Abstract—The variety of multimedia streaming platforms on the Web is ever growing, whereas (metadata) interoperability issues remain unsolved. To address this issue, the World Wide Web Consortium (W3C) launched the Media Annotations Working Group. One output of this group is the API for Media Resource, which defines a unified access to media resources on the Web.

I. MOTIVATION

Today, the public interest in data publishing and streaming platforms is tremendous. In this terms, uncountable metadata formats for annotation and query languages/APIs for retrieval of media resources emerged in analogy to the increasing number of different streaming platforms. The logical consequence of this trend is that tons of user generated multimedia content as well as metadata information are locked within these silos prevented from an unified access [1], [2].

This paper introduces an implementation of the API for Media Resource as a browser extension, to be more specific, for Firefox\(^1\). The API specification was defined by the W3C Media Annotations Working Group (MAWG) [3] with the main objective to access metadata information related to media resources on the Web in a harmonized way.

II. API FOR MEDIA RESOURCE

The API for Media Resource consists of two main parts: (i) interfaces to access media resources and (ii) a set of core properties for describing the information in an interoperable way. The interface definition is subdivided into a synchronous mode (immediate answer to every request) and an asynchronous mode. This paper focuses on the asynchronous part, where a request will be executed within a certain time period and then ready for retrieval. The API in general offers the getOriginalMetadata and getMediaProperty methods. The first method returns metadata of an associated media resource in the underlying metadata format. In contrast to that, getMediaProperty offers the possibility to define a subset of the core properties for which metadata informations should be retrieved. It is further possible to specify filter criteria, e.g., to get only metadata annotations for a specific language.

The API has been integrated into a Firefox extension, which can be called by browser-based applications. The core of the extension is based on JavaScript and the XML User Interface Language (XUL)\(^2\). Besides implementing the API, the extension is using the mapping rules defined in the MAWG Ontology for Media Resource report to support syntactical metadata transformations.

III. SHOW CASE: JOINT VIDEO PLAYLIST

The usage of the introduced MAWG API Firefox extension is shown inside a Web-based application. It is entitled “Joint Video Playlist” and serves as a mediator between various video streaming platforms enabling the user to create homogeneous video playlists. The main benefit is to break down interoperability issues between different vendors that lock videos and the corresponding metadata inside their data bins. As a proof of concept, the show case is able to process video and metadata offered by Youtube\(^3\) and Vimeo\(^4\).

Figure 1 highlights the main components of the application. Following the newest standard in web technology, it is created with HTML5 [4] and the function logic is implemented in an internal JavaScript library, e.g., gallery functionalities or API calls for video data access. All workflows related to metadata retrieval are executed inside the Firefox extension. If this extension is not pre-installed in the browser, the web-page will guide the installation process. The following functionalities are offered:

- **User administration**: In order to save a video playlist permanently, a user is able to register at the system.
- **Playlist generation**: The main focus lies on the generation of a joint video playlist. Here, a user may add videos

\(^1\)http://www.mozilla.com/

\(^2\)https://developer.mozilla.org/en/XUL

\(^3\)http://www.youtube.com/

\(^4\)http://www.vimeo.com/
from a supported streaming platform. Beside the video data, metadata will be collected and displayed using the properties defined in the **API for Media Resource** specification. The following properties are in use: Title, Creator, Date, Rating and Duration. After the import phase, a video can be added to the playlist and arranged in a global scoring using HTML5 Drag&Drop.

- **Playlist sharing:** A generated playlist can be published and shared on social networks, e.g., Twitter\(^5\) or Youtube.
- **History:** Up to twelve lastly viewed videos can be reached as short links in a history area.
- **Related search:** This functionality executes a keyword-based search on the connected streaming platforms and shows the top-eights videos with related keywords in standard mode. The set of keywords will be split into possible subsets (top-down) to ensure a minimum amount of 32 videos, which is the configurable, maximal enlargement of this view. Due to page limitations this functionality is not pictured in Figure 1.

For all videos displayed on the web-page, the logo of the corresponding source platform will be overlaid in the top right corner (except in the history).

**IV. Demonstration Procedure**

The demonstration is split in two different parts. The first part shows the basic functionalities of the show case, where the metadata interoperability functionalities of the Firefox extension have been embedded. The second part of the demonstration is based on an automatic test suite, which is able to evaluate, if an implementation of the **API for Media Resource** complies to the actual specification.

**V. Conclusion**

This paper presented the implementation of the **API for Media Resources** as Firefox extension, using a Joint Video Playlist showcase. The demonstration itself highlighted the usefulness of the extension to easily establish metadata interoperability for web-based applications.

**VI. Acknowledgment**

The authors would like to thank all the participants of the W3C **Media Annotations Working Group** for their contributions. This work has been partially supported by the German Federal Ministry of Economics and Technology under the **THESEUS Program**\(^6\) and from the European Union’s 7\(^{th}\) Framework Programme (FP7/2007-2013) under grant agreement n°215475, “2020 3D Media – Spatial Sound and Vision”\(^7\).

**References**


\(^5\)http://www.twitter.com/

\(^6\)http://www.theseus-programm.de/

\(^7\)http://www.20203dmedia.eu/