A MetaViewer for Sharing Multiple Media by WebGL-based Interfaces

L. Bergerot
UMR 3495 CNRS/MCC MAP
Marseille, France
laurent.bergerot@map.cnrs.fr

***Abstract* — In recent years, sharing images and videos on the Web has become common. Recently, the use of 3D models is also growing day by day, and the evolution of programming languages and libraries allow visualizing detailed 3D models and point clouds on the web. This opens new scenarios and possibilities for combining libraries, formats and contents (images, videos or 3D) within rich web layouts which can be visualized by multiple devices (PC, tablet, smartphone, ...). This current technological context represents a great opportunity for people involved the digital documentation of cultural heritage artifact, especially regarding the management of heterogeneous contents within digital libraries. In this paper we present the first results of an on-going development project aiming to develop a MetaViewer for sharing multiple media (images, videos and 3D) by the means of WebGL-based interfaces.**

***Index Terms*— WebGL, 3D, data, format, web platform.**

I. Introduction

The digitization of cultural heritage needs today managing multiple digital resources, presented on different media: images, videos or 3D. If the publication of digital content on the web has greatly increased in the recent years, especially regarding 3D real-time visualisation and interaction technologies, the need for simple, flexible and free solutions for managing the heterogeneous media which are generally used for documenting a heritage artifact (images, videos, 3D,…) through multiple devices (PC, tablets, smartphones, VR systems, …) still remains. This is why we started this project for developing a web MetaViewer able to adapt the interface context (visualization and interaction features) to multiple data type and according to multiple devices (see Figure 1) by basing on open source and free solutions for a complete HTML5 integration.

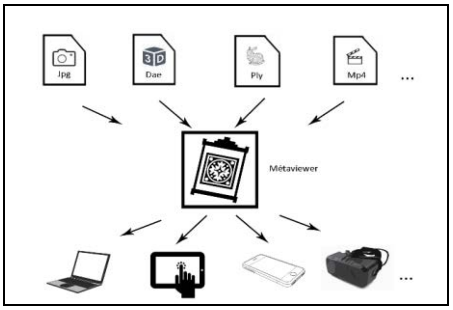

Fig. 1. MetaViewer Schema

II. Data Types

Our project intends to manage the most common formats used for Cultural Heritage documentation. Concerning 3D, we consider reality-based 3D models and cad-like 3D models by using the Collada (DAE) and the OBJ. By these formats we're able to manage structured or unstructured polygonal representations also enriched by textures (or vertex colors). We also consider the 3D raw data (sparse and dense point clouds) coming from laser scanning and photogrammetry, by using the PLY format and 3D and a compression method (potreeConverter) for preparing dense point clouds for a 3D real-time interaction on the web based on the Potree JS library (see section 3.2). Concerning 2D, JPG, PNG and BMP formats have been chosen for the images while the MP4 format has been chosen for videos.

III. Implementation

A. Javascript

We have chosen Javascript as main programming language for developing cross-browser and multimedia interfaces. The large number of libraries associated with Javascript gives us a very creative freedom, especially as the W3C standard [1] is applied to most of these libraries and will allow an optimal maintenance in the next years. Moreover, Javascript (and associated libraries) is also a free and open source solution that allows us to prevent long-term preservation and sustainability issues.

B. Main Program and features

By basing on this general framework, we have built the MetaViewer architecture by combining several JavaScript libraries, such as Three.js [2], jsc3d.js [3] and Potree.js [4] respectively to manage 3D models (reality-based and cad-like), 2D videos and 3D point clouds. A database is used for storing 2D and 3D media. PHP scripts are used for perfoming MySQL queries, while Javascript is used for managing the dynamic behaviour of the MetaViewer according to the data type and the used visualisation support. This infrastructure (PHP-MySQL + JavaScript + WebGL) is also used for managing the 2D and 3D content within HTML5 based graphic layouts.

By starting from the selection of a database category (e.g. a digital documentation project or a Heritage Asset) a first query allows to find all the related 2D and 3D content. Once a

978-1-5090-0048-7/15/$31.00 ©2015 IEEE

content is selected, an appropriate behaviour (visualisation and interaction) encapsulated into a JavaScript function, is selected and associated to the MetaViewer frame according to the data type and the viewing device. Thus the selected function add to the MetaViewer the appropriate library for visualising and interacting with media type by including this behaviour into a DIV Html tag.

IV. On-Going Experiments and Applications

Used as stand alone frame, the MetaViewer allows directly presenting 3D models (polygon-based and point-based) on a html page, as illustrated in the Figure 2 [5].

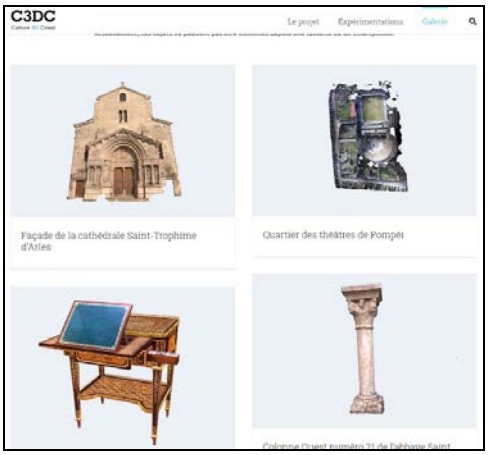

Fig. 2. The « Culture 3D Cloud » Gallery Page

The MetaViewer can also be embedded into a digital resources library (e.g. based on PhP/MySQL) for managing multiple 2D and 3D media on the web accompanied by descriptive information (e.g. typical metadata), as illustrated in the Figure 3 [6].

Fig. 3. Screen capture of our web platform with Metaviewer included (red square)

Our MetaViewer can also manage the binocular vision (with associated interaction behaviour) for visualising 3D content (3D models and point clouds) in immersive environment by using specific devices such as the Oculus Rift (Figure 4).

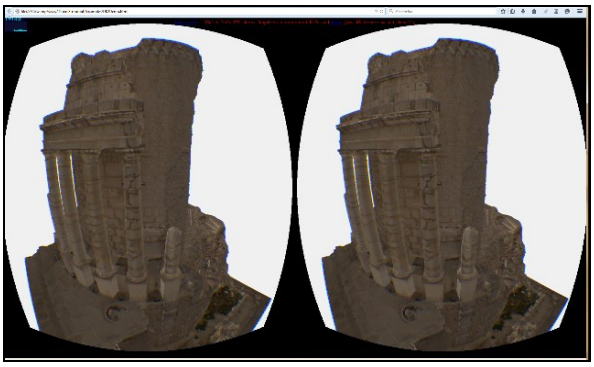

Fig. 4. Screen capture of Oculus Rift and Metaviewer view

V. Conclusion

Although the MetaViewer presents some interesting features and several potential applications, some limitations still remain.

The first of these limitations is the size of files that can be managed (especially for 3D). We noticed that we are not able to a downloading process exceeding 200 MB.

Another limitation is due to the browser refresh rate : we cannot exceed 60 fps at the moment. This limitation could be problematic within a stereoscopic environment (e.g. Oculus Rift) where the framerate is reduced to 30 fps.

The last limitation is that some browsers do not fully take into account W3C standards and some compatibility issues (especially for JavaScript libraries) could appear.

Despite these limitations, this on-going project opens various perspectives concerning the development of future platforms for collecting and sharing multiple digital resources for Cultural Heritage applications.

References

[1] W3C (https://validator.w3.org)
[2] Three.js (http://threejs.org/)
[3] jsc3d.js (https://code.google.com/p/jsc3d/)
[4] Potree (http://potree.org/)
[5] C3DC gallery (http://c3dc.fr/galerie/)
[6] Europeana 3D Icons Project (http://3dicons-project.eu/)