Metadata mechanisms: from ontology to folksonomy ... and back

Stijn Christiaens

Semantics Technology and Applications Research Laboratory
Vrije Universiteit Brussel
stijn.christiaens@vub.ac.be

Abstract. In this paper we give a brief overview of different metadata mechanisms (like ontologies and folksonomies) and how they relate to each other. We identify major strengths and weaknesses of these mechanisms. We claim that these mechanisms can be classified from restricted (e.g., ontology) to free (e.g., free text tagging). In our view, these mechanisms should not be used in isolation, but rather as complementary solutions, in a continuous process wherein the strong points of one increase the semantic depth of the other. We give an overview of early active research already going on in this direction and propose that methodologies to support this process be developed. We demonstrate a possible approach, in which we mix tagging, taxonomy and ontology.

Keywords: tagging, folksonomy, community informatics, faceted classification, ontology, Semantic Web

1 Introduction

The vast amount of information currently available can make finding the exact piece of information a tedious process. Despite todays raw search engine power, finding what you seek sometimes requires something more fine-grained, something more specific. For instance, you visit your town's website when you try to find a form for their administration, not a generic search engine. The problem in todays information sources is that the data is highly unstructured [1] or semi-structured and that meaning is only visible to human agents. Homepages, blogs, forums and others contain valuable community-produced knowledge, but the search engines have difficulties identifying and retrieving the knowledge you are looking for.

The Semantic Web [2] is the next generation of the WWW, a Web in which all content has machine-processable meaning. This Semantic Web provides all the functionality needed to build the Pragmatic Web [3, 4] on top of it. Communities will no longer search, but rather find and use information in this Pragmatic Web. The explicit meaning, understandable by both human and machine agents, attached to content is necessary for proper information retrieval and usage.

It is clear that mechanisms are needed to incorporate or annotate content with semantics, with some form of meaning in order to increase machine-understanding. Research on the Semantic Web as well as current trends in the Web (the so-called Web 2.0

[5]) resulted in several mechanisms. However these mechanisms do not all provide the same semantic depth. They have different goals, users and granularities.

In section 2 we give an overview of currently existing mechanisms. We describe their purpose and their main strong and weak points. We then discuss how these mechanisms could benefit from the others in section 2.3 and list some early research in this direction. We give an example of mechanism cooperation and how it benefits communities in section 3. Finally, we end with conclusions and suggestions for possible future work.

2 Meaning mechanisms

In this section we give a brief overview of several meaning mechanisms. It is not our intention to fully describe each of these mechanisms, but to introduce (or refresh) them to the reader for facilitation of further sections.

2.1 Overview

The earliest form of metadata to describe meaning is the introduction of keywords. These are labels with which the author (creator, publisher, ...) of the content describes his content. In the early days of the web, these keywords could (and can still) be used by search engines for information retrieval.

Tagging is the process of describing the *aboutness* of an object using a tag (a descriptive label). In the current (so-called) Web 2.0 trend (e.g., http://del.icio.us), tagging is done by the observers of the content. These tags can now be shared and used by all members of the community. The organic organization that grows through this sharing was coined folksonomy by Thomas Vander Wal [6]. Vander Wal [7] distinguishes between a broad and a narrow folksonomy. In a broad folksonomy all users can tag the visible content, while in a narrow folksonomy only the author tags his content.

A taxonomy is a hierarchical classification of things. It is mostly created by the designer of the system or a knowledge engineer with domain knowledge. Authors (or categorizers) must find a good place in this hierarchy to position their content.

In faceted classification [8], objects are described by facets. A facet is one isolated perspective on the object (e.g., color of wine). Each facet has a set of terms that are allowed (e.g., red and white for wine color). According to Kwasnick [9] there are many advantages in faceted classification, in particular the flexibility and the pragmatic appeal.

Gruber [10] writes that an ontology is *an explicit specification of a conceptualization*. An improved definition was provided by Borst [11]: *ontologies are defined as a formal specification of a shared conceptualization*. Ontologies are seen as the technology to enable the Semantic Web and many ontology languages and approaches have been developed [12], for instance RDF [13] and OWL [14].

2.2 Comparison

We will focus on the main contributions of the meaning mechanisms described in the previous subsection. It is not our intention to perform a full-scale analysis of all mechanisms, but rather to identify major strengths and weaknesses.

The author-created keyword provides a very precise (high quality) view on the content. Unfortunately, they form a one-person perspective. His information can never equal the amount that an entire community can deliver (low quantity). Narrow folksonomies suffer the same disadvantages as author-created keywords as they are very similar[1]. Broad folksonomies break free of the one-person-perspective and deliver an entire community-view on the content. The drawback is of course the quality of the metadata. For instance, a tag "toread" is not very useful except for the person who labeled the content that way. A folksonomy can also be used to compile a tag profile for users. This way, both content and people can be found (see e.g. [15]).

On the other hand, we have the more heavy-weight mechanisms. High quality taxonomies, facet classification and ontologies are costly to create and maintain. The most expensive in creation and maintenance is an ontology. It requires consensual agreement on its contents from community members. Their main benefit is that these mechanisms deliver very rich meaning. For instance, if information is annotated by means of an ontology, it can be queried for specific information using a conceptual query language (e.g., [16]), much like a database can be queried (e.g., using SQL[2]). However, given that most searches on the web are restricted to one or two keywords, it seems not likely that people will learn to use such a query language. These mechanisms seem the most distant from real-world users and are criticized as such (e.g., [17]).

2.3 Mechanism collaboration

As we explained in the subsection, each mechanism has its own advantages when compared to the others. We propose to divide the mechanisms in two groups: free and restricted. A free mechanism is one that allows anyone (both creator and observer) to annotate the content with any label he desires. A restricted mechanism is one that fixes the metadata, and all content must satisfy it. Observers can not annotate the content with their own labels. Free mechanisms are popular, but receive critique that the resulting information might not be of the desired quality. Restricted mechanisms seem less popular, but provide higher quality metadata. However, they are said to be too static, too inflexible for the ever evolving real-world situation.

In Figure 1 the listed mechanisms[3] are positioned according to our division. The feedback loop in the figure indicates *the way to go* in meaning mechanisms. This closely corresponds to the SECI model [18], where tacit knowledge is made explicit, combined, and converted again to add to the existing tacit knowledge. Ideally, a third zone should

[1] Similar, but not the same, as keywords are placed in technology (e.g., HTML) and for use by agents (e.g., a crawler) and narrow folksonomies are displayed next to the content for use by both human and machine agents.

[2] http://en.wikipedia.org/wiki/SQL

[3] Note that our list of mechanisms is not an exhaustive one. Other mechanisms may (and probably do) exists and will fit in this division as well.

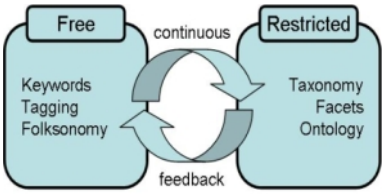

Fig. 1. Continuous feedback loop

be created, a gray area where meaning can benefit from both free and restricted mechanisms[4]. In our opinion, this will result in a system that receives the benefit from the free zone (quantity) as well as from the restricted zone (quality).

Research in this gray zone is in its early stages and seems mainly focused on pushing systems in the free zone towards more quality (e.g., [19, 20]). We believe this is because of the current popularity of these systems. Other research (e.g., [21, 22]) takes meaning in the restricted zone as the seed in order to guide users to richer annotation. In the ontology world, we mention DOGMA-MESS [23], an ontology engineering methodology for communities. It succeeds in bringing the restrictedness of ontology a step closer to the free zone in a *messy*, but structured and guided process.

Apart from improvements based on benefits of one side (either free or restricted), methodologies and mechanisms should be created that actually use and reinforce the gray zone (and thus both sides as well).

3 The Guide: a research lab's memory

3.1 Folksonomy and taxonomy

The Guide is the community portal in STARLab[5]. The community members are the people working at STARLab. The Guide is powered by Drupal[6], an open source content management system. Its modular approach and multitude of possibilities deliver an endless plethora of possibilities for *community plumbing*. At STARLab, we use the Guide to remember and retrieve all useful information (ranging from technical issues over team building to deep theoretical disscussions). Some content is completely free (e.g., a blog post can be about anything), while other information has to be placed in special containers (e.g., book pages and forum topics). All posts have to be tagged by the author herself. This way a narrow folksonomy emerges in the Guide, which we can visualize in a tag cloud[7].

[4] According to the conjunction of "folk" and "taxonomy", a folksonomy would seem to be in the gray zone already, but there is no restriction present in a folksonomy. It is simply a flat list of tags that users attach to all content. More quality is needed to move it to the gray zone.

[5] http://www.starlab.vub.ac.be

[6] http://www.drupal.org

[7] http://en.wikipedia.org/wiki/Tag_cloud

Fig. 2. From tag cloud to taxonomy in the Guide

Figure 2 displays the Guides' tag cloud at the age of month six[8]. The figure also displays part of a taxonomy we distilled from the tag cloud. We analyzed all available tags in order to get a good insight in the emerging categorization. We then grouped relevant tags together (e.g., RoadMap and RoadMap preparation) and ordered them from generic to specific. Finally, we looked for even more generic terms to label the groups (e.g., Strategy). The end-result was a basic taxonomy that brings more structure to the Guide. The approach we use here is rather ad-hoc, and if different people (or even the same person at different times) create the taxonomy in this manner, we would end up with different results. However, for our current research the end-result is satisfactory. It is clear that the content will evolve continuously implying that the taxonomy will have to follow this evolution. This update will have to occur frequently, and as such, it is important that the construction of the taxonomy is kept as light (viz. neither complex nor time-consuming) as possible.

3.2 Guide ontology

In order to take full advantage of all content present, we used the STARLab's DOGMA [24, 25] approach to build a basic ontology for our Guide. This ontology captures the meaningful relations between all information objects in the Guide. Figure 3 displays the Guide ontology in NORM tree representation [26]. The formalization of the meaning in the ontology will allow us to perform reasoning and provide easy rule creation. For instance, because the system knows that `Post causes Comment` and that `Post is categorized by Tag`, we can easily add a rule stating that if `Comment 'C' belongs to Blog Post 'BP'` and `Blog Post 'BP' is categorized by Tag 'T'`, that `Comment 'C' is categorized by Tag 'T'` is valid as well. If we combine this with knowledge present in the taxonomy, we can for instance find from `Forum Topic 'id54' is categorized by Tag 'Dogma Studio'`,

[8] Note that the topics RoadMap and RoadMap preparation are the largest in display. As we are currently in the process of construction a RoadMap at STARLab. The tag cloud correctly represents these as hot topics as a lot of people tag current posts in this area.

that `Forum Topic 'id54' is categorized by Tag 'Development'` is true as well.

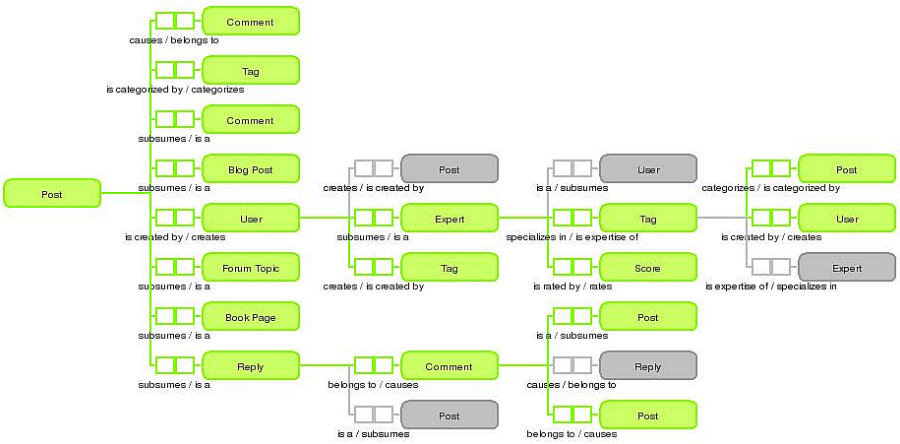

Fig. 3. Guide ontology in NORM tree representation. Gray-colored concepts represent duplicate occurrences appearing in the browsing process. These are included in the visualization in order to create a complete local context of the concept in focus.

This ontology describes the Guide overall content system. As this is relatively static[9], the ontology will also be static. We foresee that we will have to update the ontology rarely.

3.3 Example application

In the previous two subsections, we described how we enriched our Guide with both a taxonomy (to bring structure to the user-created tags) and an ontology (to describe the structure of the Guide itself). Now that this extra meaning is attached to the content, we must find ways in which we can usefully *apply* it. This way we can deliver valuable extra functionality to the users. We will demonstrate how we can achieve expert-group construction through our enriched Guide.

Suppose a new researcher starts working at STARLab. She is very interested in learning all about STARLab's previous, current and future work. The Guide holds all this information, but despite[10] both the tags and the structured posts (forum, book pages, ...), she feels a little unsure about where to start. She would like to get in touch with the right people, to ask the right questions and get the right answers. She navigates to the Expert Finder page. The new researcher is asked what she would like to know more about. She types 'Development', because she wants to talk to the people who built the

[9] Relative, as new types of posts (e.g., events) can be added in Drupal using modules.

[10] Or maybe because of the amount of structure and tags, as a large amount of such information is just as difficult to handle as a large amount of actual content.

tools for full understanding. The Guide now searches all posts that have been tagged 'Development'. This search results in too few posts, so the system adds the tags 'Dogma Studio' and 'Concept Definition Service' as these are categorized under 'Development' in the current taxonomy (see subsection 3.1). The Guide now looks at the people who *created* these posts and at the people who *commented* on these posts (see the rule example in subsection 3.2), and ranks them as experts according to their activity in these posts. The system sets up a new forum topic saying that there is a new researcher looking for information about develoment in STARLab. It invites the two highest ranked experts[11] by email to discuss development issues with the new researcher. As soon as she determines that all answers are found, she can close this topic. If another new researcher comes by and looks for experts on 'Development', he will first be guided to the existing forum topic. If he does not find what he is looking for there, the process can start again.

The approach we followed and our example application can easily be transferred to other real-world problems. Consider the problem of integration of a community of immigrants into a native population. For these people, getting started is very difficult, as the amount of information is much greater than present in our Guide. The language aspect only adds to the complexity. A visit to the webpage of the town might help, but this community would benefit much more from a separate information source, targeted to their specific problems concerning integration, much like our Guide. The town's administration could offer a basic setup like a community portal, with some high-level structure (corresponding in general to the native information pages) and a less restricted part (like blogs). In cooperation with active individuals from the integrating community, this setup could be presented in the correct language. All community members are invited to join this portal and encouraged to locate and post solutions to issues there.

For instance, someone needs a form to indicate that he desires his administrative mails in French, rather than in Dutch. After several difficult visits to several administrative services, he locates the appropriate form and procedures. He posts the specifics of his administrative adventure on the portal (in a blog or specific forum topic) and tags it appropriately in his own language. On regular intervals, the folksonomy is converted into a taxonomy, which is coupled with the native town portal's information as well. Using functionality as described in our Guide example, the details on how to otbain and use the form can be located even more easily. The original discoverer of this information will be regarded as an expert. This approach will encourage people to use the system, and as such, make it usable and turn it into an active community *driver*.

4 Conclusions and future work

In this work we gave a brief overview of several meaning mechanisms and what their main advantages are. We divided these in free and restricted mechanisms and stated that

[11] Note that these two people might not be regarded as the best develoment experts by their colleagues, but their number of posts indicate that they might be the most *helpful*. In the end, this is what is important for our current search.

free mechanisms tend to provide quantitative (but flexible) data, while restricted mechanisms could deliver more qualitative (but static) data. We identified the gray zone, which combines both sides and joins quality with quantity. Which side actually *seeds* the process of meaning generation is less important. We claim that there is need for processes and methodologies to support the continuous feedback loop legitimately. Both the community members and knowledge engineers must be active players in these processes in order to reach and benefit from the gray zone of meaning mechanisms. Only by combining quality with quantity in a legitimate manner can we achieve truly meaningful metadata. Meaningful not only for humans or machines, but for *both*. The example based on our Guide shows that it is indeed beneficial to move into the gray zone. Using only basic meaning mechanism collaboration, we transformed the tags into a taxonomy and combined it with an ontology. The end-result was a system that provides a lot of possibilities for empowering communities. We gave a brief example on how to apply our approach in other, more significant real-world problems.

Future work should focus on this gray zone, and methodologies and mechanisms that thrive there. We described research that already entered this area, and we feel that these results provide motivation and grounds for more research. In our own work, we will focus on how we can improve our Guide example. Our current approach was rather ad-hoc (e.g., conversion of tags into taxonomy). We need to research further how we can turn this into solid meaning formalization and negotiation. Furthermore, we have to look at ways to create even more integration between the different mechanisms. We will also have to keep working on how we can then bring all this to actual application, and how this approach can benefit community members.

Acknowledgments The research described in this paper was partially sponsored by the EU IP 027905 Prolix project and the Leonardo B/04/B/F/PP-144.339 CODRIVE project. We would like to thank our colleagues and Tanguy Coenen, Céline Van Damme and Eiblin Matthys from MOSI (www.vub.ac.be/MOSI) for their interesting feedback and discussions.

References

1. Robert Blumberg and Shaku Atre. The problem with unstructured data. *DM Review Magazine, February 2003*, 2003. http://www.dmreview.com/article_sub.cfm?articleId=6287.
2. Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American 284(5)*, pages 34–43, 2001.
3. Aldo de Moor, Mary Keeler, and Gary Richmond. Towards a pragmatic web. In *Proc. of the 10th International Conference on Conceptual Structures, (ICCS 2002), Borovets, Bulgaria*, Lecture Notes in Computer Science. Springer-Verlag, 2002.
4. M.P. Singh. The pragmatic web. *Internet Computing Volume 6 Issue 3*, pages 4–5, May/June 2002.
5. Tim O'Reilly. What is web 2.0: Design patterns and business models for the next generation of software, 09-30-2005. http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html.
6. Gene Smith. Atomiq: Folksonomy: social classification, Aug 3, 2004. http://atomiq.org/archives/2004/08/folksonomy_social_classification.html.

7. Thomas Vander Wal. Explaining and showing broad and narrow folksonomies, February 21, 2005. http://www.personalinfocloud.com/2005/02/explaining_and_.html.
8. B.C. Vickery. *Faceted classification: a guide to construction and use of special schemes*. London: Aslib, 1960.
9. Barbara H. Kwasnick. The role of classification in knowledge representation and discovery. *Library Trends 48(1)*, pages 22–47, 1999.
10. TR Gruber. A translation approach to portable ontology specification. *Knowledge Acquisition 5(2)*, pages 199–220, 1993.
11. WN Borst. *Construction of Engineering Ontologies*. Centre for Telematica and Information Technology, University of Tweenty. Enschede, The Netherlands, 1997.
12. Asunción Gómez-Pérez, Oscar Corcho, and Mariano Fernández-López. *Ontological Engineering*. Springer-Verlag New York, LLC, 2003.
13. Eric Miller and Frank Manola. RDF primer. W3C recommendation, W3C, February 2004. http://www.w3.org/TR/2004/REC-rdf-primer-20040210/.
14. Frank van Harmelen and Deborah L. McGuinness. OWL web ontology language overview. W3C recommendation, W3C, February 2004. http://www.w3.org/TR/2004/REC-owl-features-20040210/.
15. John Ajita and Dorée Seligmann. Collaborative tagging and expertise in the enterprise, WWW2006, Edinburgh, UK.
16. Anthony C. Bloesch and Terry A. Halpin. Conquer: A conceptual query language. In *ER '96: Proceedings of the 15th International Conference on Conceptual Modeling*, pages 121–133, London, UK, 1996. Springer-Verlag.
17. Clay Shirky. Ontology is overrated: Categories, links, and tags, 2005. http://www.shirky.com/writings/ontology_overrated.html.
18. Ikujiro Nonaka and Noboru Konno. The concept of ba: Building foundation for knowledge creation. *California Management Review Vol 40, No.3*, Spring 1998.
19. Zhichen Xu, Yun Fu, Jianchang Mao, and Difu Su. Towards the semantic web: Collaborative tag suggestions, WWW2006, Edinburgh, UK.
20. Patrick Schmitz. Inducing ontology from flickr tags, WWW2006, Edinburgh, UK.
21. Yannis Tzitzikas and Anastasia Analyti. Mining the meaningful term conjunctions from materialised faceted taxonomies: Algorithms and complexity. *Knowledge and Information Systems 9(4)*, pages 430–467, 2006.
22. Judit Bar-Ilan, Snunith Shoham, Asher Idan, Yitzchack Miller, and Aviv Shachak. Structured vs. unstructured tagging - a case study, WWW2006, Edinburgh, UK.
23. A. de Moor, P. De Leenheer, and R.A. Meersman. DOGMA-MESS: A meaning evolution support system for interorganizational ontology engineering. In *Proc. of the 14th International Conference on Conceptual Structures, (ICCS 2006), Aalborg, Denmark*, Lecture Notes in Computer Science. Springer-Verlag, 2006.
24. P. De Leenheer and R. Meersman. Towards a formal foundation of dogma ontology: part i. Technical Report STAR-2005-06, VUB STARLab, Brussel, 2005.
25. P. De Leenheer, A. de Moor, and R. Meersman. Context dependency management in ontology engineering. Technical Report STAR-2006-03-01, VUB STARLab, Brussel, March 2006.
26. Trog D. and Vereecken J. Context-driven visualization for ontology engineering. Master's thesis, Vrije Universiteit Brussel, 2006.