DreameO: An Educational Experiment on Long-Term Digital Archiving of Student Multimedia Projects at UVHC

Nicolas Viéville[1,2], François-Xavier Coudoux[1,3,4]

[1] *Université Lille Nord de France, Lille, France*

[2] *UVHC, LSC DeVisu, Valenciennes, France*

[3] *UVHC, IEMN/DOAE, Valenciennes, France*

[4] *CNRS, UMR 8520, Villeneuve d'Ascq, France*

{Nicolas.Vieville, Francois-Xavier.Coudoux}@univ-valenciennes.fr

Abstract

The DREAM (Development, Research and Education for Audiovisual and Multimedia) department at the University of Valenciennes and Hainaut Cambrésis (France) has initiated for about ten years a collaborative student project called DreameO. Originally, DreameO should serve as a simple tool for the distribution on the UVHC website of video contents created by the students of DREAM. Then, the project progressed towards a management and archiving/video streaming platform administered via a website (LAMP system – Linux Apache MySQL PHP). Today the aim of the DreameO project is to provide a fully-customized open source Media Asset Management for the DREAM students and academic staff members. The DreameO platform is entirely designed and developed by the students and use only free software solutions. It should allow the management, archiving, dissemination of student multimedia projects. A wide type of projects is covered including video products, audio products, digital graphics products (2D/3D images), but also project reports (audiovisual production/post-production or technical reports), and consequently it is needed to account for different types of data formats and technologies. In this paper, we present an overview of the DreameO project. First we describe the required features of the platform, and then we focus on some technical solutions implemented by the students for digitalization and archiving. Finally we conclude and discuss on the future of the DreameO project.

1   Introduction

The DREAM (Development, Research, Education in Audiovisual and Mul-
timedia) department at the University of Valenciennes and Hainaut Cam-
brésis (UVHC, France) has initiated for last ten years a collaborative student
project called DreameO. Originally, DreameO should be a simple tool for the
promotion and dissemination of video contents created by DREAM students
(also called DREAMers) on the UVHC website. Then, the project evolved
into a tool for managing, archiving and streaming DREAMers' video prod-
ucts as a LAMP (Linux Apache MySQL PHP) platform-based website. To-
day, the aim of DreameO project is to provide DREAM students and acade-
mic staff a complete and customized solution for managing digital resources
(a Digital Asset Management System). This solution is based entirely on
open source technologies and in full accordance with the existing and future
audiovisual work-flow. The DreameO platform is fully designed and devel-
oped by our students and should be based on free software solutions. It
should enable the management, archiving, dissemination of multimedia pro-
jects done by our students. Several types of projects should be covered in-
cluding audio/video products, digital graphics production (2D/3D), but also
the project reports (audiovisual production/post-production or technical re-
ports). Therefore, it is mandatory to account for the variety of file formats as
well as software tools used to make them.

2   Why DreameO?

2.1   Context

Because the project has matured over several years, the reasons that governed
the development of its specifications have also evolved and have become
plural. DreameO fits into the curriculum of our students through project-
based teachings that result in the production of a work. Each project is car-
ried out by students throughout their academic year and is subject to an
evaluation. Hence, each student can achieve an audiovisual product from
start to finish in relation to contents of university education and his/her pro-
fessional project.

   For technical projects such as DreameO, the work is distributed among
the first and second year's master students, respectively. This approach has
the advantage of being able to deal with a problem over a longer period of
time and therefore potentially with a little more depth. Depending on the

topic as well as the expectations and aspirations of the involved students, the project's duration can be used to complete projects from start to finish. The results depend on the motivation of students on projects. Some years, projects are progressing very significantly, while in other years the progress is much more laborious.

DreameO is a sufficiently huge project so that each student can find his/her place. It requires a set of skills ranging from software development to documentation's writing, through bibliographic research, design, ergonomics, evaluation of each implementation's step, internationalization, localization, etc. Each student is determined according to his/her own affinities or learning goals. For example, our students do not receive teaching on the Python language but the project is partly developed in this language. So, this is a personal commitment that leads students to learn this new language, given that their teachings contain an introduction to Java and C/C++. This probably facilitates learning new languages. Finally, beyond the development of software managing, disseminating and archiving tools, one of the main purposes of this project is also to learn many things that are not necessarily related to specific technical courses such as team organization to achieve a specific task, management and testing of the relationship between the team members, but also the in-depth discovery of multimedia technologies.

2.2   Educational Objectives

The supervision of the projects by the academic teaching staff requires a high number of actual hours which vary depending on the nature of the projects. Students must demonstrate autonomy, self-management and self-organization. Thus, they put to good use some teachings on related fields (project management …). Projects are preferably carried out in a group of two or more students. The larger groups of students can help address aspects of tasks synchronization and distribution. They should be able to quantify the amount of work necessary to reach the objectives while continuing to follow regular teachings lectures. Finally, putting students in semi-professional work conditions during their projects lead them to consider operational and practical issues which are difficult to address thanks to purely theoretical teachings.

2.3   Testing Asynchronous Supervision Methods

In order not to interfere with the regular teachings, the methods used for project management have been adapted. This has led to experimentations on

how to achieve effective supervision for this type of project. Asynchronous methods outside teaching hours were chosen first. For the DreameO project, particularly, the management has resulted in meetings with all project participants in the offices of the department. These meetings were programmed "when necessary" by students or academic staff, but still outside of teaching hours. The main requests come from students and concern topics such as obtaining information on the objectives, organizational issues, or sometimes redefinition of more realistic goals. The other part of the leadership is more consistent and is carried through a Google discussion group created specifically for the DreameO project. Students as well as some teachers are registered as group members. No limits are imposed on the topics, their duration, and the number of participants or the accompanying documents. All questions must be able to be discussed and debated, going from the most trivial to more technical ones. The asynchronous method allows everyone to work at his/her own pace and according to his/her availability and agenda. It is mainly used to manage and advance the work on the project, but also to confront the basic principles of the on-line discussion (e.g. netiquette) and simple (but not always taught) rules for effective communication.

These methods work relatively well in all. Over the years, students also learn a lot by browsing conversations as they are kept on the server. However, the success of these methods is not guaranteed, because it also depends on the individuals and groups of students.

2.4   Functional objectives

DreameO must have a number of basic features to meet the expressed needs. These have been identified and listed in the most comprehensive way by a single student, a year when the project was shunned by other students. The work was therefore to meet the future potential student or academic users in order to explicit the required DreameO features, but also to add some new features or modalities. Among these features, the following ones seem to be the most essential for the DreameO tool:

- Be accessible and manipulable for full functionality only through a web interface. Be light enough so that all (sometimes a little older) workstations at the department can use this tool.
- Authenticate each user wishing to change the system data according to a specific usage profile: student, moderator, examiner, administrator. The only-player users are not authenticated. Different profiles can be assigned to the same physical user.

- Archive all the productions of DREAM students such as: videos, soundtracks, 2D/3D synthesis image animations, video special effects, websites, and technical reports. Ensure data integrity and enable their conservation in their entirety while avoiding keeping the intermediate and/or large work files (e.g. backups of intermediary 3D renderings).
- Archive the whole information about each project (who, what, when, where, how, how many, why – in other words the classic credits of any audiovisual production or industrial project).
- Archive productions in the "master" format creation, at the highest quality level, as a basis for the distribution of audiovisual products.
- Perform multi-criteria research based on information relative to each project.
- Make every type of student's production available on our Intranet in broadcast conditions.
- Allow the evaluation and scoring of students productions by an academic panel (examiners).
- Enable academic staff to moderate and administer projects only through the web interface.
- Allow the system administrator to perform the tool's management via the web interface.
- Allow moderators to choose the mode of dissemination of contents depending on the editorial line and available rights. Typically broadcast worldwide, or only restricted to our Intranet.
- Accept upload of different types of projects: video products, audio products, 2D/3D digital graphics, project reports (production/audiovisual or technical post-production) including different types of data formats.
- Enable students as well as the DREAM department to value projects.

Of course this list is not exhaustive and there are other features from the functional specifications established for DreameO, but often these are extensions of the main functions described above.

3   Production and Implementation

3.1   Technological and Financial Constraints

The main constraint, which constitutes the guideline of this project, is that the implementation costs must be only related to hardware. The machine purchase can be made by the institution on the basis of argued and relevant justi-

fications, but no budget can be attributed to software. This imposes to use free open-source software solutions, but also to ensure the lightness of selected software solutions. In fact, the purchase of new machines is not considered before the completion of a working model implementing all required specifications.

One of the choices imposed in terms of specifications require a distributed architecture of the DreameO tool. Indeed, the Apache web server must be hosted by a different machine than the one hosting the MySQL server as well as the one hosting video transcoding functions. It allows load sharing more in line with the usually old materials used for prototyping. Today this architecture is retained even if for prototyping all DreameO components have been installed on a single machine. Eventually this application will be installed on Xen[1] or KVM[2] virtual machines hosted on different servers.

Video transcoding must be done using the *ffmpeg*[3] software and the dedicated machine must be able to perform conversions driven by the heart of DreameO . This constraint still exists today, but has evolved with technology and the emergence of new tools such as *ffmbc*[4] more in line with the type of media "ingested" by DreameO input (video format files suited to professional editing tools) or as *mencoder*[5]. Generally, DreameO works primarily with *ffmpeg*, but should be easily configured by the user web interface to add any new tool.

The network infrastructure that fits this system is the one offered by the Intranet of our university, but no changes are made to it. We benefit from the increased Ethernet speeds of this infrastructure (first 10 Mbps and the 100 Mbps and finally the 1 Gbps). Among the constraints, we aim at limiting as much as possible the traffic generated by the entire DreameO system, so as not to overload it.

3.2   Original DreameO Version

The initial DreameO version was developed on a LAMP platform (GNU/ Linux, Apache, MySQL, PHP) for the sake of convenience for students in

---

1 http://www.xenproject.org

2 http://www.ilnux-kvm.org/page/Main_Page

3 http://ffmpeg.org

4 http://code.google.com/p/ffmbc/

5 http://www.mplayerhq.hu/

charge of it. This initial version model has lasted a few years in this architecture. It was using VLC[6] for streaming on the UVHC Intranet. The introduction of bash scripts was needed to start and stop VLC as a classic server on a GNU/Linux Ubuntu distribution. Video transcoding was intended only three formats. Flash video and Real video formats should be used for dissemination of audiovisual contents outside of the university to meet the technical requirements and in particular the low rates. The third format was the MPEG2-TS one. MPEG2-TS was intended to allow SD (PAL 768×576) audiovisual content delivery over UVHC Intranet with broadcast quality.

3.3   Evolutions

Of course, over the years, the project has followed the technological developments, but also the skills of students in charge of it. There are three years after the almost complete specifications of the DreameO project were drafted and approved by the academic staff, students decided under the leadership of one of them to abandon PHP in favour of Python[7]. The main criteria that had led to this decision were the difficulties to perform some actions including the launch of remote tasks on other machines, full capacity in terms of object-oriented programming offered by Python, but especially high skills of one of the students in this field. So he was responsible to teach Python to his colleagues. Technically it also seemed attractive and a new challenge.

In order not to re-write code already done by others, our students use the proposed "framework" Pylons[8] whose primary objective is to provide a set of objects for the rapid development of Internet sites. It is based on the Model/View/Controller design to add functionality to a website to behave much like an application, with procedures performed on the server side to update the HTML pages. Our students chose *mako*[9] as a Python library with HTML template management (already integrated in Pylons). This tool allows accelerating and facilitating the creation of complex HTML pages dependent of data stored on the server side. Then they chose *SQLAlchemy*[10] Python toolbox for SQL and object-relational mapper. It facilitates the development

---

6 http://www.videolan.org

7 http://python.org/

8 http://www.pylonsproject.org/

9 http://www.makotemplates.org/

10 http://www.sqalchemy.org/

of procedures for handling database and allows them to be independent of the chosen database. Thus, prototyping done by our students in 2013 was based on the *SQLite*[11] database and not on *MySQL* without changing DreameO data processing code when we will activate its operation with *MySQL* or *MariaDB*.

These choices demonstrated their relevance since the re-design of the project with the functionalities of user authentication and identification with user profile could be implemented fairly quickly and tested. Of course, this was supplemented by other basic features: project creation, media addition. Some tests were also performed on the relevance and how to insert video players purely written in *JavaScript* directly integrated into the website pages.

The last two years were devoted to the complete rewriting of the existing code in Python 3 in order to switch completely DreameO to *Pyramid*, which is the new *Python* framework. This therefore needs to understand the new paradigms, and adapt the use of third-party libraries. It is also one of the reasons that led to the use of *SQLite* rather than *MySQL*, as the *SQLAlchemy* driver for it has not yet been ported to Python 3. Students paid much attention to the fact that all the DreameO code meets the specifications of Python 3.

Great care was made to develop functionality for authentication and user profile management, so as to allow the use of the tool even with reduced functionality because in development. The Pyramid code can act as an independent web server (like many Python framework for the web), it was decided to use WSGI[12] (a gateway interface for Python web server) to integrate this application written in Python to a standard Apache web server. So it loses its autonomy, but found the standard way of any website by using Apache to manage connections and traffic. This also allows relying on the functionality provided by Apache for security and the deployment of servers (virtual servers).

The first tests of DreameO installation in a virtual machine were made and conclusive. No constraint is set on the use of a specific distribution of GNU/Linux for the prototyping stage, our students are free to choose the one that suits them best. Some use Ubuntu, others ArchLinux, Debian or Fedora. Finally, the students decided to promote a recognized distribution widely used in the actual servers: CentOS distribution. Some of our students with good skills in website development mainly work on the integration of com-

---

11 http://www.sqlite.org/

12 http://wsgi.readthedocs.org/en/latest/

plementary and consistent JavaScript code to handle the dynamic elements of web pages. Changing *mako* templates associated with many tests allow for example to achieve user authentication phase. Other students focused on internationalization and localization DreameO. Indeed, this aspect has never been built before, even if it was one of the initial specifications. The entire application was therefore modified to meet this requirement. Some translations were therefore included in the project and each user can now choose the language that best suits them from those available.

Finally, students began writing in Python a motor for the complete management of video re-encoding from DreameO to machines dedicated to this task. This required the use of Python components for secure management of network connections and the definition of a basic communication protocol between two machines. Preliminary tests are promising, while still requiring complementary work. The students also decided the implementation and the use of a Git repository for asynchronous collaboration including both the code and detailed documentation. The students organized themselves for the establishment of deposit, installation of access tools and training of their colleagues to the use of this specific tool.

4 Assessment

4.1 Functionalities

In its specifications, DreameO has advanced features that we have not found in conventional or set works differently tools:

- For example, the storage of objects produced by our students in their native format coupled with information stored in a database allows to combine the advantages of both worlds (file storage and database). Indeed, database browsing is faster than files processing for meta-data extraction. In addition, the number and kind of information stored in a database is more flexible and editable.
- Another example concerns the possibility with DreameO to compose the various multimedia contents dedicated to a broadcast medium (DVD, Blu-ray, CD) by choosing from the products already archived. Then a semi-automatic "authoring" procedure of the chosen medium provides a disk image ready to burn directly to the client who made the request (not yet implemented). This type of functionality can allow academic staff to easily create a program that will be distributed at trade shows to promote

our programs, but also to establish a broadcast medium for a single product.

- One of the first DreameO objectives was to allow streaming on our Intranet of any content selected from the product catalogue archived, with broadcast quality. Originally the technical choice was the MPEG2-TS format as many sources were from video formats such as DV or DVC-Pro. We have adapted our goals to aim today formats such as full HD 1920×1080 video streaming used by our students. Of course the highest level of quality is considered, as we explore the possibilities of using standard formats offered by HTML5 (*H.264*, *WebM*, *Ogg Theora*), but also with more specific streaming solutions (H.264 streaming with software-dedicated server, or Flash video contents encoded with *ffmpeg*). The implementation of these solutions in DreameO must allow their use transparent to users of the distribution system.

4.2   Specificities

If the interest to develop this project from scratch is obvious in an educational context, it seems more difficult today to justify the use of DreameO compared to existing widespread solutions. The primary objective of this platform is to enable the archiving of audiovisual productions of students. As done by professional broadcasters, archive is ideally made from the "master" program, that is to say the finished document which serves as the basis for duplication for the purpose of dissemination. The master is stored on a medium using the best format in terms of data recording (resolution, colorimetric space, noise level, sound dynamic …). An archive is by definition meant to last, but also to keep a record to the best quality. Ideally this should be the master itself. Otherwise, the archive should be derived from the master while keeping the integrity of the original document.

YouTube, *Flickr* or other on-line multimedia storage systems do not have this concern. Broadcasting is the only objective, hence audiovisual contents are stored in formats which do not preserve the original characteristics of the master. Different lossy compression algorithms, resolution decrease, changes of colour space, various containers modify the original document. Savings in terms of storage capacity as well as network bandwidth are the most essential criteria for these platforms, even when video contents are available in full HD resolution.

The goals are different when you want to make an archive. Only the intact preservation of the document in its original state must be the first criterion.

Regardless of the volume of data generated and reading difficulties that may arise, only the intrinsic quality of the document and its sustainability over time should be considered. Any archive that meets these quality criteria can be duplicated and converted to other media or broadcast formats, with other resolutions and various containers. But it is still possible to see the quality in which it was originally designed and distributed. One constraint is to keep the means of reading it in time archiving also if necessary players in a sustainable way. Otherwise, it is always possible to re-encode contents in formats that do not alter original quality. This is what we have started through another student project which aims to digitize old audiovisual contents recorded on analog or digital tape archives.

Another reason not to use the above platforms is the copyright and/or related rights. Indeed, the work of our students, even if they are aware of these issues, may contain non original and non-royalty free elements. This may be illustration music, a pictorial work as part of a scene such as a poster, short document or a piece of work for which written permission must be obtained from the copyright holders. This is very often very expensive. The financial means available to achieve the audiovisual products of our students do not allow us to consider such acquisitions. This obstacle is not necessarily blocking for the realization of an academic product which remains the property of the university and which we know will not spread out (intranet broadcasting only), provided that the artistic quality of the project warrants.

Finally, another benefit of not using public platforms is that we can choose formats consistent with the available broadcast bandwidth, from hundred Mbps on the UVHC Intranet to lower bit rates suitable for limited bandwidths and standard readers available on public platforms.

5   Conclusion

This project is evolving even if these specifications are laid down for a few years now. Different attempts or changes are made depending on technological innovations, the aspirations of our students as well as their skills or their ability to understand the areas in which their knowledge should be reinforced. It is also in their interest to get involved in the project areas that can match their professional project to develop basic knowledge. Generally, students will remain on the discussion group set up from the beginning of the project. It is not uncommon for a former student to discuss with our current students on this discussion group. We hope the DreameO project will succeed because it corresponds to a real need in our DREAM department, al-

though this type of project which evolves over time (like a "rolling release" of a GNU/Linux) and aspirations of each other allows many aspects of collaborative work. Documentation about DreameO being now sufficiently consistent and accurate specifications defined, the implementation of the first full version should be complete soon with goodwill and organization.

Acknowledgement