Open Annotations on Multimedia Web Resources

**Bernhard Haslhofer · Robert Sanderson ·
Rainer Simon · Herbert van de Sompel**

Abstract Many Web portals allow users to associate additional information with existing multimedia resources such as images, audio, and video. However, these portals are usually closed systems and user-generated annotations are almost always kept locked up and remain inaccessible to the Web of Data. We believe that an important step to take is the integration of multimedia annotations and the Linked Data principles. We present the current state of the Open Annotation Model, explain our design rationale, and describe how the model can represent user annotations on multimedia Web resources. Applying this model in Web portals and devices, which support user annotations, should allow clients to easily publish and consume, thus exchange annotations on multimedia Web resources via common Web standards.

Keywords Annotations · Web · Linked Data

1 Introduction

Youtube and Flickr are examples of large-scale Web portals that allow users to annotate multimedia resources by adding textual notes and comments to im-

Bernhard Haslhofer
Cornell University, Department of Information Science
301 College Avenue, Ithaca, NY 14850, USA
E-mail: bernhard.haslhofer@cornell.edu

Robert Sanderson and Herbert van de Sompel
Los Alamos National Laboratory
Los Alamos, NM 87544, USA
E-mail: rsanderson,herbertv@lanl.gov

Rainer Simon
Austrian Institute of Technology
Donau-City-Str. 1, A-1220 Vienna, Austria
E-mail: rainer.simon@ait.ac.at

Fig. 1 An annotation example on Flickr.

ages or videos. Figure 1 shows an image[1] from the Flickr Commons collection contributed by the Library of Congress. Flickr users added several annotations to specific image segments, one of them telling us that this picture shows a cathedral in Bergen.

Annotations describe resources with additional information, which is valuable to other users, who are searching and browsing resource collections. They are also important for underlying information systems, which can exploit the high-level descriptive semantics of annotations in combination with automatically extracted low-level features, such as image size and color, to implement search and retrieval over multimedia resources. Taking the previous example, users who are searching for "Bergen" or even "Bergen Cathedral" will now find this particular image in Flickr, because some user provided this descriptive information in textual form.

Annotations are also becoming an increasingly important component in scholarly cyber-infrastructures (cf. [5]), which are often realized as Web systems. Therefore, a Web-based annotation model should fulfill several requirements. In the age of video blogging and real-time sharing of geo-located images, the notion of solely textual annotations has become obsolete. Instead, *multimedia* Web resources should be annotatable and also be able to be annotated onto other resources. Users often discuss multiple segments of a resource, or multiple resources, in a single annotation and thus the model should support multiple targets. An annotation framework should also follow the Linked Open Data guidelines [13] to promote annotation sharing between systems. In order to avoid inaccurate or incorrect annotations, it must take the ephemeral nature of Web resources into account.

---

[1] http://www.flickr.com/photos/library_of_congress/3175009412/

Annotations on the Web have many facets: a simple example could be a textual note or a tag (cf., [14]) annotating an image or video. Things become more complex when a particular paragraph in an HTML document annotates a segment (cf., [12]) in an online video or when someone draws polygon shapes on tiled high-resolution image sets. If we further extend the annotation concept, we could easily regard a large portion of Twitter tweets as annotations on Web resources. Therefore, in a generic and Web-centric conception, we regard an annotation as an association created between one *body* resource and other *target* resources, where the body must be somehow *about* the target.

Annotea [17] already defines a specification for publishing annotations on the Web but has several shortcomings: (i) it was designed for the annotation of Web pages and provides only limited means to address segments in multimedia objects, (ii) if clients want to access annotations they need to be aware of the Annotea-specific protocol, and (iii) Annotea annotations do not take into account that Web resources are very likely to have different states over time.

Throughout the years several Annotea extensions have been developed to deal with these and other shortcomings: Koivunnen [21] introduced additional types of annotations, such as *bookmark* and *topic*. Schroeter and Hunter [33] proposed to express segments in media-objects by using *context* resources in combination with formalized or standardized descriptions to represent the context, such as SVG or complex datatypes taken from the MPEG-7 standard. Based on that work, Haslhofer et al. [10] introduce the notion of *annotation profiles* as containers for content- and annotation-type specific Annotea extensions and suggested that annotations should be dereferencable resources on the Web, which follow the Linked Data guidelines. However, these extensions were developed separately from each other and inherit some of the above-mentioned Annotea shortcomings.

In this article we describe the *Open Annotation Model*[2], which is currently being developed in an international collaboration. It applies a Web- and resource-centric view on annotations and defines a modular architecture, which has a simple base line model in its core. The model also provides means to address segments in multimedia resources either by encoding segment information using the Media Fragment URI specification or by introducing custom segment constraints for more complex annotation use cases. By allowing fixity and timestamp information on the resources involved in an annotation, it also takes into account the ephemeral nature of Web resources. Pulling together the functionalities provided by various, partly independent Annotea extensions is a major goal of this effort.

This article extends our work previously published in [11] by the following: it explains the design rationale that lead to the specification of the current annotation model and also the technical aspects of the model in more detail. It also contains an updated and extended related work section.

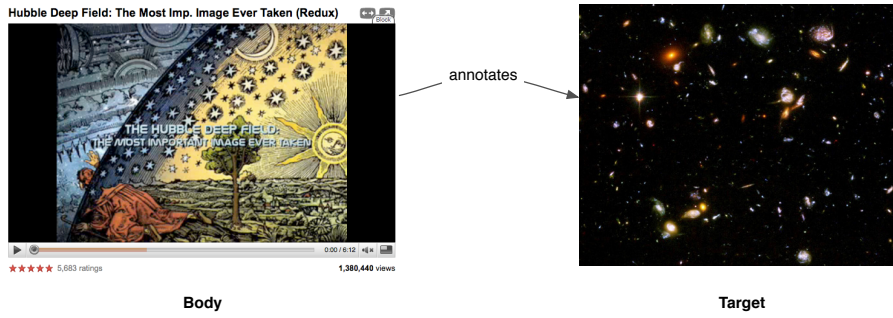[2] Open Annotation Model: Beta Data Model Guide `http://www.openannotation.org/spec/beta/`

Fig. 2 A Youtube video annotating an image on the Web.

2 Design Rationale

In this section we outline the design rationale that drives the specification of the Open Annotation Model. We give examples illustrating common requirements we found in several real-world annotation use cases (e.g., [31, 38, 35]) and describe the reasons behind our design decisions. From each design decision we derived a set of guiding design principles, which are reflected in the current Open Annotation Model design.

2.1 Annotations are qualified associations between resources

Textual notes or tags on images, as in the Flickr example in Figure 1, occur frequently on the Web and are the simplest examples for Web annotations. In Open Annotation Model terms, these are *annotations* that have a textual *body* and an image resource as *target*. However, in many scenarios the prevailing view that annotation bodies are textual is insufficient. Figure 2 shows an annotation in which the body is not a textual note, but a video, which itself is an addressable Web resource identified by a URI. In order to cover such cases, we must abstract from purely textual annotation bodies and model them as resources that can be of any media type.

From a conceptual point of view, an annotation is an association between two resources, the *body* and the *target*. However, in most cases this association needs to be qualified and addressable in some way. Indicating the creator and creation date of a resource, or replying to existing annotations, are frequently occurring scenarios that require an annotation to be expressed as a first-class entity. Web annotations are then instances of an annotation and relate together the body and target resources that are involved in the annotation association. With this approach we follow a common design pattern, which is also known as Qualified Relation in Linked Data [8] or Association Class in UML.

In contrast to existing tagging models (see [18]) the annotation *creator* is not a mandatory core model entity. We believe that adopters of the Open Annotation Model already have user models in place or rely on open Web identities (e.g., a Google or Facebook account) and are most likely not willing

to map their user models against another user model, which is specified as part of an annotation model. However, we encourage adopters to relate annotations with existing Web resources, which identify users.

From these considerations we derive the following guiding principles:

– All core entities (annotation, body, target) must be resources.
– Annotations must allow for both body and target of any media type.
– Annotations, bodies, and targets can have different authorship.

2.2 Annotations involve parts of multimedia resources

Our introductory Flickr example illustrates how annotations can target specific segments in Web resources. In Figure 2 we gave an example that involves multimedia resources: a video as body, and an image as target. There is strong dependency between these requirements because a resource's media type affects the way segments need to be addressed. Annotating an area in an image requires a different segment representation than those that target segments in the spatial and temporal dimensions of a video. Similarly, addressing text segments in a PDF document differs from addressing text in plain text or HTML documents. Since we abstract from purely textual annotation bodies, we must take into account that both the body and the target of an annotation can be resource segments. We could, for instance, refine our previous example by saying that some sequence within the video annotates a certain image segment.

The problem of addressing media segments, also known as fragment identification, is well known and will be explained in more detail in Section 6.3. Since the Open Annotation Model will be implemented in Web environments and all resources involved in an annotation are Web resources identified by URIs, it should reuse the fragment identification mechanisms that are already defined as part of the Architecture of the World Wide Web [16] and extensions thereof, such as fragment construction rules for specific media types. However, many annotation use cases require more complex segment representations such as polygon regions in images, which cannot be expressed with available standards. Therefore, our guiding principles with respect to addressing segments in multimedia resources are:

– Annotations must support resource segment addressing both on body and target resources.
– Preferably this should be done with (media) fragment URIs, but extensibility must be provided for cases in which the use of URIs for segment addressing is not possible.

2.3 Annotation resources are ephemeral Web resources

Previously we argued that all core entities of the Open Annotation Model must be first class Web resources, which are identified by URIs, if possible HTTP
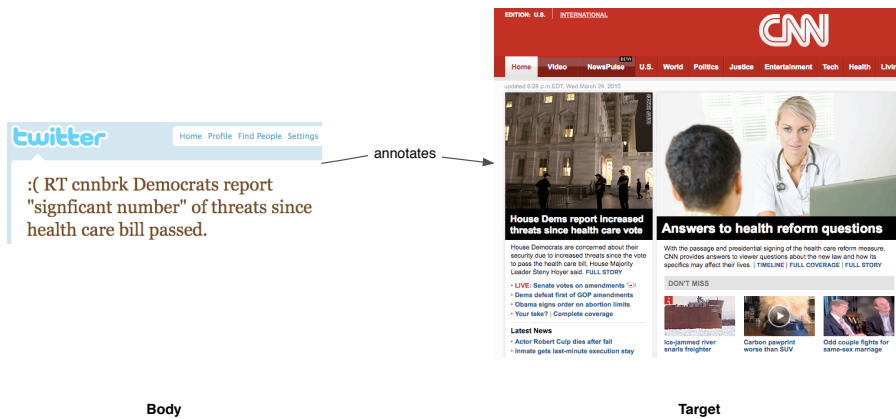
Fig. 3 A Twitter tweet annotating the CNN website.

URIs. The great benefit of this approach is that existing technologies and solutions that can be applied for Web resources (e.g., mime-types, fragments, access control, etc.) also work for resources involved in a Web annotation without the necessity to include these aspects in an annotation specification. However, there is one big problem we inherit from the Web architecture and which is severe in the context of annotations: URI-addressable Web resources are ephemeral, which means that the representations obtained by dereferencing their URIs may change over time.

The annotation example in Figure 3 shows a Twitter tweet annotating the CNN web site and illustrates the ephemerality problem: the tweet refers to the CNN page at a certain point in time and might be misinterpreted when the CNN main web site changes. Measures should be provided that can help in avoiding misinterpretations of annotations, including the expression of timestamps and fixity information for body and target resources.

2.4 Annotations should be interoperable

The focus of the Open Annotation Model is on sharing annotations on scholarly resources and therefore the model should have sufficient richness of expression to satisfy scholars' needs. However, in order to maximize the likelihood of adoption, the model should also be an interoperability framework readily applicable in other domains. One possibility to achieve this is to follow a modular and extensible modeling approach, with a generally applicable baseline model and domain- or media-type specific extensions. An annotation client should implement at least the baseline model and, if possible, provide fallback behavior for annotations that contain domain-specific extensions the client might not be aware of.

In order to increase the likelihood of adoption, and in alignment with the goal of sharing annotations, no client-server protocol for publishing, updating,
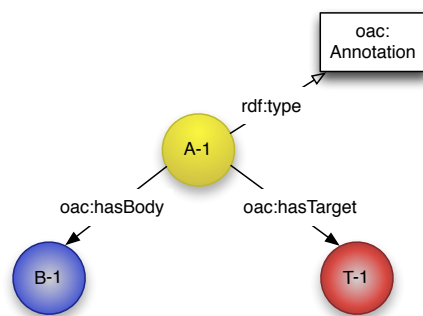
Fig. 4 Open Annotation baseline data model.

or deleting annotations will be specified. Rather, the specifications will take a perspective whereby clients publish annotations to the Web and make them discoverable using common Web approaches. Such an approach does not require a preferred annotation server for a client, yet it does not preclude one either.

Our guiding principles to achieve annotation interoperability and widespread adoption are:

− The Open Annotation Model should have a simple but expressive baseline model defining top-level classes/entities and properties/relationships.
− The baseline model should be extensible.
− Annotation protocols are out of scope.

3 The Baseline Model

The Open Annotation data model draws from various extensions of Annotea to form a cohesive whole. The Web architecture and Linked Data guidelines are foundational principles, resulting in a specification that can be applied to annotate any set of Web resources. At the time of this writing, the specification, which is available at `http://www.openannotation.org/spec/beta/`, is still under development. In the following, we describe the major technical building blocks of the Open Annotation data model and use some simple examples to illustrate how to apply the model in practice. For further examples, which also cover different use cases, such as the annotation of medieval manuscripts, we refer to the online documentation.

Following its predecessors, the Open Annotation Model, shown in Figure 4, has three primary classes of resources. In all cases below, the `oac` namespace prefix expands to `http://www.openannotation.org/ns/`.

− The `oac:Body` of the annotation (node B-1): This resource is the comment, metadata or other information that is created about another resource. The body can be any Web resource, of any media format, available at any URI.
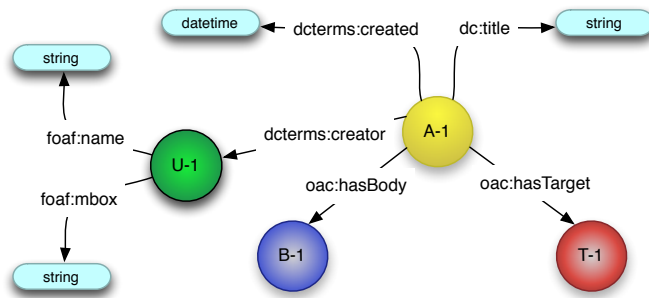
Fig. 5 Additional properties and relationships in the Open Annotation baseline model.

The model allows for either one body per annotation, or an annotation without any body, but not annotations with multiple bodies.
- The `oac:Target` of the annotation (node T-1): This is the resource that the body is about. Like the body, it can be any URI identified resource. The model allows for one or more targets per annotation.
- The `oac:Annotation` (node A-1): This resource is an RDF document, identified by an HTTP URI, that describes at least the body and target resources involved in the annotation as well as any additional properties and relationships (e.g., `dcterms:creator`). Dereferencing an annotation's HTTP URI returns a serialization in a permissible RDF format.

An annotation is a Web resource, which is identified by an HTTP URI and returns an RDF document when being dereferenced. As with any RDF data, additional properties and relationships can be associated with any of the resources. It is recommended that an annotation has a timestamp of when the annotation relationship was created (`dcterms:created`) and a reference to the agent that created it (`dcterms:creator`). Resources referenced by additional relationships may themselves have additional properties and relationships. Figure 5 gives an example of recommended and other possible (e.g., `dc:title`) properties and relationships that can be added to the Open Annotation baseline model. The set of properties and relationship in this example is by no means exhaustive. Properties and relationships from other vocabularies may also be used. It is also important to note that the creator and created timestamp of each of the three resource types above may be different. An annotation might refer to an annotation body created by a third party, perhaps from before the Open Annotation specification was published, and a target created by yet another party.

Similarly, there may be additional subclasses of `oac:Annotation` that further specify restrictions on the meaning of the annotation, such as a *reply*; an annotation where the single target is itself an annotation. This allows chaining of annotations into a threaded discussion model.

If the body of an annotation is identified by a dereferencable HTTP URI, as it is the case in Twitter, various blogging platforms, or Google Docs, it can easily be referenced from an annotation. If a client cannot create URIs for
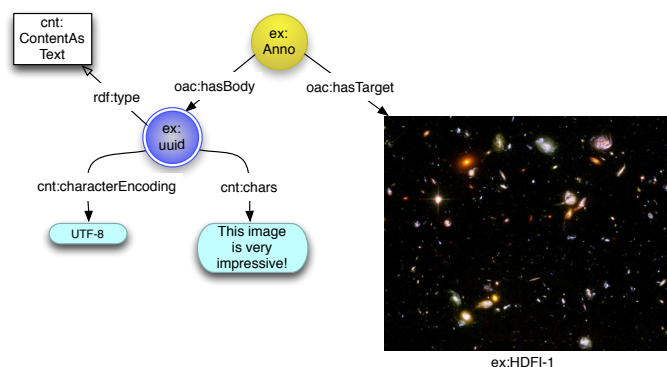
Fig. 6 An example annotation with inline body.

an annotation body, for instance because it is an offline client, it can assign a unique non-resolvable URI (called a URN) as the identifier for the body node. This approach can still be reconciled with the Linked Data principles as servers that publish such annotations can assign HTTP URIs they control to the bodies, and express equivalence between the HTTP URI and the URN.

The Open Annotation Model allows the inclusion of information directly in the annotation document by adding the representation of a resource inline within the RDF document using the *Content in RDF* specification [20] from the W3C. The example annotation in Figure 6 shows how to express this: the representation is the object of the `cnt:chars` predicate, and its character encoding the object of the `cnt:characterEncoding` predicate. Further classes from this specification include Base64 encoded resources and XML encoded resources.

4 Annotating Media Segments

Most of the use cases, which have been explored before specifying the model, involved comments that were about a segment of a resource, rather than the entire resource identified by a URI. The data model allows two different methods of identifying and describing the region of interest of a resource; either using a *fragment URI*, or a more expressive *constraint* resource. It is clearly recommended to use fragment URIs whenever possible, because this method relies on normative specifications, which brings interoperability with other applications. Only if there are no appropriate URI fragment specifications available, the creators should define their own constraints.

4.1 Describing segments with fragment URIs

A fragment URI normally identifies a part of a resource, and the method for constructing and interpreting these URIs is dependent on the media type of

the resource. In general, fragment URIs are created by appending a fragment that describes the section of interest to the URI of the full resource, separated by a '#' character (see [3]).

There are two main sources for existing fragment URI specifications, which can both identify and describe how to discover a segment of interest within a resource. The first is the set of Mime Type specification RFCs from the IETF. This includes X/HTML (RFC 2854/3236), XML (RFC 3023), PDF (RFC 3778) and Plain Text (RFC 5147). The second is W3C Media URIs specification [39], which is defined at a broader level to cover images, video and audio resources, regardless of the exact format. The following examples show how to apply these specifications to define media-type specific URIs that describe a certain resource segment:

- `http://www.example.net/foo.html#namedSection` identifies the section named as "namedSection" in an HTML document.
- `http://www.example.net/foo.pdf#page=10&viewrect=20,100,50,60` identifies a rectangle starting at 20 pixels in from the left, and 100 down from the top, with a width of 50 and a height of 60 in a PDF document.
- `http://www.example.net/foo.png#xywh=160,120,320,240` identifies a 320 by 240 box, starting at x=160 and y=120 in an image.
- `http://www.example.net/foo.mpg#t=npt:10,20` identifies a sequence starting just before the 10th second, and ending just before the 20th in a video.

We recommend that when a definition exists for how to construct a fragment URI for a particular document format, and such a fragment would accurately describe the section of interest for an annotation, then this technique should be used. It is recommended to also use `dcterms:isPartOf` with the full resource as the object, in order to make the annotation more easily discoverable. Figure 7 shows an example in which a tweet annotates a rectangular section in an image, which in turn is identified and described by a media fragment URI.

4.2 Describing segments via constraint resources

There are many situations when segments cannot be described with fragment URIs, but it is still desirable to be able to annotate a segment of a resource. For example, a non-rectangular section of an image, or a segment of a resource with a format or media type that is not covered by either fragment specification, such as a 3-dimensional model or a dataset. To handle these situations, we introduce an `oac:Constraint` resource that describes the segment of interest using an appropriate standard, and a `oac:ConstrainedTarget` resource that identifies the segment of interest. This *constrained target* is the object of the `oac:hasTarget` predicate of the `oac:Annotation`, and subsequently `oac:constrains` the full target resource. Figure 8 shows how constrained targets extend the Open Annotation baseline model.
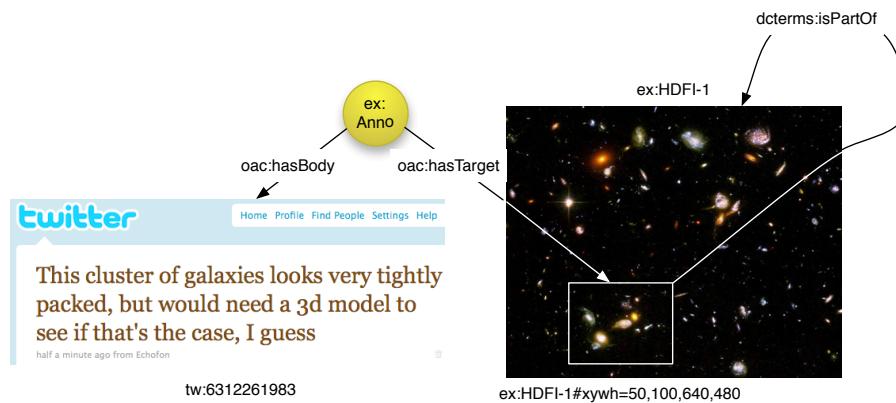
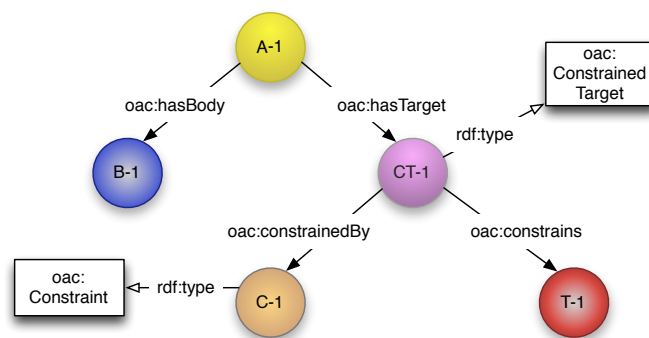**Fig. 7** Annotating media segments using a fragment URI.

Fig. 8 Constraint annotation targets in the Open Annotation Model.

The nature of the constraint description will be dependent on the type of the resource for which the segment is being conveyed. It is then up to the annotation client to interpret the segment description with respect to the full resource. Figure 9 shows an example in which an area within an image is described by an SVG path element. The document containing the SVG specification is identified by a dereferencable HTTP URI and a specialized type `oac:SvgConstraint` in combination with a `dc:format` property informs the client about the type of constraint it needs to deal with.

Alternatively, it is also possible to include the constraint information inline within the annotation document using the same technique as used for including the body. The `oac:Constraint` is given a URN (normally a urn:UUID) and then the constraint information is included as the value of the `cnt:chars` property. The requirements for doing this are the same as for including the `oac:Body` inline within the annotation document. For more complex use cases it also possible to express constraints in RDF and to apply constraints also on the body of an annotation.
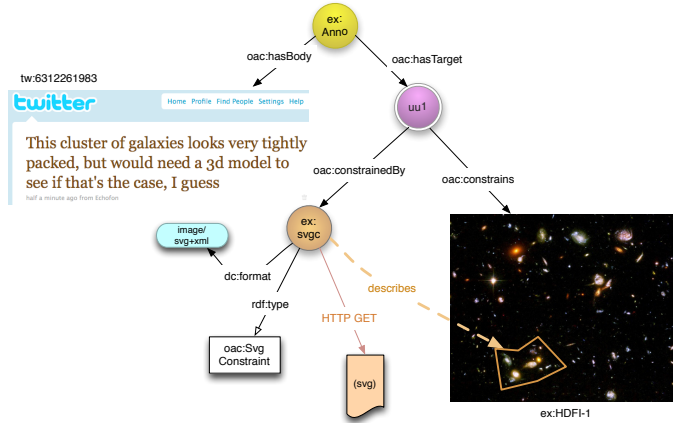
Fig. 9 Annotating media segments using an SVG constraint.

One goal of the ongoing Open Annotation demonstrator activities is to collect real-world constraint definitions from various use cases and to specify them in the context of the OA model. We hope that this also serves as as feedback loop for possible enhancements or additional URI fragment specifications.

5 Robust Annotations over Time

It must be stressed that different agents may create the *annotation*, *body* and *target* at different times. For example, Alice might create an annotation saying that Bob's YouTube video annotates Carol's Flickr photo. Also, being regular Web resources, the body and target are likely to have different representations over time. Some annotations may apply irrespective of representation, while others may pertain to specific representations. In order to provide the ability to accurately interpret annotations past their publication, the Open Annotation Model introduces three ways to express temporal context. The manner in which these three types of annotations use the `oac:when` property, which has a datetime as its value, distinguishes them.

A *Timeless Annotation* applies irrespective of the evolving representations of body and target; it can be considered as if the annotation references the semantics of the resources. For example, an annotation with a body that says "This is the front page of CNN" remains accurate as representations of the target `http://cnn.com/` change over time. Timeless annotations dont make use of the `oac:when` property.

A *Uniform Time Annotation* has a single point in time at which all the resources involved in the annotation should be considered. This type of annotation has the `oac:when` property attached to the `oac:Annotation`. For example, if Alice recurrently publishes a tweet that comments on a story on the live CNN home page, an annotation that has the cartoon as body and
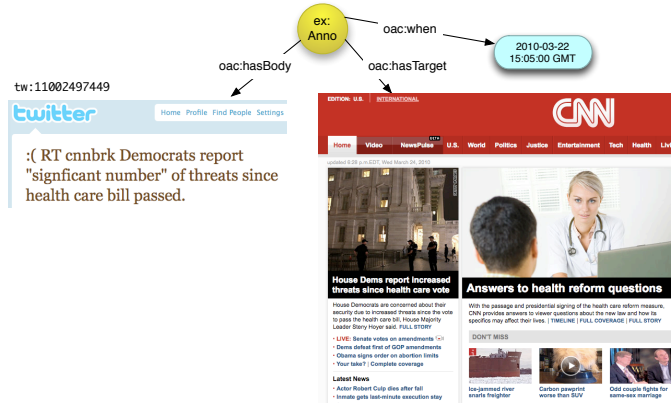
Fig. 10 A Uniform Time Annotation example.

the CNN home page as target would need to be handled as a Uniform Time Annotation in order to provide the ability to match up correct representations of body and target. Figure 10 shows how Uniform Time Annotations can be represented using the Open Annotation Model.

A *Varied Time Annotation* has a body and target that need to be considered at different moments in time. This type of Annotation uses the `oac:when` property attached to an `oac:WebTimeConstraint` node, which is a specialization of `oac:Constraint`, for both body and target. If, in the aforementioned example, Alice would have the habit to publish a cartoon at `http://example.org/cartoon` when the mocked article is no longer on the home page, but still use `http://cnn.com` as the target of her annotation, the Varied Time Annotation approach would have to be used.

This temporal information can be used to recreate the annotation as it was intended by reconstructing it with the time-appropriate body and target(s). Previous versions of Web resources exist in archives such as the Internet Archive, or within content management systems such as MediaWiki's article history, however they are divorced from their original URI. Memento [36,7], which is a framework that proposes a simple extension of HTTP in order to connect the original and archived resources, can be applied for recreating annotations. It leverages existing HTTP capabilities in order to support accessing resource versions through the use of the URI of a resource and a datetime as the indicator of the required version. In the framework, a server that host versions of a given resource exposes a TimeGate, which acts as a gateway to the past for a given Web resource. In order to facilitate access to a version of that resource, the TimeGate supports HTTP content negotiation in the datetime dimension. Several mechanisms support discovery of TimeGates, including HTTP links that point from a resource to its TimeGate(s) [32].

6 Related Work

In this section we give an overview of existing work in the area of Web annotations. After summarizing general works about annotations and annotation interoperability, we analyze the features of existing Web annotation models and compare them with those of the Open Annotation Model.

6.1 Annotations and annotation interoperability

Annotations have a long research history, and unsurprisingly the research perspectives and interpretations of what an *annotation* is supposed to be vary widely. Agosti et al. [1] provide a comprehensive study on the contours and complexity of annotations. A representative discussion on how annotations can be used in various scholarly disciplines is given by Bradley [5]. He describes how annotations can support interpretation development by collecting notes, classifying resources, and identifying novel relationships between resources.

The different forms and functions that annotations can take are analyzed by Marshall [22]. She distinguishes between *formal* and *informal* annotations, whereby formal annotations follow structural standards and informal ones are unstructured. Furthermore, Marshall divides into *implicit* annotations that are intended for sharing and *explicit* annotations of personal nature, often interpretable only by the original creator. Further divisions defined by Marshall with regard to the function of an annotation include *permanent* vs. *transient*, *annotation as writing* vs. *annotation as reading*, *extensive* vs. *intensive*, *published* vs. *private* and *institutional* vs. *workgroup* vs. *individual*. The difference between personal and public annotations in a digital environment is further investigated in a study by Marshall and Brush [23]. They derive design implications for annotation systems, e.g. regarding find and filtering requirements, and user interface strategies for processing and sharing annotations.

A taxonomy of annotation types and marking symbols used by readers of scholarly documents is presented by Qayyum [29]. His taxonomy is derived from the results of a user study conducted with students reading research articles in a private as well as a collaborative digital setting. A related recent effort is presented by Blustein et al. [4]. In their field study, conducted over the course of three years, they identify six purposes for scholarly annotation: *interpretation*, *problem-working*, *tracing progress*, *procedural annotations*, *place marking and aiding memory* and *incidental markings*.

6.2 Web annotation models

The idea of publishing user annotations on the Web is not new. Annotea [17] was specified more than a decade ago and defines a data model and protocol for uploading, downloading, and modifying annotations. Since the Web has changed over time and now also comprises non-document Web resources,

		Web Annotation Models					
		Annotea (2001)	**Co-Annotea (2008)**	**Lemo (2009)**	**M3O (2010)**	**Annotation Ontology (2011)**	**Open Annotation Model (2011)**
Features	**Annotation Representation**	HTTP-dereferencable RDF Document	HTTP-dereferencable RDF Document	Linked Open Data	Embedded in Content Representation	Linked Open Data	Linked Open Data
	Annotation Target Types	(X)HTML Web Documents	Multimedia Objects	Multimedia Web Resources	Structured Multimedia Presentations	Multimedia Web Resource	Multimedia Web Resource
	Annotation Body Types	Text	Multimedia Objects	Text	Vocabulary Terms	Vocabulary Terms	Multimedia Web Resource
	Annotation Type Extensibility	Yes	Yes	Yes	N/A	Yes	Yes
	Media Segment Identification	Xpointer	Context + MPEG-7 fragment URIs	Extensible Fragment + MPEG-21 URIs	N/A	Extensible Selectors + Xpointer	Extensible Constraints + RFC / Media Fragment URIs
	Fixity Support	Annotation	Annotation	Annotation	N/A	Annotation	Annotation + Body + Target
	Defines Protocol	Yes	No (uses Annotea protocol)	No (Linked Data)	No	No	No
	Security	No	No	No	No	No	No
	Provenance Model	Dublin Core Elements	Dublin Core Elements	Dublin Core Elements	Custom Annotation Pattern Model	FOAF / Swan / Dublin Core Elements or Terms	FOAF / Dublin Core Terms
	Base Data Model	RDF	RDF	RDF	RDF	RDF	RDF
	Definition Language	RDFS	RDFS	OWL	OWL	OWL	OWL

Fig. 11 Feature analysis of existing Web annotation models.

the Annotea model soon became insufficient for many annotation use cases, as we explained at the beginning of this article. Annotea extensions, such as Co-Annotea [34] or LEMO [10], were developed to deal with the Annotea shortcomings and to take into account emerging architectural styles, such as RESTful Web Services, or Linked Open Data. Recent Web annotation model specification efforts include the M3O ontology [30], the Annotation Ontology [6], and the Open Annotation Model, which we presented in this article.

In Figure 11 we present the results of a feature comparison we performed across the previously mentioned models. The models are timely ordered by their publication year, and the feature selection is based on the requirements and use case descriptions we found in the model documentations. Although we cannot generalize from this representative set of annotation models and fea-

tures, we can observe hat annotation models have continually been adapted to emerging standards, needs, and architectures: Linked Open Data is increasingly adopted for publishing and sharing annotations, multimedia resources can now be annotated also by other multimedia resources, extensibility has become a key requirement, security and provenance are being outsourced to other models, and standard segment identification mechanisms are being combined with custom solutions (context, fragment, selectors) to capture complex domain-specific needs.

The Annotation Ontology, which is an open ontology for the annotation of scientific documents on the Web, is technically very similar to the Open Annotation Model. However, they differ e.g., in terms of how fragments are being expressed: by representing constraints and constraint targets as first-class resources the Open Annotation Model supports direct addressing of fragments, thus enabling use cases where different users annotate the same fragment, or search scenarios where annotations are retrieved by fragment. Furthermore, the Open Annotation Model supports structured annotation bodies and allows to overlay semantic statements pertaining to one or more annotation targets, which offers potentially more flexibility, e.g., for use cases of entity and entity-relation extraction in scientific literature.

A related strand of research concerns models for *social tagging* of Web resources. Hunter [14] describes tags as "*a subclass of annotations that comprise simple, unstructured labels or keywords assigned to digital resources to describe and classify the digital resource*". A comparison of tagging ontologies is presented by Kim et al. [18]. They survey the state of the art in tagging models and identify three building blocks common to existing tagging models: *taggers*, the *tags* themselves and the *resources* being tagged.

Semantic annotations features can also be found in multimedia metadata frameworks such as MPEG-7 and multimedia metadata ontologies such as COMM [2] or the recent *W3C Ontology for Media Resources*[3] specification, which provides a core metadata vocabulary for media resources on the Web. It defines two metadata properties that can be used for the textual description of a media resource (fragment) or for relating RDF files or named graphs to a media resource. Other ontologies were designed to embed annotations directly into the multimedia content representation. The M3O Ontology [30], for instance, allows the integration of annotations with SMIL and SVG documents. On the contrary, the Open Annotation Model is more in line with the previously discussed Web Annotation models. It treats annotations as first class Web resources, which can exist independently from the content or metadata representations of media objects. This design choice is motivated by a set of scholarly use cases, which require that multimedia content objects can be annotated any time after the content production and metadata extraction process.

---

[3] `http://www.w3.org/TR/mediaont-10/#example3`

6.3 Media segment identification

Early related work on the issue of describing segments in multimedia resources can be traced back to research on linking in hypermedia documents (cf. [9]). For describing segments using a non-URI based mechanism one can use MPEG-7 Shape Descriptors (cf. [25]) or terms defined in a dedicated multimedia ontology. SVG [40] and MPEG-21 [15] introduced XPointer-based URI fragment definitions for linking to segments in multimedia resources. The Temporal URI specification [26] addresses a temporal segment in a time-based media resource through a defined URL query parameter ('t='). YouTube supports similar direct linking to a particular point in time in a video using a fragment URI. The Media Fragments URI Specification [37] is a W3C Working Draft that introduces a standard, URI-based approach for addressing temporal, spatial and track sub-parts of any non-textual media content, thus making audiovisual media segments first class citizens on the Web [12].

6.4 Robustness of Web resources

The ephemeral nature of Web resources and methods to deal with that problem have been studied from the early years of the Web on. Phelps and Wilensky [27] proposed to decorate hyperlinks with lexical signatures to re-find disappeared web resources. Recent works include Klein et al. [19], who proposed to compute lexical signatures from the link neighborhood of a Web page, and Morishima et al. [24] who describe a method to fix broken links when link targets have moved. The problem has also been realized in the Linked Data context and solutions like DSNotify [28] were proposed to re-find resources by their representations.

7 Summary and Future Directions

We apply a generic and Web-centric conception to the various facets annotations can have and regard an annotation as association created between one *body* resource and other *target* resources, where the body must be somehow *about* the target. This conception lead to the specification of the Open Annotation Model, which originates from activities in the Open Annotation Collaboration and aims at building an interoperable environment for publishing annotations on the Web.

At the time of this writing, the Open Annotation Model is still in beta stage and is currently implemented in several demonstration projects covering use cases ranging from annotating medieval manuscripts, over annotating online maps, to annotating online video segments. As part of these demonstration projects client APIs are being developed for various programming environments to ease model adoption for developers. We expect to obtain user and developer feedback from these projects, which should further refine the Open Annotation Model.

Pursuing better integration of the proposed segment identification approach with the W3C Media Fragment URI specification is on our research agenda. However, this requires an extension mechanism in that specification, which is not within the scope of the responsible working group at the moment. Also the question of how to model multiple annotation targets and how to interpret this information correctly, is currently being discussed. Finally, as a first outcome of the demonstration projects, we observed that a common set of use cases is annotating data with other data, rather than with information intended for human consumption. This raises the question of how to model *Data Annotations* in an interoperable way.

As a final result, we expect a data model, which provides an interoperable method of expressing annotations such that they can easily be shared between platforms, with sufficient richness of expression to satisfy also complex annotation scenarios.

References

1. Maristella Agosti, Giorgetta Bonfiglio-Dosio, and Nicola Ferro. A historical and contemporary study on annotations to derive key features for systems design. *Int. J. on Digital Libraries*, 8(1):1–19, 2007.
2. Richard Arndt, Raphaël Troncy, Steffen Staab, Lynda Hardman, and Miroslav Vacura. COMM: designing a well-founded multimedia ontology for the web. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007*, pages 30–43, Busan, Korea, November 2007.
3. Tim Berners-Lee, Roy Fielding, and Larry Masinter. *Uniform Resource Identifier (URI): Generic Syntax*. IETF, 2005. Available at: `http://tools.ietf.org/html/rfc3986`.
4. James Blustein, David Rowe, and Ann-Barbara Graff. Making sense in the margins: a field study of annotation. In *Proceedings of the 15th international conference on Theory and practice of digital libraries: research and advanced technology for digital libraries*, TPDL'11, pages 252–259, Berlin, Heidelberg, 2011. Springer-Verlag.
5. John Bradley. Pliny: a model for digital support of scholarship. *Journal of Digital Information*, 9(1), 2008.
6. Paolo Ciccarese, Marco Ocana, Leyla Jael Garcia Castro, Sudeshna Das, and Tim Clark. An open annotation ontology for science on web 3.0. *Journal of Biomedical Semantics*, 2(2), 2011.
7. Herbert Van de Sompel, Robert Sanderson, Michael L. Nelson, Lyudmila Balakireva, Harihar Shankar, and Scott Ainsworth. An http-based versioning mechanism for linked data. *CoRR*, abs/1003.3661, 2010.
8. Leigh Dodds and Ian Davis. *Linked Data Patterns*. Available at: `http://patterns.dataincubator.org/book/index.html`.
9. Lynda Hardman, Dick C. A. Bulterman, and Guido van Rossum. The amsterdam hypermedia model: adding time and context to the dexter model. *Commun. ACM*, 37(2):50–62, 1994.
10. Bernhard Haslhofer, Wolfgang Jochum, Ross King, Christian Sadilek, and Karin Schellner. The LEMO annotation framework: weaving multimedia annotations with the Web. *Int. J. on Digital Libraries*, 10(1):15–32, 2009.

11. Bernhard Haslhofer, Rainer Simon, Robert Sanderson, and Herbert van de Sompel. The Open Annotation Collaboration (OAC) model. In *Workshop on Multimedia on the Web 2011*, Graz, Austria, September 2011. IEEE.

12. Michael Hausenblas, Raphaël Troncy, Yves Raimond, and Tobia Bürger. Interlinking multimedia: How to apply linked data principles to multimedia fragments. In *WWW 2009 Workshop: Linked Data on the Web (LDOW2009)*, Madrid, Spain, 2009.

13. Tom Heath and Christian Bizer. *Linked Data: evolving the web into a global data space.* Synthesis Lectures on the Semantic Web: Theory and Technology, 1:1, 1-136. Morgan & Claypool, 1st edition edition, 2011. Available at: `http://linkeddatabook.com/editions/1.0/`.

14. Jane Hunter. Collaborative semantic tagging and annotation systems. *Annual Review of Information Science and Technology*, 43(1):1–84, 2009.

15. ISO/IEC. *Multimedia framework (MPEG-21) – Part 17: Fragment Identification of MPEG Resources.* International Organization for Standardization, Geneva, Switzerland, December 2006.

16. Ian Jacobs and Norman Walsh. Architecture of the World Wide Web, Volume One, December 2004.

17. José Kahan and Marja-Ritta Koivunen. Annotea: an open RDF infrastructure for shared web annotations. In *Proceedings of the 10th international conference on World Wide Web*, WWW '01, pages 623–632, New York, NY, USA, 2001. ACM.

18. Hak L. Kim, Simon Scerri, John G. Breslin, Stefan Decker, and Hong G. Kim. The state of the art in tag ontologies: a semantic model for tagging and folksonomies. In *DCMI '08: Proceedings of the 2008 International Conference on Dublin Core and Metadata Applications*, pages 128–137. Dublin Core Metadata Initiative, 2008.

19. Martin Klein, Jeb Ware, and Michael L. Nelson. Rediscovering missing web pages using link neighborhood lexical signatures. In *Proceeding of the 11th annual international ACM/IEEE joint conference on Digital libraries*, JCDL '11, pages 137–140, New York, NY, USA, 2011. ACM.

20. Johannes Koch and Carlos A Velasco. *Representing Content in RDF 1.0.* W3C, 2009. Available at: `http://www.w3.org/TR/Content-in-RDF10/`.

21. Marja-Riitta Koivunen. Semantic authoring by tagging with Annotea social bookmarks and topics. In *In The 5th International Semantic Web Conference (ISWC2006) - 1st Semantic Authoring and Annotation Workshop (SAAW2006)*, 2006.

22. Catherine C. Marshall. The future of annotation in a digital (paper) world. In Harum and Twidale, editors, *Successes and Failures of Digital Libraries*, pages 97–117. Urbana-Champaign: University of Illinois, 2000. Available at: `http://www.csdl.tamu.edu/~marshall/uiucpaper-complete.pdf`.

23. Catherine C. Marshall and A. J. Bernheim Brush. Exploring the relationship between personal and public annotations. In *Proceedings of the 2004 Joint ACM/IEEE conference on Digital Libraries*, pages 349–357, 2004.

24. Atsuyuki Morishima, Akiyoshi Nakamizo, Toshinari Iida, Shigeo Sugimoto, and Hiroyuki Kitagawa. Bringing your dead links back to life: a comprehensive approach and lessons learned. In *Proceedings of the 20th ACM conference on Hypertext and hypermedia*, HT '09, pages 15–24, New York, NY, USA, 2009. ACM.

25. Frank Nack and Adam T. Lindsay. Everything You Wanted to Know About MPEG-7: Part 2. *IEEE MultiMedia*, pages 64–73, October - December 1999.

26. Silvia Pfeiffer. Hyperlinking to time offsets: The temporal uri specification, 12 2007.

27. Thomas A. Phelps and Robert Wilensky. Robust hyperlinks cost just five words each. Technical Report UCB/CSD-00-1091, EECS Department, University of California, Berkeley, 2000.

28. Niko Popitsch and Bernhard Haslhofer. Dsnotify - a solution for event detection and link maintenance in dynamic datasets. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(3):266 – 283, 2011.

29. Asim Qayyum. Analysing markings made on e-documents. *Canadian Journal of Information & Library Sciences*, 32(1/2):35–53, 2008.

30. Carsten Saathoff and Ansgar Scherp. Unlocking the semantics of multimedia presentations in the web with the multimedia metadata ontology. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 831–840, New York, NY, USA, 2010. ACM.

31. Robert Sanderson, Benjamin Albritton, Rafael Schwemmer, and Herbert Van de Sompel. Sharedcanvas: a collaborative model for medieval manuscript layout dissemination. In *Proceeding of the 11th annual international ACM/IEEE joint conference on Digital libraries*, JCDL '11, pages 175–184, New York, NY, USA, 2011. ACM.

32. Robert Sanderson and Herbert Van de Sompel. Making web annotations persistent over time. In *Proceedings of the 10th annual joint conference on Digital libraries*, JCDL '10, pages 1–10, New York, NY, USA, 2010. ACM.

33. Ronald Schroeter, Jane Hunter, and Andrew Newman. Annotating relationships between multiple mixed-media digital objects by extending annotea. In Enrico Franconi, Michael Kifer, and Wolfgang May, editors, *ESWC*, volume 4519. Springer, 2007.

34. Jane Hunter and Ronald Schroeter. Co-Annotea: A system for tagging relationships between multiple mixed- media objects IEEE Multimedia, 15(3), pages 42–53, 2008.

35. Rainer Simon, Bernhard Haslhofer, Werner Robitza, and Elaheh Momeni. Semantically augmented annotations in digitized map collections. In *Proceeding of the 11th annual international ACM/IEEE joint conference on Digital libraries*, JCDL '11, pages 199–202, New York, NY, USA, 2011. ACM.

36. Herbert Van de Sompel, Michael L. Nelson, Robert Sanderson, Lyudmila L. Balakireva, Scott Ainsworth, and Harihar Shankar. Memento: Time travel for the web. *Arxiv preprint.*, arxiv:0911.1112, November 2009.

37. Davy Van Deursen, Raphaël Troncy, Erik Mannens, Silvia Pfeiffer, Yves Lafon, and Rik Van de Walle. Implementing the media fragments uri specification. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 1361–1364, New York, NY, USA, 2010. ACM.

38. Karin Verspoor, Judith Cohn, Cliff Joslyn, Sue Mniszewski, Andreas Rechtsteiner, Luis Rocha, and Tiago Simas. Protein annotation as term categorization in the gene ontology using word proximity networks. *BMC Bioinformatics*, 6(1), 2005.

39. W3C Media Fragments Working Group. *Media Fragments URI 1.0*. W3C, 2011. Available at: `http://www.w3.org/TR/media-frags/`.

40. W3C SVG Working Group. *Scalable Vector Graphics (SVG) — XML Graphics for the Web*. W3C, 2003. Available at: `http://www.w3.org/Graphics/SVG/`.