanism allows a user to restrict the amount of annotations visualized to only those she expressed interest in. While such notebooks management features are fully implemented by the Pundit annotation server, their full support at UI level is still under development.

4 Pundit prototype

Pundit has a client-server architecture. The client-side component comprises a set of sub-modules developed in Javascript using the dojo framework[16] 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. The storage module defines a completely generic interface, designed to support different kinds of storage systems ranging from traditional relational databases to NoSQL databases (eg. RDF triple-

[15] http://wiki.freebase.com/wiki/Freebase_API
[16] http://dojotoolkit.org/

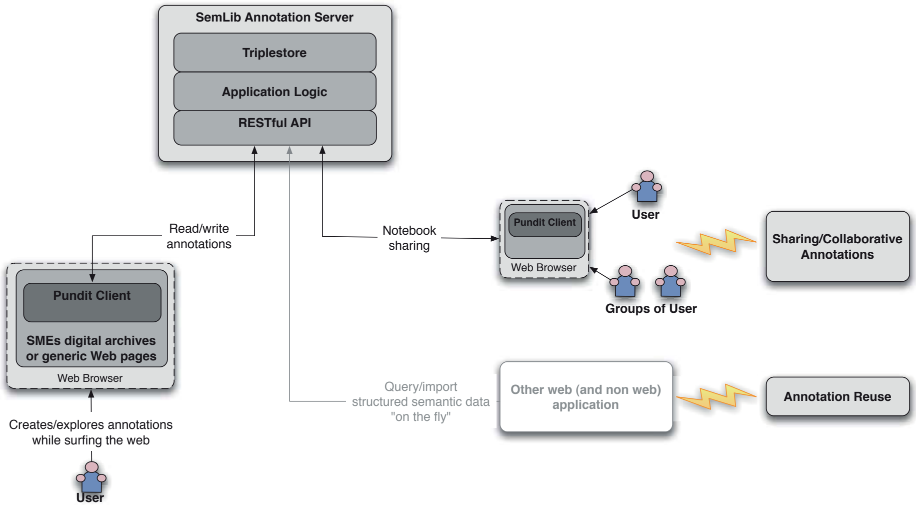

SemLib Annotation Server
Triplestore
Application Logic
RESTful API
Read/write annotations
Notebook sharing
Pundit Client
Web Browser
User
Groups of User
Sharing/Collaborative Annotations
Pundit Client
SMEs digital archives or generic Web pages
Web Browser
Creates/explores annotations while surfing the web
User
Query/import structured semantic data "on the fly"
Other web (and non web) application
Annotation Reuse

Fig. 4: Simplified architecture of the annotation system

stores). In the prototype version, the storage is implemented using the Sesame triplestore[17] as this greatly simplifies handling and exporting RDF data. The storage module, besides keeping user annotations, stores also user profiles and related contextual information (e.g.: user's metadata, user's permissions, etc.). The Annotation Server supports Open-ID[18] for users authentication with single sign-on. Different authentication systems can be easily implemented developing dedicated plugins. The use of single sign-on approach simplifies the integration of the annotation system with existing DL, which may already provide facilities for users authentication. In the following subsections, some of Pundit main features are discussed.

4.1 Annotations of different multimedia contents and at different levels of granularity

Pundit provides specific *Fragment Handlers* to assist users in selecting and highlighting parts of different contents and turn them into actual addressable resources (e.g. using XPointer or Media Fragments Uri) to be used into annotations. This means that with Pundit it is not only possible to attach annotations to single resource but also to establish semantic relations between different resources fragments also of different type. Also, selected resources can be added to "favourites' (*My Items*) and stored to the server to be displayed also in different pages other than the one in which they have been selected. This is fundamental to create cross-page and cross-domain annotations as discussed in more details in Sec. 4.4. At the time of writing Pundit prototype provides support for text fragment and image selection, while image fragment annotation is currently under

[17] http://www.openrdf.org/

[18] http://openid.net/

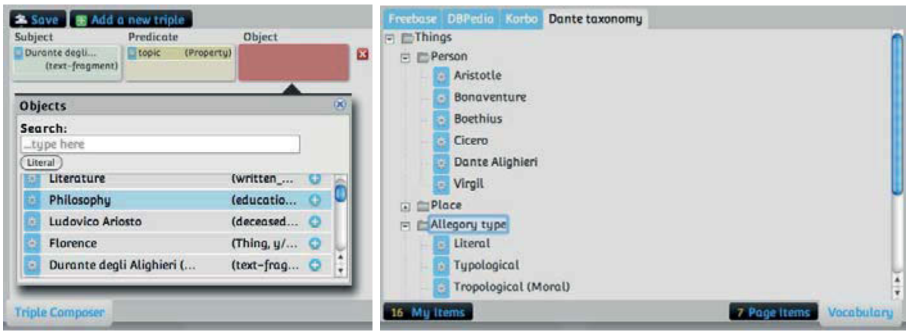


Fig. 5: The Pundit Triple Composer (a) and an example of Pundit taxonomy (b)

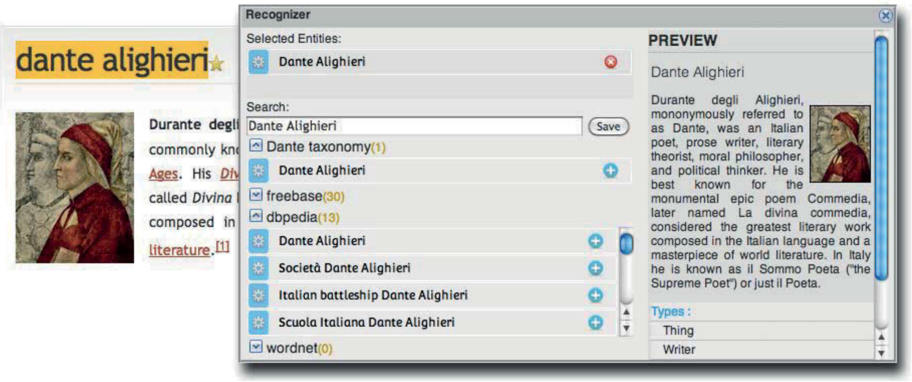
development. In addition, the annotation of video and of temporal and spatial video fragments has been already implemented in Semtube prototype[15]: a Web tool for semantic video annotation of YouTube videos, which has been recently developed basing on Pundit client API and Annotation Server.

4.2 Annotations at different levels of complexity and structure

Pundit provides support for different types of annotations, ranging from simple textual comments and semantic tags to semantic statements. Annotations can be created using different GUIs.

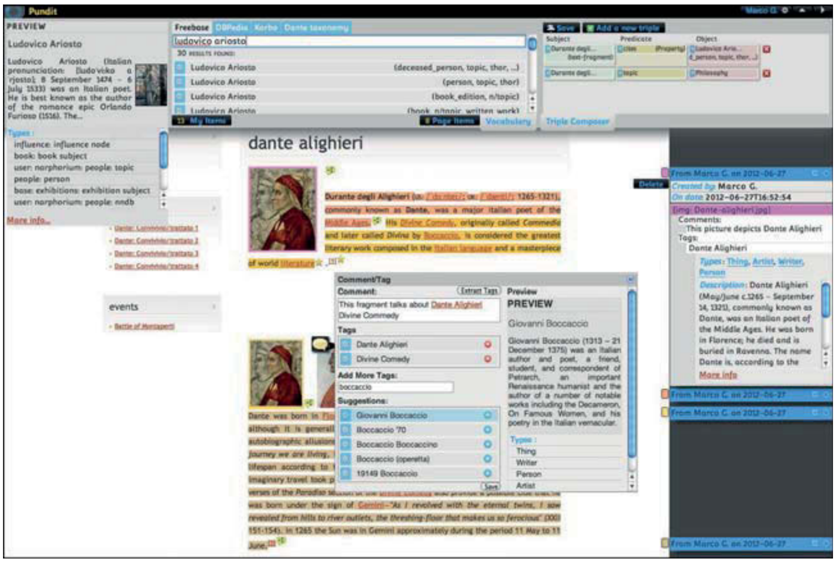
The *Comment/tags Panel* allows the user to type a comment and to automatically extract tags from it using Dbpedia Spotlight service. User can remove suggested tags that are not considered relevant or add others using Dbpedia Lookup service.

The *Recognizer Panel*, Fig. 6 is intended to be used when a user wants to mark the occurrence of a specific entity that is mentioned in text. Once one or multiple words have been selected, the recognizer searches in a set of different sources (including custom taxonomies, Freebase, DBpedia and Wordnet) and suggests matching entities. Once “recognized’, entities mentioned in the text are semantically disambiguated and enriched with structured data.



Durante degli Alighieri, mononymously referred to as Dante, was an Italian poet, prose writer, literary theorist, moral philosopher, and political thinker. He is best known for the monumental epic poem Commedia, later named La divina commedia, considered the greatest literary work composed in the Italian language and a masterpiece of world literature. In Italy he is known as il Sommo Poeta ("the Supreme Poet") or just il Poeta.

Fig. 6: The recognizer panel in action



PREVIEW

Ludovico Ariosto

Ludovico Ariosto (Italian pronunciation: [ludoˈviko aˈrjɔsto]; 8 September 1474 – 6 July 1533) was an Italian poet. He is best known as the author of the romance epic Orlando Furioso (1516). The...

Types :
influence: influence node
book: book subject
user: norphorium: people: topic
people: person
base: exhibitions: exhibition subject
user: norphorium: people: nndb

More info...

ludovico ariosto

30 RESULTS FOUND:

Ludovico Ariosto	(deceased_person, topic, thor, ...)
Ludovico Ariosto	(person, topic, thor)
Ludovico Ariosto	(book_edition, n/topic)
Ludovico Ariosto	(book, n/topic, written work)

Subject	Predicate	Object
Durante degli... Best-[fragment]	cites (Property)	Ludovico Ario... (_person, topic, thor, ...)
Durante degli...	topic	Philosophy

dante alighieri

Durante degli Alighieri 1265-1321), commonly known as Dante, was a major Italian poet of the Middle Ages. His Divine Comedy, originally called Commedia and later called Divina by Boccaccio, is considered the greatest literary work composed in the Italian language and a masterpiece of world literature

Comment/Tag

Comment:

This fragment talks about Dante Alighieri Divine Commedy

Tags

Dante Alighieri

Divine Comedy

Add More Tags:

boccaccio

Suggestions:

Giovanni Boccaccio

Boccaccio '70

Boccaccio Boccaccino

Boccaccio (operetta)

19149 Boccaccio

Preview

PREVIEW

Giovanni Boccaccio

Giovanni Boccaccio (1313 – 21 December 1375) was an Italian author and poet, a friend, student, and correspondent of Petrarch, an important Renaissance humanist and the author of a number of notable works including the Decameron, On Famous Women, and his poetry in the Italian vernacular.

Types :
Thing
Writer
Person
Artist

From Marco G. on 2012-06-27

Created by Marco G.

On date 2012-06-27T16:52:54

[img: Dante-alighieri.jpg]

Comments:
This picture depicts Dante Alighieri

Tags:
Dante Alighieri

Types: Thing, Artist, Writer, Person

Description: Dante Alighieri (May/June c.1265 – September 14, 1321), commonly known as Dante, was an Italian poet of the Middle Ages. He was born in Florence; he died and is buried in Ravenna. The name Dante is, according to the

More info

From Marco G. on 2012-06-27

From Marco G. on 2012-06-27

From Marco G. on 2012-06-27

Fig. 7: A screenshot of Pundit in action

Finally, the *Triple Composer* is the most expressive way of creating structured data, providing a specific GUI for editing semantic statements (triples) in the form of subject-object-predicate. All kinds of items (selected text, taxonomy entries and web of data resources) can be used in statements and put in relations by choosing from a customizable set of predicates. Statement can be create both dragging and dropping items or choosing between suggested items as shown in Fig. 5.a).

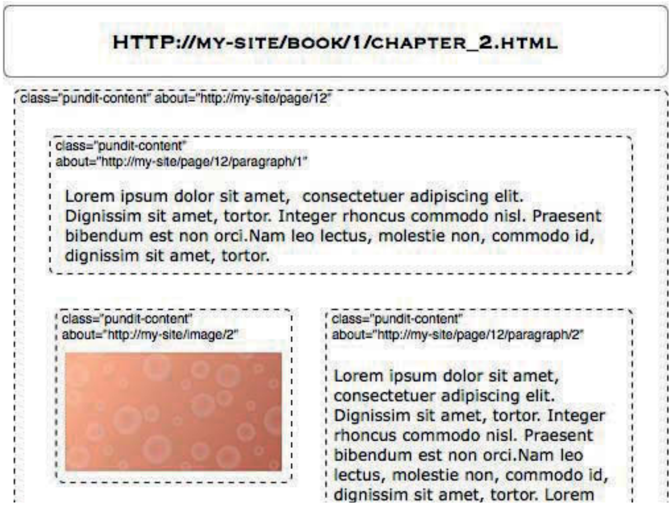
Fig. 5.b) shows the taxonomies tab. The ability of customizing Pundit with domain specific taxonomies is an important feature of Pundit. Digital Library maintainers can add custom taxonomies ”on the fly” just by adding a simple markup to their pages, linking to a JSON file containing the taxonomy.

Fig. 7 shows the overall prototypal user interface to compose semantic annotations and to display contextually created annotations.

4.3 Named Content

DLs, like other Web 2.0 applications, change over time. Presentation can be restyled, changing page layout and mark-up, and content can be re-organized and moved to different pages. 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). In order to grant annotation consistency in such cases, in particular when they are shared in communities and not under a centralized control, it is not sufficient to attach annotations to the Web page.

To overcome this issue, Pundit relies on specific page mark-up. Compliant digital libraries can benefit from a more intelligent behaviour by using the simple named content specification (documented on the web site) to mark-up atomic portions of their content as exemplified in Fig. 8. Each marked content should


HTTP://MY-SITE/BOOK/1/CHAPTER_2.HTML
class="pundit-content" about="http://my-site/page/12"
class="pundit-content" about="http://my-site/page/12/paragraph/1"
class="pundit-content" about="http://my-site/image/2"
class="pundit-content" about="http://my-site/page/12/paragraph/2"

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Dignissim sit amet, tortor. Integer rhoncus commodo nisl. Praesent bibendum est non orci.Nam leo lectus, molestie non, commodo id, dignissim sit amet, tortor.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Dignissim sit amet, tortor. Integer rhoncus commodo nisl. Praesent bibendum est non orci.Nam leo lectus, molestie non, commodo id, dignissim sit amet, tortor. Lorem

Fig. 8: Using Named Contents to allow annotations to be attached to content

have a resolvable URI associated, to which annotations are attached. In this way, annotations regarding the same content, but created in different pages, can automatically be merged and consistently displayed in all the pages where such content appears.

4.4 Cross-page and cross-domain annotations

Cross-pages annotation constitutes a key feature of the proposed annotation system that captures the distributed nature of the Web, in which information is often spread between different sources and can be augmented linking and referencing additional information beyond the boundaries of single Web site or DL. Properly structured annotations can allow weaving a semantic net in order to interconnect and merge fragments of information into a unique knowledge base. For example, an expert of literature can augment the information about Dante Alighieri appearing on a Web page of a DL with text excerpts of the Divine Comedy taken from another Web source. The implementation of such feature requires the system be able to:

- create annotations on every Web page
- create relations between different resources (as text fragment, images, etc...) belonging to different pages.

The former requirement is supported by the availability of the application as a bookmarklet, which allows running the annotation system in every Web pages injecting the required javascript. With such purpose, particular care has been required in protecting Pundit css and variable namespace, in order to avoid clashes that could result in page style and layout alteration as well as in application malfunctioning.

Regarding the latter requirement, it's worth to remark how RDF data model is perfectly suitable to cope with it, being in fact specifically conceived to create statements that connect two resources by means of a property. From an implementation point of view, such requirement is fully fulfilled by means of the *Triple Composer* and by the *MyItems* mechanism, described in the previous subsection. These allows, for example, a user to add an image to *My Items* later assert that a text excerpt selected on another page describes the image.

5 Conclusions

In this paper, Pundit annotation system, which at the time of writing has reached its first stable release, has been introduced. Pundit data model leverages on OAC annotation model and further extends it to fully support the embodiment of semantic statements in the annotation payload by means of named graphs. This provides high flexibility to annotate and interconnect heterogeneous resources over the Web and to be potentially applied in every application scenario. Pundit prototype enables the creation of semantically rich annotations at high granularity levels. These allow to interconnect different resources distributed over the Web and augment original information generating new semantically structured aggregations of knowledge. These can in turn be exploited both to provide user with a more engaging and productive experience in consuming DL and Web content, and effectively reused by other applications.

Compared with other existing semantic annotation tools, Pundit not only provides support all the main annotation approaches introduced by others tools (textual comments, semantic tagging, named entities recognition and the use of taxonomies and ontologies) but enable also more expressivity and flexibility in annotations. In particular, it allows the creation of semantic statements that enable to put in link resources, resource fragments, named entities and vocabulary resource according to semantically defined relations.

In addition, differently by other tools, Pundit has been conceived to provide specific support for annotation sharing, relying on the mechanism of notebooks to aggregate relevant information and make these available both to other users and third party applications by means of a dereferenciable URI and to be easily consumed by means of RESTfull API.

A user evaluation of the tool has been conducted for the video annotation prototype. The obtained results can be found in [15] and are driving the current Pundit development. Further user evaluations are going to be performed on the continuation of the development.

6 Acknowledgments

The research leading to these results has received funding from the European Union's Seventh Framework Programme managed by REA-Research Executive Agency[19] ([FP7/2007-2013][FP7/2007-2011]) under grant agreement n. 262301.

[19] http://ec.europa.eu/research/rea

References

1. C. Morbidoni, M. Grassi, M. Nucci, ”Introducing SemLib Project: Semantic Web Tools for Digital Libraries”. International Workshop on Semantic Digital Archives 15th International Conference on Theory and Practice of Digital Libraries (TPDL). 29.09.2011 in Berlin.
2. Andrews, P., Zaihrayeu, I., Pane, J., ”A classification of semantic annotation systems. Semantic Web Journal”. Online Available: http://www.semantic-Web-journal.net/content/classification-semantic-annotation-systems
3. V. Uren, P. Cimiano, J. Iria, S. Handschuh, M. Vargas-Vera, E. Motta, and F. Ciravegna. Semantic annotation for knowledge management: Requirements and a survey of the state of the art. Web Semant., 4(1), January 2006.
4. R. A. Arko, K. M. Ginger, K. A. Kastens, and J. Weatherley, “Using annotations to add value to a digital library for education’.
5. Rose Holley, “Crowdsourcing: How and Why Should Libraries Do It?’, D-Lib Magazine, The Magazine of Digital Library Research. March/April, 2010.
6. 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.
7. 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.
8. 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.
9. Markus Luczak-Rsch, Ralf Heese, Adrian Paschke, “Future Content Authoring’, In Nodilities The Magazine of the Semantic Web, Issue 11, pp. 17-18, 2010.
10. G. Rockwell, S. Brown, J. Chartrand, S. Hesemeier, “CWRC-Writer: An In-Browser XML Editor’ - Digital Humanities 2012 Conference Abstracts. University of Hamburg, Germany. July 1622, 2012
11. A. Gerber and J. Hunter, “Authoring, Editing and Visualizing Compound Objects for Literary Scholarship’, Journal of Digital Information, vol. 11, 2010.
12. M. L. Ralf Heese, “One Click Annotation’ in 6th Workshop on Scripting and Development for the Semantic Web, 2010.
13. M. Koivunen, R. Swick, E. Prud’hommeaux “Annotea and Semantic Web Supported Collaboration’. ESWC 2005, UserSWeb workshop. 2005
14. “Open Annotation: Alpha3 Data Model Guide’ 15 October 2010 Eds. R. Sanderson and H. Van de Sompel. `http://www.openannotation.org/spec/alpha3/`
15. M.Grassi, C. Morbidoni and M. Nucci. A Collaborative Video Annotation System Based on Semantic Web Technologie. In press: Cognitive Computation. Springer-Verlag, Berlin Heidelberg (DOI: 10.1007/s12559-012-9172-1)