Synote: Weaving Media Fragments and Linked Data

Yunjia Li, Dr Mike Wald, Dr Tope Omitola, Prof Nigel Shadbolt and Dr Gary Wills

{yl2,mw,tobo,nrs,gbw} @ecs.soton.ac.uk

School of Electronics and Computer Science

University of Southampton
What is Media Fragment?

- It is the inside content of a multimedia resource
  - Temporal, spatial dimensions
  - Track

- Sharing and Searching the WHOLE multimedia resource is easy, but PART of multimedia is difficult

“enabling the addressing of media fragments ultimately creates a means to attach annotations to media fragments”

-- W3C Media Fragment 1.0 Specification
Introduction of Synote

- User can generate annotations and synchronise them with audio-visual resources

- Synote doesn’t store video, audio, image files

- Synote stores:
  - The URL references to video, audio image files online
  - User generated annotations and synchronisation points

- Single Resource: Tag, Note, Slide, etc

- Four categories of compound resources: Multimedia, Transcript, Synmark (tags, description), Presentation Slides

- Demo, every resource is displayed in one landing page
Synote Object Model

- **UrlResource**: hasOne `url`
- **Slide**: hasMany `index`
- **TextResource**: hasOne `content`
- **Cue**: hasMany `cueIndex`, `settings`
- **Note**: hasOne `hasMany`
- **Tag**: hasMany `hasMany`
- **Synmark**: hasOne `hasMany`
- **Multimedia**: hasOne `asTarget`
- **Compound**: hasOne `asSource`
- **Presentation**: hasOne `WebVTT`
- **Annotation**: hasMany `annotation id`, `source id`, `target id`
- **Synpoint**: hasMany `source start`, `source end`, `target start`, `target end`
Tim Berners-Lee: The next Web of open, linked data

by Yunjia created 30/01/2012

Transcript

The way we use information, the way we work together. I invented the World Wide Web. Now, 20 years on, at TED, I want to ask your help in a new reframing. Time flies. It's actually almost 20 years ago when I wanted to reframe the way we use information, the way we work together: I invented the World Wide Web. Now, 20 years on, at TED, I want to ask your help in a new reframing.

00:00:36 to 00:01:02
So going back to 1989, I wrote a memo suggesting the global hypertext system. Nobody really did anything with it, pretty much. But 18 months later -- this is how innovation happens -- 18 months later, my boss said I could do it on the side, as a sort of a play project, kick the tires of a new computer we'd got. And so he gave me the time to code it up.

00:01:03 to 00:01:10

Synmarks

Tim Berners-Lee: The next Web of open, linked data by Yunjia
position: 00:00:00
tags: tim berners lee tagged linked data

20 years ago, Tim Berners-Lee invented the World Wide Web. For his next project, he's building a web for open, linked data that could do for numbers what the Web did for words, pictures, video: unlock our data and reframe the way we use it together.

TED talk by Yunjia
position: 00:00:01 to 00:00:14
tags: tim berners lee tagged linked data

This TED talk took place at Terrance Theater and it is given by Tim Berners Lee about Linked Data.

Hans Rosling's talk by Mike
position: 00:04:24 to 00:05:11
tags: data tagged Hans Rosling

Here is Hans Rosling's talk.

Linked data and linked data principles by Mike
position: 00:06:09
tags: principles tagged linked data

Tim talked about linked data principles.

Linked data article by Mike
position: 00:07:40 to 00:07:48
tags: linked data tagged article

The article Tim mentioned is here.

Government Data by Mike
position: 00:08:00

15 Presentation Slides
Goal

- Use Synote as the target application to
  - publish existing media fragments as linked data
  - publish user-generated annotations as linked data
  - link annotations with media fragments

- Improve the Online Presence of Media Fragments
  - Media fragments could be indexed through annotations
  - Search engine can locate the precise media fragment
Media Fragment + Linked Data
The Benefit

Media Fragment can act as a glue to other resources online

dc:title
The next Web of open, linked data
presentedBy
http://www.w3.org/People/Berners-Lee/card#i

ma:hasKeyword
“Linked Data Principles”

thumbnail
Grassroot diagram

rdfs:seeAlso
http://dbpedia.org/resource/DBpedia

rdfs:seeAlso
Gov Data

06:02
07:28
08:21
09:15

Another YouTube video
The Principles [1]

- Identify temporal-spatial dimensions of Media Fragments
  - HTTP URI: W3C Media Fragment URI 1.0 Specification
  - Retrieve the original representation of Media Fragments
  - Dereferencing semantic representation (RDF)

- Alignment with legacy metadata

- Interlinking Methods: manual, collaborative, (semi-)automatic

Two Types of Annotations

Type One Data
- The multimedia file
- Framerate
- Resolution
- Title, e.g. Linked Data
- Author: John

Type Two Data
- Another title?
- Thumbnail pictures
- Comments
- Reviews
- Presentation Slides
- Domain specific annotations
- Related videos, etc
Retrieve Media Fragments (1)

- Problem: Keep out of the namespace you do not control [2]
  - example.org/1.mp4 is in another domain
  - Is 1.mp4#t=3,7 dereferencable or persistent over time?
- Solution: “synote.org/resource/id#t=3,7”
  - mint our own URIs for each resource including media fragment
  - Use ma:locator (W3C Ontology for Media Resource 1.0) to indicate the exact location of media fragment
  - Use 303 redirection and content negotiation to provide both HTML and RDF representation

Retrieve Media Fragments (2)

```
<resource/1> a ma:MediaResource;
   ma:hasFragment :t=3,7;
   rdfs:seeAlso <recording/replay/1>;
   rdfs:isDefinedBy <resource/data/1>;
   ma:locator <example.org/1.mp4>.

:tt=t=3,7 a ma:MediaFragment;
   ma:hasKeyword <resource/5>;
   ma:isFragmentOf <resource/1>;
   rdfs:seeAlso <recording/replay/1#t=3,7>;
   rdfs:isDefinedBy <recording/data/1>;
   ma:locator <example.org/1.mp4#t=3,7>;
```

“resource/1#t=3,7” is the fragment of non-information “resource/1”

The real location of the multimedia a TagResource, dereferencing it will get the RDF description about this resource

the real media fragment 1.mp4#t=3,7 is related to the user generated annotation “resource/5”
Choosing Vocabularies

• Reuse current vocabularies
  – Ontology for Media Resource
  – Open Annotation Collaborative (OAC)
  – Schema.org
  – Open Archives Initiative Object Reuse and Exchange (OAI-ORE) to describe resource aggregation

• We didn’t create any new vocabulary
Interlinking Methods

- Manually embed RDFa in Synmark Note
- Using RDF content editor such as RDFaCE

\[
:t=3,7 \text{ a ma:MediaFragment;}
\text{lode:illustrate } _:\text{event1.}
_:\text{event1 a lode:Event}
\text{rdfs:seeAlso } \langle \text{tim_berners_lee_on_the_next_web.html}\rangle;
\text{lode:involvedAgent } \langle \text{http://dbpedia.org/resource/Tim_Berners-Lee}\rangle;
\text{lode:atPlace } \langle \text{http://dbpedia.org/resource/Terrace_Theater}\rangle.
\]

- Triples in RDFa are published along with media fragments
- Disadvantage: manually write RDFa
- (semi-)automatic ways: Open Calais, Zamanta, NERD
Publishing Patterns

• RESTful API Wrapper + Rich Snippet
  – RESTful API to dereference RDF representation
  – schema.org to embed semantic description
  – “itemid” attribute to point to the URI of the resource
  – Problem: No SPARQL endpoint

• Synote has its own content management system and relational database

• So it is unwise to totally abandon the existing application

• Build an extra layer on top of existing application
Improve Online Presence of Media Fragments
The Difficulties

• Media Fragments are locked in the landing page

• The landing page is not search-engine-friendly
  – Everything is on the same page
  – No semantic description of media fragments can be recognised by major search engines
  – No preview of media fragments can be displayed in the search results

• But we still need to keep the existing landing page because it offers interactive experience
Google’s Ajax Content Crawler

- The Crawler is designed to index Ajax content
- Replace token “#!” in URLs with “_escaped_fragment_”

3. Server maps from ugly URL to pretty URL: www.example.com/page?query#!key=value
4. Server creates HTML snapshot for pretty URL
5. Server returns HTML snapshot

1. Crawler maps from pretty URL to ugly URL: www.example.com/page?query&_escaped_fragment_=key=value
2. Crawler requests ugly URL
6. Crawler processes HTML snapshot, extracts pretty URLs

*Diagram from https://developers.google.com/webmasters/ajax-crawling/docs/getting-started
1: Submit pretty URL `replay/1#!t=3,7` to the crawler
2: Crawler asks server for `replay/1?_escaped_fragment_=t=3,7`
3: Redirect the request to the snapshot page generated by the server. The snapshot page only contains annotations and Microdata for “#t=3,7”,
4: The snapshot page is returned to the crawler with URL `replay/1#!t=3,7`
5: A user searches keyword “Terrace Theater”
6: Google includes `replay/1#!t=3,7` in the search results
7: The user click the link and ask for the document at `replay/1#!t=3,7`
8: The server returns the landing page containing both “Terrace Theater” and “Linked Data”
9: The landing page highlights the media fragment by start playing from 3s to 7s
Conclusions
Conclusions

• Experience to publish media fragments with user generated annotations

• Applying linked data principles
  – 303 redirection and content negotiation
  – Totally reuse current vocabularies
  – Embedding RDFa in text note

• Some initial attempt to improve the online presence of media fragments

• More media fragments could be published to both semantic and traditional search engines
Questions?