Department of
Information Engineering
and Computer Science DISI

UNIVERSITY
OF TRENTO - Italy

DISI - Via Sommarive 14 - 38123 Povo - Trento (Italy)
http://www.disi.unitn.it

A CLASSIFICATION OF SEMANTIC ANNOTATION SYSTEMS

Pierre Andrews, Ilya Zaihrayeu
and Juan Pane

December 2010

Technical Report # DISI-10-064

A Classification of Semantic Annotation Systems

Pierre Andrews [a], Ilya Zaihrayeu [a] and Juan Pane [a]
[a] *Dipartimento di Ingegneria e Scienza dell'Informazione*
The University of Trento, Italy
E-mail: {andrews,ilya,pane}@disi.unitn.it

Abstract.
The *Object-Subject-Predicate* triple annotation system is now well adopted in the research community, however, it does not always speak to end-users. In fact, explaining all the complexity of semantic annotation systems to laymen can sometime be difficult. We believe that this communication can be simplified by providing a meaningful abstraction of the state of the art in semantic annotation models and thus, in this article, we describe the issue of semantic annotation and review a number of research and end-user tools in the field. Doing so, we provide a clear classification schemes of the features of annotation systems. We then show how this scheme can be used to clarify requirements of end-user use cases and thus simplify the communication between semantic annotation experts and the actual users of this technology.

Keywords: Semantic, Semantic Annotation, Vocabulary, Classification Scheme, Tag, Attributes, Ontology

1. Introduction

Social annotation systems such as Delicious[1], Flickr[2] and others have laid the fundamentals of the Web 2.0 principles and gained tremendous popularity among Web users. One of the factors of success for these systems is the simplicity of the underlying model, which consists of a resource (e.g., a web page), a tag (normally, a text string), and a user who annotated the resource with the tag as can be seen in Figure 1. Despite its simplicity, the annotation model enables a set of useful services for the end user, e.g., searching resources using tags added by a community of users, computing the most popular tags and building the so-called tag clouds[3], finding users with common interests based on the resources they annotated and on the tags they used and providing a recommendation service on this basis, among others.

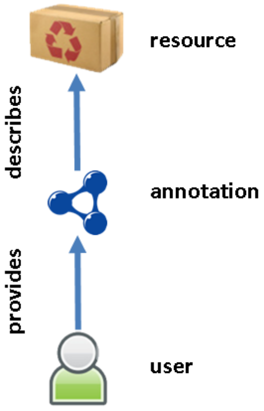


Fig. 1. A Generic Annotation Model

Due to the natural language nature of the underlying model, these systems have been criticised for not being able to take into account the explicit information about the meaning or semantics of each tag. For example, different users can use the same tag with different meanings (i.e., homonyms), different tags with the same meaning (i.e., synonyms), different tags in

[1] http://www.delicious.com/
[2] http://www.flickr.com/
[3] http://en.wikipedia.org/wiki/Tag_cloud

the same meaning but at different levels of abstraction, morphological variations of the same tag, and so on. Despite the problem of ambiguity in the interpretation of the meaning of the tag itself, there is a further problem of deciding what the tag refers to. For example, a tag "John" attached to a photo does not specify if John is a person on the photo or the photographer who took this photo.

With the advent of semantic technologies, some of the aforementioned problems were (partially) addressed in some systems. For example, Flickr introduced the so-called machine tags that use a special syntax to define extra information about the tag. For example, a tag "upcoming:event=428084" assigned to a photo encodes that it is related to an Upcoming.org event identified with "428084". Another example is Faviki[4] – a social bookmarking system similar to Delicious. However, when users enter a tag, they are asked to disambiguate its meaning to a known concept extracted from Wikipedia.

There are many types of annotation models available in the scientific state of the art and in the already existing end-user applications. In the semantic web community, these models have been abstracted by the *Subject-Object-Predicate* triple that can be used for most of the annotation kind discussed here. However, this generalisation is extremely low level and it is often hard to understand how it applies to high level application domains that interest the end-users.

In Sections 2, 3 and 4, we present an extended survey of the tools and models that were available and we provide what we believe to be a simple abstraction of the important dimensions that an annotation model can have. The features were selected based on the intention to demonstrate how much they can contribute to the definition of a semantic annotation model. The features were grouped in the following three categories:

1. the level structural complexity of annotations (e.g., tags, attributes, and relations), see Section 2;
2. the vocabulary type, i.e., the level of formality of annotations defined on the basis of the form of the underlying ontology to the elements to which the annotations can be linked (e.g., thesauri, taxonomies), see Section 3; and
3. the level of user collaboration in sharing and reusing semantic annotations and in the collaborative construction and evolution of the underlying ontology, see Section 4.

Within each category we provide a specification of the most typical approaches by giving their description, comparative analysis of their advantages and disadvantages, and examples of popular systems (both in the research field and among publicly available systems). The goal of the comparative analysis is to show the trade off between the level of user involvement and provided services in each approach.

In Section 5 we show how this annotation model classification scheme can be used to elicit requirements from end-users. To do this, we provide summaries and conclusions made from the analysis of three concrete use cases pertaining to different industrial sectors: web services, telecommunications, and online games.

2. Structural Complexity of Annotations

The first dimension that we discuss is the structural complexity of annotations. This dimension relates to the amount of information that is encoded in the annotation itself, how it is structured in the underlying storage model and how this structure can be used. This structure has great influence on what data can be displayed to the user, how it can be displayed, but also what type of back-end services can be provide. We distinguish between *tags*, *attributes*, *relations* and *ontologies*. Tags are at the beginning of the spectrum and represent the easiest form of annotation from the user point of view; whereas ontologies are at the other end of the spectrum and represent the hardest form of annotation from the user point of view. In fact, as it has been showed in [1], designing a full-fledged ontologies is a difficult and error-prone task even for experienced users.

2.1. Tags

A tag annotation element is a non-hierarchical keyword or free-from term assigned to a resource (see Figure 2). A tag *implicitly* describes a particular property of a resource as the computer and other users do not know the meaning that the annotator intended (except if the natural language used is unambiguous). Normally, a tag is a single word or a sequence of characters without spaces (which typically serve as tag separators in the user input). Examples of tags include: the name

[4] `http://faviki.com`

of the person on a picture, the name of the place where a picture was taken, or a topic of a news article. The tagging annotation systems are often discussed as part of the folksonomy annotation model [2] that links tags, resources and users following the model in Figure 1.

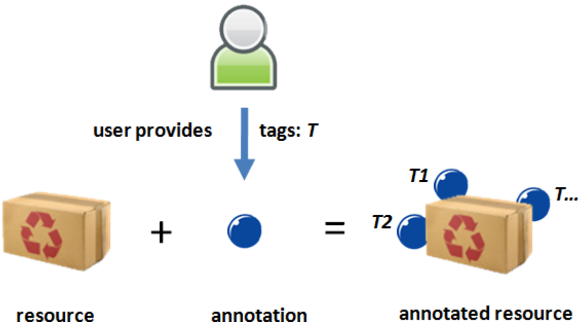


Fig. 2. The Tag Annotation Model

Pros The notion of tagging is very familiar for Web 2.0 users and, therefore, require nearly no learning curve for a typical user in order to start using them. They allow the user to easily annotate a Web resource with a free-text term and find other resources which were annotated with the same tag by browsing or searching. In fact, after four years of its existence Flickr reported to have "about 20 million unique tags" (Jan 08) and 5 billion images (September 2010) [3].

Cons Tags represent a minimal annotation model from the structural complexity point of view and, therefore, can enable only a limited number of services mainly focused on basic retrieval and browsing (e.g., retrieve resources that were assigned tag x). Because they only implicitly describe resource properties, tags are subject to ambiguity in the interpretation of these properties. For example, natural language tag "John" attached to a picture does not specify whether John is a person on the picture or if he is the photographer who took the picture.

Applications Delicious[5] is a social bookmaking service that allows its users to memorize and share URLs of Web resources such as blogs, articles, music, video, etc. It was founded in 2003 and now counts five million users and 150 million bookmarked URLs. The key idea of the service is that its users can access their bookmarks from any computer, for example at home, at work or while traveling. Delicious has been among the first popular systems that used tagging for organization and retrieval purposes. One of the important success factors of Delicious was its simplicity of use (see Figure 3). In a nutshell it works as follows: the user finds an interesting website and decides to bookmark it in Delicious; when adding the website the user assigns a set of terms that describe the website. Later, in order to retrieve a saved bookmark the system is queried with one or more of the previously assigned tags. If the answer set contains too many elements, it can be refined by adding more terms to the query. Delicious allows the user to browse not only a personal user space of resources, but also to find bookmarks saved and annotated by other users. Tagclouds (see Figure 4) and tag based search are starting points for navigation in the space of all (published) bookmarks of all users. While browsing bookmarks the system visualizes tags which were assigned to resources by other users.

Flickr[6] is a free image hosting service that stores over three billion images. It was launched in 2004 and popularized the concept of tagging together with Delicious. Flickr allows its users to upload their photos, organize and share them using tags. Even if, the users can establish relationships, form communities, comment and annotate photos of each other, the site is more used as a user's personal photo repository.

Another example of a popular social site which uses tags is Last.fm[7]. It is a UK-based Internet radio, founded in 2002, which has a thirty million active users in more than 200 countries. Last.fm allows its users to create custom playlists and radio stations from audio tracks available from the service's library. It also offers numerous social features such as recommendation of similar tracks considering user's favorites. Users can tag resources such as bands, artists, albums and tracks and retrieve them using tag based search. Examples of other systems that use tags for annotating their resources include: Youtube2009[8], CiteULike[9] and LiveJournal[10].

Note that the *Subject-Object-Predicate* model used widely in web semantic technologies (through RDF for instance) is already of higher structural complexity than the *tags* as it supposes the existence of a predicate linking the tag (Object) and the resource (Subject). In fact, in the simple tagging annotation model, there is

[5] `http://www.delicious.com/`

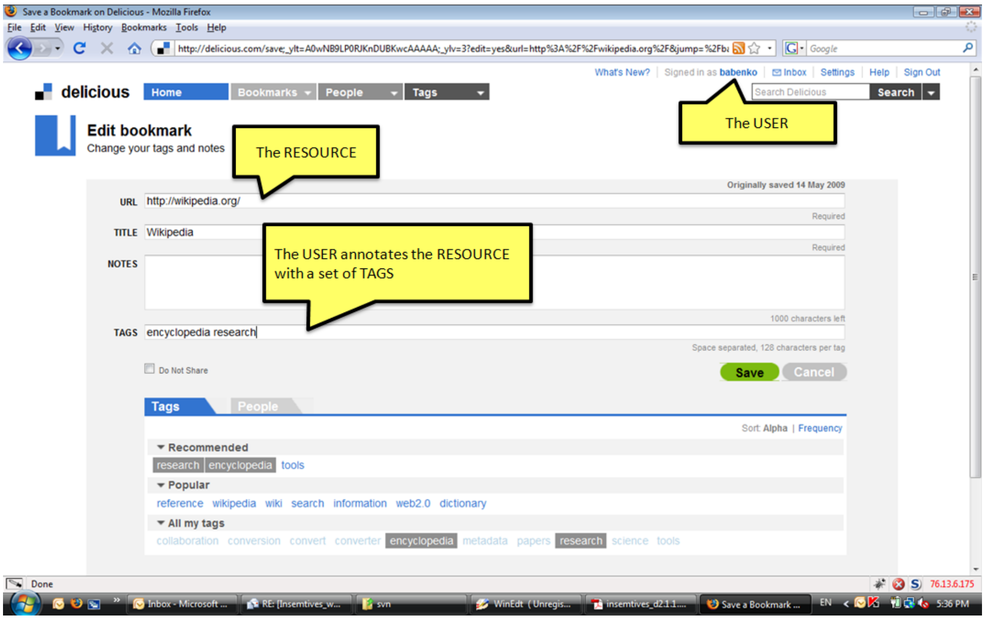
[6] `http://www.flickr.org`

[7] `http://www.last.fm`

[8] `http://www.youtube.org`

[9] `http://www.citeulike.org/`

[10] `http://www.livejournal.com/`



http://delicious.com/save;_ylt=A0wNB9LP0RJKnDUBKwcAAAAA;_ylv=3?edit=yes&url=http%3A%2F%2Fwikipedia.org%2F&jump=%2Fb

The USER

Edit bookmark

Change your tags and notes

The RESOURCE

Originally saved 14 May 2009

URL http://wikipedia.org/

Required

TITLE Wikipedia

Required

The USER annotates the RESOURCE with a set of TAGS

NOTES

1000 characters left

TAGS encyclopedia research

Space separated, 128 characters per tag

□ Do Not Share

Sort: Alpha | Frequency

Recommended

research encyclopedia tools

Popular

reference wikipedia wiki search information web2.0 dictionary

All my tags

collaboration conversion convert converter encyclopedia metadata papers research science tools

Fig. 3. Annotating a Resource Using the Tag Annotation Model

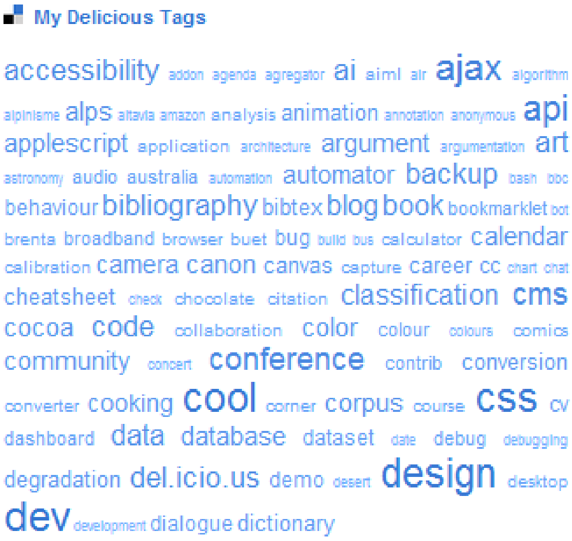

My Delicious Tags
accessibility addon agenda agregator ai aiml air ajax algorithm
alpinisme alps altavia amazon analysis animation annotation anonymous api
applescript application architecture argument argumentation art
astronomy audio australia automation automator backup bash bbc
behaviour bibliography bibtex blog book bookmarklet bot
brenta broadband browser buet bug build bus calculator calendar
calibration camera canon canvas capture career cc chart chat
cheatsheet check chocolate citation classification cms
cocoa code collaboration color colour colours comics
community concert conference contrib conversion
converter cooking cool corner corpus course css cv
dashboard data database dataset date debug debugging
degradation del.icio.us demo desert design desktop
dev development dialogue dictionary

Fig. 4. Delicious Cloud of Tags

no need for this predicate as it is always meant to be the tagging relationship.

2.2. *Attributes*

An attribute annotation element is a pair $\langle AN, AV\rangle$, where AN is the name of the attribute and AV is the value of the attribute (see Figure 5). The attribute name defines the property of the annotated resource (e.g., “location”, “event”, “starting date”) and the attribute value specifies the corresponding value (e.g., “Trento”, “birthday”, “April 1, 2009”). Apart from this, the model allows us to define the data types for attributes and, therefore, enables type checking at query time.

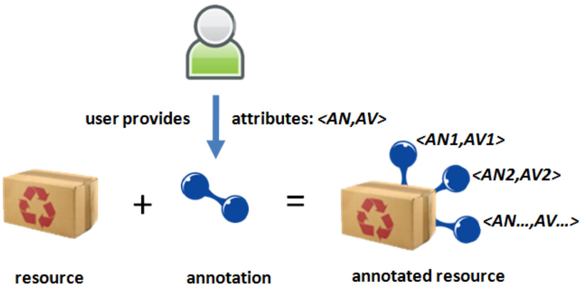


Fig. 5. The Attribute Annotation Model

Pros Attributes are pervasively used on the web and in desktop applications and, therefore, represent a well known notion for end users. Differently from tags, attributes explicitly define the described resource properties and, therefore, enable a richer resource annotation and query language. For example, one could search for images of the Eiffel Tower taken between 1890 and 1900 by a specific photographer.

CONS While enabling more services than tags do, attributes are still a limited means of annotation because they refer to single resources and, therefore, cannot be used to effectively enable services which are based on interrelationships that exist between resources (e.g., search and navigation between related resources). Furthermore, attribute annotations require a more metadata-knowledgeable user than tag annotations do.

APPLICATIONS One of the earliest systems that used attributes for resource annotation was Semantic File Systems described by [4]. The system allows the user to assign arbitrary number of name-value pairs to the user's files and then retrieve them by creating so called virtual directories. The user creates virtual directories at runtime specifying the list of attributes. According to the user input the virtual directory contains only those files whose attributes match attributes from the list. The implementation of the ideas introduced in Semantic File systems can be found in search engines integrated directly in the operating systems such as Mac OS X SpotLight.

The Web has many examples of popular systems using attributes. Almost all social networks, (e.g. Facebook[11] and MySpace[12]) consider the user as a resource and use the attribute annotation model to represent user profiles. The variety of attributes is very big and includes common attributes such as "Personal Info", "Contacts" as well as specific attributes, such as "Interests", "Traveling" (see Figure 6). Online markets such as Ebay[13] use the attribute annotation model to annotate resources which are items to be sold. The seller can assign an item with attributes such as "item location: Trento", "item price: $100", "shipping to: Italy", etc.

In the *Subject-Object-Predicate* model, the predicate would be used to link the resource (subject) and the attribute value (object) through a property name (predicate). In that sense, the RDF+OWL *Property* predicates can be considered to be a good example of the attribute complexity dimension.

2.3. Relations

a relation annotation element is a pair $\langle Rel, Res \rangle$, where Rel is the name of the relation and Res is another resource. The relation name defines how the annotated resource is related with Res. In a sense, the relation annotation model is an extension of the attribute annotation model to the domain of resources, which allows the user to interlink these resources (see Figure 7). For instance, in a scientific paper a citation referencing another paper is an example of a relation annotation which defines a relation between these documents.

PROS Relation annotations provide a way to interlink various resources through typed links. It allows the user to navigate from one resource to another and enable search and navigation based on these relation links.

CONS The user is expected to bear a higher mental load w.r.t. the previous models as, instead of describing one resource, the user has to understand what the two resources are about and what kind of relationship holds between them.

APPLICATIONS User of social network sites annotate their profiles establishing relationships between each other, which allow them to find friends and to meet new people by navigating the network of relations between users. Apart from this, some social networks, for instance Facebook, allow its users to annotate photos with links to profiles of people appearing in the picture (See Figure 8). In 2007, Facebook had around 1.7 billion uploaded photos with around 2.2 billion relation annotations.

The relation annotation model can also be used to define relations within a resource. For example, the Araucaria project [5] annotates the rhetorical structure of a document using the RST relations annotation [6]. Figure 9 illustrates an example of intra document annotation of the RST relations; segments of text are given an identifier and links between each segment are annotated with a type of argumentation relationship (justify, condition, etc.).

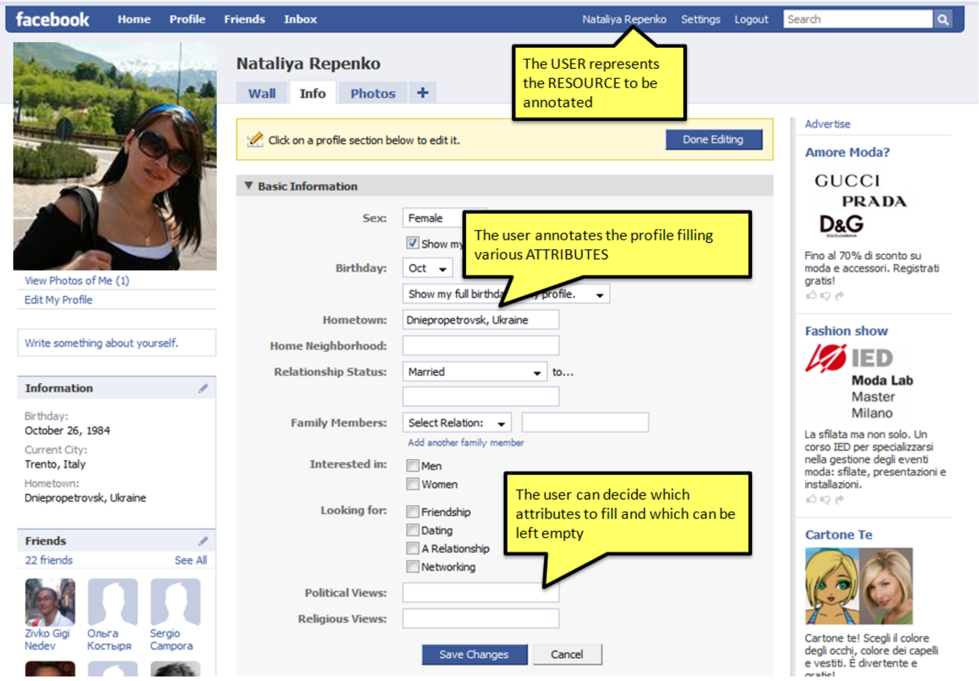
Upcoming.org[14] is a social website for listing events that can be linked to Flickr photos by annotating them with so called triple or machine tag. Triple tags use a special syntax to define extra information about the tag making it machine readable. For example, a tag like "upcoming:event=428084" assigned to a photo encodes that it is related to Upcoming.org event identified with "428084", therefore making these resources

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

[12] http://www.myspace.com/

[13] http://www.ebay.com

[14] http://upcoming.yahoo.com/



Nataliya Repenko

Click on a profile section below to edit it.

Basic Information

Sex: Female

☑ Show my

Birthday: Oct

Show my full birthday ... my profile.

Hometown: Dniepropetrovsk, Ukraine

Home Neighborhood:

Relationship Status: Married to...

Family Members: Select Relation:

Add another family member

Interested in: □ Men □ Women

Looking for: □ Friendship □ Dating □ A Relationship □ Networking

Political Views:

Religious Views:

Information

Birthday:
October 26, 1984

Current City:
Trento, Italy

Hometown:
Dniepropetrovsk, Ukraine

Fino al 70% di sconto su moda e accessori. Registrati gratis!

La sfilata ma non solo. Un corso IED per specializzarsi nella gestione degli eventi moda: sfilate, presentazioni e installazioni.

Cartone te! Scegli il colore degli occhi, colore dei capelli e vestiti. È divertente e gratis!

The USER represents the RESOURCE to be annotated

The user annotates the profile filling various ATTRIBUTES

The user can decide which attributes to fill and which can be left empty

Fig. 6. Resource Annotation Using the Attribute Annotation Model

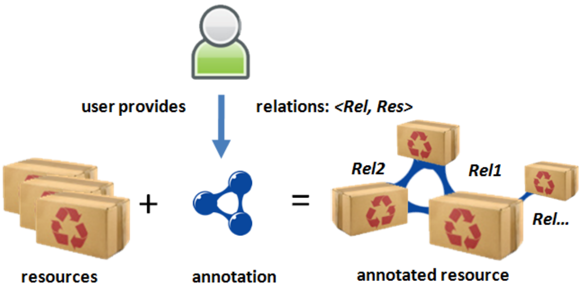


Fig. 7. The Relation Annotation Model

interlinked. Last.fm is another example of a system that use triple tags to link its tracks to Flickr photos.

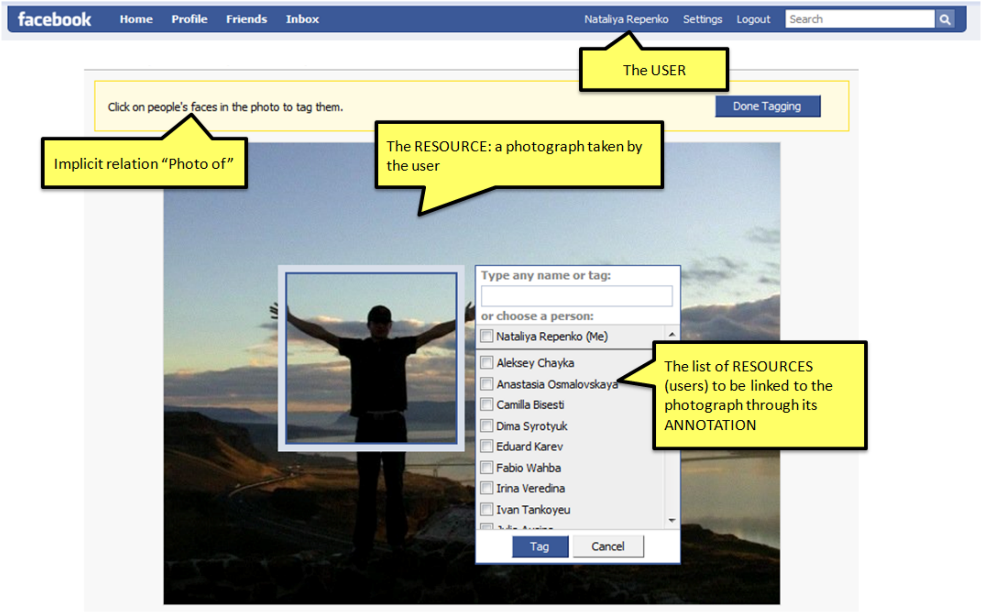
Freebase[15] is a large knowledge base containing around five million various facts about the world. It is described as "an open shared database of the world's knowledge" and "amassive, collaboratively-edited database of cross-linked data."[16]. It allows its users to annotate resources (e.g., images, text, Web pages) using the Freebase annotation schema. The schema defines aresource annotation as a collection of attributes, its kind (e.g. person, event, location), and typed relations with other resources. It provides a convenient way to perform search and navigation in the space of resources allowing the user to find them using attributes, relations, and/or schema kinds.

2.4. *Ontologies*

This model is based on the notion of semantic annotation [7], the term coined in early 2000s. It describes both the process and the resulting annotation or metadata consisting of aligning a resource or a part of it with a description of some of its properties and characteristics with respect to a formal conceptual model or ontology. Figure 10 shows an schematic representation of the model. As defined by Studer et al., "an ontology is an explicit specification of a (shared) conceptualization" [8]. In practice, ontologies are usually modeled with (a subset of) the following elements: concepts (e.g., `CAR`, `PERSON`), instances of these concepts (e.g., `bmw-2333` *`is-instance-of`* `CAR`, `Marry` *`is-instance-of`* `Person`), properties of concepts and instances (e.g., `PERSON` *`has-father`*), restrictions on these properties (e.g., `MAX(PERSON`

[15] http://www.freebase.com

[16] http://www.crunchbase.com/company/metawebtechnologies



Click on people's faces in the photo to tag them.

The USER

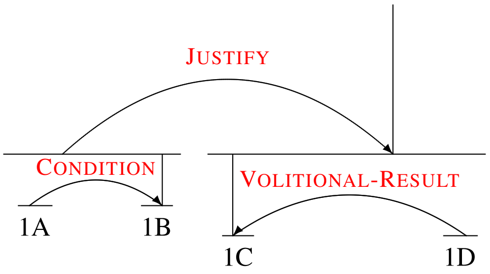
Implicit relation "Photo of"

The RESOURCE: a photograph taken by the user

The list of RESOURCES (users) to be linked to the photograph through its ANNOTATION

□ Nataliya Repenko (Me)
□ Aleksey Chayka
□ Anastasia Osmalovskaya
□ Camilla Bisesti
□ Dima Syrotyuk
□ Eduard Karev
□ Fabio Wahba
□ Irina Veredina
□ Ivan Tankoyeu

Fig. 8. Resource annotation using the relation annotation model



[And if the truck driver's just don't want to stick to the speed limits,]^{1A} [noise and resentments are guaranteed.]^{1B} [It is therefore legitimate to ask for proper roads and speed checks.]^{1C} [And the city officials have signaled to support local citizens.]^{1D}

Fig. 9. Rethorical Structure Theory Relationship

has-father) = 1[17]), relations between concepts (e.g., PERSON *is-a* BEING), relations between instances (e.g., Marry *has-father* John), etc [9]. The ontology annotation model allows the user to describe and interlink existing resources by qualifying resources as concepts or as instances and by defining relations, properties, and restrictions that hold between them.

An example of the application the ontology annotation model is shown in Figure 11. In this example, possibly different users annotated resources that represent the concepts of a Person, Country, and Political Unit as well as the entities Barack

[17] The meaning of this rather informal notation is that any instance of the concept PERSON may have one father at most.

Obama and USA. For the annotation, the user(s) used relations is-instance-of, is-president-of, and is-a defined in some ontology specification. Other examples of the application of this model can be found in [7].

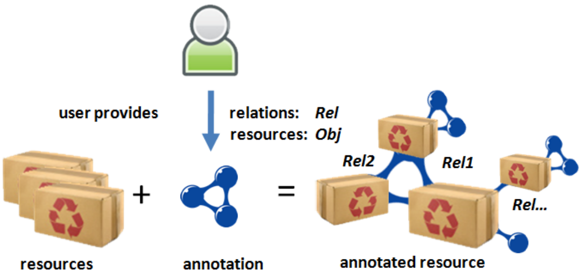


Fig. 10. The Ontology Annotation Model

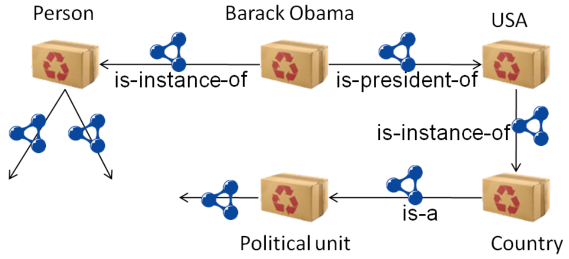


Fig. 11. An Example of the Application of the Ontology Annotation Model

Pros Ontology-based annotations or "semantic annotations" describe a resource with respect to a formal conceptual model, allowing meaning-bearing links between structured and unstructured data (such as an ontology and a text). This empowers a whole new range of retrieval techniques, which can be based on the knowledge schema expressed in the ontology, benefit from reasoning, co-occurrence of annotation or entities in the same resource or context, as well as combine this with unstructured data specific types of retrieval, such as full text search (FTS) in information retrieval (IR). The actual metadata is encoded in the annotation and usually expresses metadata automatically or manually generated about the resource. The ontology and the corresponding instance base capture background knowledge about a domain. The combination of the evidence based information about the resource and the background knowledge, allows indexing techniques, which are based on resource URIs as modeled in the ontology, ensuring retrieval and navigation through each of its characteristics (for example lexical representations such as NYC and New York City will be indexed as a single resource, despite their superficial differences, and this will lead to results containing the string "New York City", even though the user provides a query such as "NYC").

Cons Each of the annotation models described in order of increasing complexity, presents new challenges to human annotators, although disclosing richer potential for automatic processing. Semantic annotations, being the most sophisticated of this row, are no exception. The main challenges semantic annotation presents are in two major lines, namely (i) usability, and (ii) maintenance of the conceptual models.

The usability aspect is key to human involvement in the generation of semantic metadata and is also going to be the main hurdle that needs to be crossed to allow the approach weave in all forms of user interaction with software and data. Proposing a large number of entities and concepts coming form an ontology to the user when annotating is indeed an issue. There should thus be efficient search and recommendation services to help the user in providing the right sense for the controlled annotation.

This raises another issue in the use of complex ontological structures. In fact, one can experience the challenge of presenting multiple taxonomical and subsumption structures over the same model, as well as multiple description facets of each resource. Empowering users to find its way to the right concept, entity or relationship that they want to cite is a serious challenge to usability experts and visual interface designers. In addition, in real world applications (in the LOD for example) multiple instance bases and fact bases can be involved and thus, their corresponding ontologies can be inter-aligned [10], resulting in millions or even billions of individual entity descriptions. This creates a new challenge in terms of scalability, intuitive search and auto-suggest/complete methods, as well as summarization of entire knowledge bases to any useful level of granularity and with respect to any criteria.

Another complicated task an ontology provider needs to face is the maintenance and update of the knowledge, often coming from external sources, its syntactic and semantic alignment, and often, its challenging scale (e.g. bio-medical knowledge bases with billions of individual facts). In [11] study of different automatic approach to construct or extend ontologies, they conclude that most of the state of the art require some level of expert user involvement to curate the knowledge. This is also pointed out in [12].

APPLICATIONS Ontowiki [13] is a free, open-source semantic wiki application, meant to serve as an ontology editor and a knowledge acquisition system (see Figure 12). It allows its users to annotate Web resources representing them as concepts and instances of concepts. Different attributes can be assigned to a resource and interlinked with other resources, therefore, describing its characteristics. OntoWiki retrieval functionality allows to search and to generate different views and aggregations of resources based on concepts, attributes and relations.

Semantic Wikipedia [14] is an extension of Wikipedia that allows its users to annotate wikipedia articles defining an article as a class like "person", "document" or an instance of previously declared class. The users can also interlink articles by annotating them with typed links like "author" or "was born in". This brings semantic enhancement to Wikipedia that allows browsing and searching in a more specific ways, such as "Give me a table of all movies from the 1960s with Italian directors."

KIM is a semantic annotation, indexing and retrieval platform developed by Ontotext [15]. It can mainly be used to facilitate automatic semantic annotation on top of different content types, with built-in extended capabilities for semi- or non-structured text processing based on the GATE framework [16]. It is deployed on top of a native semantic database engine, currently OWLIM and/or Sesame [17,18]. KIM has been applied in various domains, some of the most interesting ones being anti-corruption and asset recovery, and analysis of bio-medical content, such as clinical study reports or scientific papers. As a result of the semantic annotation and indexing, it provides multiple retrieval paradigms, using the content, its structure, and document-level metadata, along with entity co-occurrence and graph-like patterns on the ontologies and knowledge bases to return either sets of entities or documents, or track trends and calculate timelines using content's temporal aspect. It allows both of the above described kinds of semantic annotations, in the same time providing support for the simpler annotation models explained earlier and in the next section. Despite the fact it has matured as a platform since the early 2000s and is in active use, it has never, so far, been focused on usability or visualization aspects of the manual annotation process.

3. The Vocabulary Type

When annotation elements (e.g., tags, attribute names and values, relation names) are provided by the user in the form of a free-form natural language text, these annotations unavoidably become subjects to the semantic heterogeneity problem because of the ambiguous nature of natural language [19]. Particularly, we identify the following three main issues related to the annotation process.

Base form variation This problem is related to natural language input issues where the annotation is based on different forms of the same word (e.g., plurals vs. singular forms, conjugations, misspellings) [19]. This is usually dealt with a lemmatization procedure to convert the annotation to its base form.

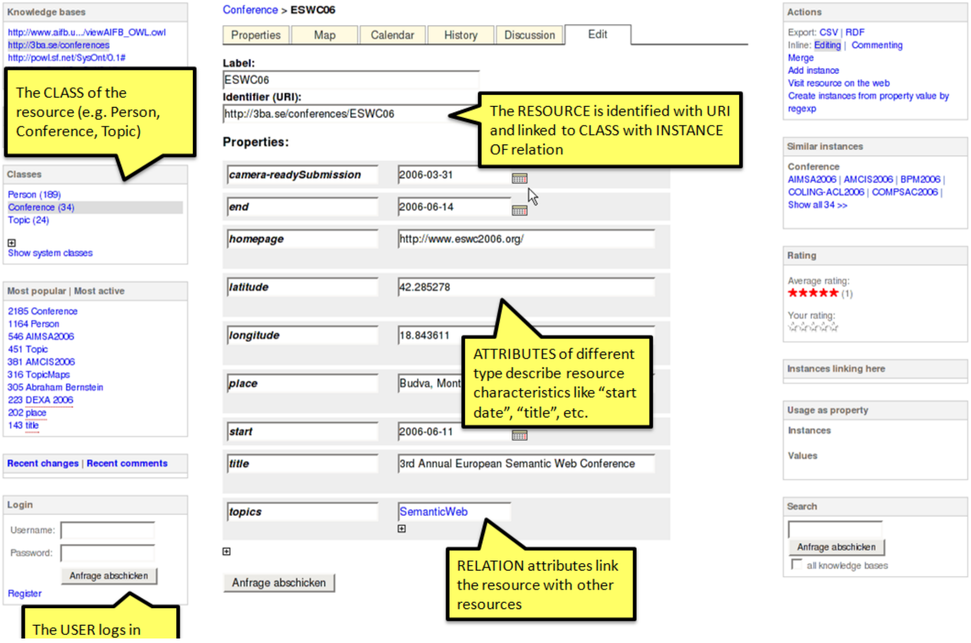
Polysemy Annotation elements may have ambiguous interpretation. For instance, the tag "Java" may be used to describe a resource about the *Java island* or a resource about the *Java programming language*; thus, users looking for resources related to the programming language may also get some irrelevant resources related to the Island (therefore, reducing the precision);

Synonymy Syntactically different annotation elements may have the same meaning. For example, the attribute names "is-image-of" and "is-picture-of" may be used interchangeably by users but will be treated by the system as two different attribute names because of their different spelling; thus, retrieving resources using only one of the attribute names may yield incomplete results as the computer is not aware of the synonymy link;

Specificity gap This problem comes from a difference in the specificity of terms used in annotation and searching. For example, the user searching with the tag "cheese" will not find resources tagged with "cheddar[18]" if no link connecting these two terms exists in the system.

Typically, the problems described above are addressed using so called *controlled vocabularies* – "*organized lists of words and phrases, or notation systems, that are used to initially tag content, and then to find it through navigation or search*" [20]. The basic underlying idea of how controlled vocabularies can be used for the annotation and search is as follows:

[18] which is a kind of cheese



Label: ESWC06

Identifier (URI): http://3ba.se/conferences/ESWC06

Properties:

camera-readySubmission	2006-03-31
end	2006-06-14
homepage	http://www.eswc2006.org/
latitude	42.285278
longitude	18.843611
place	Budva, Mont
start	2006-06-11
title	3rd Annual European Semantic Web Conference
topics	SemanticWeb

The CLASS of the resource (e.g. Person, Conference, Topic)

The RESOURCE is identified with URI and linked to CLASS with INSTANCE OF relation

ATTRIBUTES of different type describe resource characteristics like "start date", "title", etc.

RELATION attributes link the resource with other resources

The USER logs in

Fig. 12. Resource Annotation Using the Ontology Annotation Model in OntoWiki

by interpreting and linking user annotation terms to elements of a controlled vocabulary (e.g., words, concepts), the meaning of these terms is disambiguated (see Figure 13).

Controlled vocabularies are normally built in a top-down fashion by a (small) group of experts and are often compared and contrasted with the so called *folksonomies* [21] – classification systems which are generated in a bottom-up fashion through collective and massive annotation made by a (large) group of users [22,2]. Despite this difference, it has been argued that folksonomies largely conform to the requirements formulated for controlled vocabularies [23] and that both can co-exist and complement each other [24,2,22]. Given their complementary nature and the purpose of this section, we do *not* draw a difference between folksonomies and controlled vocabularies and we rather focus on differences in their models from the point of view of the resulting vocabulary structure and on their comparative advantages and disadvantages. In Section 4, we discuss models that differ in how the controlled vocabularies are actually built (e.g., top-down vs. bottom-up fashion).

Noteworthy, the term *controlled vocabulary* comes from the Library and Information Science community and in the Semantic Web community the term *ontology* is more used instead in order to describe similar kinds of knowledge organization systems. Hereinafter we will use both terms interchangeably.

Ontologies are often classified based on the level of their expressivity (or formality) which conditions the extent to which a certain form of ontology can be used in automated reasoning. [25] proposed such a classification shown in Figure 14. They note that there is a point on the scale (marked as a black bar) where automated reasoning becomes useful: this is where the ontology can be reasoned about using the subclass relations [2].

Note that the way ontologies can be used to *annotate* resources (as described in Section 2) is different from the way ontologies can be used to *support* the annotation process (as described in this section). In the former case, users build ontologies by providing pieces of the knowledge it contains as annotation elements. For example, by linking a page about Barack Obama to a page about people with the ontological relation `is-instance-of` (recall the example depicted in Figure 11), the user annotates the page about Barack Obama with an ontology annotation element. Such elements (possibly provided by different users) are then assembled into a bigger ontology which can be seen as a complex ontological annotation structure used to describe the annotated resources. In the case where the ontology is used as a knowledge base for *support* to

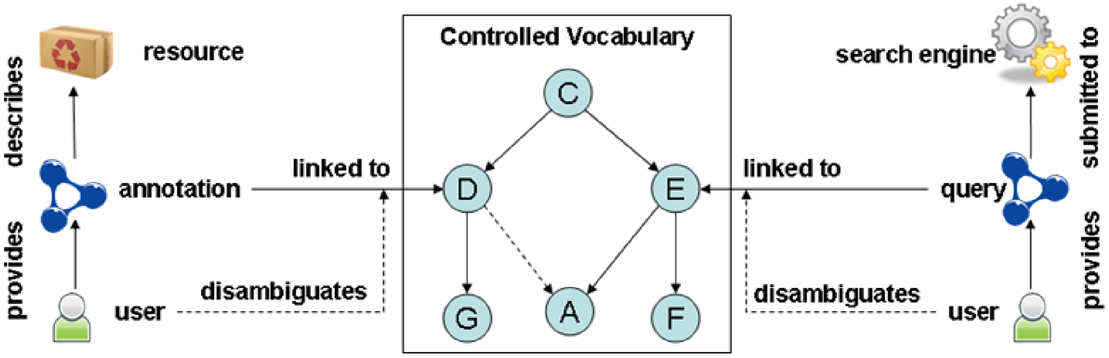


Fig. 13. Controlled Vocabulary based Annotation and Search

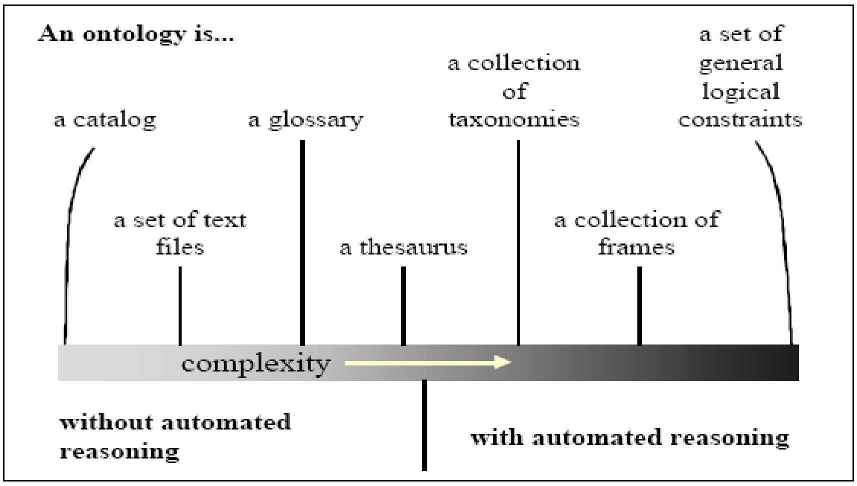


Fig. 14. A Classification of Various Forms of Ontologies According to their Level of Expressivity (Formality) [25].

the annotation process, the users provide (simpler form of) annotation elements and (semi-automatically) map them to the background ontology (see Figure 13). For instance, the user might tag a web page with the term "dog" and link it to the concept DOG in the background ontology. It is worth mentioning that both approaches can potentially enable automated reasoning.

In the rest of this section we discuss three kinds of annotation representation models which differ in the kind of vocabulary they are based on. In the selection of the models we followed the principle of staying within the spectrum of so-called *lightweight ontologies* [26]. The first model is not based on a controlled vocabulary, whereas the second and the third ones are. These last two models fall on the left and on the right side from the vertical bar shown in Figure 14 respectively. We show how the issues described above can be addressed in the last two approaches. Table 1 summarizes the semantic heterogeneity related issues and to which vocabulary types they apply.

3.1. Uncontrolled Vocabulary

In this model, annotation elements are not linked to controlled vocabulary terms and, therefore, are subjects to at least the three problems described in the introduction to this section. In other words, no controlled vocabulary exists as part of the model and free text is used instead. In principle, search in this model is implemented as syntactic matching of strings representing query terms with those used in annotation elements.

Pros The advantage of this model is that the user does not need to know about the existence of a controlled vocabulary and does not need to be involved

Type of vocabulary Issue	Uncontrolled vocabulary	Authority File	Taxonomy
Base form variation	√		
Polysemy	√		
Synonymy	√		
Specificity gap	√	√	

Table 1

Semantic heterogeneity issues and its relation to types of vocabularies

to help resolve ambiguities in linking annotation elements to controlled vocabulary terms.

The user involvement at annotation time is minimal as the simplest interface can be used: the user enters a list of free-text words in a text box.

CONS The main disadvantages of this model are the three problems described in the introduction to this section: base form variation, polysemy, synonymy, and the specificity gap.

Moreover, the minimal involvement at annotation time translates into a higher involvement required at the search and navigation time. The problem of heterogeneity described earlier produces a higher level of noise in the results of search and the users need to filter out manually the results they are interested in. At navigation time, the lack of explicit structure of the annotation provides little help and only simple clustering can be performed (such as the one illustrated in the Delicious tag cloud in Figure 4).

APPLICATIONS Delicious and Flickr are both good examples of systems that use uncontrolled vocabulary models. The users can enter any free-text tag they want to annotate their bookmarks and photos (respectively). The systems do not check for the validity of the tag or if it maps to a known meaning.

3.2. Authority file

This model is based on a simple form of controlled vocabularies called the *authority file*. In this controlled vocabulary, synonymous terms are grouped into concepts and one of the terms is selected as the concept name and used for visualization and navigation purposes (this term is called the *preferred term*) [27]. Each concept may have one or more associated terms and each term may belong to one or more concepts. In this model, user annotation elements are mapped to the controlled vocabulary terms and, consequently, to concepts which uniquely codify the meaning of the annotation elements. For example the tag “automobile” could be mapped to a concept with the preferred term “car” in the controlled vocabulary. Note that when a term belongs to more than one concept, the user may need to be involved in disambiguating the concept selection at annotation or query time.

PROS The model allows us to resolve the polysemy and synonymy problems by linking annotation elements and the user’s search queries to a predefined meaning (i.e., to a concept) in the controlled vocabulary. For example, if the user searches for the term “java” that is mapped to “programming language” in the controlled vocabulary, the search will not return resources annotated with the term “java” that refers to the “island”.

The user will also be able to navigate easily through all the documents related to a “programming language”, for example, without having to know the particular terms used to annotate these documents.

CONS Both at annotation time and at search time, the user will have to get involved more. At annotation, if there is an ambiguity in the mapping from the term the user provides and multiple concepts in the controlled vocabulary, the user will need to choose which concept the term refers to. At search time, again, if there is an ambiguity in the terms the users search for, they will have to specify explicitly which concept they are referring to. Alternatively, the user may decide to leave the concept disambiguation task to the system which could use the default concept for an ambiguous term (e.g., the one which is used more often) or use all its concepts for the annotation and search.

A second issue with this type of controlled vocabulary is the creation of the *authority file*. Each term in the vocabulary has to be mapped to one or multiple *preferred term* for the vocabulary to be useful; this requires time from the vocabulary creator(s). In addition, the coverage of the vocabulary will rarely be as wide as the full human vocabulary and the user might want to use a term that is not present in the vocabulary. Thus, this type of vocabulary is usually limited to applications in specific domains.

APPLICATIONS Faviki[19] is a social bookmarking system similar to delicious (see Figure 15). However, when users enter a tag, they are asked to disambiguate its meaning to a known concept. These concepts are automatically extracted from the contents of Wikipedia. This disambiguation allows a better convergence between the terms used for the annotation by different users as well as it proves better results at search time.

3.3. Taxonomy

This model is based on a controlled vocabulary called a *taxonomy*. A taxonomy is an extension of the authority files controlled vocabulary with the broader term (BT) and narrower term (NT) relations which are defined between controlled vocabulary concepts [27, 28]. See Figure 16 for an example of a taxonomy. Intuitively, a BT/NT relation might correspond to the *is-a* or *part-of* relation. For example, the concepts "cat" and "dog" can be linked to a higher level concept "mammal" which is itself linked to a concept "animal". Another example is a geographical taxonomy, like the one in Geonames[20], where countries are linked to the regions composing them which are in turn linked to cities, etc.

PROS In principle, this model allows us to solve the specificity gap problem because query terms can be mapped to user annotation elements through BT and NT relations and, therefore, resources which are more specific in meaning than the user query can also be retrieved. For example, the user can annotate two different resources with "cat" and "dog" and then retrieve both resources by searching for "mammal".

CONS In terms of user involvement, the taxonomy annotation model has the similar drawbacks as the previous model. The user will have to make sure that the annotation and search terms correspond to the correct node in the taxonomy if there is an ambiguity.

The taxonomy creation time is also an issue as it requires specific time, from an expert or from the actual user to create a meaningful classification of terms that can then be used for annotation.

Another disadvantage of taxonomies is the problem of the vocabulary granularity. If a third party (an expert) produces the taxonomy to the user, there is no guarantee that this taxonomy will be detailed enough for the user. For example, in the biological classification example earlier, the expert might not include the "vertebrate" vs. "invertebrate" split. When the user wants to perform a generic search for all "vertebrate", only a search for the higher term "animal" will be possible. In opposition, if the taxonomy provided by the expert is too detailed, the disambiguation task at annotation and search time will be more complex for the user that might not know all the differences between the same meaning of terms or might have to search between many different – but very close in sense – meanings.

APPLICATIONS Shoebox[21] is a photo management software that allows the users to annotate photos with annotations which are nodes in a taxonomy. The software comes with predefined categories (such as countries/regions/cities, or animal kingdom) but the users can also freely extend the taxonomy by creating their own categories and sub-categories. The user can then search for all photos tagged with an annotation corresponding to a category or one of its sub-categories as illustrated in Figure 17.

4. Level of User Collaboration

In this section we will describe a dimension related to how users contribute to the creation of different types of annotations and the vocabulary used in the process (see Section 3). We can roughly distinguish between two approaches:

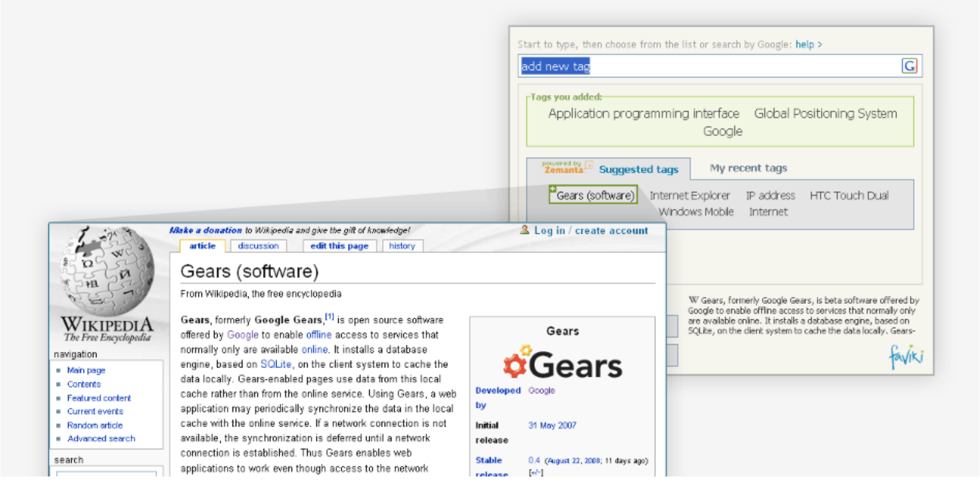
1. the single-user model, where a single user performs the task of either annotating resources or creating the vocabulary (or both),
2. the community model, where a set of users collaborate in the task of either annotating resources or creating the vocabulary (or both).

In this section we will focus on the interactions between users, and how these interactions affect, positively or negatively, the other two elements of our classification of annotations; these elements are the structural complexity of the annotation, i.e., how users interact with each other using some degree of structured

[19] http://faviki.com

[20] http://www.geonames.org/

[21] http://www.kavasoft.com/Shoebox



Start to type, then choose from the list or search by Google: help >

add new tag

Tags you added:

Application programming interface Global Positioning System Google

Gears, formerly Google Gears, is beta software offered by Google to enable offline access to services that normally only are available online. It installs a database engine, based on SQLite, on the client system to cache the data locally. Gears-

Make a donation to Wikipedia and give the gift of knowledge!

Gears (software)

From Wikipedia, the free encyclopedia

Gears, formerly **Google Gears**,[1] is open source software offered by Google to enable offline access to services that normally only are available online. It installs a database engine, based on SQLite, on the client system to cache the data locally. Gears-enabled pages use data from this local cache rather than from the online service. Using Gears, a web application may periodically synchronize the data in the local cache with the online service. If a network connection is not available, the synchronization is deferred until a network connection is established. Thus Gears enables web applications to work even though access to the network

Gears	
Developed by	Google
Initial release	31 May 2007
Stable release	0.4 (August 22, 2008; 11 days ago) [+/−]

Fig. 15. Faviki: Tagging with Concepts from Wikipedia (screenshot provided by faviki.com)

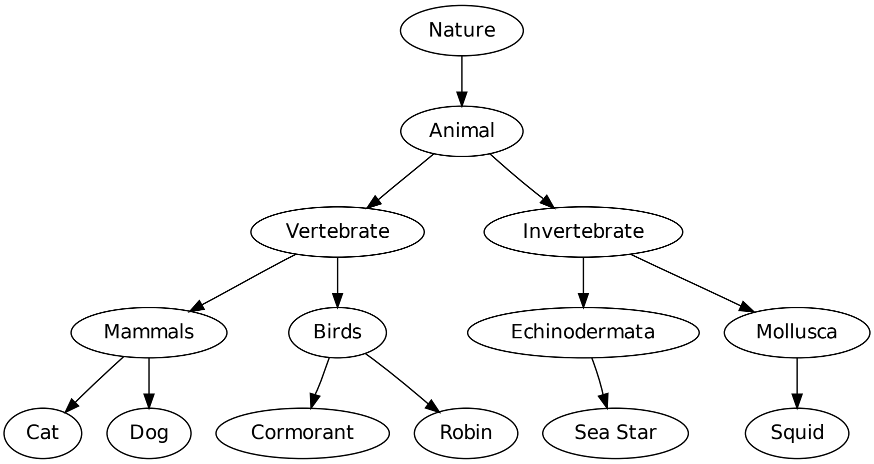


Fig. 16. Example of a Taxonomy

annotation (see Section 2) to annotate resources, and the type of vocabulary used to annotate the resources (see Section 3).

Considering the annotation perspective, in the single user model we can identify two sub-dimensions by considering who uses the annotations (see Table 2). A single user can annotate a resource for personal use only, or can annotate a resource considering that these annotations will be later used by many other users. In the remainder of this section we will explain how this consideration affects the annotation process.

Considering the vocabulary perspective we will focus on controlled vocabularies, since in the free-form annotation model (uncontrolled vocabulary), anyone is free to use any term (even terms not corresponding to a vocabulary) and hence there is little point on studying how interactions between users affect the vocabulary (as it is free). We will make clear how the interactions between users affect the construction of controlled vocabularies in the remainder of this Section.

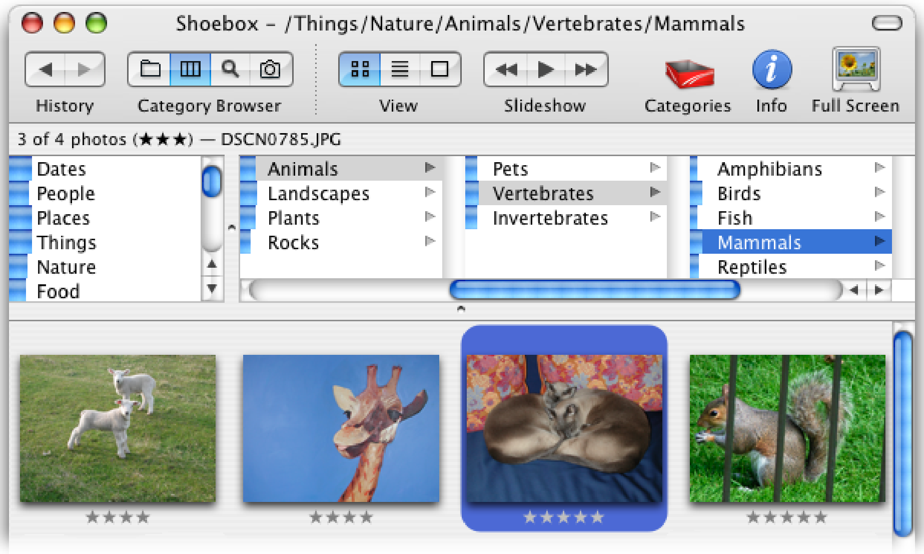


Fig. 17. Searching of Photos by Categories in Shoebox: Here Things > Nature > Animal > Vertebrate > Mammals.

Feature \ Type of annotation	Single User		Collaborative
	Private use	Public use	
User interaction during the annotation process	none	none	encouraged
Type of vocabulary	personal	personal	shared

Table 2

Types of Collaboration for Annotating Resources and Building Vocabularies

4.1. Single-user (Private use)

In this model a user annotates resources for personal purposes, usually to organize these resources for future search or navigation. Single users could also build their personal controlled vocabularies, but as we will see in the CONS part, the necessary work does typically not justify this task.

PROS The advantage of the *single-user annotation model for private use* is that each user is in full control of the annotations and since there is no sharing, the issues related to the semantic heterogeneity problem (see Section 3) is reduced to the scope of the single user. One can also argue that since the annotator and the user of the annotation are the same person, the annotations will reflect the personal taste and knowledge of the user, which in turn can be translated into more accurate results at search time.

CONS In this model, each single user has to annotate all the resources that he wants to have annotated for future use. These resources can be private resources (e.g., local files), resources shared by other users (e.g., shared photos), o publicly available resources such as Web pages. This annotation process has the following disadvantages:

- annotating all of the resources takes time and requires motivation (building personal controlled vocabularies to annotate the resources requires even more time and motivation). If the user has no strong incentives, the quality and coverage of the annotations and the controlled vocabulary cannot be guaranteed;
- the single annotator might not be an expert in the annotation process (or vocabulary building) and might miss relevant annotations (or terms) for a resource;
- users have to remember which terms they normally use to annotate a particular concept in a resource and use it consistently across the whole set of resources. If this is not done properly the user

will not be able to find the desired resources at search time (see Section 3).

APPLICATIONS Bookmarking is a well known example of annotation of Web resources for personal use. Currently. most Web browsers such as Mozilla Firefox[22], Google Chrome[23] and Internet Explorer[24] include this feature by default.

Other examples of personal annotation systems are Picasa[25] and Shoebox, where users tag their photos with free-form keywords. Picasa also adds the possibility of geo-tagging photos while Shoebox provides a controlled vocabulary in the form of taxonomy for the annotation.

Another system for the annotation of Web resources is Zotero[26], where users have the possibility to manage Web resources related to scientific research by classifying them and/or by adding free-form annotations.

4.2. Single-user (Public use)

In this model, a single user (normally an expert, or a small set of experts representing an institution) annotates resources with the goal of organizing the knowledge for a broader set of users. Library catalogs are perhaps the most well known example of this kind of model.

Considering the construction of the controlled vocabulary, in this model also a single user (also normally an expert or a small set of experts representing an institution) builds the controlled vocabulary to be used by a broader set of users in the annotation task and later for search. In this case, library classification schemes or thesauri are well known examples.

PROS Many people benefit from the work performed by a few experts of the field; the work (either the annotation or the resulting controlled vocabulary) is considered to be of good quality resulting in a good organization of the resources. In addition, experts in the annotation task are also able to use more complex and well structured vocabularies (see Section 3) as they are better trained in this task. Well studied and consistent controlled vocabularies developed by experts are very useful for solving the issues related to the semantic heterogeneity problem (see Section 3).

CONS It is normally costly to have dedicated experts to annotate and organize resources or to build controlled vocabularies. Moreover, in highly dynamic domains, the time lag between the publication of the resource and its annotation can be considerable [29]. This delay can be due to, either the scalability of the approach (work overload of the experts) making the classification of vast amounts of resources in short times impracticable, or to the fact that the field of knowledge is new or very dynamic and thorough studies have to be performed in order to reach consensus on the right vocabulary to be used for the annotations. Another issue with this approach is that the experts who annotate the resources are neither the authors nor the users of the annotations, therefore the annotations might not always reflect the users' perspective or the authors' intention [30]. Some possible solutions to these issues, still at research stage, are being proposed; for example, to let annotators define their classes and keywords that will later be approved by domain and annotation experts [31].

APPLICATIONS Library catalogs, such as the Library of Congress[27] and many other libraries fall into this model of annotation. Library classifications schemes used in library catalogs are examples of expert designed controlled vocabularies with the purpose of classification of books. There are also more general purpose controlled vocabularies designed by experts such as WordNet [32].

4.3. Collective annotation

In the collective annotation model, a set of users share their annotations about publicly available resources. These annotations are later also used by a set of users (possibly larger than the set of annotators) for navigation and search. In this model, the workload of annotating resources is distributed (in contrast to the previous 2 models). The users do not necessarily work together in the process of annotation, neither do they have to reach an agreement on the resource annotations (but obviously this could be the case).

Collaborative Tagging (or social tagging) is a well known model to annotate resources[28] with free-text tags [19]. When a critical mass of user-generated tags is available, these annotations can be used to generate folksonomies [21]. The main characteristic of folk-

[22] http://www.mozilla.com/firefox/

[23] http://www.google.com/chrome

[24] http://www.microsoft.com/windows/internet-explorer

[25] http://picasa.google.com/

[26] http://www.zotero.org/

[27] http://www.loc.gov/index.html

[28] mainly on the web.

sonomies (folk + taxonomies) is the bottom-up approach (i.e., from individuals to groups) for reaching consensus on emerging concepts, without any a priory agreement or controlled vocabularies.

Considering the construction of controlled vocabularies, the basic idea is to let users collaboratively build a controlled vocabulary using a bottom-up approach. This approach is reflected in several proposals in the literature. These proposals differ in whether there should be a pre-defined controlled vocabulary which will be collaboratively extended [33], or the controlled vocabulary should emerge bottom-up from scratch [2,29]. Depending on the model, users could work together in order to reach agreements in the construction process [29], or they could work independently and an automatic process could address the evolution of knowledge [2]. [24] proposed several generic characteristics and issues that any system for *emergent semantics* should consider and address. The notions presented in these proposals allows us to address disadvantages present in centrally built controlled vocabularies.

PROS The main advantage of the collaborative annotation model and construction of the controlled vocabulary is the distribution of the workload among all users of the system. This, in turn, lowers the barrier of user participation in the annotation process since one user can adopt the annotations that were assigned to a resource by others — thus simplifying the annotation task —, or even, more importantly, can search and discover new resources that are annotated with the same tags as the user's tags [34]. Another advantage is that users can express their own views when annotating resources, which in contrast with annotations made by experts, could simplify search and discovery of resources for other users with the same interests, considering that these resources were annotated with many potentially different points of views and interests.

In addition, the mass of data provided through the annotations in a collaborative model as well as the users' relations with these annotations can be used for automatic extraction of new knowledge (new terms and relations in the controlled vocabulary); for example, a recommendation system can be built to propose relevant annotations for a new resource, based on existing annotations of similar resources [35] or to extract co-relations from unstructured annotation schemes [36,37,2].

This model also provides the system with behavioral information about the users' interests through their interaction with other users' annotations and their own annotations. By using this information the system could infer a user model containing, for example, their education, background, interests or culture. Based on this user model the system can suggest and discover related resources or can discover hidden communities and sub-communities of users based not only on social relations (such as friends or coworkers) but also on interests and shared knowledge or background [2]; ebay and Amazon[29] are systems that use these techniques to suggest their consumer related products.

Building the controlled vocabulary in a bottom-up fashion can address the time lag problem for the inclusion of new concepts in the controlled vocabulary, therefore increasing steadily the precision and recall of any systems that is based in these kind of controlled vocabularies as these evolve dynamically and in direct response to the knowledge evolution of the involved users [24].

CONS Folksonomies and other collaborative tagging systems based on free-text annotations suffer from the lack of formal semantics that leads to the semantic heterogeneity problem, which makes indexing, search and navigation difficult [38]. This heterogeneity problem can be addressed by using controlled vocabularies or community-built controlled vocabularies that are better related to the domain of the resources.

However, even with the use of these vocabularies, social annotation approaches could suffer from biases that express personal likes or dislikes of the users, and not the actual intention of the resource author, examples of these annotations are "best paper", "don't bother with this one", "to read". This type of annotations cannot be avoided since the primary goal of any annotation model is to allow users to find the resources they are interested in. These kind of biased annotations could be also important for ranking purposes of the resources as well as for the users. The system could try to limit such biased annotations by automatically separating the personal annotation from the general annotations or by avoiding the mix between annotations that express biases and the ones that are not associated to the user's relative taste.

An important issue which remains open and needs to be treated in the community-built controlled vocabulary is the model to be used to allow knowledge to emerge from the contribution of each single user. The process of agreement (either automatic or user driven)

[29] http://www.amazon.com/

in knowledge evolution is still an interesting research issue to be addressed [24,29].

APPLICATIONS Diigo[30] and Delicious are examples of applications implementing this model of annotation to build a shared collection of bookmarks. Flickr is a social website for sharing and organizing photos and videos, users can comment resources, tag or add them to their own personal favorite list. These tags are later used for search and navigation. [38] proposed and extension for searching Flickr content where WordNet [32] is used to expand queries, using this approach users are able, for example, to find results about "automobiles" when searching for "cars".

The work presented by [39] incorporates the ideas of collaboration to library catalogs, where the users can become more than simple content consumers by annotating resources in digital libraries and become content creators. Facetag [40] is another initiative following this direction. The rationale in this system is to try to integrate top-down (i.e., from experts to individuals) and bottom-up classification in a semantic collaborative tagging system, incorporating the ideas of faceted classification, which helps users organize their resources for later search, navigation and discovery. Facetag incorporates the idea of collaborative annotation and collaborative controlled vocabularies in a single system.

Considering community-built controlled vocabularies, [29] proposed an ontology maturing process by which new knowledge is formalized in a bottom-up fashion going through several maturity stages: from informal emergent ideas using input from social annotation systems such as folksonomies, to formal lightweight ontologies via a learning process involving users. They also introduced[31], a system for collaborative ontology engineering and annotation.

Another related work to community-built controlled vocabularies was presented by
[2] where a model for formalizing the elements in an ontology evolution scenario was proposed. The model consists of three elements, Actor-Concept-Instance. This model has a straightforward parallel to the model adopted in this section to explain the annotation process as seen in Figure 1; where Actor is equivalent to *users*, Concepts to *annotations* and Instances to *resources*.

Tagpedia [33] tries to build a general purpose controlled vocabulary to overcome the fundamental problems of Wordnet and Wikipedia as standard controlled vocabularies. The main issues with WordNet are the lack of knowledge about entities (specially people) and the lack of support for incorporating new knowledge. Wikipedia contains a lot of information about entities but suffers from the lack of a more formal/ontological structure [33]. The idea of Tagpedia is to initially mine Wikipedia to construct an initial set of *syntags* (as opposed to synsets of WordNet) and to allow users to extend this initial set of *syntags* dynamically in a bottom-up manner.

5. Using the Annotation Model Dimensions

As we have mentioned earlier, the different dimensions that are discussed in the previous section are not only useful to classify the current state of the art but also to specify new requirements to create annotation models. In the current section, we describe the methodology to use this annotation dimensions to analyze end-users' requirements.

As an example, in Table 3 we provide a summary of the requirements of different online end-users systems for the *vocabulary type* and *structural complexity* dimensions. In addition, as part of the INSEMTIVES project, three different industrial use cases required annotation models without explicitly knowing what exactly they required. Presenting to these users the *Object-Subject-Predicate* general annotation model is actually quite difficult as it is abstract and distant to what the end-users might want to achieve in an annotation interface.

In the development of the annotation models for these use cases (discussed each separately in the following sections), we followed a methodology that consists of five steps which are schematically depicted in Figure 18 and described below:

1. **Analysis of the State-of-the-Art.** The state-of-the-art in the area of annotation models and systems has been throughoutly studied. A particular attention was given to systems operating on resources' kind that the use case partners are most interested in: textual documents (Use Case 2, see Section 5.2), web services (Use Case 1, see Section5.1), and multimedia (Use Case 3, see Section 5.3).

[30] http://www.diigo.com/
[31] http://www.soboleo.com/

Structural Complexity Vocabulary type	**Tags**	**Attributes**	**Relations**	**Ontologies**
Uncontrolled Vocabulary	*Flickr* annotates photos with free-text tags. *Delicious* annotates bookmarks with free-text tags.	*Facebook* allows users to describe their interests in books, movies, music with free-text fields.	*Freebase* allows its users to connect freebase topics with unlimited number of user defined relations.	*Semantic Wikipedia* users build ontology out of Wikipedia articles by annotating them with categories, attributes and typed links that can be created on the fly.
Authority File	*Faviki* annotates bookmarks with free-text tags mapped to a controlled vocabulary built from wikipedia[32].	*The "Encyclopaedia of Life"*[33] provides a set of controlled attribute based annotations (species, genus, family, order, etc.) for users to tag photos and videos of flora and animals with their corresponding biological classification.	*Facebook* allows the user to annotate photos and videos with links to people present in these media. *Flickr* ↔ *Last.fm/Upcoming.org* by adding a special tag to a flickr photo, users can link it to the events (described on last.fm or upcoming.org) where it was taken.	
Taxonomy	*Shoebox* allows users to tag their photos with tags present in a taxonomy. The user can then do "generic" searches for all the photos tagged under a node in this taxonomy.		The *Rhetorical Structure Theory* provides an annotation model to describe the argumentation relation between segments of text. The relations are annotated with a type that can be chosen from a taxonomy of 25 different types of relations classified under two categories "nucleus ↔ satellite" or "multinuclear".	

Table 3
Examples of Systems in the Complexity and Vocabulary Dimensions.

2. **Grouping and generalisation of key characteristics.** Based on the analysis of specific systems, three key groups of characteristics were identified and formulated as described in Sections 2, 3 and 4: *structural complexity*, *vocabulary type*, and *collaboration type*. These groups represent dimensions in which the existing systems can be classified based on their particular characteristics and features. The definition of the characteristics within the three groups was made through the analysis and generalisation of specific characteristics of each of the analysed single systems. As we show in the following sections, these generalised characteristics can be used to define a generic annotation model that would address the possibly diverse needs of use cases in all kind of domains.
3. **Requirements identification and analysis.** In order to identify the needs of the use case partners for the different characteristics of the annotation model, a semi-structured interview was performed with a representative of each use case partner. Before the interviews, each use case partner was given the report on the state-of-the-art

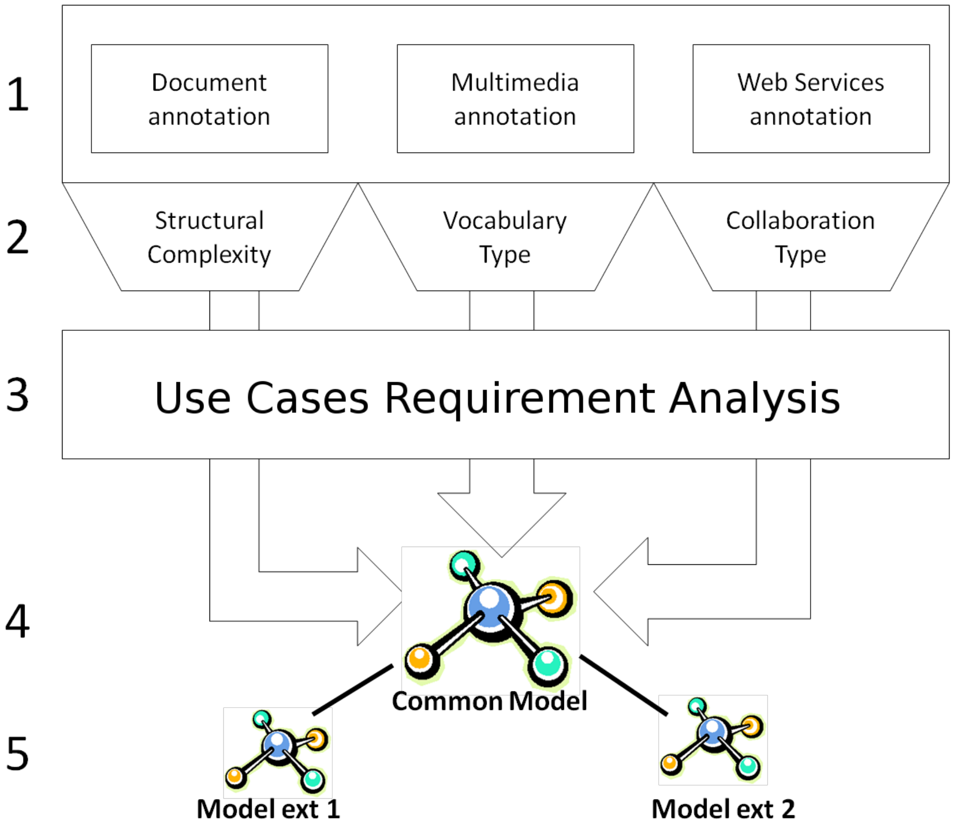


Fig. 18. Annotation Model Development Methodology

on annotation systems (as presented in the Sections 2, 3 and 4). This allowed us to increase the awareness of the use case partners of the existing systems and their features and helped the partners make more informed decisions about their needs. The questions formulated during the interviews were made as nontechnical as possible but with clear connections to the elements of the annotation model they referred to. The questions were clarified during the interviews. The answers were then rigorously analysed with the goal of the identification of the requirements for the annotation model for each use case partner as reported in [41].

4. **Definition of a common annotation model.** Based on the analysis of the requirements, an annotation model that covers common needs of all the use case partners was identified (see Table 4) and formalized in [42]. It is noteworthy that the proposed annotation model is generic in the sense that it does not depend on a particular resource kind, application domain or knowledge domain; it allows the model to be used in a large spectrum of annotation applications going beyond those of the use case partners.
5. **Definition of model extensions.** The requirements of the use case partners that could not be addressed by the common annotation model were addressed within model extensions presented in [42].

As the classification of the requirements is independent from lower level implementation details, it is easy to present to the use case partners that might not be experts in RDF, database schemas, etc. However, with an understanding of this classification and these implementation technology, it is then also easy to decide, depending on the specific requirements of the use case partners, how the annotation models developed from the requirements were mapped to lower level implementation details, in particular to decide if the use of an RDF based store was needed or if simpler dedicated technologies could be used.

	STRUCTURAL COMPLEXITY				VOCABULARY TYPE			COLLABORATION TYPE
	Tag	**Attributes**	**Relations**	**Ontologies**	**Free-Text**	**Authority file**	**Taxonomy**	**Provider**
Use Case 1	√	√	√		√		√	experts
Use Case 2	√	√	√		√		√	experts+users
Use Case 3	√	√	√	√			√	experts

Table 4

Summary of the Requirements for each Use Case

5.1. Case Study One – Web Service Annotation

Web Service annotation was first approached in the area of Semantic Web services (SWS) [43]. The aim of SWS is the automation of specific Web service usage tasks such as discovery, selection, composition by using semantic descriptions to make them machine processable.

The most popular SWS approaches are WSMO [44], OWL-S [45], and SWSF [46] each of which puts a semantic layer on top of existing (technical) Web service descriptions. WSMO for instance allows the "annotation" of functionality and behavior of Web services. Further it allows the attchement of so-called non-functional properties whose intention is to describe non-functional aspects of Web services such as the creator, rights, or quality of service.

Approaches which directly integrate semantic annotations into the WSDL files are WSDL-S [47] and its successor SAWSDL.[34]. Both provide means to point to semantic descriptions outside of the WSDL or to conversion services to obtain a semantic description of various aspects described in it.

More end-user based annotations are provided on platforms such as Strikeiron[35], the ProgrammableWeb[36], or the Seekda[37] portals. StrikeIron hosts a Web service marketplace which foremost groups services in pre-defined categories. While there are several features for the web services, such as pricing, sample code and other, this features are maintained by internal experts and there is no option for end users to provide more annotations. ProgrammableWeb (PW) hosts a directory of Web services offering detailed descriptions of APIs which can be annotated using natural language and also some controlled vocabularies.

We analyze the one of the aforementioned web service annotation portals, mainly the Web service annotation services provided on the portal offered by Use Case 1 (cf. Figure 19). The portal provides an attribute-based description of Web services as can be seen in Figure 19. Besides that, the platform provides means to add Wiki-based comments, ratings, or free tags to Web services as such it realizes a collaborative approach to Web service annotation.

We report on the requirements for the semantic annotation model which conforms to the requirements of a platform that supports end-user based semantic Web service annotation. These requirements were developed from the results of the interview reported in [41] and are structured according to the annotation features presented in Sections 2, 3 and 4.

The use case demands for a solution to annotate Web services, their properties, and different aspects of stakeholders engaged in the Web services such as their providers. The annotation should, but not exclusively, point to aspects already described in WSDL files. As such annotations might point to functional aspects or to descriptive elements.

5.1.1. Structural Complexity

The Web service annotation use case requires a mixed complexity in the annotation model. Due to considerable differences in user characteristics, an annotation procedure should support quick and easy annotations but also very detailed annotations. The annotators are familiar with *simple annotation mechanism* which supports tagging, rating, and commenting of resources.

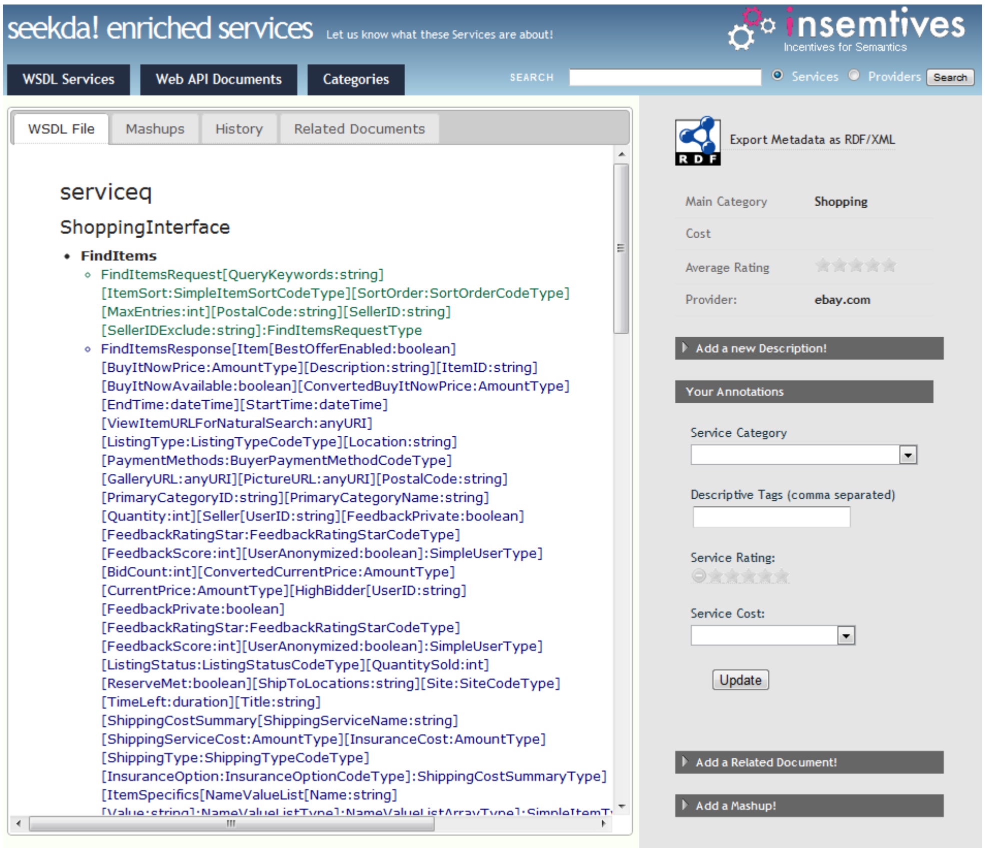
Users should be able to annotate specific attributes which are consequently used in search for filtering and navigation. Furthermore, semantic relations between

[34] `http://www.w3.org/TR/sawsdl/`

[35] `http://www.strikeiron.com`

[36] `http://www.programmableweb.com/`

[37] `http://www.seekda.com`



serviceq

ShoppingInterface

- **FindItems**
  - FindItemsRequest[QueryKeywords:string][ItemSort:SimpleItemSortCodeType][SortOrder:SortOrderCodeType][MaxEntries:int][PostalCode:string][SellerID:string][SellerIDExclude:string]:FindItemsRequestType
  - FindItemsResponse[Item[BestOfferEnabled:boolean][BuyItNowPrice:AmountType][Description:string][ItemID:string][BuyItNowAvailable:boolean][ConvertedBuyItNowPrice:AmountType][EndTime:dateTime][StartTime:dateTime][ViewItemURLForNaturalSearch:anyURI][ListingType:ListingTypeCodeType][Location:string][PaymentMethods:BuyerPaymentMethodCodeType][GalleryURL:anyURI][PictureURL:anyURI][PostalCode:string][PrimaryCategoryID:string][PrimaryCategoryName:string][Quantity:int][Seller[UserID:string][FeedbackPrivate:boolean][FeedbackRatingStar:FeedbackRatingStarCodeType][FeedbackScore:int][UserAnonymized:boolean]:SimpleUserType][BidCount:int][ConvertedCurrentPrice:AmountType][CurrentPrice:AmountType][HighBidder[UserID:string][FeedbackPrivate:boolean][FeedbackRatingStar:FeedbackRatingStarCodeType][FeedbackScore:int][UserAnonymized:boolean]:SimpleUserType][ListingStatus:ListingStatusCodeType][QuantitySold:int][ReserveMet:boolean][ShipToLocations:string][Site:SiteCodeType][TimeLeft:duration][Title:string][ShippingCostSummary[ShippingServiceName:string][ShippingServiceCost:AmountType][InsuranceCost:AmountType][ShippingType:ShippingTypeCodeType][InsuranceOption:InsuranceOptionCodeType]:ShippingCostSummaryType][ItemSpecifics[NameValueList[Name:string][Value:string]:NameValueListType]:NameValueListArrayType]:SimpleItemT

Main Category	Shopping
Cost	
Average Rating	☆☆☆☆☆
Provider:	ebay.com

Service Category

Descriptive Tags (comma separated)

Service Rating:

Service Cost:

Fig. 19. Web Service Description on the Use Case 1 Marketplace

different resources should be expressible. The annotation model requires support *tags* which are not associated to any particular property of the resource, but also *attributes* to describe specific metadata and semantic *relations* between resources for navigation, applicable to the web service as a whole, and also to its parts (operations, parameters, ...).

5.1.2. Vocabulary Type

Different types of vocabularies are required, ranging from free-text to ontology-based annotations. Different user groups might either enter free-text comments or tags, while others provide structured and detailed knowledge about properties of a resource.

The controlled vocabularies are predefined and will provided and maintained by the use case being the current category based on the *Service Finder Service Category Ontology*[38]. Current experiments within the use case use an ontology for annotation[39].

The use case interest in resolving of synonymy and polysemy and to expand queries demands at least for the representation of an *authority file*. Free text tags should be mapped to general purpose vocabularies such as Wordnet [32] or DBPedia [48] in order to enhance search and navigation capabilities; however, this mapping process should be performed as seamlessly as possible for the annotating user. Furthermore, users might be interested in seeing detailed classification information or clustering of results, which will require a *taxonomy*.

5.1.3. Collaboration Type

The use case requires the annotation of shared resources. Access to them is restricted on the level of an-

[38] http://www.service-finder.eu/ontologies/ServiceCategories

[39] athttp://www.service-finder.eu/ontologies/ServiceOntology

notations which means that every user can view a resource, but he should not be able to see every annotation available for it. Search for resources is consequently not limited.

Resources can be annotated by many users. Some properties can however only be described by the owner of a resource. The prospected collaborative annotation model should thus further embed provenance information as well as versioning information. The use case demands for a collaborative annotation model in which users can also extend the underlying general purpose vocabulary, but only experts can maintain the classification scheme.

5.2. Case Study Two – Corporate Portal

In this section we report on the requirements for the annotation model to be used in a Intranet Corporate Portal domain. The use case aims to enrich the target portal with automatic, semi-automatic and user-guided semantic annotation capabilities as a way to increase the performance of the portal providing enhanced features such as semantic search, faceted navigation and intelligent recommendations.

Users will be able to annotate blog entries and forum posts in HTML format but will also be able to annotate news videos. For all type of resources, the use case requires to be able to annotate parts of the resource (i.e., sentences, paragraphs for text and segments for videos).

5.2.1. Structural Complexity

The use case requires a mixed complexity in the annotation model. Users are only interested in an *easy and quick* annotation procedure and the annotators are already used to a *simple tagging interfaces*.

However, in the use of the annotation for searching and navigating, the users should be able to *filter out* searches by using particular attributes (e.g., the author, a date range, etc.). There is also interest in having an annotation model that supports relational annotations to link resources together, but also parts of resources for navigating between them.

The annotation model should thus be able to support *tags* which are not associated to any particular property of the resource, but also *attributes* to describe specific metadata and *relations* between (and within) resources for navigation.

5.2.2. Vocabulary Type

During the interview, users made clear that the annotation should not be limited to a specific controlled vocabulary and that users will enter free-text annotations. However, after further discussion and by looking at the intended usage, it appears that the free-text annotation will be combined with a controlled vocabulary of some sort, provided by experts. The controlled vocabulary might take the form of a taxonomy to ease the navigation and search, and the users will be ready to align their free-text annotations to the concepts in the controlled vocabulary.

A set of expert users, with a good knowledge of the domain as well as of the annotation system, will provide an initial controlled vocabulary that can be used to bootstrap the recommendation system. This vocabulary is then extended by the users when providing free-text annotations.

There is also a strong requirement that the issues of synonymy and polysemy should be resolved for improving the search accuracy. This will either require an *authority file* or a *taxonomy* and resources that are annotated with unambiguous metadata. Hence, this use case will require a controlled vocabulary where users will help disambiguating the annotation to map *free-text tags* to the corresponding concept in the vocabulary.

5.2.3. Collaboration Type

The use case will have many shared resources that every user can access, annotate, and search for. With the exception of some of the resources where there is a need for access control, all the resources and annotations will be available to everyone.

Moderators will be controlling the user's annotations to make sure they are correct, where traceability of the annotations is needed for auditing purposes. The collaborative annotation model thus should embed provenance information as well as versioning information.

In addition, the controlled vocabulary required will be built both in a *top-down* fashion, with a small set of experts creating the initial vocabulary but also in a *bottom-up* fashion, with users and moderators extending the existing vocabulary and annotating resources on their own.

5.3. Case Study Three – Online Games

The use case will provide an annotation platform for users of an online virtual world. The annotators are supposed to annotate (astrological) images, videos, and multi-layer Flash content representing virtual landscapes. The content formats used are Flash and various image formats. There is a requirement to annotate parts of all content types which means spatial regions in images, spatio-temporal regions in videos, and regions in layers of the Flash content.

5.3.1. Structural Complexity

The use case requires a mixed complexity in the annotation model. The user characteristics of possible annotators demand for an *easy to use* annotation procedure which hides most of the complexity behind the annotation interface. There is no indication that the prospected annotators are used to annotation interfaces.

The annotators should in some cases be able to annotate specific aspects or attributes of the content and annotations shall be attached to both resources and parts of it. Furthermore, there is an interest in a model that supports relationships to be able to identify semantically related resources.

The annotation model should thus be able to support *tags* which are associated to the *whole resource, but also parts with it*. Further it should be possible to describe specific *attributes* of resources which refers to the *attribute*-model. Finally, *relationships* between resources shall be described (cf. the *relation*-model).

5.3.2. Vocabulary Type

As expressed in the interview, the annotations in the online game use case should be solely based on *controlled vocabularies* which can only be defined by administrators of the platform.

The use case also considers offering thematical browsing based on taxonomies or query expansion strategies which requires a *taxonomy* with which resources are annotated with.

The use case requires annotation based on more structured knowledge as narrative structures shall be represented. Hence this use case will require a *controlled vocabulary* which might be organized in a *taxonomy* or richer knowledge structures.

5.3.3. Collaboration Type

The game use case will have different types of resources that any user of the platform can annotate. The created annotations will always be available to everyone and access will not be restricted. Annotators and searchers might be different depending on the sub-use case.

There is no need to keep track of a version history for the annotations, but the username and a timestamp have to be stored to allow their traceability.

The game use case thus demands for a *collaborative annotation* model in which the vocabulary used will be provided.

6. Conclusions

This article investigates existing models for representing annotations, and analyses their different characteristics, forms, and functions. Based on this analyses, a classification scheme for annotation models was developed that distinguishes three main dimensions:

- structural complexity of annotations,
- the type of the vocabularies used,
- the collaboration type supported.

Furthermore a comprehensive overview of existing annotation approaches is given to provide examples of annotation models and their different characteristics.

The article further reports on a methodology to use this annotation model classification scheme to elicit requirements for new annotation models in end-user applications. This methodology is then illustrated with three use cases from which requirements for annotation models have been extracted. As an extension to this work, we further abstract from these requirements in order to be able to address needs of a wider spectrum of applications.

To conclude, below we provide our recommendations for the configuration of a balanced semantic annotation system that, on one hand, encapsulates enough semantics to allow the development of new semantics-based services for the end-user and, on the other hand, does not require an explicit expertise from such a user to provide semantic annotations, which would facilitate a wider adoption of such systems. Our recommendation is based on the analysis and findings reported in this article and we detailed for each of the three dimensions, as reported below:

Structural Complexity of Annotations While the users should be able to provide annotations of the tag, attribute, and relation kinds (see Section 2), they should be driven towards providing more relation and attribute to improve the semantic quality of service. In this case, the user will provide qualified resource descriptions and interlinks resources with qualified links. This results in a structure that can be naturally mapped into RDF and, therefore, can enable the adoption and reuse of services developed by the Semantic Web community so far.

In addition, these lightweight but structure-rich annotations can form the basis for the automatic extraction of data sets to be uploaded to the LOD cloud. The use of the full fledged ontology for annotation is not recommended when the user is not an expert in semantic web technologies due to high learning barriers [1];

Vocabulary type We recommended that, during the annotation process, the user is *supported* by a controlled vocabulary of *taxonomy* type. This will enable one of the mostly used reasoning tasks, the subsumption on the hierarchy of concepts (see Section 3.3). From the user perspective it means, among other things, that the semantic gap problem can be addressed at the application level (see the introduction to Section 5.3.2). While the users should be driven towards providing controlled annotations, they should not be restricted from providing free text annotations. This is to support the users who are well familiar with the existing Web 2.0 annotation systems and to overcome the potential problem of a partial coverage of the controlled vocabulary, which can be addressed, at least in part, by adopting the recommendation below;

Level of User Collaboration Considering the annotation process, whenever possible, it is recommended that the workload of annotating resources is distributed among the users of the system. This is to leverage the "wisdom of crowd" effect – the concept that has been successfully implemented in many Web 2.0 applications such as Delicious. Considering the vocabulary type, it is recommended that the underlying controlled vocabulary is evolved dynamically by the users in order to address the problem of partial knowledge and time lag between creation of concepts inside communities and their inclusion in the controlled vocabulary by experts. From the user perspective, this means that they are more in control of the vocabulary, but from the system design perspective, this means that new methodologies for evolving knowledge are required. In order to avoid the "cold start" problem, it is recommended to provide an initial controlled vocabulary (possibly designed by experts). Because the users will have to deal with the semantics of annotations, proper user interface and interaction tools need to be developed that would expose to the user these semantics in a simple and natural way. For example, when the user enters "java" as a tag annotation, the system can show the recognised meaning of the tag as, for example, "java (island)" that would help the user understand if the meaning is the intended one and correct it otherwise.

Acknowledgements

This work has been partly supported by the INSEMTIVES project (FP7-231181, see `http://www.insemtives.eu`).

The authors would like to thank Denys Babenko, Tobias Bürger, and Borislav Popov for their valuable contribution made at early stages of this work.

References

[1] A. L. Rector, N. Drummond, M. Horridge, J. Rogers, H. Knublauch, R. Stevens, H. Wang, and C. Wroe. OWL pizzas: Practical experience of teaching OWL-DL: Common errors & common patterns. In *EKAW*, pages 63–81, 2004.

[2] Peter Mika. Ontologies are us: A unified model of social networks and semantics. In *International Semantic Web Conference*, pages 522–536, 2005.

[3] Flickr. Flickr blog. `http://blog.flickr.net/en/2010/09/19/5000000000/`, September 2010.

[4] David K. Gifford, Pierre Jouvelot, Mark A. Sheldon, James W. O'toole, and Jr. Semantic file systems. In *In Proceedings of the 13th ACM Symposium on Operating Systems Principles*, pages 16–25, 1991.

[5] C Reed and G Rowe. Araucaria: Software for puzzles in argument diagramming and XML. Technical report, Department of Applied Computing. University of Dundee Technical Report., 2001.

[6] Bill Mann. An introduction to rhetorical structure theory (RST). `http://www.sil.org/\`, 1999.

[7] Atanas Kiryakov, Borislav Popov, Ivan Terziev, Dimitar Manov, and Damyan Ognyanoff. Semantic annotation, indexing, and retrieval. *Elsevier's Journal of Web Semantics*, 2(1), 2005.

[8] Rudi Studer, Richard Benjamins, and Dieter Fensel. Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1-2):161–198, MAR 1998.

[9] Web Ontology Working Group. W3c recommendation for owl. `http://www.w3.org/TR/2004/REC-owl-features-20040210/`, February 2004.

[10] J. Euzenat, A. Mocan, and F. Scharffe. Ontology alignments. In Martin Hepp, Pieter De Leenheer, Aldo De Moor, and York Sure, editors, *Ontology Management: Semantic Web, Semantic Web Services, and Business Applications*, volume 6 of *Semantic Web and Beyond*. Springer, 2007.

[11] A. Gomez-Perez and David Manzano-Macho. A survey of ontology learning methods and techniques. Technical report, OntoWeb Deliverable 1.5, 2003.

[12] Katharina Siorpaes, Christian Hofer, and Elena Simperl. Analysis of semantic content authoring processes. Technical report, UIBK, September 2010.

[13] Soren Auer, Sebastian Dietzold, , Thomas Riechert, and Thomas Riechert. Ontowiki - a tool for social, semantic collaboration. In *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006*, pages 736–749. Springer, 2006.

[14] Markus Krötzsch, Denny Vrandečić, Max Völkel, Heiko Haller, and Rudi Studer. Semantic Wikipedia. *Journal of Web Semantics*, 5(4):251–261, 2007.

[15] Borislav Popov, Atanas Kiryakov, Damyan Ognyanoff, Dimitar Manov, and Angel Kirilov. Kim - a semantic platform for information extraction and retrieval. *Journal of Natural Language Engineering*, 10(3-4):375–392, September 2004.

[16] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, 2002.

[17] Atanas Kiryakov, Damyan Ognyanov, and Dimitar Manov. Owlim âĂŞ a pragmatic semantic repository for owl. In *Int. Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005), WISE 2005*. Springer-Verlag LNCS series, November 2005.

[18] Sesame. `http://openrdf.org`. (last accessed on 01.06.2009).

[19] Scott Golder and Bernardo A. Huberman. The structure of collaborative tagging systems. *Journal of Information Science*, 32(2):198ÂŰ208, April 2006.

[20] Amy J. Warner. A taxonomy primer. `http://www.ischool.utexas.edu/~i385e/readings/Warner-aTaxonomyPrimer.html`. (last accessed on 01.06.2009).

[21] Thomas Vander Wal. Folksonomy: Coinage and definition. `http://www.vanderwal.net/folksonomy.html`. (last accessed on 01.06.2009).

[22] Thomas Gruber. Ontology of folksonomy: A mash-up of apples and oranges. *International Journal on Semantic Web & Information Systems*, 3(2):1–11, 2007.

[23] Louise Spiteri. Structure and form of folksonomy tags: The road to the public library catalogue. *Webology*, 4(2), 2007.

[24] Karl Aberer, Philippe C. Mauroux, Aris M. Ouksel, Tiziana Catarci, Mohand S. Hacid, Arantza Illarramendi, Vipul Kashyap, Massimo Mecella, Eduardo Mena, Erich J. Neuhold, and Et. Emergent semantics principles and issues. In *Database Systems for Advances Applications (DASFAA 2004), Proceedings*, pages 25–38. Springer, March 2004.

[25] Barry Smith and Christopher Welty. Fois introduction: Ontology—towards a new synthesis. In *FOIS '01: Proceedings of the international conference on Formal Ontology in Information Systems*, New York, NY, USA, 2001. ACM Press.

[26] F. Giunchiglia and I. Zaihrayeu. Lightweight ontologies. In Ling Liu and M. Tamer Ozsu, editors, *Encyclopedia of Database Systems*. Springer, July 2009.

[27] Lars M. Garshol. Metadata? thesauri? taxonomies? topic maps! making sense of it all. *Journal of Information Science*, 30(4):378–391, 2004.

[28] Fred Leise, Karl Fast, and Mike Steckel. What is a controlled vocabulary? `http://www.boxesandarrows.com/view/what_is_a_controlled_vocabulary_`, 2002.

[29] Simone Braun, Andreas Schmidt, Andreas Walter, Gabor Nagypal, and Valentin Zacharias. Ontology maturing: a collaborative web 2.0 approach to ontology engineering. In *Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge (CKC 2007) at the 16th International World Wide Web Conference (WWW2007)*, 2007.

[30] Michael Buckland. Vocabulary as a central concept in information science. In *Digital Libraries: Interdisciplinary Concepts, Challenges, and Opportunities. Proceedings of the Third International Conference on Conceptions of Library and Information Science*, Dubrovnik, Croatia, May 1999.

[31] Adam Mathes. Folksonomies - cooperative classification and communication through shared metadata. Technical report, Graduate School of Library and Information Science. University of Illinois Urbana-Champaign, December 2004.

[32] George A. Miller. WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[33] Francesco Ronzano, Andrea Marchetti, and Maurizio Tesconi. Tagpedia: a semantic reference to describe and search for web resources. In *SWKM 2008: Intl. Workshop on Social Web and Knowledge Management @ WWW*, 2008.

[34] M Ames and M Naaman. Why we tag: motivations for annotation in mobile and online media. In *CHI 2007 - Tags, Tagging and Notetaking*, pages 971–980, San Jose, 2007.

[35] S Sood, K Hammond, S Owsley, and L Birnbaum. Tagassist: Automatic tag suggestion for blog posts. *International Conference on Weblogs and Social Media*, 2007.

[36] Tye Rattenbury, Nathaniel Good, and Mor Naaman. Towards automatic extraction of event and place semantics from flickr tags. In *SIRIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 103–110, New York, NY, USA, 2007. ACM Press.

[37] P. Schmitz. Inducing ontology from flickr tags. In *Proc. of the Collaborative Web Tagging Workshop (WWW ÂŠ06)*, May 2006.

[38] Josef Kolbitsch. WordFlickr: a solution to the vocabulary problem in social tagging systems. In *Proceedings of I-MEDIA*, 2007.

[39] Rich Gazan. Social annotations in digital library collections. *D-Lib*, Volume 14 Number 11/12, December 2008.

[40] Manuele Quintarelli, Andrea Resmini, and Luca Rosati. Facetag: Integrating bottom-up and top-down classification in a social tagging system. In *IASummit*, Las Vegas, 2007.

[41] Tobias Bürger, Ilya Zaihrayeu, Pierre Andrews, Denys Babenko, Juan Pane, and Borislav Popov. Report on the state-of-the-art and requirements for annotation representation models. Technical report, UIBK, UNITN, ONTOTEXT, June 2009.

[42] Tobias Burger, Olga Morozova, Ilya Zaihrayeu, Pierre Andrews, Juan Pane, and Borislav Popov. D2.1.2: Specification of models for representing single-user and community-based annotations of web resources. Technical report, UIBK, UNITN, ONTOTEXT, September 2009.

[43] S. MCIllraith, T. C. Son, and H. Zeng. Semantic web services. *IEEE Intelligent Systems. Special Issue on the Semantic Web*, 16(2):46–53, 2001.

[44] D. Roman, U. Keller, H. Lausen, J. de Bruijn, R. Lara, M. Stollberg, A. Polleres, C. Feier, C. Bussler, and D. Fensel. Web service modeling ontology. *Applied Ontology*, 1(1):77–106, 2005.

[45] David Martin, Mark Burstein, Jerry Hobbs, Ora Lassila, Drew McDermott, Sheila McIlraith, Srini Narayanan, Massimo Paolucci, Bijan Parsia, Terry Payne Evren Sirin, and Naveen Srinivasan Katia Sycara. Owl-s: Semantic markup for

web services. W3C Member Submission, 22. November 2004, 2004.

[46] Steve Battle, Abraham Bernstein, Harold Boley, Benjamin Grosof, Michael Gruninger, Richard Hull, Michael Kifer, David Martin, Sheila McIlraith, Deborah McGuinness, Jianwen Su, and Said Tabet. Semantic web services framework. W3C Member Submission 9 September 2005, 2005.

[47] Rama Akkiraju, Joel Farrell, John Miller, Meenakshi Nagarajan, Marc-Thomas Schmidt, Amit Sheth, and Kunal Verma. Web service semantics: Wsdl-s. W3C Member Submission, 07. November 2005, 2005.

[48] S. Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. DBpedia: a nucleus for a web of open data. In *The Semantic Web*, pages 722–735. Springer, 2008.