What are Semantic Annotations?*

Eyal Oren[1], Knud Hinnerk Möller[1], Simon Scerri[1], Siegfried Handschuh[1], and Michael Sintek[2]

[1] Digital Enterprise Research Institute, National University of Ireland, Galway
{eyal.oren, knud.moeller, simon.scerri, siegfried.handschuh}@deri.org
[2] German Research Center for Artificial Intelligence (DFKI)
sintek@dfki.uni-kl.de

Abstract. Annotations of Web resources can be created using traditional document annotation tools or more recent approaches such as semantic wikis, semantic blogs and collaborative tagging. Currently no unified model exists for all these different kinds of annotations, making it difficult both to compare and assess annotation tools and to integrate the various kinds of annotation data.

We analyse annotations in various domains and present a unified formal model for semantic annotations. We evaluate existing annotation tools from these different domains and show how to map the data these tools produce onto our formal model, thus allowing to access and represent this data in a unified way.

1 Introduction

The Semantic Web enables machines to interpret, combine, and use data on the Web. Whereas the current "eye-ball" Web is only understandable for humans, the Semantic Web can be used by computers as well. The basis for the Semantic Web are computer-understandable descriptions of resources. We can create such descriptions by annotating resources with metadata, resulting in "annotations" about that resource.

Metaphorically, we can see URIs[3] as the "atoms" of the Semantic Web and semantic annotations as the "molecules". The Semantic Web is about shared terminology, achieved through consistent use of URIs. Annotations create a relationship between URIs and build up a network of data.

Several tools and paradigms exist to create annotations of Web resources, both manually, semi-automatically and fully automatically. Traditionally the domain of (shallow) linguistic analysis in document annotation, we observe the current trend of tagging, semantic blogs and semantic wikis as forms of annotations as well. Currently, no unified formal model exists to describe these different kinds

* This material is based upon works supported by the Science Foundation Ireland under Grants No. SFI/02/CE1/I131 and SFI/04/BR/CS0694 and by the European Commission under the Nepomuk project FP6-027705.

[3] http://www.w3.org/Addressing/

of annotations. The lack of such a model limits the comparison of annotation tools and complicates the integration of annotation data from multiple sources.

Bringing all annotations into a common RDF[4] model would partially solve the data integration. However, to map annotations into RDF we need to first understand their similarities and differences: we need a common model of annotations, and we need to unify the terminology and modelling choices of the various tools.

In this paper, we analyse the notion of "annotation" in the context of the Semantic Web: we investigate annotations in different domains (document annotation, tagging, semantic wikis and semantic blogs), we define a formal and unified model of annotations, derive categorising dimensions and evaluate existing annotation tools according to these dimensions. Finally we show how existing annotation tools can be mapped to our unified formal model, allowing data interoperability and integration.

Our work is based on several earlier approaches to understand, classify and compare document annotations such as [3,7,17,20]. We align and extend these works with new paradigms such as Semantic Wikis and Semantic Blogs and introduce a formal model for annotations. Our contributions are: (i) a formal model of annotations with characterising dimensions, (ii) a clear characterisation of annotation tools, and (iii) a unified model for semantic annotations.

2 Creating annotations

The term "annotation" implies, very generally speaking, to attach data to some other piece of data. During the course of this paper, we will elaborate on this rather simple statement with respect to a number of different domains. The following sections will give a short introduction to each of these domains and specify the role of annotations for each domain.

2.1 Document annotations

The traditional domain of *document annotation* covers the annotation of arbitrary textual documents, or parts of them. Annotations can be manual (performed by one or more people), semi-automatic (based on automatic suggestions), or fully automatic.

Manual annotation tools allow users to add annotations to web pages or other resources, and share these with others. An example annotation would relate the text "Paris" to an ontology, identifying it as a city and as capital of France. Automatic tools can perform similar annotations (such as named-entity recognition) without manual intervention.

[4] `http://w3.org/RDF`

2.2 Semantic Wikis

Wikis are collaborative hypertext authoring environments and allow people to collaboratively collect, describe, and author information. Semantic Wikis allow users to make formal descriptions of resources by annotating the pages that represent those resources. Where a regular Wiki enables users to describe resources in natural language, a Semantic Wiki enables users to additionally describe resources in a formal language. By adding metadata to ordinary Wiki content, users get added benefits such as improved retrieval, information exchange, and knowledge reuse [14].

2.3 Semantic Blogs

Blogs (or weblogs) [22] are online journals or diaries. Blogs consist of individual posts, habitually created and presented in reverse chronological order.

An annotation in blogs is, most commonly, a statement about a post. For example, many current blogging solutions allow to classify posts with simple categories or topics such as "sports", "cinema" or "Sigmund Freud" (a form of *tagging*, discussed next) — we can say that blog posts are annotated with these categories. In Semantic Blogging [4], and to some extent in structured blogging[5], these annotations are extended, and allow association on an ontological basis.

2.4 Tagging

Tagging systems such as del.icio.us[6], Technorati[7] or Flickr[8] allow users to associate one or more tags to a web resources [11]. The tags express some unspecified relation between the resource and whatever the term refers to.

For example, in Flickr a photograph of a cat might be tagged "cat", indicating that the photograph depicts a cat. Similarly, in del.icio.us the website for ISWC 2006 might be tagged "conference semantics iswc2006", indicating that the website is about the ISWC conference and about semantics. Which resources can be tagged depends on the particular system: e.g. del.icio.us allows tagging of arbitrary web pages, whereas Flickr allows tagging of uploaded pictures, and Technorati allows tagging of blogs.

3 Definition of annotations

We can arrange annotation approaches in a three-dimensional space, c.f. [1,6]: effort of the annotation, completeness of the result (i.e. how well does it capture the real-world situation) and (ontological or social) commitment to the result (i.e. how many commit to this model of the world and understand it).

[5] `http://structuredblogging.org`
[6] `http://del.icio.us`
[7] `http://www.technorati.com`
[8] `http://www.flickr.com`

For example, tags require little effort (they are easy to assign) and result in high commitment (through the collaborative tagging process the community agrees on the complete results, if not on single annotations), but they have a low completeness (one can not make complex statements about the real world, but only assign shallow tags).

3.1 Conceptual model

The term "annotation" can denote both the process of annotating and the result of that process [9]. Where we say "annotation" we mean the result. An annotation attaches some data to some other data: it establishes, within some context, a (typed) relation between the annotated data and the annotating data.

We differentiate three types of annotations: informal, formal and ontological. Informal annotations are not machine-readable because they do not use a formal language. Formal annotations are machine-understandable, but do not (per se) use ontological terms. In ontological annotations the terminology has a commonly understood meaning that corresponds to an shared conceptualisation called ontology [8].

Whether a term is ontological is a social matter and not a technical or formal matter. It is sometimes mistakenly understood that using a formal ontology language makes terms ontological. An ontology however denotes a shared (social) understanding; the ontology language can be used to formally capture that understanding, but does not preclude reaching an understanding in the first place.

Summarising, we can distinguish three types of annotations:

1. informal annotations,
2. formal annotations, that have formally defined constituents and are thus machine-readable, and
3. ontological annotations, that have formally defined constituents and use only ontological terms that are socially accepted and understood.

3.2 Formal model

Investigating the nature of annotation further, we can model it as a quadruple:

Definition 1 (Annotation). *An annotation A is a tuple (a_s, a_p, a_o, a_c), where a_s is the subject of the annotation (the annotated data) a_o is the object of the annotation (the annotating data) a_p is the predicate (the annotation relation) that defines the type of relationship between a_s and a_o, and a_c is the context in which the annotation is made.*

The *annotation subject* can be formal or informal. For example, when we put a note in the margin of a paragraph, the informal convention is that the note applies to the paragraph, but that pointer is not formally defined. If we however use a formal pointer such as a URI[9] to point to the paragraph then the subject is formally specified.

[9] One can use XPointer to point to a paragraph in a document and XPointer can be used as a URI, as discussed in `http://www.w3.org/TR/xptr-framework/#escaping`.

The *annotation predicate* can be formal or informal. For example, when we put a note in the margin, the relation is not formally defined, but we may informally derive from the context that that the note is a comment, a change-request, an approval or disapproval, etc. If we use a formal pointer to an ontological term that indicates the relation (e.g. `dc:comment`) then the predicate is formally defined.

The *annotation object* can be formal or informal. If an object is formal we can distinguish different levels of formality: textual, structural, or ontological. For example, then string "This is great!" is a textual object. A budget calculation table in the margin of a project proposal is a structural object. And an annotation object that is not only explicitly structured but also uses ontological terms is an ontological object.

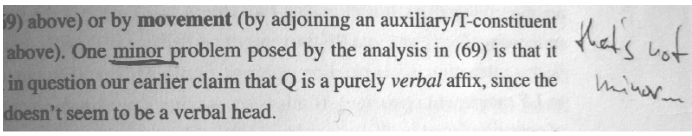
The *annotation context* can be formal or informal. Context can could indicate when the annotation was made and by whom (provenance), or within what scope the annotation is deemed valid, for example in a temporal scope (it is only valid in 2006) or in a spatial scope (it is only valid in Western Europe). Often context is given informally and implicitly. If we use a formal pointer such as a URI then the context is formally defined.

Definition 2 (Formal annotation). *A formal annotation A_f is an annotation A, where the subject a_s is a URI, the predicate a_p is a URI, the object a_o is a URI or a formal and the context a_c is a URI.*

Definition 3 (Ontological annotation). *A ontological annotation A_s is a formal annotation A_f, where the predicate a_p and the context a_c are an (arbitrarily complex) ontological term, and the object a_o conforms*[10] *to an ontological definition of a_p.*

We show an example for each kind of annotation. Example 1 shows an informal annotation: a handwritten margin annotation in a book. Example 2 shows the same annotation, but now formally expressed in N3[11] and thus machine readable. Example 3 then shows the same annotation while using ontological (socially accepted) terms. None of the examples defines an explicit context.

Example 1 (Informal annotation).


9) above) or by **movement** (by adjoining an auxiliary/T-constituent above). One minor problem posed by the analysis in (69) is that it in question our earlier claim that Q is a purely *verbal* affix, since the doesn't seem to be a verbal head.

that's not minor

Example 2 (Formal annotation).

[10] The notion of "conformance" is rather weak in some ontology languages (such as RDFS or OWL) since these are not constraint-based languages (as opposed to e.g. database schemas). However, we use the notion of conformance to differ between "good" usage of textual objects, for example to indicate the name of a person, and "bad" usage of textual objects, for example to indicate the friends of a person.

[11] `http://www.w3.org/DesignIssues/Notation3.html`

```
<http://papers.org/minimalism#minor>
 <disagree> "that's not minor!".
```

Example 3 (Ontological annotation).

```
<http://papers.org/minimalism#minor>
  ibis:con
   [ rdf:type ibis:Argument;
     rdf:label "that's not minor!" ].
```

4 Evaluation

In this section, we evaluate several typical annotation tools based on the model developed in the previous section. We first describe classifying dimensions of annotations and then analyse the tools in each annotation domain.

We have selected and combined several dimensions from the literature as well as our own to classify annotations (we again focus on the annotation result, not the annotation process).

Association ([17]) The way an annotation is associated with the annotated resource, viz. whether the annotation is embedded in the annotated resource, or references the resource externally. For example, in a Wiki annotations usually only describe the page on which they appear.

Subject granularity (also called “lexical span” [16] or “scope” [17]) Indicates the granularity of the annotation subject: e.g. is the annotation about a document, a section inside a document, a sentence, or a word?

Representation distinction (also called “instance identification vs. reference” [3]) Indicates whether the tool distinguishes annotations about documents from annotations of the concept described in or otherwise related to the document?

Terminology reuse (also called “heterogeneity” and “interoperability” [17]) Indicates whether an annotation is self-confined with its own terminology, or whether an annotation uses terms from (one or more) existing ontologies, and are thus interoperable and understandable for others.

Object type (also called “annotation form” [7]) Indicates the type of annotation object: is it a literal or textual object, a structural object (including a hyperlink to another page), or an ontological object?

Context Indicates the context of the annotation: when was it made, by whom, and within what scope: the annotation could for example be temporally scoped (it is only valid in 2006) or spatially scoped (it is only valid in Western Europe). If the annotation is not about a document, then the context could also be the document the annotation is derived from (see also Sect. 5.2).

4.1 Document annotations

Document annotations target annotation of arbitrary textual documents, or parts of them. We analyse Annotea [10], OntoMat [9], COHSE [3] and GATE [5]. The first three are manual annotation tools, whereas GATE is a framework usually used for (semi-)automatic annotation. A summary of these tools is given in Table 1.

All tools discussed here store the annotations **externally**, i.e. no manipulation of the original document is needed. Different mechanisms are used to achieve this, such as XPointer in Annotea and OntoMat, or the GATE-specific way of document duplication. Also, each of the tools offers a very high **subject granularity**, allowing for any document fragment. Both OntoMat and typical GATE annotation makes statements about the world and anchor these statements in the annotated documents (in this sense they could also be considered ontology population tools, rather than document annotation tools). Annotea on the other hand focusses on free text annotations about the documents themselves. COHSE allows both kinds of annotations. With respect to terminology reuse, Annotea uses a "fixed" annotation schema, while the other tools support multiple **arbitrary ontologies** or vocabularies. Annotation **objects** in Annotea are literal comments, ontological objects in OntoMat, literals or pages in COHSE and any of these in GATE (typically though, only literals are used). Finally, Annotea and COHSE make metadata such as author and date explicit as the annotation **context**. GATE theoretically allows any kind of context to be generated, but out-of-the-box only stores information regarding the internal processes that were performed during the annotation process (mainly for debugging the automatic annotation).

dimension	**OntoMat**	**Annotea**	**COHSE**	**GATE**
association	current	current	current	current
granularity	any fragment	any fragment	any fragment	any fragment
repr. distinction	focus: concepts	focus: document	both	focus: concepts
term. reuse	yes	single ont.	yes	yes
object type	onto.	comment	literal or page	literal
context	no	yes	yes	yes

Table 1: Classical document annotations

4.2 Semantic Wikis

Annotations in Semantic Wikis are formal and possibly semantic, i.e. they are formally defined, and possibly use ontological terms. We have earlier evaluated the level of annotation in current Semantic Wiki approaches [15]; we analysed WikSAR [2], Semantic MediaWiki [21], IkeWiki [18] and SemperWiki [13,14] as the most prominent systems under ongoing development. A summary is given in Table 2.

Most existing Wikis attribute statements only to the page on which they appear, allowing only annotations of the current page. In SemperWiki the user can explicitly state the subject of the annotations, because we separate the page and the thing it describes, and annotations can thus be attributed to arbitrary URIs.

Most existing Wikis only allow annotation of complete pages and no finer **granularity**, for the same reason (the implicit annotation subject) as mentioned above.

Of the discussed Wikis only SemperWiki clearly **separates** the page from the concept that it describes, and offers a syntax that distinguishes annotations of the page from annotations of the concept. IkeWiki also separates pages from the concepts that they describe (a concept can be represented on multiple pages), but does not, as far as we know, offer a syntax to manually express this distinction.

IkeWiki and SemperWiki allow existing **terminology to be reused** in annotations (through namespace definitions or full URIs), the rest can only create annotations using internal Wiki pages and can thus not make use of existing terminology.

All discussed Wikis allow an **object** to be a literal or an internal Wiki page. Of the discussed Wikis, only SemperWiki allows the object of an annotation to be an arbitrary URI. No Semantic Wiki allows unnamed resources (blank nodes) as objects.

The **context** of annotations is ignored in all existing Wikis.

dimension	**WikSAR**	**Sem. MediaWiki**	**IkeWiki**	**SemperWiki**
association	current	current	current	current, any URI
granularity	page	page	page	page, any fragment
repr. distinction	no	no	yes	yes
term. reuse	no	no	yes	yes
object type	literal, page	literal, page	literal, page	literal,page, URI
context	no	no	no	no

Table 2: Annotations in current Semantic Wikis

4.3 Semantic Blogs

We analyse several Semantic Blogging systems, namely semiBlog [12], which allows users to annotate blogs with desktop items such as addressbook entries; the HP Semantic Blogging demonstrator [4], which enables users to attach bibliographic metadata to a blog, and Semblog [19]), which allows users to annotate blogs with their personal taxonomy. A summary is given in Table 3.

These systems have quite similar characteristics: All systems annotate a **complete** blog post with some object. Furthermore, none of the systems makes any fine-grained distinction between different annotation **predicates**; instead, the annotations simply denote that the subject, the post, is somehow about or otherwise related to the object, expressed with a rather generic property such as `foaf:topic`. As explained above, the supported annotation **objects** differ, from various desktop items, to bibliographic metadata, to taxonomic concepts.

Dimension	semiBlog	SB Demonstrator	Semblog
association	current post	current post	current post
granularity	whole post	whole post	whole post
repr. distinction	n/a	n/a	n/a
terminology reuse	yes	BiBTEX	DMOZ and FOAF
object type	literal	ont.	ont.
context	author (implicit)	author (implicit)	author (implicit)

Table 3: Annotations in current Semantic Blogging tools

4.4 Tagging

The **subject** of tagging annotations is always a URL, but each system targets different types of URLs, as explained in Sec. 2.4.

With the exception of del.icio.us-like systems, which allow tagging of arbitrary URIs including document fragments, the **subject granularity** in tagging is always a complete document.

In the analysed tagging systems, the **predicate** is always implicit. The **object** of a tagging annotation is always a literal, without **reuse of formal terminology or ontologies**. On the other hand, the collaborative nature of tagging systems coupled with the tagging suggestion mechanism employed by all tagging systems, ensures a convergence into common tags, which can be considered a reusable terminology.

The annotation **context** is not always explicitly expressed, but similarly to blogging, we can implicitly assume the person who tagged the resource as the context.

Dimension	Flickr	del.icio.us	Technorati
association	current	current	current
granularity	Flickr picture	any URI	Blog
repr. distinction	document	document	document
terminology reuse	yes/no	yes/no	yes/no
object type	literal	literal	literal
context	author	author	author

Table 4: Annotations in collaborative tagging systems

5 Unified model

In order to allow agents to access, represent and otherwise process the various kinds of annotations presented in Sect. 4, these annotations will have to be mapped onto a unified model. We will now present such a model, which is based on our discussion in Sect. 3.

Graphically, the unified model can be expressed as shown in Fig. 1. Here, the annotation is represented as a named graph (the triple in the center) which has a type (some subclass of `AnnotationType`) and other metadata (the examples in the figure are creator, date and scope) attached to it. Both type and metadata

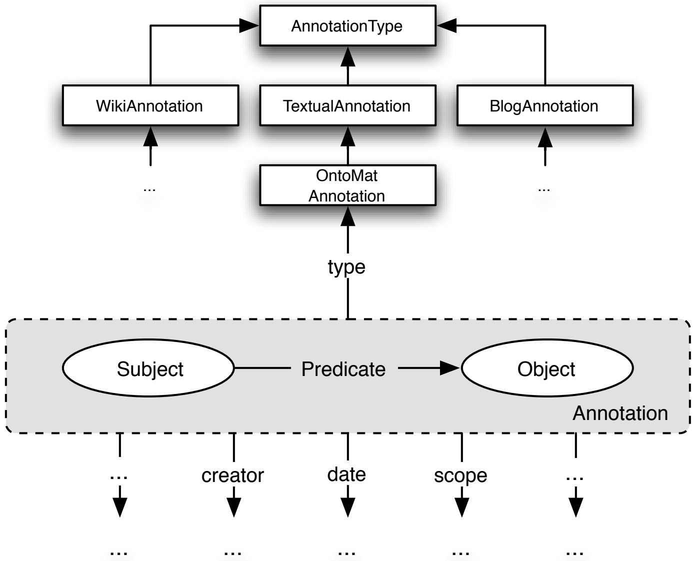


Fig. 1: Graphical representation of the Annotation Model

make up the annotation context. The values for the various annotation dimensions discussed in 4 are expressed as attributes of the types. To represent our model syntactically, we will use RDF N3 syntax, as shown in Listing 1.1. Since ordinary RDF can only express triples, we will use N3's graph quoting mechanism to allow for the representation of the full model, consisting of Annotation subject, predicate, object and context.

```
@prefix annot: <http://www.deri.org/ontology/annotation#> .

{ AnnotationSubjecct
    AnnotationPredicate AnnotationObject . }
    rdf:type          annot:OntoMatAnnotation ;
    annot:creator     ... ;
    annot:date        ... ;
    annot:scope       ... .
```

Listing 1.1: Syntactical representation of the Annotation Model

In the remainder of this section we will look at a number of illustrative examples from the various annotation domains and show how they can all be mapped into the same model.

5.1 Mapping Example: Flickr tags

As a first example we choose the rather trivial case of mapping Flickr tags to our model. We have said that in this case, the annotation subject is a picture within Flickr, the predicate is a generic predicate expressing a tagging relation (here we will simply call it `flickrTaggedWith`) and the object is a tag in the form of a literal. We can consider the person who applied the tag (the *creator*) to be the annotation context. If we query Flickr for the tags of a particular picture (e.g. by using the Flickr API[12]), we will get a response such as the one shown in Listing 1.2.

```
<photo id="144543922">
  <tags>
    <tag id="1568518-144543922-675" author="79255326@N00"
         raw="galway">galway</tag>
    <tag id="1568518-144543922-6519" author="79255326@N00"
         raw="jose">jose</tag>
    <tag id="1568518-144543922-26135" author="79255326@N00"
         raw="gonzalez">gonzalez</tag>
    <tag id="1568518-144543922-2757" author="79255326@N00"
         raw="concert">concert</tag>
  </tags>
  <urls>
    <url type="photopage">
        http://www.flickr.com/photos/dunken69/144543922/</url>
  </urls>
</photo>
```

Listing 1.2: Flickr XML source format (extract)

From the XML we can infer the picture's URL, as well as four tags that have been assigned to it ("galway", "jose", "gonzalez" and "concert"). Each tag that was applied to the picture has, among other things, an author, which in this case is identified with the id `79255326@N00`. We now have all components needed to represent this data in our unified model, as shown in listing 1.3. The core annotation triples are (pictureURL `flickrTaggedWith` tag). Each of these triples is assigned the type `FlickrAnnotation` and some metadata identifying the author of the tagging operation.

```
@prefix annot: <http://www.example.org/annotation#> .

{ <http://www.flickr.com/photos/dunken69/144543922/>
    annot:flickrTaggedWith "jose", "gonzalez", "concert", "galway" . }
    rdf:type          annot:FlickrAnnotation ;
    annot:creator     [ annot:flickrPersonID "79255326@N00" ] .
```

Listing 1.3: Flickr Annotation in the Unified Model

5.2 Mapping Example: OntoMat

Another and more complex example shows how to map annotations made with the OntoMat annotation tool. Here, the user has annotated the `www.example.org` web page. A string on the page ("Siegfried F. Handschuh") was used to assert that there exists some instance of the class `iswc:Researcher` (namely Siegfried

[12] `http://www.flickr.com/services/api/`

Handschuh). The data that OntoMat produces from this annotation process is shown in Listing 1.4. There are two parts to this data — statements A about things in the world derived from the annotated web page ("there is a researcher", here expressed using the `iswc` namespace), and meta-statements A_M about the annotations themselves ("this annotation is rooted in this piece of text", expressed using the `ontomat` namespace). In terms of our formal model, these meta-statements can be interpreted as the annotation context (but see below). In this example, the meta-statements A_M are provenance information specifying where the annotations A were originally derived from (referenced through an xpointer[13]).

```
@prefix iswc: <http://annotation.semanticweb.org/2004/iswc#> .
@prefix omat: <http://annotation.semanticweb.org/ontologies/cream/ontomat#> .

<http://www.example.org#siggi> rdf:type iswc:Researcher .

[ rdf:type             omat:ReificationDataIndividual ;
  omat:creationSource  <http://www.example.org#xpointer(84-107)> ;
  omat:aboutIndividual <http://www.example.org#siggi> ] .
```

Listing 1.4: OntoMat source format in N3

There are actually two ways of expressing the OntoMat data in the unified model: (i) Focussing on the document fragment which is the origin of A — this would result in a nested annotation ("{*xpointer isSourceOf* {*siggi isa researcher*}}"), and would reflect the fact that the authors of OntoMat consider their tool to be a "webpage annotation tool"[14]; (ii) Focussing on A, and thereby interpreting A_M as the annotation context. This interpretation would imply that OntoMat is more of an ontology population tool. Both intepretations are valid, but we will here only show the second one (see Listing 1.5), which we think is simpler and less convoluted.

```
@prefix annot: <http://www.deri.org/ontology/annotation#> .
@prefix iswc:  <http://annotation.semanticweb.org/2004/iswc#> .
@prefix omat:  <http://annotation.semanticweb.org/ontologies/cream/ontomat#> .

{ <http://www.example.org#Siggi> rdf:type iswc:Researcher }
    rdf:type             annot:OntoMatAnnotation ;
    omat:creationSource  <http://www.example.org#xpointer(84-107)> .
```

Listing 1.5: OntoMat annotation in the Unified Model

5.3 Mapping Rules

We have sofar shown by example how various, different kinds of annotations can be mapped onto the same unified model. Naturally, any kind of agent assessing and comparing such annotations would have to perform these mappings

[13] The xpointer shown here is abbreviated for formatting reasons.
[14] `http://annotation.semanticweb.org/ontomat`

automatically. Listing 1.6 shows how the previous OntoMat example could be generalized, using a simply N3 rule expression. An agent would have access to a number of similar rules, each of which covers a specific kind of annotation.

```
@prefix omat: <http://annotation.semanticweb.org/ontologies/cream/ontomat#> .
@prefix annot: <http://deri.org/ontologies/annotation#>.

{ ?s ?p ?o .
 [ omat:aboutIndividual ?s ;
   omat:creationSource  ?source ]} =>
{ { ?s ?p ?o }
  rdf:type annot:ontomat ;
  annot:source ?source . } .
```

Listing 1.6: Mapping from OntoMat to the unified model

6 Conclusion

Several tools and approaches exist to create annotations of both Web resources and abstract concepts, but there is currently no formal model that can capture all these approaches. The lack of such a model complicates the comparison and evaluation of tools and makes integration of annotations difficult.

We have analysed semantic annotations in the domains of traditional document annotations, semantic wikis and semantic blogs and in collaborative tagging systems. We have then presented a formal model that is capable of capturing the different notions of *semantic annotation* in all these approaches.

We have characterised annotations in different dimensions, and analysed existing tools acccording to these dimensions. A summary of our analysis is shown in Table 5, displaying the rough differences between the various approaches.

Finally, we have shown how our unifying model can be used to integrate various kinds of annotations, and have shown how some example annotations can be mapped into a common model, thereby giving a first insight in how to use Semantic Web technology to bridge traditional methods (document annotation), new developments such as Semantic Wikis and Blogs, as well as the so-called "Web 2.0" (collaborative tagging).

dimension	**Document**	**Semantic Wiki**	**Semantic Blog**	**Tagging**
terminology reuse	yes	yes/no	yes/no	no
association	flexible	current page	current post	n/a
granularity	fragment	page/fragment	post	resource
repr. distinction	yes/no	yes/no	no	no
object type	ont.	literal/ont.	ont.	URI
context	yes/no	no	author (implicit)	author (implicit)

Table 5: Comparison of annotation approaches

References

1. A. Abecker and L. van Elst. Ontologies for knowledge management. In S. Staab and R. Studer, (eds.) *Handbook on Ontologies*, pp. 435–454. Springer, 2004.
2. D. Aumueller. Semantic authoring and retrieval within a wiki. In *ESWC*. 2005.
3. S. Bechhofer, *et al.* The semantics of semantic annotation. In *ODBASE*. Irvine, California, 2002.
4. S. Cayzer. Semantic blogging: Spreading the semantic web meme. In *XML Europe 2004, Amsterdam, Netherlands, Proceedings*. Apr. 2004.
5. H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *40th Anniversary Meeting of the Association for Computational Linguistics*. 2002.
6. L. van Elst and A. Abecker. Ontologies for information management: balancing formality, stability, and sharing scope. *Expert Systems with Applications*, 23:357–366, 2002.
7. J. Euzenat. Eight Questions about Semantic Web Annotations. *IEEE Intelligent Systems*, 17(2):55–62, Mar/Apr 2002.
8. T. R. Gruber. Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli, (eds.) *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers, 1993.
9. S. Handschuh. *Creating Ontology-based Metadata by Annotation for the Semantic Web*. Ph.D. thesis, University of Karlsruhe, 2005.
10. J. Kahan, M. Koivunen, E. Prud'Hommeaux, and R. Swick. Annotea: An open RDF infrastructure for shared web annotations. In *WWW Conf.*, pp. 623–632. 2001.
11. C. Marlow, M. Naaman, D. Boyd, and M. Davis. Position paper, tagging, taxonomy, flickr, article, toread. In *Collaborative Web Tagging Workshop at WWW Conf.* May 2006.
12. K. Möller, J. G. Breslin, and S. Decker. semiBlog - Semantic Publishing of Desktop Data. In *14th Conference on Information Systems Development (ISD2005), Proceedings*. Karlstad, Sweden, August 2005.
13. E. Oren. SemperWiki: a semantic personal Wiki. In *SemDesk in ISWC*. 2005.
14. E. Oren, J. G. Breslin, and S. Decker. How semantics make better wikis. In *WWW (poster)*. 2006.
15. E. Oren, *et al.* Annotation and navigation in semantic wikis. In *SemWiki in ESWC*. 2006.
16. F. Rinaldi *et al.* Multilayer annotations in Parmenides. In *Proc. of the K-CAP2003 workshop on Knowledge Markup and Semantic Annotation*. 2003.
17. P. Sazedj and H. S. Pinto. Time to evaluate: Targeting annotation tools. In *Proc. of Knowledge Markup and Semantic Annotation at ISWC 2005*. 2005.
18. S. Schaffert, A. Gruber, and R. Westenthaler. A semantic wiki for collaborative knowledge formation. In *Semantics 2005*. 2005.
19. H. Takeda and I. Ohmukai. Semblog Project. In *Activities on Semantic Web Technologies in Japan, A WWW2005 Workshop*. 2005.
20. V. Uren *et al.* Semantic annotation for knowledge management: Requirements and a survey of the state of the art. *Journal of Web Semantics*, 4(1):14–28, 2006.
21. M. Völkel, *et al.* Semantic wikipedia. In *WWW*. 2006.
22. J. Walker. Weblog. In D. Herman, M. Jahn, and M.-L. Ryan, (eds.) *Routledge Encyclopedia of Narrative Theory*, p. 45. Routledge, London and New York, 2005.