STAG: Smart Tools and Apps in computer Graphics (2016)
Giovanni Pintore and Filippo Stanco (Editors)

Automating Large 3D Dataset Publication in a Web-Based Multimedia Repository

M. Potenziani[1], B. Fritsch[2], M. Dellepiane[1], R. Scopigno[1]

[1]Visual Computing Lab, ISTI CNR, Pisa, Italy
[2]Excellence Cluster Topoi, HU, Berlin, Germany

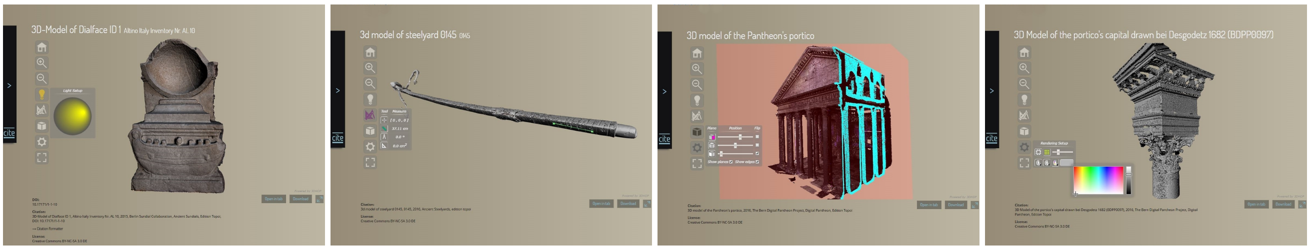

Figure 1: *An overview of the 3D items selected from the multimedia online repository (http://repository.edition-topoi.org/).*

Abstract

Online publishing of almost every type of 3D data has become a quasi-standard routine. Nevertheless, the integration in a web page of a single 3D model, or of a predefined restricted set of models, raises different issues compared to an efficient and effective integration of thousands of them in an online repository. In this case it is mandatory to have an automatized pipeline to prepare and homogenize the dataset. The pipeline should be able to automatically wrap 3D data in all conditions, and display every single model with the best scene setup without any (or with a minimal) interaction by the database maintainers.
This paper, retracing the steps of a recent real application case, aims at showing all the faced issues (and the adopted solutions) to publish a large and heterogeneous three-dimensional dataset in a web specialized repository. We want to introduce a valid and reusable strategy, starting from the description of the pipeline adopted for data pre-processing and moving to the choices made in the 3D viewer implementation. The paper concludes with a discussion on the actual state of the integration of 3D data with the other multimedia informative layers.

Categories and Subject Descriptors (according to ACM CCS): H.3.7 [Information Storage And Retrieval]: Digital Libraries— H.5.1 [Information Interfaces And Presentation (I.7)]: Multimedia Information Systems H.5.1 [Information Interfaces And Presentation (I.7)]: Group and Organization Interfaces—Web-based interaction I.3.4 [Computer Graphics]: Graphics Utilities— Application packages

1. Introduction

In the last years, the practice of sharing information on the web has touched almost all aspects of human life, becoming an hot topic not just in the social activities but also in other more specialized contexts. One of these is certainly the scientific one: the possibility to share data, to spread information, to link knowledge, opened the doors to collaborations and improvements which were unimaginable until a few years ago.
In the last decade, there has been a proliferation of every type of online portal, platform, database and repository. Nevertheless, the proposed solutions are very heterogeneous, addressed to different expertises (from humanities to science), developed following opposite philosophies (from restricted to Open Access), and with heterogeneous contents (from text to videos).
All these differences heavily affect the design choices to build the informative structure containing the published data, so as to make them accessible with the best user experience. From this specific point of view, the type of information people want to share plays a fundamental role, because of the inherently different nature of the various media (text, images, audio, and video) and also because of the different tools available to handle them (especially in a

DOI: 10.2312/stag.20161369

particular environment such as the web).

Among the group of multimedia contents, 3D data deserve a special mention. While it is becoming more and more familiar in plenty of application fields in these last years, this type of data is just the latest arrival in the information landscape. Also, the use of 3D in many specialized scientific areas began to give promising results long before the recent mass penetration of three-dimensional models in everyday life. In the last decade, we have seen the birth of several applications addressed to peculiar working groups (e.g. in the architecture and archaeological field), some of which have become an inseparable daily working tool.
The online publication of these new data and their integration with the usual set of information is important for several reasons:

- it results fundamental in the working activity centered on 3D dataset, but it also adds realism to that fields that makes just a marginal use of 3D data;
- it allows users to study specific information, sometimes viable just through three dimensional dataset, in a real-time and collaborative way;
- it provides a new way to show the information and to query a database, given the possibility to link directly or indirectly the other media on the 3D layer support;
- it can establish an extra level in the long term data preservation of (real world) digitized 3D models, acting both as an "informative bank" and as a facilitator in the spread of the data.

While the necessity and usefulness of shared 3D data online is clear, at the moment a fast, efficient and effective publishing of 3D Data on the web is not available. This is especially true when the number of models and the level of required process automation drastically increases.

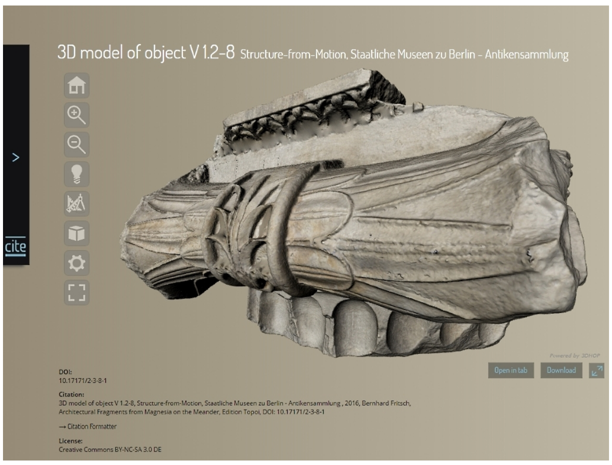

Figure 2: *A screenshot showing the final implementation of integration procedure, with the uploaded 3D model in the center of the interactive online 3D scene and other textual information about the selected repository item in the bottom of the web page (example available at http://repository.edition-topoi.org/collection/MAGN/single/0005/0).*

This paper describes the actions taken in a recent project aimed at integrating a large dataset of heterogeneous 3D models in the context of an Open Access archaeological multimedia database.
In the next sections we'll present two different working stages: the first one deputed to introduce a rough 3D data pre-processing, exploited to re-orientate the 3D object in the space and to adjust other technical issues, and the second one addressed to describe the design and the development of a "general purpose" web viewer, which allows users to visualize and analyze a large range of 3D model with different basic technical features.
Then, the final results of the publishing routine will be shown (see Figure 2). The paper has the aim of opening a discussion about the condition of 3D data as real multimedia layer and focusing on what we still need to move to another level of automation and abstraction in the integrated online publishing of large 3D dataset.

1.1. The repository

As part of the interdisciplinary research association of *Topoi* (http://www.topoi.org/) in Berlin, which combines different kinds of studies about ancient cultures, the *Edition Topoi* (http://www.edition-topoi.org/) was launched in April 2016. It is an innovative digital publication platform which also holds, besides books and articles, a number of digital repositories for all the research data which has been generated within the scope of the project. It is strictly Open Access and the digital collections (see Figure 3) aim to be a long-term archive for the research on ancient cultures . All digital resources are considered as independent publications (called "citables"), thus are quotable and constantly reachable through the use of DOIs.

In the first phase eight repositories of the collections contain 3D Data. Since the research association covers quite a number of different subjects like Philosophy, Epigraphy, Archaeology or History of Science, and the data were acquired in different projects over a longer period in many different places all over Middle Europe, the Mediterranean Sea and the Near East, the data are quite heterogeneous. 3D models were created by Laser scanning, Structure-Light scanning or Structure-from-Motion (SfM), resulting in objects of different size and types like point clouds or meshes, with or without color and other attributes. For that reason, a flexible and easy to use Open Source 3D Web viewer is needed. Additionally, the performance (loading time, etc.) in the viewer is an important issue so that also users equipped with slow machines can use the repositories.

A short description of the heterogeneous collections involved may clarify the need for a robust and reliable publication strategy. The collections contain:

- the Digital Pantheon Project, which is the only complete laser scan model of the Pantheon in Rome. The model was split into over 200 separated point clouds. This helps in handling different research questions, like for example that one concerning the shape of the columns of the *portico* [GB11]. For that reason, only the point cloud is needed for further analyzing. Thus the visualization has to pay attention to the correct display of the points, especially the light and normals conditions;
- a collection of cylinder seals of the *Vorderasiatisches Museum - Staatliche Museen zu Berlin* . The over 1200 objects of an average size from 2-3 cm resulted in point clouds about 200 000 and

900 000 vertices . The scans were used to create digital impressions of the seals where now different figures can be carved out. The technique of the fabrication of the seals can be analyzed as well;

- a collection of ancient sundials, which covers a wider range of 3D models. About 60 percent of the approximately 230 3D models were acquired with a Structure-Light scanner, the other 40 percent were obtained through the method of Structure-from-Motion. Both techniques provide meshes; the SfM models do additionally have color information. Since the objects are partly fragmented and partly quite large, the number of vertices and faces exhibits a lot of differences;
- a large number of architectural fragments of Magnesia on the Meander which are hold by the *Antikensammlung - Staatliche Museen zu Berlin* . The outcome of the project is available in the repository including models of two different laser scanners (Structure-Light scanning) and SfM.

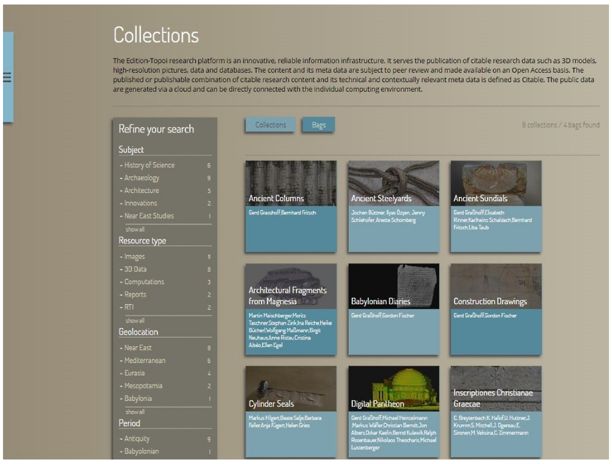

Figure 3: *The picture shows the landing web page of the repository, with all the various items composing the online database ordered in the established reference categories (http://repository.edition-topoi.org/).*

All in all the Edition Topoi Collections cover a broad spectrum and a large number (around 2000) of 3D models based on the different subjects that have different research questions to digital data and the different methods that were used over the last ten years to create the models.

2. Related work

As already stated, several communities (i.e. modelling and 3D printing) usually share or sell 3D models. The structure of repositories is heterogeneous, given the aim of use and the typology of data. Even in the research community, there have been several efforts (i.e. the Visionair project [AGPS13]) to provide integrated infrastructures.

In the context of Cultural Heritage (CH), a certain amount of 3D models is available, and some effort has been done to create repositories which could contain different data types. Some of them were mainly devoted to create an infrastructure from scratch [PSS*12], while others aimed at creating metadata schemes that could be integrated with existing repositories, like Europeana [DF13]. This task proved to be challenging, due to the variability in the CH domain, and difficulty in extending existing schemes.

Additionally, several of the proposed solutions are "plain" repositories, where the 3D model can be only downloaded, while the advent of WebGL [Khr09] gave the possibility to further exploit 3D graphics directly on a browser. This still poses a lot of challenges [ERB*14], related not only to data format and rendering, but also to interaction paradigms and data protection.

The interactive visualization on the web was one of the major issues: the proposed solutions, mainly based on progressive rendering [LJBA13, LCD13, PD15], give now the possibility to handle very complex geometries.

Among the CH-related tools for remote presentation of 3D models, an interesting example is the Smithsonian X3D explorer [Smi11], developed as a "branch" application of the Autodesk Memento engine [Aut11]. Unfortunately, the tool has not been distributed, and in any case it needs a certain amount of manual work to set up the presentation. Similarly, the free 3DHOP tools [PCD*15] allow to create simple visualizations with a limited effort, but cannot be directly used for automatic web publication.

Seamless publication is ensured by some services that focused part of their work on the support to the CH community: for example, SketchFab [Ske14] enabled the main museums in the world to access the "Pro" functionalities when publishing 3D models coming from their collections. Similarly, the Visual Media Ariadne Services [PPD*15] was created to support users in publishing advanced media content. Considering the aim of the proposed work, these tools are limited because the metadata associated to the 3D models are basic and no integration with more complex databases is possible. Additionally, several aspects like the visualization modalities, the interaction paradigms and the additional features must be set by users by hand after the publication.

A recent work by Galeazzi [GCD*16] aimed at a similar task to the ones taken into account in this paper: automatically integrating 3D models in an already defined and complete data structure for archaeological data. The integration works in a seamless way, creating a navigation canvas for the 3D model. However, some of the aspects tackled in the next sections (i.e. batch pre-processing, visualization modality) are not taken into account.

3. 3D dataset integration

The publication of a full dataset of objects in a repository requires an elevated level of automation. It's clear that when the number of elements reaches the order of magnitude described in Section 1.1, a routine which contemplates to work with the single object is unfeasible, so what is really needed is a strategy that attacks the problem in its generality.

When the data to treat are three-dimensional (possibly big complexity, huge disk space usage, poor standardization and elevated intrinsic heterogeneity) and the database to implement is a web-based database (limited computing resources, unknown available bandwidth, expensive server disk space), the issues increase significantly.

3D data indeed, frequently need to be processed to be prepared for a ready-to-use visualization, and these modifications generally re-

quire onerous computational actions, that can't be delegated to the network infrastructure, especially whereas they should be multiplied by the thousand instances of the repository objects.
Hence, it's mandatory to employ a working strategy able to move all the complex calculus on the dataset in a pre-processing stage (as automated as possible, given the amount of data involved) to execute locally. Only after this step, aimed to smooth the coarse differences between the input data, it will be possible to feed the database with the new output data, still considerably heterogeneous between them, but ready to be visualized in a viewer able to adapt itself to these differences.
From the point of view of this paper, if on one hand the composition of the database taken in consideration (entirely composed by acquired digitized CH items) helps to narrow the field of action within certain limits, on the other hand it doesn't detract anything to the complexity of the problem. The origin of the data indeed, acquired around the world over a long period using various technique and different specialized tool, by itself introduces a level of dissimilarity (basically typical of all the large generic repositories) which only serves to confirm the proposed pipeline (Figure 4).

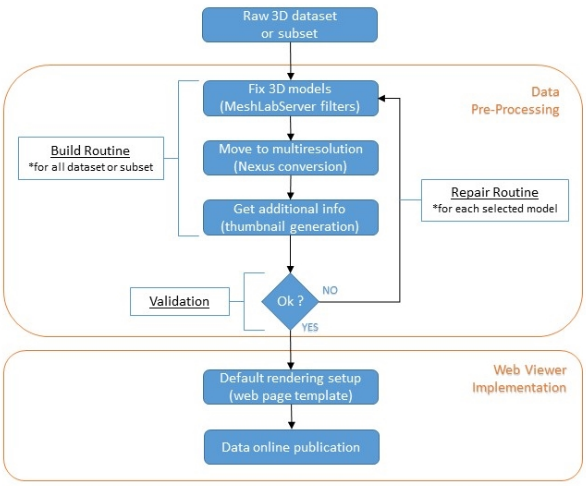

Figure 4: *The schematized flowchart of the adopted pipeline, basically composed by two blocks: a first one deputed to 3D dataset pre-processing and the latter to the web viewer configuration.*

3.1. Pre-processing pipeline

Several considerations have to be taken into account when dealing with 3D models. Among them, data format and navigation are critical.
Despite some attempts to define a common standard 3D format, [Rag95, DB07], a myriad of alternatives (PLY, STL, OBJ, OFF, PTX, VMI, DAE, PTS, XYZ, GTS, PDB, TRI, ASC, X3D, and WRL, just to name a few) is still around. While several conversion tools are available, it's necessary to be careful in preserving the attributes of 3D elements, and sometimes the limited standardization of the formats can lead to errors and loss of data.
Additionally, single resolution 3D models can be used on the web only when a limited amount of data has to be visualized. Otherwise, multi-resolution data structure are needed to provide a seamless navigation in a short time.
Another big issue is the definition of the interaction paradigm associated to a model. The solutions adopted for instance to visit from inside a building reconstruction are almost certainly inappropriate to appreciate a digitization of a piece of ancient pottery, and *viceversa*.
These premises suggest that a pre-processing step may be needed to "prepare" the 3D models, and that this step could need some validation. Hence, an analysis of the input 3D data may help in creating the pre-processing stage.

The repository chosen for this test case can help to reduce the working area. First of all, all the models were associated to a similar origin (CH related) and to a similar intended use (scientific study or preservation). These simple constraints are fundamentals, because they cut off all the huge world of the modeled 3D data, where data format and topology could devise for different solutions in the processing stage.
Additionally, another important assumption concerns the navigation paradigm of the 3D scene: since the great majority of the models are archaeological findings digitized as single separated objects, all them well suite to a "turntable trackball" paradigm, where the object is treated as it is on the hand of the user, who is able to rotate the view around the object, and zoom in and out. The trackball paradigm can be defined using a few parameters that can be easily pre-calculated by analysing the bounding box of the object. Other more accurate [Mal13] or *ad-hoc* [CDS15] approaches may be hard to extend to more complex surfaces.
For the trackball paradigm, the only important factor is that the model must be "aligned" to the axes of the reference system, otherwise the navigation is not realistic.

In order to take into account the needs and assumptions regarding the collections, we designed a smart batch pre-processing stage able to work on all the original data, implemented as semi-assisted routine, able to provide a graphical feedback to the database developer and also to be programmable depending on the dataset needs. This procedure (Figure 4, top), applying mathematical transformation on the entire 3D dataset, could easily need a lot of computational resource, so it needs to run locally, preferably on the same machine where the data are hosted.
The batch procedure is expected to:

- Fix some basic issues related to the models acquisition/generation (to do this we used the MeshLabServer service, a powerful highly programmable scripting tool provided by MeshLab [Mes]);
- "Prepare" the models for the online environment, converting them from the original data format to a web ready format (to do this we exploit the Nexus [Nex] package, a free software able to implement a functional multiresolution and compressed 3D data format [PD15]);
- (Optional) extract from the models some useful information to link in the database enriching the informative content of the related web page (this can be obtained with several software/tools, depending on which information is needed).

More in detail, what we have done with the specific 3D data in our

repository has been: the correction of the objects spatial positioning (since their alignment with the reference system was different w.r.t. the one used by the adopted web viewer), the re-calculation of the normal vectors orientation, and the export in a 3D format (*.ply*) suitable to be used as input for the subsequent conversion.
Subsequently, the PLY files were converted in a multiresolution compressed format (*.nxs*), called "Nexus". Then, a rendering of the 3D model from the initial point of view (given a basic turntable trackball paradigm) is generated. The image is intended to be used as a thumbnail, and as an output for user validation (see later).
These actions have been implemented in a script file, called "*build*". In detail, running the "*build*" batch procedure the system:

1. enters in the dataset location and visits all the directories and the sub-directories looking for the input model files;
2. for each 3D model found: it applies the MeshLabServer scripts with two selected filter (rotation and normal vectors calculation), saving the obtained fixed model in PLY format;
3. it feeds the Nexus converter with the obtained PLY files, getting as output the corresponding multiresolution compressed NXS model files;
4. for each final Nexus files: it opens it locally and take a screenshot of the model saving a copy of it in an dedicated folder (called "*build*") unique for the entire database.

At the end of the first phase, in addition to the original model, and the generated PLY and NXS models, there will be a "*build*" folder containing the thumbnails of every single object of the dataset.
The thumbnails are used by those who operate on data to understand which models need to be further processed (Figure 5).

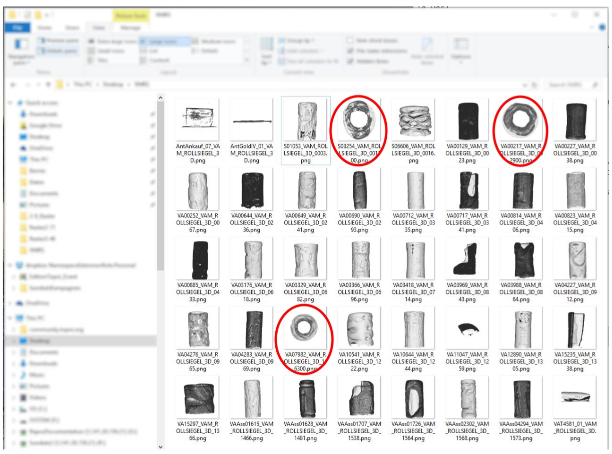

Figure 5: *A figure showing a sample "build" folder with the thumbnails of 3D models after the first stage of the pre-processing pipeline. Examining the pictures of the models (almost all cylinder seals, in this case) it's easy to see which of them after the first rotation are still in an wrong initial position (thumbnails circled in red).*

The second stage of the pre-processing is again implemented as a script. If the data operator finds a model that needs repairing (i.e. a further alignment in the reference space is needed), he moves the associated thumbnail from the "*build*" to another dedicated folder (called "*repair*"), then he selects the right MeshLabServer filter to apply (i.e. "*rotation*"), and finally he runs the "*repair*" batch script. In the case of the collections taken into account, the MeshLabServer filters that could be applied were dedicated to two possible actions: rotation, and normal estimation. These operations could be needed to fix the right initial position of the model or have better estimated normals. The "*repair*" script, acting only on the repository models with the same name of the the thumbnails placed in the homonymous folder, follows the same steps as the "*build*" one, finally updating with the new thumbnails the "*build*" folder where the user can check and validate the results again.

This pipeline structure, composed by sequential blocks, is highly configurable. The only mandatory constrains indeed are the kind of output format from the MeshLabServer stage (i.e. PLY) and the final 3D model file format (i.e. NXS) which as we shall see is fundamental for operating in an efficient way in the viewer.
All the other actions on the data can be decided by the data operator depending on the 3D dataset, just changing the associated pipeline block.

3.2. Web viewer implementation

The pre-processing pipeline introduced in the previous subsection is fundamental to smooth the principal rough differences in the original dataset. However the previous step, while preserving the basic and fine features of the data (geometric structure, color information, etc), still preserves their differences and peculiarities.
Hence, it's important to choose an appropriate web viewer, which should be able to support large amount of 3D data and to automatically get (for every single model) the best scene setup, minimizing the intervention by the platform developer.
So, planning an efficient and effective integration of the proposed set of data in the proposed online database, we decided to employ a 3D viewer which was:

- specifically designed for the web (able to guarantee a solid and fast data streaming in "all terrains" conditions);
- able to support quite wide range of data setup possibilities (point clouds and meshes, textured model, with per vertex color or without any color, etc);
- simple enough to be modified according to the needs of the developer (so as to use it, for instance, in an automatized pipeline where the HTML element containing the 3D scene have to be generated with an unsupervised routine).

The choice has fallen on 3DHOP [PCD*15], a free set of tools able to handle even complex 3D data (like the ones coming from acquired CH 3D artifacts), to ensure a web friendly streaming and rendering of these data (thanks to a proprietary multiresolution compressed format already cited in subsection 3.1), and easy to be modified to get the wanted behavior and to produce a homogeneous output for all the repository.
Starting from the basic implementation of the viewer, which natively provides a set of tools addressed to the CH field, and also supports virtually all the data configuration of the repository dataset, we just modified it in a couple of features, adding more flexibility and adaptability as required by a project like this, where the creation of the web page cannot be supervised step by step.

The key point of this implementation (Figure 4, bottom) has been to act on the default settings of the scene, moving them from an hard coded choice to an exposed and programmable feature, giving more decision-making power both to the repository developer (who can set his preferred default values in the viewer template passed to the web page) and to the final user (who can change the default setup interacting with a dedicated control panel, as shown in Figure 6).
The source code of the viewer was edited, both at GLSL (OpenGL Shading Language) and JavaScript level, adding portions of code able to switch between different rendering modes and modifying the preexisting data structures. Appropriate informative flags, added to the JavaScript objects related to each model, are used to describe the different structural characteristics of the 3D instances. The introduction of these flags gives the possibility to manage a couple of characteristics which are closely related with the virtual scene basic appearance: the geometric and the color representation of the 3D models.
This basically allows to overcome the need to set the best scene configuration model by model.
For example, a 3D viewer may be set by default factory settings to render preferably texture color rather than vertex color. Now, this feature could make sense (usually texture resolution gives better visual appearance to a 3D model), but maybe the two color information modalities could be able to convey different informative content, or maybe the data owner could prefer to shown a feature rather that another. While fixing this behavior for a single model publishing is not hard, dealing with a full dataset needs a certain amount of automation.

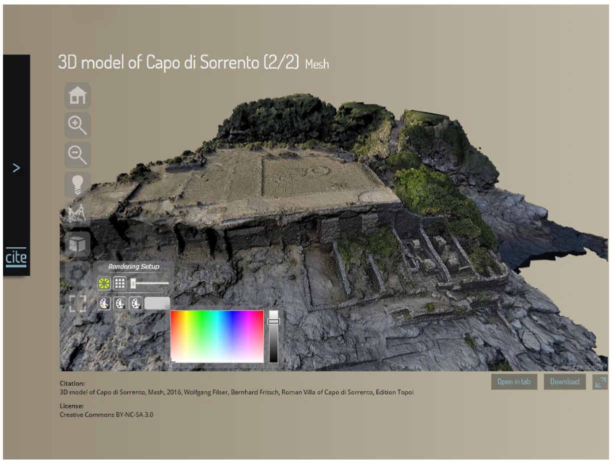

Figure 6: *A sample image showing the implemented setup panel working: using the panel the final user can modify the default setup automatically defined by the web viewer. The panel makes possible to select different rendering options changing the geometric primitives (triangles or points) and the color features (texture, vertex or solid). This example is available at http://repository.edition-topoi.org/collection/SORR/single/00002/1.*

So, next to the automatic default choice, we need to add other options to the web viewer if employed in a large repository:

- The database developer can create his own setup default, by editing the template used to generate the web repository page with the viewer. So, exactly as he could decide which tools (see next Section) will be put on the web visualization page, he can choose the geometric primitives and the color feature to render at the start;
- The final user, in a dedicated interface panel, can select his own preferred setup and explore all the possibility given by the 3D model representation, for instance switching between model and point cloud representation, or also between texture and solid color.

The use of a pre-defined template actually gives the possibility to adapt the visualization to the need of large groups of similar 3D models.

4. Results

The entire web repository was populated using the proposed pipeline in a short time, and it's currently working online since the project release (April 2016).
The pre-processing of the full 3D dataset, carried up on a common machine, allowed to prepare the data with few loops of "*build*" and "*repair*" batch procedures, saving time for the database maintainers.
Using the output of this first working stage, the second stage of the proposed workflow has been performed without complications. The generation of the web pages related to each single object has been obtained with standard web routines used in the development of similar online structure (i.e. using templates) with the only addition of the design pattern able to produce the 3D viewer implementation.

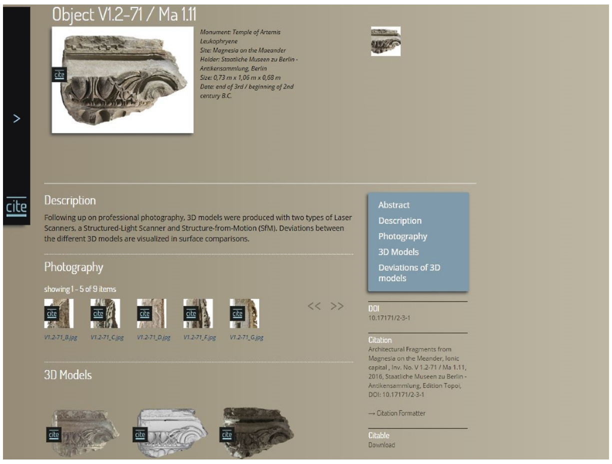

Figure 7: *An example of web page referring an object currently integrated in the online repository (http://repository.edition-topoi.org/collection/MAGN/object/V1.2-71).*

Also, thanks to a preliminary minimal knowledge of the 3D dataset, composed by subsets of objects coming from separated campaigns, it has been possible to improve both the working stages, subdividing data in big groups of models produced/acquired with the same workflow (and so affected by the same issues or sharing the same features), hence calibrating the pre-processing phases and the virtual scene setup with ad-hoc settings studied to enhance the final result.

In the current implementation, the web repository allows the access to 12 different collections of items, containing a mixed set of data: images, reports, RTI, computations and also 3D models (present in 8 of the mentioned collections).
For each collection, the user gets information about the selected subset objects. Subsequently he can refine the search until landing on the page of a single object. Here all the informative multimedia layers (text, pictures, 3D, RTI, etc) are shown, organized in a rational and sensible way (Figure 7).

The thumbnail of the 3D object gives access to the web viewer page, where besides to other textual information related to the model, there is the pre-defined virtual scene.
Thanks to the multiresolution implementation of the viewer the interactive 3D model is soon available, ready to be explored and analyzed with the provided toolset, which contains, aside from the basic navigation buttons (home and zoom controls), a set of 4 different tool, in the following shortly described.

The light control tool enables the control of the light source on the virtual scene: through a dedicated panel the user can interact with an intuitive HTML canvas (the sphere in Figure 8), moving and positioning the punctual source everywhere around the 3D scene. In the CH domain, radent light is a powerful mean to visualize fine geometric details.

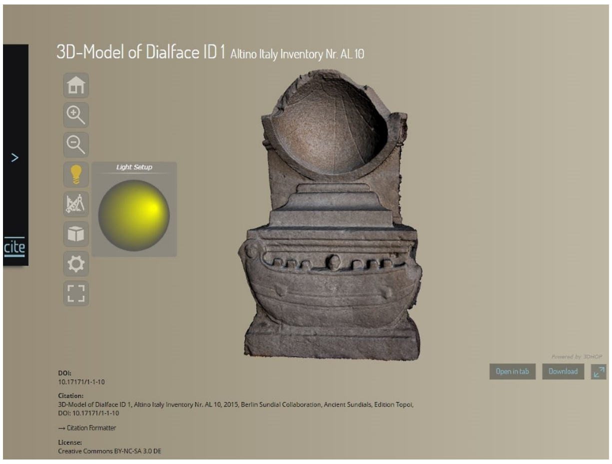

Figure 8: *The light tool: interacting with the light sphere the final user can modify the position of the light source present on the virtual scene (example available at http://repository.edition-topoi.org/collection/BSDP/single/0515/0).*

The measurement button gives access to several tools: the retrieval of the point coordinates in the 3D scene space, the calculation of the linear distance between two points, the measurement of the plain angle between two lines, and the estimation of the area defined by a polyline (Figure 9).

Using the sectioning tool is possible to get planar sections of the model on the scene (Figure 10). Three orthogonal sectioning

Figure 9: *The measurement tool: by clicking on the panel icons the repository user can select four different interactive measurement tool: point picker, distance tool, angle tool, area tool (example available at http://repository.edition-topoi.org/collection/BSYP/single/0722/0).*

planes are provided, the proper use of them not only give accurate references points (for measurement or other needs, for instance), but also allows to visualize the internal parts of the 3D objects.

Figure 10: *The sections tool: interacting with the sliders in the panel the final user can independently move each orthogonal section around the virtual scene. In addiction he can also control the section edges and planes visibility (respectively the cyan and red regions in this example, available at http://repository.edition-topoi.org/collection/BDPP/single/0753/0).*

Finally, the setup tool, already introduced in Section 3.2, allows user to modify in real time the scene rendering parameters, giving the possibility to change the displayed primitives (switching between triangles and points), or to move from a textured to a color per vertex 3D model representation. It also allows to modify the size of the points in a point cloud rendering and, at last, to select a desired solid color for visualization (Figure 11).

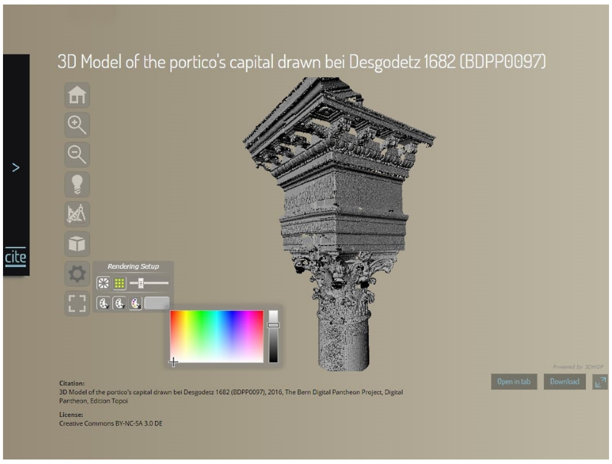

Figure 11: *The setup tool: the picture shows the color setup feature in action. By selecting the desired value in the displayed color picker the final user can select the solid color to use during the 3D model rendering (example available at http://repository.edition-topoi.org/collection/BDPP/single/1010/0).*

Of course, this implementation of the 3D web viewer interface is just one among all the possibilities. The tools to add or to remove can be easily chosen by the database developer, who can freely select, for instance, to delete the sectioning tool, to separate the color setup panel from the rendering primitive setup panel, or even to integrate some hotspot features to link external information directly inside the virtual scene.

5. Conclusion

The pipeline introduced in this paper aims to drive, automate and speed up the publishing of a large set of 3D data in a web repository.

The first critical issue to be faced was the heterogeneity situation in which 3D data are. Acquired with different tools, or using different techniques, or sometimes long time ago, possibly they represent a different state of art in the data. Furthermore, since the creation of models was undertaken in different disciplines, the focus of an acquisition campaign can cover a wide variety of objects. Moreover, the 3D models could be handed over to the repository in miscellaneous (or in the worst case in proprietary) file formats, with extra texture files or without, and so on.
All these dissimilarities, together with the increasing number of software developed to provide 3D rendering, are problematic for the consistency in the graphic representation.

In addition to these technical issues, the background of the users must be taken into account. In this case, the users from the CH community could have a limited experience with 3D navigation. Hence, very simple structures and easy handling tools were needed.

In order to solve these (and other) issues we proposed a two stage work flow, where a first (supervised) pre-processing of the entire raw dataset is performed, and then an online virtual scene setup is automatically generated.
Although this pipeline brings to real benefit and to a fast deployment of the database structure, some issues are still not solved. For instance, the procedure is currently partially supervised .
Additionally, in the future development of this project it would certainly help to add more interaction with the models, giving for instance the possibility to mark annotation, hotspots and links, in a easy way inside the 3D scene.
Hence, further research efforts are needed to address the urgent needs for the publication 3D dataset online. The final goal is to achieve a real and full integration of the 3D multimedia layer.

Acknowledgements

The research leading to these results was partially funded by EU INFRA Project Ariadne (GA n. 313193, http://www.ariadne-infrastructure.eu/).

References

[AGPS13] Attene M., Giannini F., Pitikakis M., Spagnuolo M.: The visionair infrastructure capabilities to support research. *Computer-Aided Design and Applications 10*, 5 (2013), 851–862. doi:10.3722/cadaps.2013.851-862. 3

[Aut11] Autodesk: Project Memento. http://memento.autodesk.com, 2011. 3

[CDS15] Callieri M., Dellepiane M., Scopigno R.: Remote visualization and navigation of 3d models of archeological sites. In *ISPRS Archives, Proceedings of 3D-ARCH Conference* (March 2015), vol. XL-5/W4, pp. 147–154. URL: http://vcg.isti.cnr.it/Publications/2015/CDS15. 4

[DB07] Don Brutzmann L. D.: *X3D: Extensible 3D Graphics for Web Authors*. Morgan Kaufmann, 2007. 4

[DF13] D'Andrea A., Fernie K.: CARARE 2.0: A metadata schema for 3d cultural objects. In *2013 Digital Heritage International Congress, Marseille, France, October 28 - November 1, 2013, Volume II* (2013), pp. 137–143. URL: http://dx.doi.org/10.1109/DigitalHeritage.2013.6744745, doi:10.1109/DigitalHeritage.2013.6744745. 3

[ERB*14] Evans A., Romeo M., Bahrehmand A., Agenjo J., Blat J.: 3d graphics on the web: A survey. *Computers & Graphics 41*, 0 (2014), 43 – 61. URL: http://www.sciencedirect.com/science/article/pii/S0097849314000260, doi:10.1016/j.cag.2014.02.002. 3

[GB11] Grasshoff G., Berndt C.: *Die Entasis der Säulen des Pantheon*. Berlin: Exzellenzcluster 264 Topoi, 2011. URL: http://edition-topoi.org/articles/details/die-entasis-der-saeulen-des-pantheon. 2

[GCD*16] Galeazzi F., Callieri M., Dellepiane M., Charno M., Richards J., Scopigno R.: Web-based visualization for 3d data in archaeology: The {ADS} 3d viewer. *Journal of Archaeological Science: Reports 9* (2016), 1 – 11. URL: http://www.sciencedirect.com/science/article/pii/S2352409X16303261, doi:10.1016/j.jasrep.2016.06.045. 3

[Khr09] Khronos Group: WebGL - OpenGL ES 2.0 for the Web, 2009. 3

[LCD13] Lavoué G., Chevalier L., Dupont F.: Streaming compressed 3d data on the web using javascript and webgl. In *Proceedings of the 18th International Conference on 3D Web Technology* (New York, NY, USA, 2013), Web3D '13, ACM, pp. 19–27. 3

[LJBA13] LIMPER M., JUNG Y., BEHR J., ALEXA M.: The pop buffer: Rapid progressive clustering by geometry quantization. *Computer Graphics Forum 32*, 7 (2013), 197–206. 3

[Mal13] MALOMO L.: Generalized trackball and 3d touch interaction. *Master thesis, Università degli Studi di Pisa* (2013). 4

[Mes] MESHLAB: An open source, portable, and extensible system for the processing and editing of unstructured 3d triangular meshes. http://meshlab.sourceforge.net. 4

[Nex] NEXUS: Multiresolution visualization. http://vcg.isti.cnr.it/nexus. 4

[PCD*15] POTENZIANI M., CALLIERI M., DELLEPIANE M., CORSINI M., PONCHIO F., SCOPIGNO R.: 3dhop: 3d heritage online presenter. *Computer & Graphics 52* (Nov 2015), 129–141. URL: http://vcg.isti.cnr.it/Publications/2015/PCDCPS15. 3, 5

[PD15] PONCHIO F., DELLEPIANE M.: Fast decompression for web-based view-dependent 3d rendering. In *Web3D 2015. Proceedings of the 20th International Conference on 3D Web Technology* (2015), ACM, pp. 199–207. URL: http://vcg.isti.cnr.it/Publications/2015/PD15. 3, 4

[PPD*15] PONCHIO F., POTENZIANI M., DELLEPIANE M., CALLIERI M., SCOPIGNO R.: The ariadne visual media service. In *CAA 2015* (2015), p. 12. URL: http://vcg.isti.cnr.it/Publications/2015/PPDCS15. 3

[PSS*12] PAN X., SCHIFFER T., SCHRÖTTNER M., HAVEMANN S., HECHER M., BERNDT R., FELLNER D. W.: A scalable repository infrastructure for CH digital object management. In *18th International Conference on Virtual Systems and Multimedia, VSMM 2012, Milan, Italy, September 2-5, 2012* (2012), pp. 219–226. URL: http://dx.doi.org/10.1109/VSMM.2012.6365928, doi:10.1109/VSMM.2012.6365928. 3

[Rag95] RAGGETT D.: Extending WWW to support platform independent virtual reality. *Technical Report* (1995). 4

[Ske14] SKETCHFAB: Publish and find the best 3d content. https://sketchfab.com, 2014. 3

[Smi11] SMITHSONIAN I.: Smithsonian X 3D. http://3d.si.edu, 2011. 3