MUTO: The Modular Unified Tagging Ontology

Steffen Lohmann
Universidad Carlos III de Madrid, Avenida de la Universidad 30, Leganés Madrid, Spain
slohmann@inf.uc3m.es

Paloma Díaz
Universidad Carlos III de Madrid, Avenida de la Universidad 30, Leganés Madrid, Spain
pdp@inf.uc3m.es

Ignacio Aedo
Universidad Carlos III de Madrid, Avenida de la Universidad 30, Leganés Madrid, Spain
aedo@ia.uc3m.es

ABSTRACT

Tagging has become a popular indexing method within the last years and can be considered one of the cornerstones of the Social Web. Several ontologies have been developed that aim to formally describe tagging and folksonomies in order to improve their interoperability and processability. However, each of these ontologies covers different aspects of the domain; finding the right ontology for a certain purpose and aligning it correctly with other ontologies is difficult. This paper critically reviews available tagging ontologies and presents a unified vocabulary that combines the 'best of' these ontologies in one consistent schema. A central goal was to ensure high concept reuse by simultaneously avoiding redundancies. A modular design was chosen to reduce complexity and prevent inconsistencies. It groups the different elements of tagging and separates advanced concepts from the core ontology. Key design decisions are justified and modeling alternatives are discussed, not only to explain the unified ontology but also to contribute to a better understanding of the conceptual space of tagging and folksonomies.

Categories and Subject Descriptors

D.2.13 [**Software Engineering**]: Reusable Software—*domain engineering, reuse models*; H.2.1 [**Database Management**]: Logical Design—*data models, schema*

General Terms

Design, Languages, Standardization

Keywords

Tagging, folksonomy, ontology, unification, modularization, review, MUTO, Semantic Web, RDF, OWL.

1. INTRODUCTION

It is now more than seven years since tagging – i.e., the allocation of freely chosen text labels to digital resources by users – emerged on the Web [48]. Meanwhile, tagging has become a popular indexing method in many software programs, not only but especially in Social Web applications, such as social bookmarking or media sharing websites [30, 40]. Tagging in these environments is often called *social* or *collaborative tagging* as it creates useful links between tags, resources, and users. The resulting link structure has come to be known as *folksonomy* and is a valuable source for social navigation, collaborative filtering, and information retrieval, among others.

The full potential of folksonomies, however, cannot be exploited due to their low formal semantics. Ambiguity and synonymity are not the only problem of folksonomies [29], but also the lack of a common representation and exchange format that facilitates interoperable processing of folksonomies. In response, several ontologies have been developed within the last years that aim to overcome these limitations by providing an explicit specification for the representation of folksonomies based on Semantic Web technologies.

The large number and variety of proposed tagging ontologies, however, makes it difficult and time consuming for developers to find the ontology that meets their needs best. Since each tagging ontology focuses on a different aspect, one ontology is not sufficient in most cases but a combination of ontologies is required [33]. To further complicate matters, many of the existing tagging ontologies are only hardly alignable with each other due to conceptual incompatibilities. A unification of these different approaches in one consistent schema is thus highly demanded, not only to ease the development of semantically interoperable tagging systems but also to contribute to a better understanding of the domain of tagging and its conceptualization.

The development of such a unified ontology was the main goal of the work presented in this paper. We critically reviewed available tagging ontologies and combined the 'best of' these ontologies in one consistent conceptual schema that we call the *Modular Unified Tagging Ontology (MUTO)*. MUTO's modular architecture enables developers to use only the parts they need. It furthermore ensures high stability and scalability of the core vocabulary with regard to future evolutions in the domain of tagging. Further goals in the design of MUTO were to capture all essential tagging information, to reuse related ontologies, to avoid redundancies and to support different forms of tagging, in particular *semantic tagging* (as described e.g. in [38, 47]).

The paper starts with the review of existing tagging ontologies in Section 2. We then classify the tagging concepts

I-SEMANTICS 2011, 7th Int. Conf. on Semantic Systems, Sept. 7-9, 2011, Graz, Austria

found in the review and discuss basic design considerations for a unified vocabulary in Section 3. Section 4 describes the MUTO core ontology in detail and explains central design decisions and possible alternatives. The paper concludes with a discussion of the approach and a comparison to related work in Section 5.

2. REVIEW OF TAGGING ONTOLOGIES

Relevant work on the semantic representation of folksonomies started in 2005. Among the most influential attempts at that time were the creation of a first tagging ontology by Newman et al. [46], the formal definition of a tripartite model for folksonomies by Mika [42], and two talks about basic ideas and concepts for an "ontology of folksonomy" by Gruber [27, 28], followed by the formation of the *TagCommons* initiative [16] to further develop these ideas. These early attempts identified and defined the key elements of tagging ontologies, namely *resources*, *tags*, and *users* that are all interconnected by *taggings*. Some further concepts of tagging, such as date and time information [46], relations between tags [46, 42], or the source of tagging (i.e. the used tagging system [27]) were also considered.

2.1 Newman's Tag Ontology (TAGS)

The early "Tag Ontology" (`TAGS`[1]) by Newman et al. [46] provides a good starting point for any modeling attempt, as it defines the fundamental conceptual structure of tagging. A basic decision in the design of `TAGS` was to assign all key elements, including tags, a URI for unique identification on the Web [46]. Accordingly, `TAGS` defines a class `tags:Tag` for the representation of tags, instead of using simple literals. Likewise, a class `tags:Tagging` is defined which reifies "the n-ary relationship between a tagger, a tag, a resource, and a date." The definition of such a class is fundamental to any tagging ontology in order to capture the tripartite character of folksonomies, as it is formally described by Mika [42].[2]

Another fundamental decision in the development of `TAGS` was to reuse existing ontologies for concepts that are not tagging-specific. For instance, `foaf:Agent` from the "Friend of a Friend" (`FOAF`) vocabulary is used to represent users, the tagging date (`tags:taggedOn`) is a subproperty of `dc:date` from the "Dublin Core Metadata Element Set" (`DC`) and `tags:Tag` a subclass of `skos:Concept` from the "Simple Knowledge Organization System" (`SKOS`) vocabulary.

2.2 Knerr's Tagging Ontology (TO)

The early `TAGS` ontology was followed by a number of further tagging ontologies in the subsequent years, each focusing on different aspects of tagging. The "Tagging Ontology" (`TO`) by Knerr [35], for instance, defines a `to:ServiceDomain` class to represent the used tagging system, as proposed by Gruber [27]. `TO` furthermore allows to define the visibility of taggings (via the class `to:VisibilityEnum` with instances `to:Private`, `to:Public`, and `to:Protected`) and the type of the tagged resource by linking the "DCMI Type Vocabulary" (`DCTYPE`). Although Knerr mentions the `TAGS` ontology, he does not align `TO` to it. However, similar to `TAGS`, `TO` imports concepts from the popular `DC`, `FOAF`, and `SKOS` vocabularies.

[1] Vocabularies are abbreviated by their common namespace prefixes in the following. Since there are no common namespace prefixes for most tagging ontologies so far, the prefixes used in the MUTO ontology and this paper are summarized in Table 1. The namespace URIs and prefixes for all vocabularies referenced in this paper are given in Table 3 in the appendix

[2] To emphasize the importance of the tagging class, we prefer the term "tagging ontology" over "tag ontology".

2.3 Echarte's Ontology of Folksonomy (OF)

The "Ontology of Folksonomy" (`OF`) published by Echarte et al. [26] in 2007 does also not explicitly refer to `TAGS`. In contrast to `TO`, `OF` does not even link to any other vocabulary, though it reuses some popular concepts, such as the `SKOS` distinction of *preferred*, *alternative*, and *hidden labels* that it applies to tag labels. Similar to `TO`, `OF` considers the source of tagging (`of:Source`) and additionally incorporates Gruber's idea of giving tags a polarity [27] (via `of:hasPolarity`).

A novel aspect brought into play by `OF` is the representation of the tags' positions within the list of tags that a tagging consists of. Regardless of whether the position of a tag might have a specific meaning to users or not, users would expect the ordering of the tags in a tagging to remain the same whenever they access the tagging. Thus, the position of a tag is essential information that should be included in any comprehensive tagging ontology. Strangely, none of the reviewed tagging ontologies except from `OF` is capable to represent this information.

2.4 Social Semantic Cloud of Tags (SCOT)

The "Social Semantic Cloud of Tags" (`SCOT`) [32] and "Meaning of a Tag" (`MOAT`) [8, 47] ontologies from the years 2007 and 2008 are the first two tagging ontologies that explicitly reuse and extend the `TAGS` ontology. They are also the first tagging ontologies that integrate the "Semantically-Interlinked Online Communities" (`SIOC`) vocabulary.

`SCOT` focuses on "collective" tagging activities and reuses `SIOC` to represent groups of users (via `sioc:Usergroup`). It defines two new classes with several properties for the representation and sharing of *tag clouds* (`scot:Tagcloud`) and tag co-occurences (`scot:Cooccurrence`). However, most of this information can be easily inferred from basic tagging concepts and thus does not need to be redundantly represented. A separate storage as proposed by `SCOT` might make sense in certain cases – e.g., to facilitate querying or to shorten query response times – but is better avoided in a general conceptualization of tagging that we aim for with MUTO, since the ontology gets unnecessary complex and prone to inconsistencies.

2.5 Meaning of a Tag (MOAT)

`MOAT` [8, 47] is the second tagging ontology that explicitly reuses and extends `TAGS`. In particular, it adds a `moat:Meaning` class and corresponding properties to the `TAGS` vocabulary. This advancement is of special interest to the Semantic Web community and the idea of semantic tagging [38], as it allows to disambiguate tags by linking to well-defined concepts, such as *DBpedia* resources [23] or other instances from the *Linking Open Data* project [7, 22]. `MOAT` is also the first tagging ontology that considers *automatic tagging*, e.g. via keyphrase extraction [41], by defining a `moat:TagType` class with instances `moat:AutomatedTagging` and `moat:ValidatedTagging` (the latter being for manual taggings).

Name	Abbr.*	Authors	1st publication (latest update)	Newly introduced concepts	Reused vocabularies**
Tag Ontology	TAGS	Newman et al.	2005-03-23 (2005-12-21)	Fundamental structure, restricted tagging	DC, FOAF, SKOS
Tagging Ontology	TO	Knerr	2006 (2007-01-15)	Tagging source, private tagging	DC, DCTERMS, DCTYPE, FOAF, SKOS, XSD
Ontology of Folksonomy	OF	Echarte et al.	2007 (—)	Tag position and polarity	XSD
Social Semantic Cloud of Tags	SCOT	Kim et al.	2007-03-23 (2008-06-13)	Tag clouds and co-occurrences	FOAF, SIOC, XSD (DC, SKOS via TAGS)
Meaning of a Tag	MOAT	Passant & Laublet	2008-01-15 (—)	Tag meaning, automatic tagging	FOAF, SIOC, (DC, SKOS via TAGS)
Upper Tag Ontology	UTO	Ding et al.	2008 (—)	Tagging note, voting via tags	DCTERMS, FOAF, SIOC, SKOS, XSD
Common Tag	CTAG	Tori et al.	2009-06-08 (—)	Author vs. reader tags	DCTERMS, (MOAT, SIOC, SIOCT, SKOS, TAGS)
TAGora Tagging Ontology	TT	Szomszor et al.	2009 (2010)	Automatic tag sense disambiguation	XSD
NiceTag Ontology	NT	Limpens et al.	2009-01-09 (2010-09-09)	Named graphs, tag functions and forms	FOAF,IRW, SIOC, RDFG
Modular Unified Tagging Ontology	MUTO	Lohmann et al.	*2011-09* (—)	Unification, modularization	DCTERMS, SIOC, SKOS, XSD (FOAF, all tagging ont.)

* Abbreviations as used in this paper and in the MUTO ontology.
** Vocabularies in brackets are not directly integrated but via other ontologies or separate schemas.

Table 1: Basic information on the reviewed tagging ontologies in chronological order.

2.6 Common Tag (CTAG)

The idea of linking tags with well-defined concepts from the Semantic Web is also adopted by the "Common Tag" (`CTAG`) ontology that was released in 2009 [3]. `CTAG` is intended to be a "minimal" tagging vocabulary for embedding into XHTML via RDFa. It does not distinguish between taggings and tags but represents all information in one single `ctag:Tag` class. It also misses a specification of how users are represented or linked, as the authors expect `CTAG` to be extended as needed with "additional information from other RDF vocabularies" [3]. Hence, `CTAG` rather relies on the emergence of conventions than offering a comprehensive specification of the domain of tagging.

Conceptually, `CTAG` extends `MOAT`'s dichotomy of *manual* and *automatic tagging* by distinguishing between `ctag:AuthorTag` and `ctag:ReaderTag` in addition to `ctag:AutoTag`. `CTAG` defines all concepts in one namespace, without integrating concepts from other vocabularies (except from `dcterms:created`). Mappings to related concepts of `SIOC`, `SIOCT`, `TAGS`, `MOAT`, and `SKOS` are, however, listed in a separate schema.

2.7 Upper Tag Ontology (UTO)

Similar to `CTAG`, the "Upper Tag Ontology" (`UTO`) [25] defines a basic vocabulary of consistently named tagging concepts. The main difference is that `UTO` serves as an upper ontology instead of a minimal vocabulary like `CTAG`. As such, it aims to provide a general description of the domain of tagging that other ontologies can be aligned to. However, `UTO` misses some important concepts, such as private taggings or tag positions (see above).

Instead, `UTO` introduces the new classes `uto:Comment` and `uto:Vote`. The first represents user notes that are entered along with a tagging, as supported by several tagging systems. The latter captures the aspect of "voting by tagging" which can be differently realized, for instance, via tag labels (e.g. "`*****`" indicates a star rating, "5/10" a scale rating) or by counting the number of users who tagged a resource.

Similar to `CTAG`, `UTO` defines all concepts in one namespace that is enriched by mappings to `DCTERMS`, `FOAF`, `SIOC`, and `SKOS`. However, whereas `CTAG` uses purely subsumptions (`rdfs:subClassOf`, `rdfs:subPropertyOf`) and separates all mappings from the core ontology, `UTO` uses also equivalence relations (`owl:equivalentClass`, `owl:equivalentProperty`) and includes these relations in the core ontology.

2.8 Tagora Tagging Ontology (TT)

Another tagging ontology has been developed within the larger context of the TAGora project [17], as part of the "TAGora Sense Repository (TSR)" [18]. Similar to `MOAT` and `CTAG`, TSR centers around semantic tagging and the mapping of tags to well-defined resources from the Semantic Web, in particular to *DBpedia* resources and *WordNet synsets* [44]. However, whereas `MOAT` and `CTAG` support the manual disambiguation of tag meanings, TSR tries an automatic matching based on information about tag frequencies and usage data [49], similar as it is represented in `SCOT`.

The TSR services are based on three ontologies: A tagging ontology (the "Tagora Tagging Ontology", `TT`), an ontology for enriching DBpedia resource descriptions, and an ontology that describes the tag sense disambiguation. All three ontologies were primarily designed for the TAGora project and do not provide a general conceptualization of the domain of tagging.

2.9 NiceTag Ontology (NT)

The most recently released tagging ontology that we could find for this review is the "NiceTag Ontology" (`NT`) [12, 45, 37] by Limpens et al. Its first version was published in January 2009 under the name "Semantically Related Tag Ontology"; since then it underwent several changes with the latest update in September 2010. Limpens et al. emphasize the manifold forms and functions of taggings that they aim to detail with NT. Each tagging is represented as a *named graph* in their approach (i.e., `nt:TagAction` is defined as a subclass of `rdfg:Graph` from the "Named Graphs" vocabulary) that can be enriched with additional information. In addition, several subclasses of `nt:TagAction` are defined to capture the type, form, and function of taggings (e.g., `nt:SetTask` represents tags like "todo" or "toread", `nt:Evaluate` tag-based ratings like "nice" or "***", similar to `uto:Vote`). Likewise, several properties are defined to distinguish the relations that can exist between resources and tags (or "signs" according to the NT terminology). However, the question of how to derive these more nuanced distinctions from common taggings has not (yet) been answered sufficiently.

Limpens et al. also aim at a more nuanced description of the resources being tagged. For this purpose, NT integrates the "Identity of Resources on the Web" (IRW) ontology to distinguish, for example, between a resource (`irw:Resource`) and its web representation (`irw:WebRepresentation`). It is, however, again not further detailed how these distinctions can be derived in practice.

2.10 Related Ontologies

As shown in the review, many tagging ontologies reuse concepts from related vocabularies, in particular from DCTERMS, FOAF, SKOS, and SIOC (see Table 1). They either directly link the concepts (e.g. represent users via `foaf:Person`) or derive tagging-specific classes and properties from more general ones (e.g. define tags as subclass of `skos:Concept`).

The related ontologies alone are not capable to represent tagging. Most importantly, they do not provide a class that links the domains of resources, tags, and users and could be reused as tagging class. But also other domain-specific concepts (e.g., private tagging, semantic tagging, or tag position) cannot be represented with the related ontologies.

2.10.1 NEPOMUK Annotation Ontology (NAO)

The same is true for general annotation ontologies, such as they have been proposed in the *Annotea* [1] and NEPOMUK [10] projects. The "NEPOMUK Annotation Ontology" (NAO) [11] has been developed for the "Social Semantic Desktop" [10]. Since tagging is a specifically supported type of annotation in this project, NAO provides a class for tags (`nao:Tag`). However, like the above general vocabularies, NAO misses a tagging class and other key elements (e.g. a class for users), which disqualifies NAO as tagging ontology. Instead, it is rather comparable with SKOS by offering preferred and alternative labels as well as descriptions for tags.

2.10.2 Annotea's Bookmark Schema (AB)

The "Bookmark Schema" (AB) [39] of the well-known *Annotea* project, by contrast, is more appropriate to represent taggings. It defines the two classes `ab:Bookmark` and `ab:Topic` that are interlinked via `ab:hasTopic` and assigned to resources via `ab:recalls`. Despite their different terminology, these classes can be used to represent taggings and tags, as demonstrated by Koivunen in [36]. Koivunen also shows how AB can be combined with DC and FOAF to link users, tagging notes, and creation dates of taggings and tags.

However, since AB has not been designed specifically for tagging (it was developed in 2003), it lacks some important concepts, such as private tagging or tag position. Like CTAG, it also misses a clear specification of how users are linked with tags. Last but not least, AB provides only textual descriptions but no formal specifications for the domains, ranges, and cardinalities of properties, limiting the possibilities for machine interpretation and automatic validation of the folksonomy data.

3. TOWARDS A UNIFIED TAGGING ONTOLOGY

The nine reviewed tagging ontologies introduce a number of concepts that must be considered by any comprehensive representation of tagging. Thus, they also provide the basis for the MUTO ontology that unifies these ontologies and adds missing concepts and links. For this purpose, we classified the found concepts into four categories:

1. **Core concepts**: These concepts are tagging-specific and essential for an interoperable representation of folksonomies. They must be part of the core ontology.

2. **Generic concepts**: These concepts are essential but not tagging-specific. They are already defined in related ontologies and better reused than redefined.

3. **Inferable concepts**: These concepts are also important, but there is no need to represent them in the ontology since they can be inferred from the other concepts.

4. **Rare concepts**: These concepts have been proposed as extensions to tagging but are seldom used in practice. They do not need to be part of the core ontology but can be included via extensions where appropriate.

Table 2 summarizes the results of our categorization. The listed concepts are rather abstract and can consist of several classes and properties. They often group similar concepts from different tagging ontologies (e.g. `moat:AutomatedTagging` and `ctag:AutoTag` are grouped to "automatic tagging") or are abstractions of very specific concepts (e.g. `uto:Vote` is included in "tag function"). "Tag meaning" was considered a core concept (though an optional one) because it is essential for semantic tagging, which is a key application area for tagging ontologies (see Section 5).

1. Core	**2. Generic**
Tagging	Access control (ACL, SIOC)
Private tagging	Date (DCTERMS)
Autom. tagging	User group (FOAF, SIOC)
Tag	Hierarchical relation (SKOS)
Tag position	Note (SIOC, SKOS)
Tag meaning	Resource (DCTYPE, IRW, RDFS)
	Source (FOAF, SIOC)
	User (DCTERMS, FOAF, SIOC)
3. Inferable	**3. Rare**
Tag cloud	Restricted tagging
Tag co-occurrence	Tag polarity
Tag frequency	Tag function
Author vs. user tag	Tag spelling

Table 2: Tagging concepts derived from the review and classified into core, generic, inferable, and rare concepts.

None of the reviewed tagging ontologies defines all the essential concepts (i.e. all concepts listed under the categories "core" and "generic" in Table 2). Taking one tagging ontology and extending it is difficult due to conceptual limitations. For instance, many of the ontologies define direct relations between tags and resources that complicate the integration of private tagging. An integration and alignment of different tagging ontologies has similar problems. The only exceptions are `MOAT` and `SCOT` that have already been aligned to `TAGS` during development [33]. But even in these cases result unnecessary complex conceptualizations.

Thus, we decided to develop a unified ontology that takes the best parts of the reviewed tagging ontologies and combines them in one consistent conceptualization. We chose a modular design that separates the core concepts from the generic and advanced ones. For the generic parts, we reused concepts from popular ontologies instead of defining them once again. Advanced concepts can be added via modules, as it is well-known from other vocabularies, such as `SIOC` (with its "access", "types", and "services" modules [14]) or the "RDF Site Summary" (`RSS`) specification (with its "Dublin Core", "Syndication", and "Content" modules [13]). Furthermore, we defined mappings between MUTO and related concepts from the reviewed tagging ontologies, but separated these mappings from the core ontology, similar as it was done in `CTAG`.

The modularization reduces the complexity and leads to a compact and understandable core ontology. It also avoids conceptual inconsistencies and different levels of expressiveness. Finally, it helps to keep the core ontology stable with regard to future evolutions and advancements in the domain of tagging.

4. THE MUTO ONTOLOGY

Figure 1 depicts the MUTO core ontology as an UML diagram according to OMG's *Ontology Definition Metamodel (ODM)* standard [21] (with class notation for properties and special compact notations for the built-in RDFS and OWL properties `rdfs:domain`, `rdfs:range`, `owl:inverseOf`, and `owl:unionOf`)[3].

The classes of the four key elements of tagging – resources, tags, users, and taggings – are marked in bold in Figure 1. The two domain-specific classes `muto:Tagging` and `muto:Tag` form the center of the core ontology. The other two key elements are not unique to tagging; here, classes from the `RDFS` and `SIOC` vocabularies (namely `rdfs:Resource` and `sioc:UserAccount`) are reused. Based on the two main classes `muto:Tagging` and `muto:Tag`, we will describe the MUTO ontology in more detail in the following.

4.1 Tagging

The central `muto:Tagging` class reifies the tripartite relationship between resources, tags, and users, similar as it was formally described by Mika [42] and first defined in Newman's `TAGS` ontology [46]. In contrast to Mika and Newman, MUTO does not limit the number of tags per tagging to one single tag. Mika and Newman make this restriction mainly due to architectural reasons: Mika's model requires for the ternary relations and `TAGS` defines a `tags:RestrictedTagging` subclass for taggings with "precisely one associated resource, and one associated tag" [46] (unfortunately, without further explanation). `MOAT` reuses `tags:RestrictedTagging` to disambiguate taggings and `NT` applies a similar restriction in its named graph approach.

We designed MUTO so that the number of tags per tagging is theoretically unlimited, since we consider this the most accurate and understandable conceptualization. MUTO even allows for taggings without tags, to support cases where users first simply index a resource and add tags later (like it is supported by the social bookmarking service [4], among others). As we will show in the following, our decision of not restricting the number of tags per tagging does not reduce but increase MUTO's expressive power.

4.1.1 Cardinalities

While the number of tags per tagging is not restricted in MUTO, the number of resources and users is (see cardinalities in Figure 1). This is consistent with one of the key principles of folksonomies: *A tagging must always be linked to exactly one resource and one user account* (the latter can be omitted in case of automatic taggings; therefore the cardinality of "0..1"). If this key principle is violated, the folksonomy cannot be processed as usual (e.g. to generate tag clouds [31] or allow for *pivot browsing* [43]). Defining these restrictions in a tagging ontology is important to ensure high processability and interoperability of the folksonomy. Strangely, none of the reviewed tagging ontologies specifies these restrictions accordingly. `TAGS`, `MOAT`, and `NT` restrict either the number of tags or the number of resources or both, but they make no restrictions on the number of users that are linked to a single tagging.

4.1.2 User Account

MUTO does not link to the user as such (e.g. via `foaf:Agent` as in `TAGS` or via `foaf:Person` as in `TO`) but to `sioc:UserAccount`. This is more accurate and flexible, as it allows one user to have several accounts (e.g. one for work-related and one for personal taggings). An alternative to `sioc:UserAccount` would be `foaf:OnlineAccount` (as used e.g. in `SCOT`), which is conceptually roughly the same. We decided for `sioc:UserAccount` mainly because we also use several other `SIOC` concepts along with `muto:Tagging` and can thus stay in the same namespace, which facilitates linking. However, as `sioc:UserAccount` is a subclass of `foaf:OnlineAccount`, concepts from the `FOAF` vocabulary can also be used (e.g. `foaf:accountServiceHomepage` as in the example of Figure 2). Other useful concepts from the `SIOC` vocabulary are, for instance, `sioc:email`, `sioc:avatar`, or `sioc:follows`. The latter allows one user to follow the taggings of another by linking their accounts. Instead of using `sioc:has_creator` directly to link the user account, MUTO subclasses it in order to explicitly define `muto:Tagging` as `rdfs:domain`. Furthermore, the restriction of one user per tagging is set on this property, as discussed above.

If `SIOC` and/or `FOAF` is also used to represent metadata about the `resource` that is tagged, it can easily be checked if the author of a tagging is also the author of the resource (provided that the same user profile has been used). So we do not need to include a concept to distinguish between author and user tags in MUTO, as proposed by `CTAG`, but can easily infer this information when needed.

[3]The complete MUTO specification, including serializations in RDF/XML format, is available at [9].

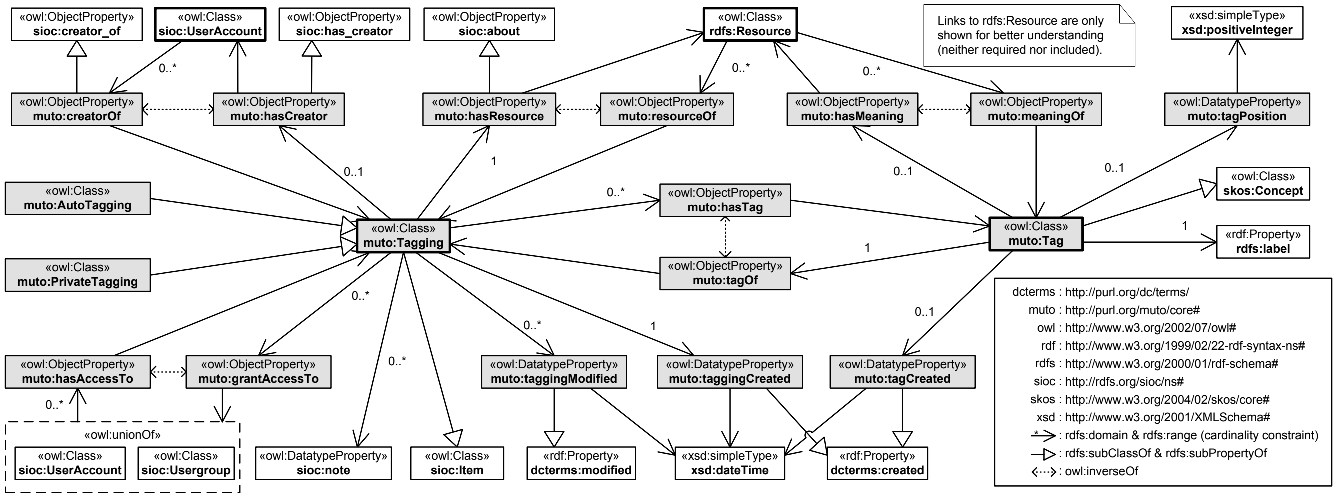

Figure 1: UML diagram of the MUTO core ontology.

4.1.3 Group Tagging

A comprehensive tagging ontology must also be capable to represent group tagging, which is a popular activity in tagging systems. However, the only reviewed ontology that explicitly considers group tagging is SCOT. Basically, three ways of group tagging can be distinguished: The simplest (but most insecure) is to agree upon a unique "group identifier" tag that is used to aggregate taggings from the individual group member accounts in one shared view. An alternative is to create a shared user account that members can log-in to perform group taggings. The third and most advanced solution is to create a group account that the user accounts of all group members are linked to (as supported e.g. by Flickr [6] or Bibsonomy [2]).

Since MUTO links user accounts and not users, it supports all three variants of group tagging. For instance, the third one can be realized with the class `sioc:Usergroup` that `sioc:UserAccount` is linked to via `sioc:member_of` according to the SIOC specification. Since `muto:Tagging` links to single user accounts, not to group accounts, it remains transparent which user added which taggings (assuming that each user account is used by only a single user).

4.1.4 Private Tagging and Access Control

MUTO uses the SIOC classes `sioc:UserAccount` and `sioc:Usergroup` also for access control in private tagging. Every tagging can be linked to either a single account (a friend, a family member, etc.) or a set of accounts (all friends, the whole family, etc.) via `muto:grantAccessTo`. Although this property is mainly meant to grant access in private tagging, it can also be used in public tagging, for instance, to state that a tagging has been sent to another user (e.g. via a send option included in many tagging systems). Private taggings are always represented with the subclass `muto:PrivateTagging`, whether shared with other users or not.

4.1.5 Automatic Tagging

Likewise, MUTO defines a subclass `muto:AutoTagging` for automatic taggings, i.e. taggings that are not created by a user but the tagging system itself or some external service, following the idea of MOAT, CTAG, and NT. The creator of automatic taggings must not but can be captured with MUTO, for instance via `sioc:UserAccount` which is linked to `foaf:Agent` in the SIOC vocabulary. Representing manual and automatic taggings in the same ontology makes sense, as it avoids a redundant conceptualization and allows for an easier transformation of automatic tags into manual (i.e. user validated) ones.

4.1.6 Date and Time of Tagging

Date and time of tagging are important context information that must be captured by any tagging ontology. Many tagging systems use this information, for instance, to order taggings reverse chronologically. Most of the reviewed ontologies reuse concepts from the "Dublin Core" vocabularies DC or DCTERMS here, which are also used in MUTO. But MUTO goes one step further: It allows not only to track the creation date of taggings but also every single edit, which can also be useful information for tagging systems (e.g. to sort taggings not by creation date but date of last modification). In addition, MUTO sets an explicit range (`xsd:dateTime`) for the `muto:taggingCreated` and `muto:taggingModified` properties, in order to force a standardized format and increase interoperability. This is also the main reason why we have not directly used `dcterms:created` and `dcterms:modified` but defined own subproperties.

4.1.7 Tagging Source

Representing the source of tagging is important, for example, if folksonomies from different tagging systems are merged. MUTO does not define an extra concept for this purpose (such as TO, OF, and UTO), but reuses SIOC. Generally, it makes `muto:tagging` a subclass of `sioc:Item`, which is an adequate conceptualization and allows to group several taggings in one `sioc:Container`. The source can then be represented via `sioc:Space` (see example in Figure 2). Thus, several taggings can be grouped and linked to the same source, which is an efficient means to store taggings and source information.

4.1.8 Tagging Note

As many tagging systems allow to enrich taggings by a note, MUTO furthermore links to `sioc:note`. The only other tagging ontology that explicitly considers notes is `UTO` (with `uto:Comment`). However, `UTO` does not align its conceptualization of tagging notes with existing ontologies.

An alternative to `sioc:note` would be `skos:note`, but we decided to use only the `SIOC` vocabulary here, which has been specifically designed "for representing rich data from the Social Web" [15] and is very appropriate for `muto:Tagging`. Accordingly, `muto:hasResource` has been defined as subproperty of `sioc:about`. Concepts from the `SKOS` vocabulary are stronger related to `muto:Tag`, as we will detail in the following.

4.2 Tag

The second core class of the MUTO ontology is `muto:Tag`. Each tag is an instance of this class with its own URI, as in most of the reviewed tagging ontologies. Representing tags as class instances and not as simple literals is necessary for the definition of tag properties (see Section 2.1). Tags with the same label are not merged in MUTO, as this would not only affect the labels but also other tag properties. Aggregations of tags with the same label (e.g. to generate tag cloud visualizations) are not part of the representation itself, but are performed by the tagging system or some external service on the basis of the folksonomy data that is provided by the representation.

4.2.1 Tag Relations

MUTO's tag class is defined as a subclass of `skos:Concept`, like in `TAGS`, `TO`, and `CTAG`. This opens up many possibilities to enrich tags with concepts from the SKOS vocabulary. Especially `skos:semanticRelation` can be well reused to describe relations between tags. For instance, hierarchical tag relations – as defined by some tagging systems (e.g., Delicious [4] with its *tag bundle* feature or Bibsonomy [2] with its *subtag* and *supertag* relations) – can be represented via `skos:broader` and `skos:narrower` (which are subproperties of `skos:semanticRelation`). Likewise, `skos:related` can be used to describe tag relations of a more general nature. In contrast to `TAGS`, we decided to not include specific tag relations (like `tags:equivalentTag` or `tags:relatedTag`) in the core ontology but to reuse SKOS relations where appropriate and leave a detailed description of tag relations to future modules for the MUTO ontology.

4.2.2 Tag Label

However, not all `SKOS` concepts can be reasonably used along with `muto:Tag`. Especially the application of the subproperties of `skos:label` (i.e. `skos:altLabel`, `skos:hiddenLabel`, and `skos:prefLabel`) to `muto:Tag` makes no sense, as MUTO restricts the number of labels per tag to one. This is consistent with another key principle of folksonomies that must not be violated if common data processing should be guaranteed (see Section 4.1.1): *Every tag has exactly one label.* Accordingly, a tag that has more than one label (e.g., a multi-language tag) is strictly speaking not a tag but a concept. Even though MUTO follows this strict distinction, it supports mappings between tags and concepts via its `muto:hasMeaning` property (see below).

4.2.3 Separate Date and Time Information

The date and time information for tags (`muto:tagCreated`) is conceptually separated from the date and time information for taggings (`muto:taggingCreated`). This separation is useful if only certain tags of a tagging are edited, or if tags are not added with the creation of a tagging but at a later time (see example in Figure 2). Omitting the separate date and time information in these cases can result in biased tag statistics and wrong conclusions about the folksonomy's evolution.

However, since the creation date and time of a tag are usually equal to the creation date and time of the associated tagging, we defined `muto:tagCreated` as an optional property to prevent storing redundant information. This means that if no separate date and time information is given for a tag, it is assumed that the tag has been created at the same date and time as the tagging (i.e., `muto:tagCreated` = `muto:taggingCreated`).

As discussed above, a tag is essentially its label. Hence, editing a tag label means, strictly speaking, the substitution of one tag by another. This is the reason why MUTO does not define a separate modification date and time for tags. If a tag label is edited by the user, a new tag is created, having mostly the same property values as the old one but a different label and creation date. If other properties of a tag are changed (e.g. its position, see below), this is considered as a change of the tagging and can be recorded via `muto:taggingModified`. Prohibiting changes to tag labels can prevent misuse and wrong interpretation of the folksonomy data.

4.2.4 Tag Meaning

MUTO's approach of disambiguating tags by linking them to well-defined concepts from the Semantic Web (via `muto:hasMeaning`) is similar to the attempts of `MOAT` and `CTAG`. However, MUTO distinguishes strictly between *taggings* and *tags* and allows only single tags (not complete taggings) to be linked to concepts. In general, it can be differentiated between two kinds of disambiguation: Mapping a tag to an existing concept (such as a DBpedia resource [23] or Wordnet term [44]) or transforming it into a new one. Both forms can be expressed with `muto:hasMeaning` which links to the generic `rdfs:Resource` class. This property can also be used if tags are converged (e.g. synonyms) by linking several tag instances to the same concept.

4.2.5 Tag Position

Last but not least, MUTO defines the optional `muto:tagPosition` property to represent a tag's position within the list of tags that a tagging consists of. An alternative to using a property would have been RDF concepts for the representation of lists, such as RDF containers (e.g. `rdf:Seq`) or RDF collections (e.g. `rdf:List`), or some OWL workaround (e.g. the "Ordered Lists Ontology", `OLO` [19]). However, since we do not want to unnecessarily complicate the MUTO ontology, we decided to adopt the practical solution that was proposed by `OF` and linked `muto:tagPosition` to an integer value that represents the list position of the tag. We decided for `xsd:positiveInteger` to prevent that some tagging systems start counting at zero while others start at one. Furthermore, we decided to specify `muto:tagPosition` as an optional property, as the semantics of a tag's position is not important enough to force storing this information.

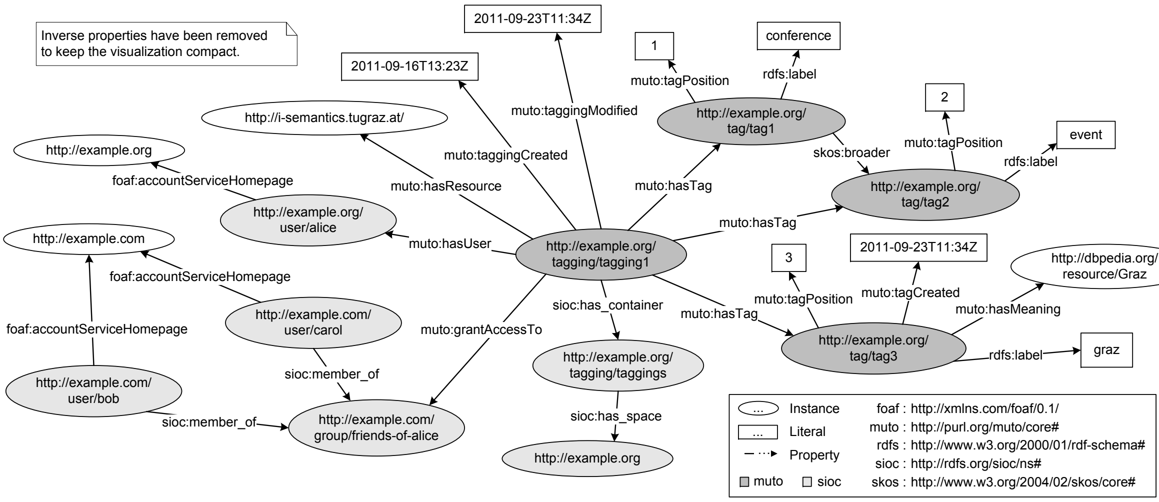

Figure 2: Example of using MUTO to represent a tagging from a social bookmarking service.

4.3 Example

Figure 2 provides an example of using MUTO with the fictional social bookmarking service *Example.org*. It shows the RDF graph of a tagging by user Alice who bookmarked the homepage of the I-SEMANTICS conference (`http://i-semantics.tugraz.at`). She first used the tags "conference" and "event" (in this order, as given by `muto:tagPosition`) and later added the tag "graz" (as indicated by the timestamps of `muto:tagCreated`). She also set a hierarchical relation between the first two tags, and 'semantified' the third by linking it to the DBpedia resource for the city of Graz (`http://dbpedia.org/resource/Graz`). Finally, she decided to not make this bookmark publicly accessible but to share it only with her friends Bob and Carol. Bob and Carol are not registered with the bookmarking service, but they have accounts at the fictional social networking service *Example.com*. Both are members of the group "friends-of-alice" there, allowing Alice to set only one link to the group account instead of linking the individual accounts of Bob and Carol separately.

Note how tagging-specific concepts from the MUTO ontology are used in combination with more general concepts from the integrated ontologies in this example. The hierarchical relation between the first two tags is represented via `skos:broader` (and the inverse `skos:narrower`), while the used tagging system is linked via `sioc:has_space`. Furthermore, `foaf:accountServiceHomepage` – which is a property of `foaf:OnlineAccount`, a superclass of `sioc:UserAccount` according to `SIOC` – is used to represent the services the user accounts belong to. Finally, `sioc:member_of` is used to link the user accounts of Bob and Carol to the "friends-of-alice" group.

Also note that both common and semantic tagging are used in combination in this example. The tags "conference" and "event" are not further disambiguated, while the tag "graz" is linked to the corresponding DBpedia resource and thus enriched with a lot of additional information.

Since the first two tags are added with the creation of the tagging, no separate date and time information must be saved. The third tag, by contrast, requires a separate timestamp, as it has been created at a later date. Because this was the only edit of the tagging, the timestamp of `muto:taggingModified` is equal to the one of `muto:tagCreated`.

5. DISCUSSION

The main goal of this work was not to propose yet another tagging ontology but to unify existing attempts in one consistent conceptualization. We carefully reviewed available tagging ontologies, derived and classified the found concepts and enriched them by adding missing concepts and links. We justified our main design decisions and discussed possible alternatives and extensions. The modular design of the resulting MUTO ontology reduces complexity, avoids redundancies, and increases stability with regard to future evolutions and extensions in the domain of tagging. It reflects the tripartite character of folksonomies [42] by connecting the independent domains of *resources*, *tags*, and *users*.

While *tags* are tagging-specific and thus a core part of the MUTO ontology (though linked to the related `skos:Concept` class), *users* and *resources* are generic concepts and better separated from the tagging ontology. For this reason, MUTO sets links to `sioc:UserAccount` and `rdfs:Resource` which are good starting points for a more detailed description of the respective domains: `SIOC` and the related `FOAF` vocabulary have been proven useful in the representation of Web users and social networks [24], whereas the generic `rdfs:Resource` class can be further detailed with vocabularies like `IRW`, `SIOC`, or `DCTYPE`. Finally, also some more advanced concepts from the reviewed tagging ontologies can be reused to extend MUTO. Examples are the subproperties of `scot:spelling_variant` for the representation of spelling variations in tags, or the subclasses of `nt:TagAction` for capturing specific forms and functions of taggings. But as these advanced concepts are not (yet) considered in tagging systems, we decided not to include them into the MUTO core ontology in favor of a more compact design.

However, we included the advanced form of semantic tagging, as it is important for the Semantic Web and increasingly supported by tagging systems (see e.g. Faviki [5] or Zigtag [20]). Disambiguations of tags can be represented by links to well-defined resources of the Semantic Web in

MUTO, similar as proposed by MOAT, CTAG, and TT. It is important to note that MUTO has not been designed specifically for semantic tagging. The disambiguation links are optional, making MUTO capable to represent both common and semantic tagging in the same conceptualization.

This is in line with the overall goal of this work to design an ontology for broad applicability. Although we aimed for a simple and consistent design, MUTO is not intended to be a "minimal" ontology (like CTAG) but rather aims at a complete conceptualization of the key elements of tagging. It should provide sufficient representation capabilities for most use cases and can be easily extended for others due to its modular structure.

If we compare MUTO with the reviewed ontologies, it is maybe most closely related to TAGS (plus the advancements of MOAT), but includes further concepts from other ontologies that are missing in TAGS and MOAT, such as private tagging, access control, modification date, or tag position. Furthermore, MUTO does not restrict the number of tags per tagging to one, which is a more natural representation and facilitates access (e.g. to tag co-occurrences), but it defines other important cardinalities (e.g. number of users per tagging).

Another goal of this work was to contribute to a better understanding of the conceptual domain of tagging and its formal representation on the Web. We tried to achieve this not only with the MUTO ontology but also with the survey and discussion of existing tagging ontologies. Former reviews of a part of the ontologies can be found in [33] and [34]. Even though these works already aligned own developments with the TAGS ontology, MUTO is the first attempt to develop a unified conceptualization based on a comprehensive review of available tagging ontologies.

Future work concerns the development of modules that extend the MUTO core ontology by advanced tagging concepts to represent, for instance, specific types of tags (hashtags, geotags, tag-based star ratings, etc.) or advanced tag relations (synonymy, part-of, etc.). Even though this is already possible by extending the MUTO core ontology (as we have illustrated in the example), we aim for interoperable conceptualizations that are described in well-designed modules and can be seamlessly reused along with MUTO.

6. ACKNOWLEDGMENTS

This work has been conducted in the context of the UrThey project funded by the Spanish Ministry of Science and Innovation (TIN2009-09687).

7. REFERENCES

[1] Annotea Project. http://www.w3.org/2001/Annotea.

[2] Bibsonomy. http://www.bibsonomy.org.

[3] Common Tag. http://www.commontag.org.

[4] Delicious. http://www.delicious.com.

[5] Faviki. http://www.faviki.com.

[6] Flickr. http://www.flickr.com.

[7] Linking Open Data – W3C SWEO Community Project. http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData.

[8] MOAT: Meaning Of A Tag. http://moat-project.org.

[9] MUTO: Modular Unified Tagging Ontology. http://purl.org/muto.

[10] NEPOMUK – The Social Semantic Desktop. http://nepomuk.semanticdesktop.org.

[11] NEPOMUK Annotation Ontology Specification. http://www.semanticdesktop.org/ontologies/nao.

[12] Nice Tag Ontology. http://ns.inria.fr/nicetag/2010/09/09/voc.html.

[13] RDF Site Summary 1.0 Modules. http://web.resource.org/rss/1.0/modules/.

[14] SIOC Core Ontology Specification. http://rdfs.org/sioc/spec/.

[15] SIOC: Semantically-Interlinked Online Communities. http://sioc-project.org.

[16] TagCommons. http://tagcommons.org.

[17] TAGora. http://www.tagora-project.eu.

[18] TAGora Sense Repository. http://tagora.ecs.soton.ac.uk/tsr.

[19] The Ordered List Ontology 0.72. http://purl.org/ontology/olo/core#.

[20] Zigtag. http://zigtag.com.

[21] Ontology Definition Metamodel. http://www.omg.org/spec/ODM/1.0, 2009.

[22] C. Bizer, T. Heath, and T. Berners-Lee. Linked data – the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009.

[23] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. Dbpedia - a crystallization point for the web of data. *J. Web Sem.*, 7(3):154–165, 2009.

[24] U. Bojars, A. Passant, J. G. Breslin, and S. Decker. Social network and data portability using semantic web technologies. In *BIS 2008 Workshops Proceedings*, volume 333 of *CEUR-WS.org*, pages 5–19, 2008.

[25] Y. Ding, E. K. Jacob, M. Fried, I. Toma, E. Yan, S. Foo, and S. Milojević. Upper tag ontology for integrating social tagging data. *J. Am. Soc. Inf. Sci. Technol.*, 61:505–521, 2010.

[26] F. Echarte, J. J. Astrain, A. Córdoba, and J. E. Villadangos. Ontology of folksonomy: A new modelling method. In *Proc. of SAAKM'07*, volume 289 of *CEUR-WS.org*, 2007.

[27] T. Gruber. Ontology of folksonomy: A mash-up of apples and oranges. http://tomgruber.org/writing/mtsr05-ontology-of-folksonomy.htm, 2005. Keynote at MTSR'05.

[28] T. Gruber. TagOntology – a way to agree on the semantics of tagging data. http://tomgruber.org/writing/tagontology.htm, 2005. Presentation to Tag Camp.

[29] M. Guy and E. Tonkin. Folksonomies: Tidying up tags? *D-Lib Magazine*, 12(1), 2006.

[30] T. Hammond, T. Hannay, B. Lund, and J. Scott. Social bookmarking tools (i): A general review. *D-Lib Magazine*, 11(4), 2005.

[31] O. Kaser and D. Lemire. Tag-cloud drawing: Algorithms for cloud visualization. In *Proc. of WWW 2007 Workshop on Tagging and Metadata for Social Information Organization*, 2007.

[32] H. L. Kim, J. G. Breslin, S.-K. Yang, and H.-G. Kim. Social semantic cloud of tag: Semantic model for social tagging. In *Proc. of KES AMSTA'08*, volume 4953 of *LNCS*, pages 83–92. Springer, 2008.

Vocabulary name	Prefix	URI reference
(Annotea) Bookmark Schema	AB	`http://www.w3.org/2002/01/bookmark#`
Basic Access Control Ontology	ACL	`http://www.w3.org/ns/auth/acl#`
Common Tag	CTAG	`http://commontag.org/ns#`
DCMI Metadata Terms	DCTERMS	`http://purl.org/dc/terms/`
DCMI Type Vocabulary	DCTYPE	`http://purl.org/dc/dcmitype/`
Friend of a Friend	FOAF	`http://xmlns.com/foaf/0.1/`
Identity of Resources on the Web	IRW	`http://www.ontologydesignpatterns.org/ont/web/irw.owl`
Meaning of a Tag	MOAT	`http://moat-project.org/ns#`
Modular Unified Tagging Ontology	MUTO	`http://purl.org/muto/core#`
NEPOMUK Annotation Ontology	NAO	`http://www.semanticdesktop.org/ontologies/2007/08/15/nao#`
NiceTag Ontology	NT	`http://ns.inria.fr/nicetag/2010/09/09/voc`
Ontology of Folksonomy	OF	`http://www.eslomas.com/tagontology-1.owl`
Ordered Lists Ontology	OLO	`http://purl.org/ontology/olo/core#`
Named Graphs	RDFG	`http://www.w3.org/2004/03/trix/rdfg-1`
RDF Schema	RDFS	`http://www.w3.org/2000/01/rdf-schema#`
RDF Site Summary	RSS	`http://purl.org/rss/1.0/`
Semantically-Interlinked Online Communities	SIOC	`http://rdfs.org/sioc/ns#`
SIOC Types	SIOCT	`http://rdfs.org/sioc/types#`
Simple Knowledge Organization System	SKOS	`http://www.w3.org/2004/02/skos/core`
Social Semantic Cloud of Tags	SCOT	`http://scot-project.org/scot/ns#`
Tag Ontology	TAGS	`http://www.holygoat.co.uk/owl/redwood/0.1/tags/`
Tagging Ontology	TO	`http://bubb.ghb.fh-furtwangen.de/TagOnt/tagont.owl` (Available at: `http://tagont.googlecode.com/files/tagont.owl`)
TAGora Tagging Ontology	TT	`http://tagora.ecs.soton.ac.uk/schemas/tagging`
Upper Tag Ontology	UTO	`http://info.slis.indiana.edu/~dingying/uto.owl` (Aligned UTO version without URI in [25])
XML Schema	XSD	`http://www.w3.org/2001/XMLSchema#`

Table 3: Alphabetical list of the names, namespace prefixes, and URIs of all referenced vocabularies.

[33] H. L. Kim, A. Passant, J. G. Breslin, S. Scerri, and S. Decker. Review and alignment of tag ontologies for semantically-linked data in collaborative tagging spaces. In *Proc. of ICSC'08*, pp. 315–322. IEEE, 2008.

[34] H. L. Kim, S. Scerri, J. G. Breslin, S. Decker, and H. G. Kim. The state of the art in tag ontologies: a semantic model for tagging and folksonomies. In *Proc. of DC'08*, pages 128–137. DCMI, 2008.

[35] T. Knerr. Tagging ontology – towards a common ontology for folksonomies. `http://tagont.googlecode.com/files/TagOntPaper.pdf`.

[36] M.-R. Koivunen. Semantic authoring by tagging with annotea social bookmarks and topics. In *Proc. of SAAW'06*, volume 209 of *CEUR-WS.org*, 2006.

[37] F. Limpens, A. Monnin, D. Laniado, and F. Gandon. NiceTag ontology: Tags as named graphs. In *Proc. of SNI'09*, 2009.

[38] A. Marchetti, M. Tesconi, F. Ronzano, M. Rosella, and S. Minutoli. Semkey: A semantic collaborative tagging system. In *Proc. of WWW 2007 Workshop on Tagging and Metadata for Social Information Organization*, 2007.

[39] E. P. Marja-Riitta Koivunen, Ralph Swick. Annotea shared bookmarks. In *Proc. of K-CAP'03*, volume 101 of *CEUR-WS.org*, 2003.

[40] C. Marlow, M. Naaman, D. Boyd, and M. Davis. Ht06, tagging paper, taxonomy, flickr, academic article, to read. In *Proc. of HYPERTEXT'06*, pages 31–40. ACM, 2006.

[41] O. Medelyan, E. Frank, and I. H. Witten. Human-competitive tagging using automatic keyphrase extraction. In *Proc. of EMNLP'09*, pages 1318–1327. ACL, 2009.

[42] P. Mika. Ontologies are us: A unified model of social networks and semantics. In *Proc. of ISWC'05*, volume 3729 of *LNCS*, pages 522–536. Springer, 2005.

[43] D. R. Millen and J. Feinberg. Using social tagging to improve social navigation. In *Proc. of AH 2006 Workshop on the Social Navigation and Community-Based Adaptation Technologies*, 2006.

[44] G. A. Miller. Wordnet: a lexical database for english. *Commun. ACM*, 38:39–41, 1995.

[45] A. Monnin, F. Limpens, F. Gandon, and D. Laniado. Speech acts meet tagging: NiceTag ontology. In *Proc. of I-SEMANTICS'10*, pages 31:1–31:10. ACM, 2010.

[46] R. Newman. Tag ontology design. `http://www.holygoat.co.uk/projects/tags`, 2005.

[47] A. Passant and P. Laublet. Meaning of a tag: A collaborative approach to bridge the gap between tagging and linked data. In *Proc. of LDOW'08*, volume 369 of *CEUR-WS.org*, 2008.

[48] G. Smith. *Tagging: People-Powered Metadata for the Social Web*. New Riders, 2008.

[49] M. Szomszor, C. Cattuto, W. Van den Broeck, A. Barrat, and H. Alani. Semantics, sensors, and the social web: The live social semantics experiments. In *Proc. of ESWC'10*, volume 6089 of *LNCS*, pages 196–210. Springer, 2010.

APPENDIX

Table 3 lists the names and URIs of all vocabularies referenced in this paper. In addition, all namespace prefixes as used in this paper and in the MUTO ontology are given.