

International Journal of Semantic Computing
© World Scientific Publishing Company

Applying Linked Data in Multimedia Annotations

Yunjia Li

*School of Electronics and Computer Science
University of Southampton, UK
yl2@ecs.soton.ac.uk*

Mike Wald

*School of Electronics and Computer Science
University of Southampton, UK
mw@ecs.soton.ac.uk*

Gary Wills

*School of Electronics and Computer Science
University of Southampton, UK
gbw@ecs.soton.ac.uk*

Received 3rd June, 2011

Revised (1st November 2011)

Accepted (Day Month Year)

Current multimedia applications in Web 2.0 have generated large repositories for multimedia resources and annotations, so there is an urgent requirement to interlink annotations of these resources across different repositories to achieve better indexing and searching. To solve this problem, many researchers have been trying to apply semantic Web technologies to media fragments and annotations. Linked data has brought forward a promising way to expose, index and search media fragments and annotations which used to be isolated in different applications. This paper discusses in depth three key research problems when applying linked data principles in multimedia annotations: choosing URIs for media fragments, dereferencing media fragments and ontology alignment. An architecture is designed based on the possible solutions of the research problems. The key idea of the architecture is that it should act like an extra layer built on top of old applications when publishing linked data. A demo is built as an implementation of the architecture to show that media fragments can be published and linked to various datasets in the linked data cloud. In the future, some algorithms should be designed to make full use of the interlinked media fragments and annotations for indexing and searching.

Keywords: linked data; multimedia annotation; media fragment; semantic Web.

1. Introduction

The Web applications today have been enriched with various multimedia resources and annotations. In the broad sense, multimedia annotations include a resource's own metadata (e.g. creator, created date, copyright information of the multimedia

resource) and other resources which annotate this resource for content enrichment purposes (e.g. user generated comments, images which annotate a video). The success of multimedia storage and sharing applications, such as Flickr and YouTube, has proved that, instead of plain text resources, multimedia resources are being raised to the first class citizen of the Web. Social network applications like Facebook, which allow sharing multimedia resources from various repositories, further generate massive quantities of annotations stored independently from the original media repositories.

The term "media fragment" refers to the inside content of multimedia objects, such as a certain area within an image, or a video clip between 20 and 30 minutes within an one-hour video. Media fragments usually denote the temporal and spatial nature of multimedia resources. There is still a lack of semantic interlinking between media fragments and annotations, which leads to the insufficient index of inside content of multimedia resources. Thus most search results for multimedia resources in major search engines stay on the whole multimedia resource level. Some research has been devoted to address media fragments and semantic multimedia annotations, such as MPEG-7 [30] and Core Ontology of Multimedia (COMM) [2], but it is still difficult to achieve the interlinking of various multimedia resources on the Web due to the fact that these standards are usually format dependent and there is lack of mutual understandings among applications.

The Linked Data principles [4] offer guidelines of publishing linked data on the Web so that data can be better connected to each other and explored by machines. The publishing of Public Sector Information (PSI) data for the UK government [35] has shown how the open Web of data can help to connect massive data from various backgrounds. Linked data principles do not restrict the object they could be applied to on the Web. Therefore, this paper discusses the problems and solutions when publishing multimedia fragments and annotations using linked data principles.

Figure 1 briefly explains the research question. There are many raw multimedia data in varieties of repositories on the Web and there are also applications, which enable users to add annotations (tags, comments) to media resources they do not own. Some examples of these applications are Delicious, Synote [25, 27] and video post functions on Facebook^a. Usually, the annotations are saved in relational databases or in structured form like XML and can be accessed through APIs provided by different applications. On the semantic annotations level, all the annotations are represented as machine-readable data. Different from traditional data, the semantic multimedia annotations require an explicit identification for not only the parent media resources, but also the content inside them (media fragments). It has been stated in [39] that "enabling the addressing of media fragments ultimately creates a means to attach annotations to media fragments". Then the annotations are ready to be further published and be integrated with other datasets in the linked data cloud. This paper will focus on the research problems at semantic annotation level

^a<http://www.facebook.com/?sk=video>

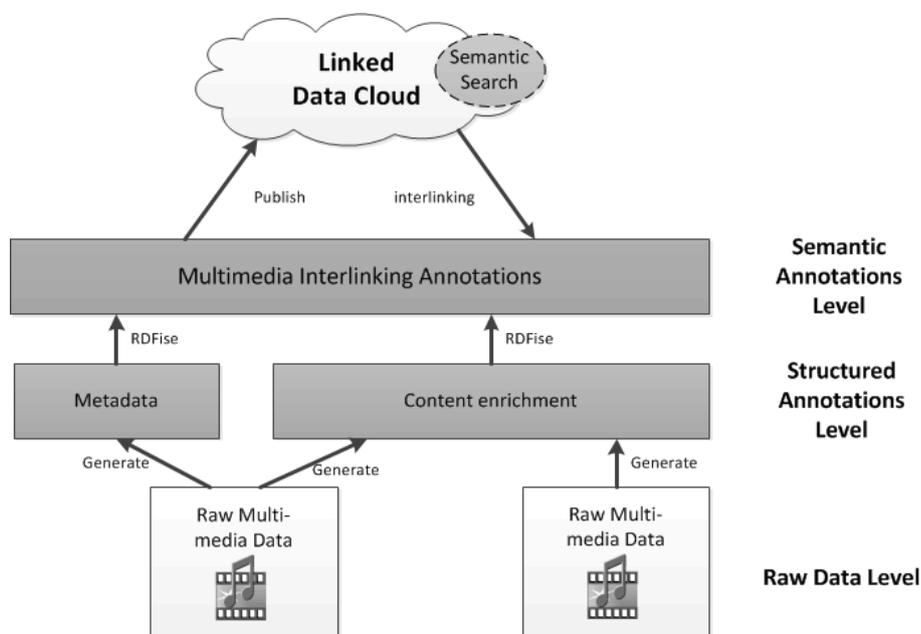


Fig. 1. The research question

and the mechanisms to publish, as well as consume, annotations as linked data.

This paper firstly reviews some related work in Section 2 and then summarises the research questions when applying linked data principles in multimedia annotations in Section 3. Section 4 introduces a architecture and demo to solve the research questions. The solutions are based on analysing the current conditions of multimedia applications on the Web. So the discussion will mainly focus on the temporal nature. Some future work to improve the publishing of multimedia annotations and media fragments searching will also be included in this section. Section 5 gives the conclusions.

2. Literature Review

This section briefly introduces the principles of linked data and the research about semantic multimedia annotations. Related applications which integrate linked data and multimedia annotations will also be discussed in this section.

2.1. *Linked Data*

The initiative of linked data describes a series of methods of publishing structured data using semantic Web technologies and other related standards, such as Resource Description Framework (RDF) [29] and SPARQL query language [32]. Linked data

enables people or machines to automatically discover more data from the data they already know. Generally, there are four rules which must be followed when publishing linked data on the Web [4]:1) Use URIs as names for things 2) Use HTTP URIs so that people can look up those names 3) When someone looks up a URI, provide useful information 4) Include links to other URIs, so that they can discover more things.

By September 2010, there have been more than 200 major datasets published with around 25 billion triples and 395 million RDF links according to ESW wiki^b. DBpedia [3] is one of the most famous datasets, which have been acting as a central hub of linked data. DBpedia extracts structured information from Wikipedia and makes them available as linked data for reuse and query. Other important datasets include, Linked GeoData, datasets harvested for RKBExplorer [14], data.gov.uk, etc.

The data.gov.uk is a paradigm of adopting linked data as recommended practice to improve transparency of local government data and the interaction between citizens and government in the UK as well as around the world [35]. The datasets already published in UK's Public Sector Information (PSI) include: government expenses, crime rate, parliament information, etc. Some of the government data, such as energy consuming, have been published within EnAKTing project^c, which is trying to address fundamental problems in "achieving an effective web of linked data". Most of the datasets in data.gov.uk use the geographic data as the key dataset to tie other datasets together and a single point of access is established for all public UK datasets. The parliament dataset in data.gov.uk is dumped from TheyWorkForYou.com^d, which is a Website publishing information about MPs, Lords and parliament debates.

Google combines the Rich Snippets [38] technologies with traditional documents search. The Rich Snippets feature currently is built on open standards or community-agreed approaches, such as RDFa [1], Microformats and recently published HTML5 Microdata [20]. All of these technologies use semantic mark-ups, which can be embedded into current Web pages. Google search results then can recognize and display the structured data and highlight that bit of information along with the result item. Google believes that Rich Snippets can improve the efficiency of metadata harvesting and interlinking, which has great business impact.

Since much effort has been devoted to publish and consume various linked datasets, some summaries have been made as guidelines for linked data developers. [19] has summarised the common steps that developers have to take from choosing URIs to the final testing and discovering linked data according to different situations. It has been pointed out that developers should not totally abandon "existing data management system and business applications", but add an "extra technical

^b<http://esw.w3.org/TaskForces/CommunityProjects/LinkingOpenData/DataSets>

^c<http://enakt.org>

^d<http://www.theyworkforyou.com>

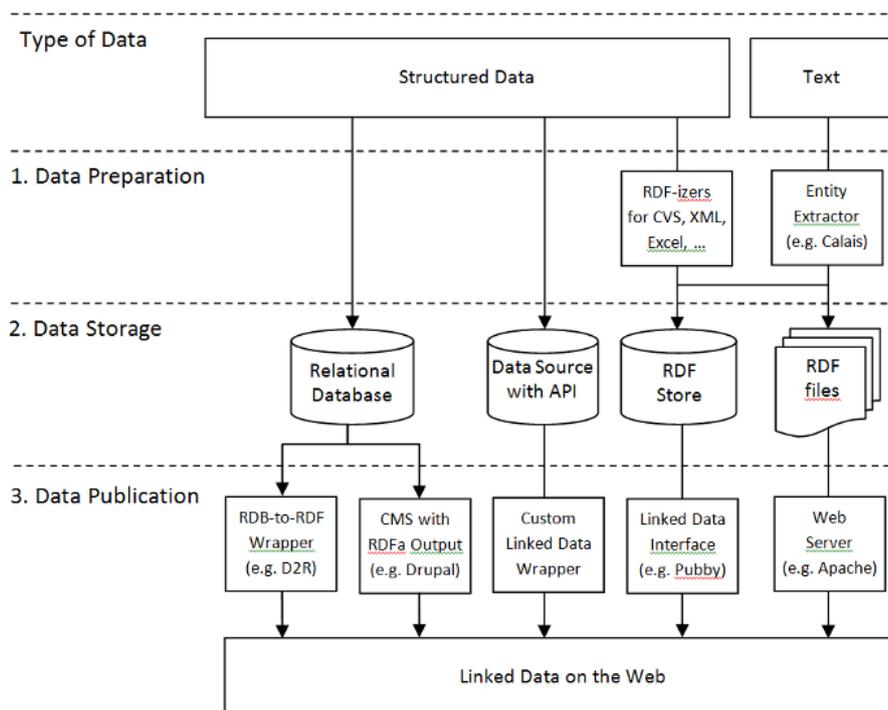


Fig. 2. Linked Data Publishing Patterns

layer of glue to connect these into the Web of Data". Figure 2 [19] shows the patterns of publishing linked data based on different original datatypes and storages.

Tim Berners Lee proposed a 5-star process [4] to encourage data owners, especially government, to share their data on the Web. From a developer's point of view, this process is trying to break down the task into achievable steps so as to lower the barrier for developers.

2.2. Semantic Multimedia Annotations and Media Fragment

There are many applications involving semantic multimedia annotations on the Web, but applying linked data principles to media fragments and annotations is still an area that is not fully explored. Stamou et al [37] explain the gap between multimedia annotation and the Semantic Web. In sub-symbolic level, the multimedia information is represented in various formats, which are usually binary and optimized for compression. Symbolic level offers a structural layer on top of the binary media stream. Information at this level is serialized in xml. The problem at this level is the semantics of the information encoded in the xml are only specified within each standard's framework. The third layer provides semantics for the sec-

ond layer. This layer defines mapping between the structured information source and the domain's formal knowledge.

Many standards try to expose the media fragments for annotations, but some of them cannot be applied under linked data principles as they are non-URI based mechanisms, such as MPEG-7 [30], Synchronized Multimedia Integration Language (SMIL)^e and SVG^f. In these standards, the descriptions of temporal and spatial dimensions are divided into several attributes, thus the media fragment is not represented by a single URI.

URI-based standards have also been proposed and there are generally three kinds of implementations to represent temporal, spatial and other dimensions of media fragments: URI query, URI fragment (hash namespaces) and URI slash (slash namespaces). MPEG-21 [10] specifies a normative URI fragment syntax to address fragment in MPEG compatible files. But the syntax of MPEG-21 is so complex that it is difficult to fully follow the standard. Core Ontology for Multimedia (COMM) aims to add formal semantics to MPEG-7 [2], but again, the ontology relies on the MPEG-7 description and is not compatible with other formats. Another attempt is to use "temporal URI" to specify time intervals in URI queries and fragments for time-based web resources [31]. The query parameter ("?" character) can generate a new fragment of resource on the basis of its parent resource. But this solution does not follow the "Cool URIs" [34], so it is difficult to be applied in linked data. The EnAKTing project develops reusable URIs to represent temporal entities [11]. They applied slash namespaces and developed the concept of Linked Timelines, which adopts the OWL time ontology^g to describe temporal entities and relationships between them. The proposed URIs for Linked Stream Data also applied slash namespaces to include real time and space information in URIs together or separately.

The W3C Media Fragment Working Group^h and Media Annotation Working Groupⁱ in Video in the Web Activity^j have collected a wide range of use cases of using media fragments [39] and proposed Media Fragment URI 1.0 draft (MFURI 1.0)[40], Ontology for Media Resource 1.0 draft (OMR 1.0)[28] and API for Media Resource 1.0 (APIMR 1.0)[24], which underpin the interlinking of media fragments and annotations in linked data era. OMR 1.0 also has APIs for user agent (UA) to access objects in media resources. MFURI 1.0 supports the addressing of image, audio and video along four dimensions: temporal, spatial, track and named section (such as chapter 1, section 3, etc). The information about each dimension is encoded in URIs using hash fragments following a certain format. The main motivation of MFURI 1.0 is that by parsing the URI fragments, a byte stream of media sub-part

^e<http://www.w3.org/TR/SMIL/>

^f<http://www.w3.org/TR/SVG/>

^g<http://www.w3.org/2006/time#>

^h<http://www.w3.org/2008/WebVideo/Fragments/>

ⁱ<http://www.w3.org/2008/WebVideo/Annotations/>

^j<http://www.w3.org/2008/WebVideo/>

can be returned through HTTP with the collaboration of UAs, proxy and original web server, instead of the whole media. Some issues regarding URI dereferencing and multimedia representations using MFURI 1.0 in linked data are also pointed out in [18].

Multimedia Metadata Ontology (M3O) [33] provides a framework to describe and annotate complex multimedia resources. It fills the gap between the structured metadata models, such as SMIL and EXIF^k, and semantic annotations. M3O can be integrated with various standards to provide semantic annotations with some further development. The key concept of M3O is separation of information objects and information realizations. Based on this idea, M3O comes up with a core set of ontology design patterns, including annotation, decomposition, collection and provenance patterns.

2.3. *Linked Data in Multimedia Annotation Applications*

Many multimedia authoring applications on the Web have implemented their URIs for media fragments and managed to relate resources from various datasets for discovering and reasoning. In [9], authors introduced two general ways of setting RDF links: manually or automatically. In the manual way, developers need to know which datasets they want to link to with the assistance of official CKAN datasets registry^l, or RDF crawler like Sindice. The automatic way usually needs an algorithm to map items in both datasets. Ideally, a one to one relationship can be generated if the item in one dataset can be found with no ambiguity in the other one. The BBC tries to integrate its music and programme data with MusicBrainz and DBpedia. As the data in BBC have connections with varieties of domains, they have to provide a content categorization system called CIS [22] to select the best match resource.

The methods for interlinking multimedia annotations can be divided into four categories: totally manual, collaborative editing, semi-automatic and totally automatic [17]. Manual methods mainly refers to user contributed linking [15] and game based interlinking, in which end users can generate high quality links with fun. Semantic Wiki, which contains multimedia objects and allows annotation of part of the multimedia objects, is the chief manner of collaborative interlinking. Metavid-Wiki extends the famous semantic MediaWiki^m and supports interlinking between temporal fragments. Semi-automatic method is quite similar to the applications which create tags with controlled vocabulary. Users will be given some suggestions and asked to accept, modify, reject or ignore the suggestions. Automatic interlinking can be realized by purely analysing the content of datasets. However, it has been suggested that after the automatic interlinking generation, a community needs to review and modify the links to improve the accuracy [17].

^k<http://www.exif.org/Exif2-2.pdf>

^l<http://ckan.net/group/lodcloud>

^mhttp://metavid.org/wiki/MetaVidWiki_Features_Overview

Annotation is a tool to handle the input annotations from users [23] developed by the Open University in the UK. The annotations are saved in RDF quad store with users' own privacy and provenance data. All the videos and annotations in the store are assigned with globally unique URIs so that they can be published in Linked Open Data cloud. SugarTube browser is another linked data application developed by the Open University to help learners navigating through resources using linked data. When doing a term search, SugarTube can invoke RESTful services provided by DBpedia, GeoNames and Open University Learning Resources RDF Repositories to get semantically relevant data about the searched terms [23].

Yoviso.com host large amount of recordings of academic lectures and conferences for users to search in a content-based manner. Some researchers augment Yovisto open academic video search platform by publishing the database containing video and annotations as linked data [42]. Yovisto uses Virtuoso server [13] to publish the videos and annotations in the database and MPEG-7, COMM to describe multimedia data.

Europeanaⁿ is a platform for users to share and annotate multimedia resources about culture and history all over Europe. Europeana integrated LEMO multimedia annotation framework [16], in which media fragments are published using MPEG-21 vocabulary. LEMO has to convert existing video files to MPEG compatible version and stream them from LEMO server. LEMO also derived a core annotation schema from Annotea Annotation Schema^o in order to link annotations to media fragments identifications.

NinSuna (Metadata-driven media adaptation and delivery) is a format-independent RDF Model-driven media delivery system, which implements MFURI 1.0 [12]. The system can "ingest" different formats of multimedia resource and save information about temporal, spatial and track dimensions to semantically represented data blocks. When delivering the media resource, one or more data blocks can be returned to UA through HTTP live streaming (HLS) according to the Range header in HTTP request as well as the adaptive context of the UA. As NinSuna model is partially implemented in OWL, the metadata and media fragments can be further linked to other resources.

3. Applying Linked Data Principles in Multimedia Annotations

Following the brief arguments in [26], this chapter will summarise the key issues and have a deep discussion on the basis of the four linked data principles.

The four linked data principles underlie all applications which want to publish their data. Different from the number and text-based data on the Web, multimedia data has a more complex structure. Multimedia data usually has multiple dimensions, such as temporal, spatial and tracks. These dimensions are referred to as the

ⁿ<http://www.europeana.eu>

^o<http://www.w3.org/2000/10/annotation-ns#>

term "media fragment". To follow the first and second rules of linked data principles, HTTP URIs needs to be used to identify media fragments and they should be applicable to all commonly used media formats on the Web. Each media fragment could be a single dimension or a combination of different dimensions. Some dimensions are not applicable to some resources. Images, for example, usually do not have a temporal dimension and audio resources do not have spatial a nature. Video content can have the most complex combination of dimensions compared with images and audio. As "media fragment" is not a concept for image or audio only, information in each dimension should be able to be encoded in a single HTTP URI in order to identify the media fragment in linked data.

Multimedia resources online are using varieties of coders and wrapped in different containers, which require proprietary software or decoders to display on the Web page. There is no agreement yet as to which format is the dominant. The third rule of linked data principles requires that the URIs should be dereferenceable. Applications have to be able to either directly return or redirect the request to both original representations (the original file of image, audio and video) and RDF representations. So there must be a mechanism to decide which representation should be returned in different conditions and how the representations can be returned. It is possible that the RDF representation can be embedded into the original representation.

The third and fourth rules of linked data principles actually ask data providers to choose appropriate vocabularies to describe the data and link it to other datasets. So it is necessary to find vocabularies to describe media fragments and their relationships with multimedia annotations. When different vocabularies are chosen, mappings or ontology alignment between different vocabularies will be needed in order to achieve better interlinking.

From the discussion above, the research problems regarding linked data and multimedia annotations can be summarises as followings:

- Q1** How to choose URIs to identify media resources, especially media fragments
- Q2** How to return appropriate representations when dereferencing the URIs of media fragments in different contexts
- Q3** How to tackle problems of ontology alignment related to media fragments and annotations

This section will discuss the three research questions and come up with the solutions to each question based on the analysis of current available standards and the current situation of multimedia applications on the Web. It has been pointed out in Section 1 that some multimedia applications on the Web have multimedia repositories but some of them do not. So the discussion will consider both situations. Another important guideline that must be followed is on the application level, developers should not totally abandon "existing data management system and business applications", but add an "extra technical layer of glue to connect these into the Web of Data" [19].

3.1. Choosing URIs for Media Fragment

Using a single URI to identify media fragments is not a new idea, but in the background of choosing URIs for linked data use, the mechanisms have to be reconsidered and evaluated for the usability in real applications. According to the URI definition in RFC3986 [6] and "Cool URIs for the Semantic Web" [34], there are three valid ways to add fragment information into URIs: URI query, slash namespace, URI hash fragment (hash namespace). URI with query is not widely used to identify a resource in RDF. The query will return a completely new resource from the server and thus lose the affiliation between parent and children resources [40]. Hash fragment and slash better fits the RESTful nature of linked data API, so these two methods are commonly used in RDF to identify resources.

In multimedia contexts, the slash namespace has several merits. Firstly, each annotation will be able to (but not necessarily) have an individual RDF representation (see Recipe 5 in [7]), which is flexible in that necessary HTTP 3XX redirection can be configured on the server side for each annotation. Secondly, server side can easily get the parameters indicating the temporal and spatial dimensions of the media fragment, and return the corresponding byte ranges. One big issue of using slash namespace is applications have to largely change server side programmes to handle media fragment information behind the slash. Otherwise, a "404 not available" response will be returned to the client. For example, to use the Timeline URI <http://time.psi.enakting.org/instant/2007-01-10T10:00:00Z/ttl>, proposed in [11], the application needs to add extra scripts to handle `/2007-01-10T10:00:00Z/ttl`.

The hash namespace method is adopted by MFURI 1.0 draft [40]. Compared with MPEG-21, different dimensions in multimedia are expressed in simple URI fragments and there is no strict limitation of the media format that is applicable to this draft. This solution perfectly fits the requirement that it can work as an "extra technical layer" and old applications can stay as they are because the hash fragments will not be passed to the server, so that server side do not need to add any programme to deal with this information. The UA, if it could not understand the semantics of the fragment, will just ignore the hash fragment and no further error will happen.

There are several weaknesses with this solution as has been pointed out in [18]. The first problem is that the semantics of URI fragments for most multimedia formats are un-defined^P. Therefore, UAs cannot understand the media fragment just like URI fragments in HTML document and the manipulation of media fragments (such as parsing and visualization media fragments) can only be processed with the assistance of UA plugins or client-side scripts (javascript). According to section 2.2.2, 3.2.1 and 3.2.2 of [21], new registrations must be done to IANA URI scheme so that the semantics of a fragment for each media type can be clarified for UAs.

^P<http://www.iana.org/assignments/uri-schemes.html>

Unfortunately, few media types have been registered yet. The second problem is that RDF representation of every media fragment will be retrieved in the same file as the RDF representation of the primary media resource. As the media annotation is frequently changed, media fragment URI `http://example.org/video.ogg#t=30,43` may not be valid 5 minutes later just because the user has deleted the corresponding fragment `#t=30,43`. It is against the best practice that "hash URIs should be preferred for rather small and stable sets of resource" and "the ideal case are RDF Schema vocabularies and OWL ontologies, where the terms are often used together, and the number of terms is unlikely to grow out of control in the future" [34].

Considering the downside of hash namespaces, many standards and applications also bring in URI queries as a complementation of hash URIs. MFURI 1.0 specifies the resolving of URI because "URI queries have to be used since they result in a server interaction and can deliver a transcoded resource" [40]. It is the same case for Temporal URI [31]. YouTube also allows users to attach both query and hash to indicate the start playing time.

From linked data point of view, solutions of choosing URIs are acceptable provided that the media fragment can be universally identified by the URI. However, the encoding of URIs will affect the way it could be dereferenced. URI query and slash namespaces would be more convenience for servers to get fragment information and return a suitable media fragment as well as RDF representation. But applications which do not own the media repository could not adopt this method unless the repository server could dereference query and slash URIs. In this condition, hash namespaces are more appropriate. Since media fragment is universally identified, it is the user agent's (or the application developers') responsibility to present the media fragment in a sensible way. Usually, applications have to serve both original and RDF representations of the resource. So the URIs for media fragments will decide if it is easy for the server to return and for the client to display a suitable media fragment representation, which is a big concern for MFURI 1.0 and applications like NinSuna. More discussion about this problem can be found in Section 3.2.

According to the second rule, HTTP should be chosen as the URI schema, but MFURI 1.0 also discusses the possibility of applying RTSP in linked data^q. RTSP has merits when delivering streamed media and has been widely used on the Web, such as BBC iPlayer^r. RTSP provides better media control when delivering the original representation of multimedia resources, so sending a bit stream is relatively easier compared with mechanisms provided by MFURI 1.0. There is also the possibility that the RDF representation can be included in the response of "DESCRIBE" activity. However, the dereferencing of RTSP URIs is quite different from HTTP URIs. In addition, proprietary servers need to be used to stream the audio/video.

^q<http://www.w3.org/TR/media-frags/#rtsp-media-fragment-processing>

^r<http://www.bbc.co.uk/iplayer/>

3.2. *Dereferencing URIs for Media Fragment*

According to the Web architecture [21], a representation is a stream of bytes in a particular format, such as HTML, PNG, RDF/XML, etc. A single resource can have different representations. This process can be done through content negotiation, where `Accept: application/rdf+xml` or `Accept:text/html` can distinguish if the response is RDF or HTML document.

The original proposed ways to dereference ("process" is the word they use in the draft specification) MFURI 1.0 is to map the timeline with the byte stream with the help of "smart enough" UAs and servers, so that only the byte ranges corresponding to the media fragment will be returned [41]. However, most of the media formats cannot fully satisfy conditions, so both client and server sides are inevitably to be re-developed. NinSuna implements MFURI 1.0, but all the media resources need to be ingested and broken down into units which can be mapped to byte ranges [12]. So even though this method of dereferencing brings flexibility to fulfil some requirements [39], it is not realistic to convert all multimedia resources online to a suitable format. More importantly, what linked data principles care about is the dereference of the semantic representation, such as RDF. The principles DO NOT require returning the media fragment of the primary media resource, so this problem can be left for further discussions.

The traditional methods of dereferencing URIs introduced in [7] are still applicable to URIs of media fragments. Content negotiation is used to distinguish the request for original resource from the RDF representation and 303 redirection is used to locate the correct RDF file. But the choosing of URIs will affect the dereferencing process as it decides whether or not the fragment information can be passed to the server.

Another problem is the "fuzzy querying" for media fragments. For example, when `#t=20s,40s` is requested, the server can choose to return RDF representations for the exact fragment, or within the interval such as `#t=25s,35s`, or near the fragment interval such as `#t=18s,38s`. This is due to the reason that when users connect annotations to media fragments, their choosing of fragment might be slightly different, even though they are trying to annotation the same fragment. It is the same case for spatial fragments in images, as the exact matching data may not be available. One lazy solution of this problem is whatever fragment is requested, the server just returns the RDF graph for the whole media resource, which contains the reference to the media fragment. The ideal result is returning a smaller RDF graph containing only triples related to the requested media fragments. One possible way is sending the fragment information via the Range header in HTTP request [18]. The server then can compose a RDF file only about that media fragment as a response. The problem about this method is that currently it cannot be done automatically and some client-side programme has to insert the Range header. Another way to tackle this problem is similar to "Redirecting to a SPARQL endpoint" recipe [7] (Figure 3). Client side has to encode the media fragment URI into the



Fig. 3. Dereferencing from SPARQL endpoint

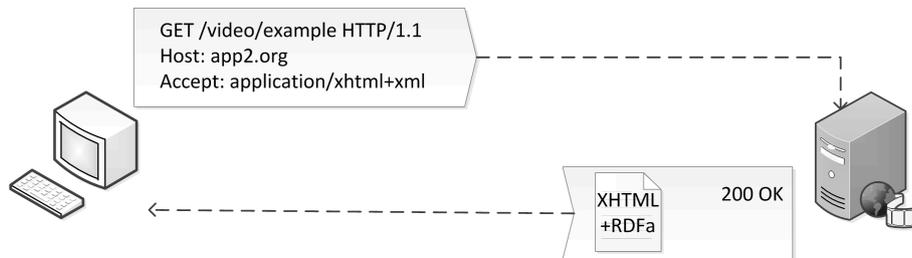


Fig. 4. Embed RDFa in XHTML

query string and server side only needs to expose a SPARQL endpoint to handle the HTTP GET query. In this solution, the server-side still has to deal with the fuzzy querying problem and client-side script is necessary to form the query string.

It is also possible to embed media fragment RDF representation into an XHTML page in form of RDFa (Figure 4). In some applications (such as YouTube) the visualization of media resources are embedded within a Web page. This offers opportunities to embed RDF about media fragments and annotations into the original displaying multimedia resources and their annotations. Client-side in this method does not need to change anything unless the visualization of media fragments is required, while server-side has to add RDFa into the web pages.

3.3. Ontology Alignment

The fourth rule of linked data indicates that links to other URIs should be included. When multimedia annotations on the Web are identified by URI, the next step will be interlinking related resources together for discovery and reasoning. Actually, ontology alignment is not a problem only for topics of multimedia annotations, but within the multimedia domain, the problem becomes more complex.

Vocabularies need to be chosen to describe media fragments and annotations.

What should be included in the RDF representation depends on the requirement of each application. Generally speaking, to publish media fragments and annotations, two kinds of vocabularies are necessary: the vocabularies to describe media resources (metadata of media resources including media fragments) and domain specific vocabularies to describe other annotations.

There are many vocabularies existing to describe multimedia resources, such as MPEG-7, EXIF. It is not realistic to select a dominant vocabulary. In order to achieve the mutual understanding among these vocabularies, some mappings are necessary to be developed. OMR 1.0 defines the core vocabulary for multimedia resources and provides ontology (or vocabulary) mappings to other existing formats. Applications, which serve metadata in MPEG-7 for example, do not need to re-write their metadata format, but use a mapping to publish MPEG-7 metadata in OMR 1.0 format. Much alignment work has also been done in M3O, which offers rich semantic description of media resources.

It is a domain specific problem to choose an appropriate vocabulary to describe the relationships of media fragments and annotations. For example, if an application makes annotations on UK Parliament debates, except for the vocabularies used to describe media fragments, domain vocabularies about debate events, parliament decision making processes and MPs' profiles may also be used to semantically describe the debates. A demonstration will be shown later in this paper based on this scenario.

Many automatic and manual annotation methods can be applied to interlink annotations to other datasets. For example, if media fragments are annotated by text-based annotations (transcript or comments), it is possible to use DBpedia Spotlight^s and Open Calais^t to semantically annotate these text-based annotations in an automatic manner. Then, through the annotation relationship, the media fragments are related to these automatic annotations. There are many other methods of extracting metadata and other data representing content in the media such as mountain, trees, people, sea, etc. In addition, some fixed relationships on the temporal or spatial dimensions can be explored automatically. However, human interference is still needed to point out which media fragments are valued and should be interlinked to other datasets.

3.4. Summary of the Discussion

The discussion above lists the possible solutions for each research problem. There are many standards and best practices that developers can follow, but it is obvious that none of them is perfect for all situations. The best combination of solutions for the three problems depends on several factors, such as the format of multimedia resources (images, audio or video), what media fragments should be published

^s<http://dbpedia.org/spotlight>

^t<http://www.opencalais.com/>

(temporal or spatial information), the extent to which the legacy applications could be changed, etc. For example, an application hosting Flash videos may find that MFURI 1.0 and OMR 1.0 are sufficient to identify media fragments and describe the resources, even though the dereferencing of media fragments needs extra work. An application, which provides image services only, may think SVG is good enough for it.

4. A Demonstration of Solutions

This section will show a demonstration of publishing media fragments and annotations as linked data using some of the technologies or solutions mentioned in Section 3. The demo is based on the UK Parliament Debate. Transcripts about the debate have been synchronised with the debate video through media fragments. The media fragments and annotations are further interlinked to other datasets, such as data.gov.uk and DBpedia. It is anticipated that users, who are interested in some topics in the debate can easily, find the original words MPs and Lords say, thus resulting in a improved transparency of the government. This demo is trying to prove that by answering the three research questions in Section 3 current multimedia applications can be extended to publish data into and consume data from the linked data cloud.

4.1. Scenarios

The following scenarios illustrate some of the functions of the demo:

Scenario 1: Michael has a Website that collects recordings of the sittings of his government's parliament. These recordings tend to be very long - generally on the order of 7 hours in duration. Instead of splitting up the recordings into short files by manual inspection of the change of topics or some other segmentation approach, he prefers to provide many handles to a unique video resource. As he publishes the files, however, he provides pagination on the videos such that people can watch them 20 min at a time (this scenario is from [39]).

Scenario 2: Bob does a keyword search on House of Common's debate about "police reform". This keyword is found in several debate videos. All the debate videos are very long and cover many other topics apart from police reform. Thanks to the media fragments and transcript annotation attached to them, Bob is able to navigate to the exact offset of the video rather than searching the full recording to find the relevant clips.

Scenario 3: Alice is watching a debate in the House of Common. When the video is playing, the speakers' information, which is fetched from the Parliament datasets in data.gov.uk, are displayed along with the transcript. The topic of this debate is about the police's aim in UK, which is coined as "to reduce the crime". She knows that the speaker has made the same point in Association of Chief Police Officers (ACPO) conference 2010. So she finds the ACPO 2010 speech on YouTube and annotates the media fragment with this YouTube video. Alice also notices that

one speaker mentioned the budget cuts and reducing bureaucracy for UK police. She knows that there is a document online explaining the detail of budget cuts. She then puts the URI of that document as the annotation of this speech.

Scenario 4: Tim is not sure what is the Association of Chief Police Officers. But with the help of DBpedia spotlight, the transcript of the speeches are automatically annotated or linked to the explanation of that term in DBpedia, so that Tim can click on the link and check that term in DBpedia.

4.2. *Architecture of the Demo*

Figure 5 shows the general architecture of the demo. This demo is built based on the work of Synote [25, 27], which is a synchronised annotation system for continuous multimedia resources (audio and video). Users of Synote could generate transcript and make notes, which are synchronised with audio and video resources. Synote does not host any audio or video resources, but refers to them through the resource URL. User generated notes and the synchronisation points will be stored in local Synote database.

The basic idea of this architecture design is that the publishing and consuming of linked data should be an extra layer built on the original application. The database, business logic (BL) and user interface in Figure 5 can reflex the basic architecture of Synote, as well as most Web applications serving multimedia resources and annotations. The publishing of linked data can happen in three levels: data storage level, business logic level and user interface level. Data Publisher and Data Integrator will not affect the original functions of legacy applications. Necessary changes will be made to the three levels in order to add linked data related functions, but the original application will not be totally re-designed or abandoned. This architecture is inspired from the concept of linked-data-driven applications [17].

In Data Publisher, RDB-to-RDF wrapper will generate RDF from database. BL-to-RDF can convert the data in BL to RDF, which is similar to the function of Linked Data Wrapper in Figure 2. Then the RDF data can be further published through SPARQL endpoint, dereferenceable RESTful API or RDFa in web pages.

As Figure 2 demonstrates, there are several publishing patterns for Data Publisher. One is writing a database dump or other dump programme to extract information from other resources as RDF triples. The triples can be served from a file system or saved in a triple store, which is accessible via SPARQL endpoint. This way of publishing data is suitable for applications which do not modify old data frequently but only add new data to datasets. The dumping programme can be run every week or month to keep the dataset update. DBpedia and some data.gov.uk data follow this way. The other way is publishing database dynamically to RDF using RDB-to-RDF server, such as D2R server [8] and Open Virtuoso server [13]. Usually, a mapping between database schema and RDF vocabulary or ontology has to be provided for the dynamic publishing. The change of database will immediately be reflected on the published datasets. D2R and Virtuoso server also provides

input for business logic. The data integration can happen on both client side and server side. The client side can use javascript to access datasets or sent query to SPARQL endpoints. This process is similar to the mashups in Web 2.0, but the difference is media fragments are involved in this process as part of the linked data cloud. It is important to emphasise that the created media fragments and annotations will not directly go to the RDF store, which means the RDF store only publishes what the database has according to the RDB-to-RDF mapping. This design is considering the fact that moving the whole database into a triple store involves massive re-development for business logic.

Another important function for Data Publisher is to deal with the "fuzzy query" when a URI of media fragment is to be dereferenced. Data publisher can refine the request query to return a better result if necessary. For applications, which do not host the multimedia resources themselves, sometimes it is also necessary to further dereference more data about that resource from the host application.

The user interface in this architecture can also embed media fragments and annotations as RDFa (as well as microformat and microdata) as introduced in Figure 4. If the user interface connects with the server via RESTful APIs or Web services, other applications could use the business logic but provide a different user interface for media fragments.

It is not necessary to implement every function block in Data Publisher and Integrator as long as it is sufficient to satisfy the publishing and consuming requirements of the specific application. In this demo, only functions shown in a grey background in Figure 5 are implemented.

4.3. *Implementation*

On the server side, the implementation directly uses Synote's database. The server-side programme in Data Integrator fetch the transcript text and the speaker's information from the Theyworkforyou API and save them in the Synote database. Theyworkforyou API does not provide the synchronisation information for each text block, so the author synchronised the speeches with the debate video manually and save the synchronisation data in the database. When the demo starts, BL programme reads the transcript text, speaker's information and the synchronisation points from the database. Then The BL-to-RDF function constructs RDF data from BL and embed it as RDFa in user interface. In this demo, the sample debate resource is about the bill presentation of police reform on 13th Dec, 2010. The debate video is hosted remotely in another application and the demo only refers to it through URL.

MFURI 1.0 and OMR 1.0 (namespace "maont") are chosen to represent media fragments and describe the debate video. Linking Open Descriptions of Events (LODE) [36] and DOLCE+DnS Ultralite^u (DUL) are used to generally model the

^u<http://www.loa-cnr.it/ontologies/DUL.owl>

The screenshot displays a web interface for a video player. On the left, there is a profile for **Keith Vaz**, Labour, from Leicester East. Below his name is a URL: <http://reference.data.gov.uk/id/mp/leicester-east/keith-vaz>. The main text of the video is: "Will the right hon. Lady join me in commending the work of Jan Berry, who was appointed by the previous Government but completed her report under the present Government, and her recommendations to reduce police bureaucracy? Will the right hon. Lady give the House an undertaking that that work will continue, and that Jan Berry or someone like her will continue to monitor the reduction in the bureaucracy that is hampering the police in doing their job?". Below the text is a "play this fragment" button. To the right is a video player showing Keith Vaz speaking, with a progress bar at 12596.513s. Below the video player is a "Speaker Information from data.gov.uk" section containing a table of properties and values for Keith Vaz.

| predicate | object |
|---------------|------------------------------------------------------|
| type | Person |
| description | Keith Vaz represents the constituency Leicester East |
| partyMemberOf | Labour |
| uriSet | Dataset of Ministers in the House of Commons |
| uriSet | Dataset of MPs |
| seeAlso | 1118281 |
| seeAlso | leicester_east |
| name | Keith Vaz |
| hasMembership | seat |
| hasMembership | 6 |
| holdsSeat | Leicester East |
| lastName | Vaz |

Fig. 6. Screenshot of the demo

debate events, such as who is the involvedAgent (involvedAgent property in LODÉ) of a speech and a speech is illustrated ("illustrate" property in LODÉ) by some media fragment. User generated annotations (scenario 3 in Section 4.1) are modelled by Open Annotation Collaboration (OAC) ontology^v. Server-side programmes are developed to construct the URIs for media fragments and related events out of the database. The server also provides RESTful APIs to serialize debate resource and annotations about this resource into XML format, which contains the URI for debate video, debate video's media fragments, speeches and the speakers. On the client side, the XML response is parsed by javascript (this demo uses jQuery libraries^w) and all the information about the debate video, speakers and speeches

^v<http://www.openannotation.org/ns/>

^w<http://jquery.com/>

are displayed in a synchronised manner (Figure 6). URIs for different resources are related to each other via media fragments and the vocabularies introduced earlier.

The visualization of media fragments are represented by the behaviour that the clicking on the speech will invoke the embedded player (VLC player ^x in this demo) on the Web page to play from the start time the media fragment denotes. Actually, each speech annotates a media fragment and the the client-side javascript programme directly calls the player's APIs to change the current playing time. This solution ignores the problems of caching and delivering media fragments from the server, but still makes sure that media fragment are exposed through the annotation of speech text.

In order to show that media fragments can be linked to other datasets in linked data cloud, the demo implements some javascript mashups. One is to retrieve speaker's information from the UK Parliament datasets in data.gov.uk. The demo firstly gets the speaker's full name when generating the recording for the whole debate session. Then a javascript programme is used to send the SPARQL query to find the speaker's information by his or her name. As the bottom-right table in Figure 6 shows, through lode:involvedAgent (the speaker is the lode:involvedAgent of a transcript) and lode:illustrate (a media fragment lode:illustrate the transcript), media fragment URIs are linked to the speaker and further linked to the party and region he or she represents. Another mashup is using DBpedia spotlight to automatically annotate the speech text (part 1 in Figure 7). People who do not know what is the Association of Chief Police Officers will be easily to follow the link and to find the definition in DBpedia. Again, through this annotation, the media fragment is linked to the URI representing ACPO in DBpedia. The problem of using DBpedia spotlight is it often generate meaningless or worthless annotations (such as the "key" and "police" links in the figure).

This demo also shows how media fragment URIs could link to other resources on the Web by user generated annotations. Users can choose which transcript text he or she wants to annotate (the subject in part 2 and 3 of Figure 7) and then provide the URI as the object of the annotation. The annotation in part 2 of Figure 7 illustrates that the user is willing to link the speech about "Reducing Bureaucracy in Policing" to a related pdf document online. The annotation in part 3 links the Theresa May's speech to another media resource in YouTube, because she repeated her point in that video. In this way, media fragments are firstly connected to the transcript text, and then further linked to remote resources via user generated annotations. The RDF graph about media fragments, speeches, speakers and annotations are embedded into the web page using javascript as RDFa. Figure 8 is a segment of the RDF file dumped from the RDFa in web page. The speaker Theresa May has a URI from data.gov.uk and she is related to a speech in this debate session by lode:involvedAgent predicate. Her speech is identified by a URI from the UK's office

^x<http://www.videolan.org/vlc/>

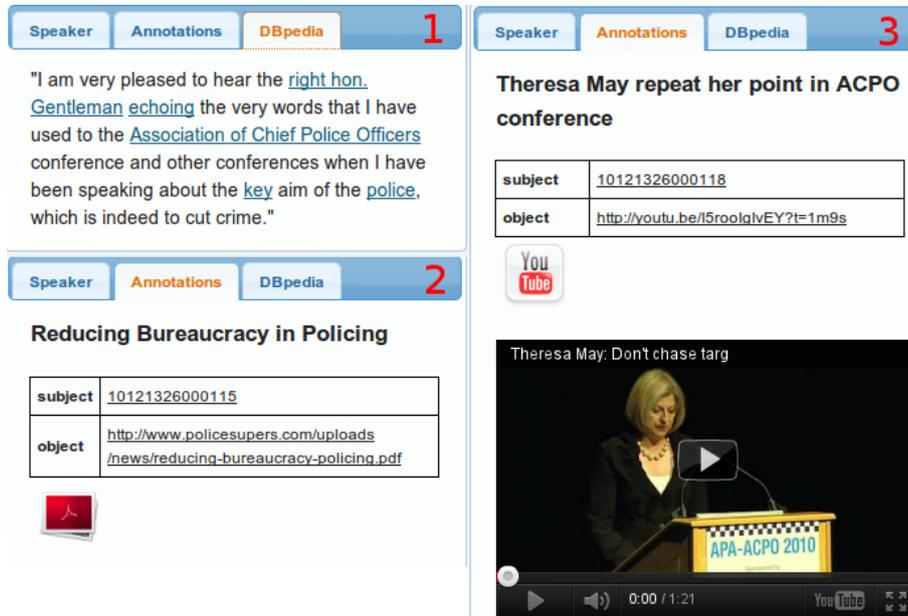


Fig. 7. User generated annotations

```
<rdf:RDF>
...
<rdf:Description rdf:about="http://reference.data.gov.uk/id/mp/maidenhead/theresa-may">
  <foaf:name>Theresa May</foaf:name>
</rdf:Description>
...
<lode:Event rdf:about="http://www.publications.parliament.uk/pa/cm201011/cmhansrd/cm101213/debtext/101213-0002.htm#10121326000118">
  <lode:involvedAgent rdf:resource="http://reference.data.gov.uk/id/mp/maidenhead/theresa-may"/>
</lode:Event>
<maont:MediaFragment rdf:about="http://www.twofourdigital.net/UKParliament/Archive/0000014318.wmv.aspx?t=12658,12677">
  <lode:illustrate rdf:resource="http://www.publications.parliament.uk/pa/cm201011/cmhansrd/cm101213/debtext/101213-0002.htm#10121326000118"/>
  <maont:isMediaFragmentOf rdf:resource="http://www.twofourdigital.net/UKParliament/Archive/0000014318.wmv.aspx"/>
...
<oac:Annotation rdf:about="http://www.synote.org/oore/annotation/33">
  <oac:hasBody rdf:resource="http://www.publications.parliament.uk/pa/cm201011/cmhansrd/cm101213/debtext/101213-0002.htm#10121326000118"/>
  <oac:hasTarget rdf:resource="http://youtu.be/I5rooIqIvEY?t=1m9s"/>
</oac:Annotation>
</maont:MediaFragment>
</rdf:RDF>
```

Fig. 8. RDF file dumped from the demo page

Parliament website^y. The speech URI is related to the media fragment URI, which represents the video clip containing this speech, via lode:illustrate predicate. Finally, a user generated annotation (<http://www.synote.org/core/annotation/33>) linked to this speech URI to a YouTube video clip through oac:hasBody and oac:hasTarget predicates. The screencast of this demo can be found on YouTube^z.

^y<http://www.publications.parliament.uk>

^z<http://www.youtube.com/watch?v=KkWI-DHLD.M>

Even though this demo focuses on temporal aspect of the video content, the rational behind the demo can be easily applied to image resources. The chosen URIs (MFURI 1.0) are also capable of identifying spatial and track fragments, so the same format of URIs can be applied to images. The dereferencing and ontology alignment process will also be similar to video resources except that temporal information does not need to be handled by the server. On the user interface level, displaying media fragments for images will be different from the methods used for video resources. Server-side can choose to return only that fragment of image or the client-side can highlight the region (e.g. with a rectangle).

4.4. Future work

The demo gives basic ideas of implementing the solutions proposed for the three research questions when applying linked data principles in multimedia annotations. There is some future work to be done so that the datasets about media fragments and annotations can be fully published into the linked data cloud and, more importantly, benefit the indexing and searching of media fragments.

Firstly, the relational database has not been published by RDB-to-RDF wrapper yet. The media fragment URIs in the demo are still not dereferenceable (because the host server of debate video cannot provide RDF representations) and there is no SPARQL endpoint for other application to access the triple store. When the database is published, it is better that each media fragment has an identification with in the application server's domain (<http://synote.org/video/1#t=20,30> for example), but at the same time keeps a `rdfs:seeAlso`, `owl:sameAs` or other relationships to the original host server. When dereferencing this media fragment URI, Data Publisher can return the RDF graph with in the local server, or further dereference it from the original repository if the current application does not save the resources locally. Data publisher also has to use some algorithm to deal with the "fuzzy query" problem mentioned in section 3.2.

Secondly, developers have to think about how to display the dereferencing information of media fragments and annotations in an appropriate manner on the user interface level. This demo allows users to link media fragments URIs to any resources on the Web. The servers of some resources will not provide the RDF representation for the dereferencing. From a linked data principles' point of view, whether or not the dereferencing results are displayed on the Web page does not affect the fact that the link between URIs has been setup and published online. However, it will be still necessary to develop a user interface to guide users to create annotations for media fragments, even though the annotation may point to some resources which are not dereferenceable. It is also helpful to create contextual information, which helps Data Integrator to dereference and display the annotation, when the annotation is saved into database. With the help of this information, Data Integrator can have some default settings to dereference and display the annotation. For example, inserting "``" tag into web page if the annotation is an image.

Thirdly and most importantly, the goal of publishing multimedia annotations is to improve the indexing of media fragments and finally provide more efficient search. The demo has shown that media fragments can be published and linked to other datasets. However, it is still unclear about how to use these interlinkings to benefit search and what search user interface should be provided for end users. In the semantic search, media fragments will be the first class entities, which can act as join points for interesting, contextual and other related entities on the Web. Some algorithms should be developed and evaluation will be needed to prove (if the hypothesis is true) that the searching results using linked data have an advantage compared with traditional multimedia search results, such as Google and YouTube.

One example of media fragments' contributions to semantic search would be to use the annotation properties to refine the search. Users can specify the predicate (such as "speechOnTopicOf") and the keywords (such as "police reform") in the search. Then the search engine can traverse the RDF graph, which includes media fragment URIs and user generated annotations, and find out the URIs related to "speechOnTopicOf" predicate. A text-based matching can be carried out at the same time with the traversing to reveal which media fragments are related to "speechOnTopicOf" predicate and also annotated by the text containing "police reform" keywords. The media fragment entity then could be returned and users would be able to further explore other properties about the media fragment.

5. Conclusions

Current multimedia applications in Web 2.0 have generated large repositories for multimedia resources and annotations. But each application is "a silo, walled off from the others" [5]. A lot of research has been trying to apply semantic Web technologies to media fragments and annotations. Linked data has provided a way forward to expose, index and search media fragments and annotations which used to be isolated in different applications. So this paper identifies three research questions when publishing multimedia annotations into as linked data. The three research questions are closely related to each other and there is no perfect solution to fit into all occasions. The discussion in Section 3 would be helpful for developers to compare different solutions based on what the applications have got and how they want to publish data.

This paper also proposed a architecture to publish multimedia annotations, which is an extra layer built on top of old applications, thus minimising the workload to publish traditional data into the linked data cloud. Old applications can still work well without this layer and end users will not notice the change unless a user interface related to linked data is added. The key components in the architecture are Data Publisher and Data Integrator, which connect the old application to linked data. Both components can involve (but not necessary) client-side and server-side programmes to implement the publishing and consuming functions. The demo uses various open standards and datasets to accomplish the interlinking of

media fragments to other datasets. As the demo shows, technologies and solutions mentioned in Section 3 can be applied in the architecture according to individual application's own requirements.

Some future work needs to be done. The demo mainly embeds RDFa on the Web pages, but media fragments could also be published on the database and business logic level. Since media fragment URIs have been created and linked to annotations, some methods (such as SPARQL endpoint) need to be provided to enable other applications to dereference the URIs. An appropriate design of user interface is necessary in order to highlight the media fragments and allow users to make annotations against them. The most important future work is to design an algorithm to make full use of the interlinked media fragments and annotations for indexing and searching. The big vision of this work is to encourage multimedia authoring and hosting applications to publish their data on the Web. All the media fragments and annotations in different repositories should include links to each other and provide RDF representation dereferencing services. The exposure of media fragments is the key to achieve better index for multimedia resources, so it is a good practise to attach annotations to media fragments instead of the whole multimedia resource.

References

- [1] Adida, B., Birbeck, M.: RDFa Primer (2008). URL <http://www.w3.org/TR/xhtml-rdfa-primer/>
- [2] Arndt, R., Troncy, R., Staab, S., Hardman, L., Vacura, M.: COMM: Designing a Well-Founded Multimedia Ontology for the Web. In: International Semantic Web Conference, *Lecture Notes in Computer Science*, vol. 4825, pp. 30–43. Springer Berlin Heidelberg (2007). DOI 10.1007/978-3-540-76298-0_3
- [3] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: A Nucleus for a Web of Open Data. In: K. Aberer, K.S. Choi, N. Noy, D. Allemang, K.I. Lee, L. Nixon, J. Golbeck, P. Mika, D. Maynard, R. Mizoguchi, G. Schreiber, P. Cudré-Mauroux (eds.) *The Semantic Web, LNCS*, vol. 4825, pp. 722–735. Springer (2007)
- [4] Berners-Lee, T.: Linked Data (2006). DOI 10.4018/jswis.2009081901. URL <http://www.w3.org/DesignIssues/LinkedData.html>
- [5] Berners-Lee, T.: Long Live the Web: A Call for Continued Open Standards and Neutrality. *Scientific American* **303**(6), 1–8 (2010). URL <http://www.scientificamerican.com/article.cfm?id=long-live-the-web>
- [6] Berners-Lee, T., Fielding, R.T., Masinter, L.: RFC 3986 - Uniform Resource Identifier (URI): Generic Syntax (2005). URL <http://tools.ietf.org/html/rfc3986>
- [7] Berrueta, D., Phipps, J.: Best Practice Recipes for Publishing RDF Vocabularies (2008). URL <http://www.w3.org/TR/swbp-vocab-pub/>
- [8] Bizer, C., Cyganiak, R.: D2R Server: SPARQL Endpoint for Relational Databases (2006)
- [9] Bizer, C., Cyganiak, R., Heath, T.: How to Publish Linked Data on the Web (2007). URL <http://www4.wiwi.fu-berlin.de/bizer/pub/LinkedDataTutorial/>
- [10] Bormans, J., Hill, K.: MPEG-21 Overview v.5 (2002). URL <http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm>

- [11] Correndo, G., Salvadores, M., Millard, I., Shadbolt, N.: Linked Timelines: Temporal Representation and Management in Linked Data. 1st International Workshop on Consuming Linked Data COLDC 2010 Shanghai (2010)
- [12] Deursen, D.V., Lancker, W.V., Neve, W.D., Paridaens, T., Mannens, E., Walle, R.V.D.: NinSuna: a fully integrated platform for format-independent multimedia content adaptation and delivery using Semantic Web technologies. *Multimedia Tools and Applications* **46**(2), 371–398 (2009). DOI 10.1007/s11042-009-0354-0
- [13] Erling, O., Mikhailov, I.: RDF Support in the Virtuoso DBMS. *Proceedings of the 1st Conference on Social Semantic Web CSSW 221* (ISBN 978-3-88579-207-9), 59–68 (2007). DOI 10.1007/978-3-642-02184-8_2
- [14] Glaser, H., Millard, I.C., Jaffri, A.: RKBExplorer . com : A Knowledge Driven Infrastructure for Linked Data Providers. *Proceedings of the 5th European Semantic Web Conference ESWC 2008* pp. 797–801 (2008)
- [15] Halb, W., Raimond, Y., Hausenblas, M.: Building Linked Data For Both Humans and Machines. *WWW 2008 Workshop Linked Data* (2007)
- [16] Haslhofer, B., Jochum, W., King, R., Sadilek, C., Schellner, K.: The LEMO annotation framework: weaving multimedia annotations with the web. *International Journal on Digital Libraries* **10**(1), 15–32 (2009). DOI 10.1007/s00799-009-0050-8
- [17] Hausenblas, M.: Exploiting Linked Data to Build Web Applications. *IEEE Internet Computing* **13**(4), 68–73 (2009). DOI 10.1109/MIC.2009.79
- [18] Hausenblas, M., Troncy, R., Bürger, T., Raimond, Y.: Interlinking Multimedia: How to Apply Linked Data Principles to Multimedia Fragments. *Linked Data on the Web Workshop* (2009)
- [19] Heath Tom Bizer, C.: *Linked Data: Evolving the Web into a Global Data Space* (2011)
- [20] Hickson, I.: *HTML Microdata* (2010). URL <http://dev.w3.org/html5/md/>
- [21] Jacobs, I., Walsh, N.: Architecture of the World Wide Web. *arq Architectural Research Quarterly* **2**(01), 1–56 (2004). DOI 10.1017/S1359135500001172. URL <http://www.w3.org/TR/webarch/>
- [22] Kobilarov, G., Scott, T., Raimond, Y., Oliver, S., Sizemore, C., Smethurst, M., Bizer, C., Lee, R.: Media Meets Semantic Web How the BBC Uses DBpedia and Linked Data to Make Connections. In: L. Aroyo, P. Traverso, F. Ciravegna, P. Cimiano, T. Heath, E. Hyvönen, R. Mizoguchi, E. Oren, M. Sabou, E. Simperl (eds.) *The Semantic Web Research and Applications, Lecture Notes in Computer Science*, vol. 5554, pp. 723–737. Springer (2009). DOI 10.1007/978-3-642-02121-3_10
- [23] Lambert, D., Yu, H.Q.: Linked Data Based Video Annotation and Browsing for Distance Learning. In: *The 2nd International Workshop on Semantic Web Applications in Higher Education (SemHE'10)* (2010)
- [24] Lee, W., Stegmaier, F., Poppe, C.: *API for Media Resource 1.0* (2010). URL <http://www.w3.org/TR/mediaont-api-1.0>
- [25] Li, Y., Wald, M., Khoja, S., Wills, G., Millard, D., Kajaba, J., Singh, P., Gilbert, L.: Synote: enhancing multimedia e-learning with synchronised annotation, pp. 9–18. *ACM* (2009)
- [26] Li, Y., Wald, M., Wills, G.: Interlinking multimedia annotations. In: *Web Science 2011* (2011)
- [27] Li, Y., Wald, M., Wills, G., Khoja, S., Millard, D., Kajaba, J., Singh, P., Gilbert, L.: Synote: development of a web-based tool for synchronized annotations. *New Review of Hypermedia and Multimedia* **0**(0), 1–18 (2011)
- [28] Malaisé, V., Bürger, T., Sasaki, F., Stegmaier, F., Joakim Soderberg: *Ontology for Media Resource 1.0* (2010). URL <http://www.w3.org/TR/mediaont-10/>

- [29] Manola, F., Miller, E.: RDF Primer (2004). URL <http://www.w3.org/TR/rdf-primer/>
- [30] Martinez, J.M.: MPEG-7 Overview (2004). URL <http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm>
- [31] Pfeiffer, S., Parker, C.D., Pang, A.T.: Specifying Time Intervals in URI Queries and Fragments of Time-Based Web Resources (2005)
- [32] Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF (2008). URL <http://www.w3.org/TR/rdf-sparql-query/>
- [33] Saathoff, C., Scherp, A.: Unlocking the semantics of multimedia presentations in the web with the multimedia metadata ontology. Proceedings of the 19th international conference on World wide web WWW 10 p. 831 (2010). DOI 10.1145/1772690.1772775
- [34] Sauer mann, L., Cyganiak, R.: Cool URIs for the Semantic Web. W3C Note **49**(681), 1–15 (2008). URL <http://www.w3.org/TR/cooluris/>
- [35] Shadbolt, N.: data.gov.uk - The Linked Data Revolution (2010)
- [36] Shaw, R., Troncy, R., Hardman, L.: Lode: Linking open descriptions of events. The Semantic Web **5926**(Lecture Notes in Computer Science), 153–167 (2009). URL <http://www.springerlink.com/index/C3385U550313P281.pdf>
- [37] Stamou, G., Van Ossenbruggen, J., Pan, J.Z., Schreiber, G., Smith, J.R.: Multimedia annotations on the semantic Web. Ieee Multimedia **13**(1), 86–90 (2006). DOI 10.1109/MMUL.2006.15
- [38] Steiner, T., Troncy, R., Hausenblas, M.: How Google is using Linked Data Today and Vision For Tomorrow. In: S. Auer, S. Decker, M. Hauswirth (eds.) Linked Data in the Future Internet 2010. Ghent, Belgium (2010)
- [39] Troncy, R.: Use cases and requirements for Media Fragments (2009). URL <http://www.w3.org/TR/media-frags-reqs>
- [40] Troncy, R., Mannens, E., Pfeiffer, S., Deursen, D.V.: Media Fragments URI 1.0 (2010). URL <http://www.w3.org/TR/media-frags>
- [41] Van Deursen, D., Troncy, R., Mannens, E., Pfeiffer, S., Lafon, Y., Van De Walle, R.: Implementing the media fragments URI specification. Proceedings of the 19th international conference on World wide web WWW 10 p. 1361 (2010). DOI 10.1145/1772690.1772931
- [42] Waitelonis, J.: Augmenting video search with linked open data. Proc of Int Conf on Semantic Systems (2009)